UNIVERSITY OF RIJEKA
FACULTY OF MARITIME STUDIES

Matthew Sumner

DYNAMIC COLLISION AVOIDANCE OF SEA SURFACE VEHICLES WITH A HIDDEN MARKOV MODEL

Doctoral Thesis

Rijeka, 2021

UNIVERSITY OF RIJEKA
FACULTY OF MARITIME STUDIES

Matthew Sumner

DYNAMIC COLLISION AVOIDANCE OF SEA SURFACE VEHICLES WITH A HIDDEN MARKOV MODEL

Doctoral Thesis

Thesis Supervisor: izv. prof. dr. sc. Igor Rudan

Rijeka, 2021

SVEUČILIŠTE U RIJECI

POMORSKI FAKULTET RIJEKA

Matthew Sumner

DINAMIČKO IZBJEGAVANJE SUDARA POMORSKIH OBJEKATA PRIMJENOM SKRIVENOGA MARKOVLJEVA MODELA

DOKTORSKA DISERTACIJA

Rijeka, 2021.

Thesis Supervisor: izv. prof. dr. sc. Igor Rudan

Doctoral Thesis defended on 24th November 2021 at University of Rijeka, Faculty of Maritime Studies. Committee members:

1. prof. dr. sc. Serđo Kos
2. prof. emer. dr. sc. habil. Tibor Poganj
3. izv. prof. dr. sc. Đani Mohović
4. izv. prof. dr. sc. Marko Valčić

Dedicated to the memory of my son Maro

You were too beautiful for Earth

Abstract

In this thesis an integrated dynamic collision avoidance and hazard alerting system is proposed and identified as Marine Collision avoidance and Alerting System (MCAS). It is comprised of four integrated models that aid navigators in making appropriate decisions to prevent collisions at sea. Before autonomous sea surface vehicles would be allowed to navigate on commercial routes, a robust collision avoidance system has to be developed. Even though MCAS system is feasible for autonomous navigation, development of a decision support system that can be used within the current legal frameworks is in focus.

Research problem is formulated within the Hidden Markov Model (HMM) framework and solutions that are based on various Partially Observable Markov Decision Processes (POMDP) and Reinforcement Learning (RL) solvers are proposed. This approach is based on offline development of robust look-up tables, rules, and protocols that aid online computation of conflict resolutions while preserving overall feasibility and reduce computational expense.

To ensure feasibility of trajectories that are generated by collision avoidance algorithms, autopilot and auto-telegraph models are developed for the motion control of sea surface vehicles. Motion control algorithms thrive under dynamic environmental loads and are capable to control underactuated sea surface vehicles. Heading, course, and throttle algorithms are proposed to allow for larger action space when avoiding collision. Even though model-free approach is envisioned, Model-Predictive Control framework is exploited to propagate signals to motion control actuators. With the intention of reducing uncertainties and improve input data stability, a non-linear dynamic state estimator is proposed, named Foraging Particle Filter, that is based on swarm algorithmic approaches, and is utilized to filter input signals to motion control algorithms.

With the purpose of developing a robust and feasible collision-avoidance system, it is important to ensure that it can be used within the legal framework of collision avoidance at sea. COLREGs classification algorithm that quantifies requirements of collision regulations by reducing vagueness and uncertainties is proposed. Quantification is based on empirical studies and case laws. COLREGs classification algorithm is used to

produce input signals to collision avoidance algorithm and in that way decentralize computation. In order to generate evasive trajectories, predictor is developed that takes feasibility of turns into account and ensures trajectories are hazard free. Simulation results confirmed that the proposed system is capable to avoid complex close-quarter situations.

Previous research has focused on egocentric resolutions where only own vehicle is equipped with collision avoidance systems, while in this research, a holistic collision risk resolution model for multiple targets in mixed equipage situations is developed. Communication protocols are utilized to share intent and other relevant information that is required to reduce uncertainties and computational complexities of trajectory generations. Simulation results demonstrate feasibility and have shown that intent-aware approach outperforms egocentric conflict resolutions, as well as leads to reduction of close-quarter situations as it is possible to foresee collision risk in early stages of passage exploitation. Proposed multi-objective optimization-based collision avoidance method allows conflict resolutions with higher CPAs, reduces distance travelled to avoid collision and shortens time required to go back to the original route.

To further reduce computational complexity of the collision avoidance algorithm, benefit of having decentralized unit for hazard alerting is investigated. This research showed that nuisance alerts onboard commercial sea surface vehicles are a substantial problem that has to be confronted by exploiting design of trajectory generator and hazard alerting algorithm that managed to considerably reduce nuisance alerts and ensure that only relevant alerts are triggered in collision-avoidance situations.

KEYWORDS: dynamic collision avoidance, hidden Markov model, partially observable Markov decision processes, reinforcement learning, ship motion control, intent-aware navigation, early detection of collision risks.

Sažetak

U ovom radu predložen je integrirani sustav za uzbunu i dinamičko izbjegavanje sudara na moru, pod nazivom Pomorski Sustav za Uzbunu i Izbjegavanje Sudara (MCAS). MCAS sustav se sastoji od četiri integrirana modela čija je namjena pomoć pri donošenju odluka u situacijama izbjegavanja sudara na moru. Prije nego se autonomnim plovilima dopusti plovidba u komercijalnom okruženju, potrebno je razviti robustan sustav za izbjegavanje sudara na moru. Iako se MCAS sustav može koristiti na plovilima sa autonomnim upravljanjem, cilj je ovog istraživanja razvoj sustava za podršku pri odlučivanju koji je u skladu sa trenutno važećim pozitivnim propisima i zakonskim okvirima međunarodne plovidbe.

Problem istraživanja definiran je u okvirima Skrivenih Markovljevih Modela (HMM), te su predložena rješenja koja se temelje na okosnicama metoda za rješavanje djelomično vidljivih Markovljevih procesa odlučivanja (POMDP) i podržanog učenja (RL). Pristup ovog istraživanja se temelji na izgradnji dvostrukog sustava koji dopušta da se određeni procesi vrše pomoću robusnih interpolacijskih tablica, pravila i protokola koji služe podršci izračunavanja optimalnih trajektorija za izbjegavanje sudara na moru koja se dešava u realnom vremenu, te se na takav način osigurava izvedivost sustava, ali i smanjuje računalno opterećenje.

Da bi trajektorije koje generira algoritam izbjegavanja sudara na moru bile izvedive, u ovom radu se razvija model automatskog upravljanja plovilom i sa automatskom kontrolom pogonskog postrojenja. Algoritmi upravljanja uspješno se nose i sa dinamičkim atmosferskim i morskim opterećenjima, te su u mogućnosti kontrolirati plovne objekte bez dinamičkih sustava za upravljanje. Predloženi su algoritmi za održavanje smjera plovidbe, kursova, i brzine plovnog objekta, kako bi se omogućila bolja upravljivost prilikom izbjegavanja sudara na moru. S namjerom smanjenja šuma i u cilju povećanja točnosti ulaznih podataka, u radu se predlaže nelinearni filter čestica za određivanje stanja pod nazivom Foraging Particle Filter (FPF) koji se temelji na genetskim algoritmima, te se koristi za filtriranje ulaznih signala u algoritme upravljanja kretanjem plovnih objekata.

Kako bi se predloženi model izbjegavanja sudara na moru mogao koristiti u praksi, potrebno je osigurati da se sustav može koristiti u skladu sa svim pozitivnim propisima i zakonima koji uređuju plovidbu međunarodnim morima. Predložen je algoritam klasifikacije pravila o izbjegavanju sudara na moru koji kvantificira zahtjeve pravila o izbjegavanju sudara na moru i na taj način smanjuje nejasnoće koje u današnjim pravilima postoje. Analiza pravila o izbjegavanju sudara na moru obuhvaća empirička istraživanja i sudsku praksu. Klasifikacijski algoritam se koristi za analizu trenutne situacije u okruženju i generiranje ulaznih signala za algoritam izbjegavanja sudara na moru, te se na takav način vrši decentralizacija računalnih resursa. U svrhu generiranja izvedivih trajektorija prilikom izbjegavanja sudara na moru, predlaže se prediktor koji uzima u obzir manevarske sposobnosti plovnog objekta, te osigurava da trajektorije budu bez sudarnih opasnosti. Rezultati istraživanja su potvrdili da je predloženi sustav u mogućnosti riješiti složene situacije izbjegavanja sudara plovnih objekata.

Prethodna istraživanja su usredotočena na slučajeve gdje je samo naše plovilo opremljeno sustavima za izbjegavanje sudara, dok je cilj ovog istraživanja razviti holistički model smanjenja rizika od sudara za više plovila od kojih su neki opremljeni sa sustavom za izbjegavanje sudara, dok drugi nisu. Komunikacijski protokoli se koriste za izmjenu informacija i namjere kako bi se olakšala generacija optimalnih trajektorija. Rezultati istraživanja pokazuju da su generirane trajektorije izvedive i da je pristup kod kojeg se rješava situacija za više plovnih objekata bolje rješenje jer dolazi do smanjenja rizičnih situacija, te pruža mogućnost ranog otkrivanja rizika sudara. Predložena metoda omogućuje veće rastojanje među plovnim objektima, predlaže trajektorije sa manjim devijacijama od planirane rute, te vrši povratak plovnog objekta na planiranu rutu u kraćem vremenskom razdoblju.

Da bi se dodatno rasteretio računalni proces pri generiranju optimalnih trajektorija za izbjegavanje sudara, istražena je korisnost decentralizirane jedinice za uzbunu od sudara. Istraživanjem je ustanovljeno da postoji veliki broj uzbuna prilikom korištenja navigacijskih uređaja koji uznemiruju navigacijske časnike i ne vode ka sigurnijoj navigaciji, već stvaraju dodatni rizik gubitka koncentracije. Iz tog razloga predložen je sustav koji smanjuje nepotrebnu uzbunu tako da koristi prediktor da bi odredio kada je uzbuna potrebna, a kada ne. Predloženi sustav je uvelike smanjio broj nepoželjnih uzbuna i osigurao da se samo relevantna upozorenja aktiviraju u situacijama izbjegavanja sudara na moru.

KLJUČNE RIJEČI: dinamičko izbjegavanje sudara, skriveni Markovljevi Modeli, djelomično vidljivi Markovljevi procesi odlučivanja, podržano učenje, kontrola kretanja plovnog objekta, pravila o izbjegavanju sudara na moru, izbjegavanje sudara uz propagiranu namjeru, rano otkrivanje rizika od sudara.

Acknowledgments

This dissertation has been partially supported by the Croatian Science Foundation under the project HRZZ-IP-2018-01-3739 (Decision Support System for Green and Safe Ship Routing).

First and foremost, I am extremely grateful to my supervisor, Prof. Rudan for all his efforts, invaluable advice, selfless support, and limitless patience. Igor has been there for me not only as mentor, but also a true friend who spent numerous hours building relationship, confidence, and trust. I will never forget sleepless nights chasing deadlines, quarantine hotels in China and administrative efforts to reach the end. Countless thanks are owed to members of defense committee: professors Kos, Poganj, Mohović and Valčić for their inordinate amount of time and energy to guide and aid my progress. Special thanks to professor Srđan Žuškin, an excellent teacher and fantastic friend, for spending hours with me in the lab and on the simulator, and for genuine support on this journey. I would also like to thank Prof. Alen Jugović for believing at me all this time.

I would also like to thank my primary school mathematics teacher Mrs. Marija Loncaric for giving me one of the most important lessons of my life. Even though I've never received any words of encouragement, but rather foresights that nothing but failure is destined on my path, I am appreciative to have someone to strive against. Special thanks also go to my dear friends, Zvjezdan and Ivan, who both patiently accepted all my socializing rejections because of the work I needed to do. Their warmth and support reached levels outside of the thesis-related dimension.

Above all, the biggest thanks go to my family. To my mom and dad, Ida and Nebojša, who supported all my wishes and dreams since I was toddler and guiding me on choosing right paths. I am thankful to my mother-in-law, Olja, who selflessly provided her support anytime needed. I am appreciative for selfless support, patience and affection received from my sons, Bartol and Maro. Finally, I am eternally grateful to my wife Ana, who never doubted my abilities and provided nothing but support throughout the years. Innumerable amounts of travails, uneasiness, impatience, and absences were always greeted with a smile. Thank you, my love.

Contents

Chapter 1

Introduction

Many research hours have been devoted to make land, sea, and air vehicles safer to operate. Operating environments are complex and dynamic; therefore, any effort invested in developing systems that aid users to safely control operations is a step closer to hazard free environments. Various alerting and autonomous systems are in use as a result of dedicated and thoughtful research, be it drowsiness of the driver, line crossing warnings, traffic alerts within approximated airspace, and similar [Bevan and O'Reilly, 2007; Cheng et al., 2006; Gumaste et al., 2007; Mertz et al., 2000; Mukai et al., 2009]. Even though collision avoidance is topic of many authors in the previous two decades, there is still an area of improvement in developing practical hazard avoidance and alerting solutions for commercial sea surface vehicles.

The severity of impact between two vehicles largely depends on the weight and speed of the involved vehicles; hence it is imperative to assist operators in determining collision risks as early as possible. Considering that aircrafts are both heavy and fast, academia has brought many solutions for the early detection of the airborne collision risks. Shipping industry requires same attention considering that the number of sea surface vehicles is rapidly increasing and that consequences of collisions are often catastrophic for passengers, environment, and vehicles. Unlike the air traffic, decision-making is autonomy of a navigator. Even though International Convention on Standards of Training, Certification and watchkeeping for seafarers (STCW) provides standards of training for all crewmembers operating vessels, differences in competence still exist. This discrepancy in knowledge, competence, abilities, and challenges of human factors, oblige researchers to find solutions that would assist navigators in determining risks of collision and provide hazard avoidance resolution advisories.

In this thesis, dynamic programming and collision avoidance algorithms producing alerts and resolution advisories to navigators are introduced to minimize collision risk during encounters with other sea surface vehicles and hazardous objects at sea. Developing robust Marine Hazard and Collision Avoidance System (MCAS) with logic that reliably prevents collision without excessive alerting is challenging due to sensor errors, sensor quality, and the uncertainty of predicted trajectories of encountered targets.

Even though Moore's law would allow for computational solutions that would support full autonomy, quality of sensors and computing equipment installed onboard commercial sea surface vehicles is poor, while communication standards remain heavily restrictive. With cheaper computational solutions, restrictive communication protocols, and outdated regulations, focus of this thesis is decision support system, rather than full autonomy, even though same models could be used for autonomous vehicles. Decision-support in this context is defined as resolution advisories to human operators in dynamic and uncertain environments by mapping the environment, determining the pose of a vehicle, generating trajectories, and providing optimal action advisories. In order to reduce computational complexity, fused and filtered sensor information is utilized and used as an input to the decision-support system. Instead of finding optimal policies in state and action spaces that are commonly computationally intractable, latent Q-states are exploited and chains of optimal actions found to generate risk-free trajectories. That is why model-free approach with offline and online learning model is selected that can adjust to various disturbances and apply acquired knowledge in future state estimations.

This thesis provides an extension of prior research on maritime collision avoidance, exploring cooperative programming and comprehensive hazard recognition solutions. In situations where a sea surface vehicle encounters other vehicle with a collision avoidance system, it is important that the resolution advisories provided to navigators be coordinated to avoid same direction maneuvers. In order to make collision avoidance system feasible and cost worthy, existing sensors should be used as much as operationally possible. Finally, because many of the navigating areas are becoming increasingly dense with traffic, it is important to design resolution advisories as an aid to the navigator, not a burden. Nuisance collision alerts can discourage users to exploit benefits of the system.

In this chapter challenges of the research, research goals, and hypothesis outlines are defined. Also, we deliver overview of the previous work, present human factor as the main motivation for the research, provide summary of the scientific methods used in the research, and conclude with the thesis outline.

1.1 Problem statement

In many industries human-machine interaction is getting increasingly more complex. Even though many maritime electronic aids to navigation were introduced in previous decades, number of incidents and significant near misses demonstrate that it is becoming difficult for navigators to understand and process all information available on navigating bridge to effectively make appropriate navigating decisions. In order to aid navigators in managing dynamic navigational situations and selecting the most optimal routes to avoid collisions, a decision-supportive collision avoidance system is warranted to minimize undesirable effects of the human element.

Automatic alerting system monitors traffic situation and, when necessary, generates warnings (alerts) to prevent undesirable incidents. Modern ocean-going vehicles are equipped with large number of sensors, all monitored by the Integrated Automation Systems, to simplify operational control of machinery, cargo, or navigation. Alerting systems in aircrafts have appeared to be particularly complex, use sophisticated decision algorithms, and even provide guidance to pilots through various advisories. Examples of successful alerting systems are the mid-air collision [Harman, 1989; O'Hara, 1998; Nordwall, 2002] and terrain [Phillips, 2001; Feith, 2002] avoidance systems installed on many aircrafts.

Newest research has produced alerting systems where decisions are based on real time prediction metrics, rather than on unrefined criteria [Yang & Kuchar, 2002; Kuchar & Yang, 2000]. A shift is made towards clearer alerting requirements, rather than adjusting logic by means of trial and error. A system based on random dynamics, where inputs consist of various data available on navigating bridge (radar, echo sounder, trim and list indication, speed log, etc.) and available intruder’s data is proposed. While the algorithm handles cooperative intruders with lowest communication latency possible, uncooperative targets need more dynamic approach, as their behavior is stochastic in nature and based on probabilistic measures (prediction of forthcoming behavior).

Development of collision avoidance systems is lengthy and rigorous process that can take several years or even decades to be certified [RTCA, 2008]. The challenge is not only to mandate installation of the system, but also to unify performance quality. Commercial sea surface vehicles are highly diverse in performance characteristics and sensor capabilities. From pleasure crafts to Very Large Crude Carriers (VLCC), sea surface crafts vary vastly in maneuvering dynamics. A common system that would accommodate different sensor configurations and maneuvering characteristics would significantly reduce the cost of development and certification.

With a main goal of presenting alerting issues, introducing cooperative protocols, and improving alerting system performance, this thesis continues the path of previous research in designing COLREGs compliant last-minute hazard avoidance system.

1.2 Research objectives, challenges, and hypothesis

The main **objective** of this research is to establish theoretical foundation of COLREGs compliant dynamic collision avoidance method for sea surface vehicles based on probabilistic mathematical models while maintaining cooperativeness of equipped vehicles. Furthermore, there are several areas of hazard awareness and recognition where improvement of existing research results is sought:

- When a specific event triggers an alert (such as low-pressure alarm on a gas compressor) it informs a user that a certain event in the designated process is out of its normal reach. Sometimes an alert is connected with logic that will prevent continuing operation of machinery in order to preserve its functionality and prevent breakdowns. However, there are distinct dynamic behaviors that system can exhibit. Certainly, it is less complex to monitor one or several parameters to trigger the alert, but when the system is as complex as human awareness of collision avoidance, there are many uncertainties that have to be considered. As commercial vehicles sail on different drafts depending on the cargo, ballast water, or fuel quantities loaded, navigator needs to select safety margins to get the alert if a vehicle is approaching shallow water zone. If a navigator selects incorrect safety contours, consequences could be running aground, which can lead to total loss and significant environmental degradation. Uncertainty about correct selection makes it unclear whether an alert is needed or not. If a vehicle is safe and alerting occurs, it can cause irritation of a human operator and lack of trust. However, failing to alert when it is necessary could result in grounding. This presents a challenge for the alerting system, as there is a choice of weighing the cost of different errors to make a probability trade-off between them, or the choice of reducing the alerting uncertainty to allow for more precise decision-making. Furthermore, there is a challenge of timely alerting. If the system can predict based on a current state, this would be of great assistance to the navigator. For example, as the maritime sector allows for longer timeframes for decision-making than in the mid-air close encounters, it is possible to design alerting system to notify navigator that a vehicle would run aground in XX minutes if hazard is not avoided. MCAS system can be considered as an early alerting and last-minute collision avoidance system. The term "last-minute" will depend on many factors, but in general, finding the point in the time domain where navigator has a last chance of acting in order to avoid collision or near miss situation is of interest of this research. In this thesis, modeling of selection uncertainties, simulation of upcoming events, and examination of benefits of probabilistic approaches in development of those models is investigated.

- Depending on the architecture and design, all sensors have inherent measurement noises. Taking into consideration that there are many different shipbuilders and sensor manufacturers, there are numerous accuracy uncertainties. If the information on which the collision avoidance is based is faulty, the algorithm is likely to compute incorrect advice, and can even lead to collision, as proposed system relies on accuracy of data. Also, it is necessary to take into consideration uncertainties about the intention of intruder sea surface vehicles. There is a possibility of intentional collision by hostile target, or an equipped target vehicle that is following advisory of the resolution advisory, or even oblivious target that is just following its passage plan without regards to the traffic around. Proposed algorithms within MCAS resolution advisories have to account for various intruder behaviors and be parametric to accommodate various sensor modalities, sensitivity levels, and sea surface craft maneuvering characteristics. Computational speed is less of a factor in commercial shipping, as usually there is enough time before incident occurs, but nevertheless intention is to present algorithms that can resolve traffic disputes in real time.

- Sea surface vehicles collision avoidance problem is presented using decision planning framework where the ideal performance requires balancing the collision avoidance and passage plan adherence. It was noted earlier that multiple sensor inequalities make significant amount of sensor noise, and that intruding vehicles behave stochastically. This is the main reason why Hidden Markov Model (HMM) is selected to represent collision avoidance problem, where system is described as Markovian, while performance depends probabilistically on a present state. Q-state reinforcement learning is utilized to find optimal solutions without incurring large computational expenses. Whenever discretization is used, it is logical and efficient.

- Contemplating a situation in which own sea surface vehicle encounters hazard or intruder and needs to perform evasive maneuver to avoid collision, it is necessary to consider nonholonomic constraints [Bryant, 2006]. A nonholonomic motion constraint is mathematical structure that explains state changes by including path selection. This basically fathoms that it is not sufficient just to alter the position of a certain object in geometrical space, but a path from one position to another has to be examined, while constraints of that path evaluated. Therefore, it is not enough to only satisfy the resulting safe position of the collision-avoiding maneuver, but also to ensure that the passage towards the resolution position is safe for navigation as well. This challenge introduces hazard avoidance in proposed collision avoidance algorithms. Therefore, MCAS system has to satisfy not only optimization of safe route selection that is compliant with COLREGs, but also verify that there are no hazards, such as shallow water, island or any significant NO-GO areas established in that navigating area. It is necessary to emphasize at this stage that collision avoidance has priority over Temporary and Preliminary notices and other navigational warning that are not permanently embedded within Electronic Chart Display Systems (ECDIS) installed onboard of sea surface vehicle. Therefore, MCAS system will allow vessels to enter areas ordinarily restricted for vessels if there is enough depth and if this is a last-minute resolution advisory as a result of an imminent collision threat.

- The MCAS system has to be risk averse. This relates to the worst-case scenarios while avoiding impact. The algorithm has to be able to optimize selection of the safe trajectory having in mind failure of rudder, unavailability of engine, shallow water or proximity of navigational danger. We envision a situation where own vehicle is avoiding a group of fishing vessels with an island and shallow waters on the port side, and open sea on the starboard side. Avoidance maneuver can be done from the port or the starboard side; however, if done from the port, own vehicle will be closer to an island and have less water available for the maneuvers, while starboard offers safe waters. Turning to port in this situation is not deemed the safer choice, as water is shallower and there is always a possibility that other vehicle will depart from the port or nearby anchorage. It is, therefore, imperative that the resolution advisory prioritizes options with lower level of risk for the vehicle. Quantification of a collision and hazard avoidance risk is utilized to compute dynamic points where vehicle has to maneuver in order to avoid hazards. As the motion control algorithm is taking external disturbances into account, the maneuvering point is dynamic and changes in relation to changes of vehicle speed and pose.

- In order to achieve cooperative collision avoidance resolution advisories, it is necessary to ensure that equipped vehicles communicate effectively. At first, this is possible by introducing specially designed transponders. Modifying already installed Automatic Identification Systems (AIS) would be costly but installing new AIS equipment tuned for the MCAS use is the most cost-effective solution for the future. Considering that the transponder communication (or AIS information sharing in the future) will allow MCAS system to receive passage plans, intruder intentions, and plan escape maneuvers; it is possible to develop a model of an early collision avoidance detection algorithm that can be incorporated within the MCAS. In this work both coordinated and uncooperative sea surface vehicles are captured in differentiating algorithms.

- Interesting fact about collision regulations, or COLREGs, is that they mandate behavior of vessels in various situations, but also leave space for acting outside the framework if it will avoid collision. For example, vessels facing head-on situation should turn to starboard to avoid collision, but if one vessel is unresponsive, the other one has to do everything possible to avoid collision, even if it means turning to port when starboard turn is not an option available anymore. Rather than being in a form of “IF THEN” algorithms, COLREGs are developed to fit human judgments. In the real sector it is possible to face unresponsive vehicles regularly. Among navigators there is lack of trust that other person will follow the rules, which significantly increases level of uncertainty when navigating vehicles. Equipping commercial vessels with MCAS system that consistently follow COLREGs is an effective way to reduce navigating uncertainties and effectively avoid collision. COLREGs have evolved over the years to satisfy technological advances of the shipping industry, however one thing that remained consistent is the necessity to guide navigators in navigation. Future will certainly convey computational advances within navigating platforms, for which COLREGs will have to be adapted once again. With MCAS system protocol-based algorithms that include predictable collision avoidance advisories governed by the COLREGs are explored.

- The main reason why many navigators choose to use VHF communication to aid the collision avoidance is the uncertainty that they feel when navigating in confined waters. Even though COLREGs suggest that expert navigators refrain from using radio communication for the purpose of collision avoidance, many experts of ocean navigation still communicate intent, as there is lack of trust among professionals. To reduce uncertainty of the navigator’s intent is one of the objectives of this study.

As noted above, **research challenges** could be defined as:

Many research hours have been conducted in relation to the maritime collision avoidance, however it is hard to obtain holistic models of hazard avoidance systems. In other transportation fields implementation and further development of machine learning as an aid to navigators is witnessed. In the maritime sector, there is a gap between current research that utilizes fuzzy logic, neural networks, or genetic algorithms, and development of credible collision avoidance systems. This study attempts to bridge this gap by introducing probabilistic methods to collision avoidance in dynamic environments with main focus on development of cooperative hazard avoidance system that would reduce collision risks and minimize uncertainties of targets' behavior. Therefore, there is a need to consistently investigate, diagnose, improve, and appropriately simulate collision avoidance issues in dynamic maritime environments.

With research challenges being determined as said, **research subject** could be defined as:

To investigate underlying determinants of maritime collision avoidance, theoretical sources and practical application; to portray foundations of hazard avoidance and why industry necessitate implementation of machine-learning based systems to guide navigators; to describe application of Hidden Markov Models and especially Reinforcement learning techniques in probabilistic determination of state space, such is collision avoidance problem, and present benefits of employing these probabilistic models in solving intruder uncertainties and resolve various sensor noises; to simulate and depict results of dynamic hazard avoidance system based on HMMs; to propose a solution for cooperativeness and robustness of the collision avoidance system; to design a risk-averse algorithm for hazard alerting that will incorporate COLREGs into resolution advisories and guide navigators to safety.

Constrained by the research problems and research subjects, the **fundamental research hypothesis** is portrayed:

Proceeding from the fundamental determinants of stochastic modeling, and by having in mind constraints of dynamic environment and collision regulations, it is possible to design risk-averse hazard avoidance system that can cooperate with equipped sea surface vehicles and avoid other intruding objects deemed dangerous by the model.

Fundamental scientific hypothesis is directly related to the research subject and implies several supplementary hypotheses:

1. Having in mind uncertainty in which navigators make their decisions, it is possible to develop hazard alerting system based on Hidden Markov Models;

2. It is possible to design collision resolution advisories taking into account various sensory noises and maneuvering characteristics to ensure safe navigation within congested areas;

3. Taking into account both equipped and non-equipped sea surface vehicles, it is possible to develop solutions for collision avoidance in mixed equipage situations by introducing communication protocols between equipped participants;

4. Considering the constraints of recent technological development, especially ECDIS, it is possible to design a collision avoidance model with better understanding of COLREGs to enhance decision making when selecting optimal trajectories;

5. In order to reduce uncertainty of collision avoidance problem, it is possible to adapt an early collision detection solution that will induce observability in latent states of the HMM model.

Taking in consideration complexity of the proposed model, this thesis has a main objective of confirming the fundamental and supplementary hypothesis. Further optimization and development of legal framework would allow development of robust Marine Hazard Alerting and Collision Avoidance System (MCAS) in the future.

1.3 Related work and preliminary research

In the years and decades to come fleet expansion is expected within the maritime sector. Many large commercial sea surface vehicles will occupy common ship routes and increase concentration of maritime traffic, which for a consequence has elevated risk of collisions. Considering a fact that some type of human error causes 75–96 % of marine accidents and casualties [Rothblum et al., 2002; Antao and Guedes, 2008], and that unmanned marine vehicles will eventually vacate same waters, the need for resolving collision avoidance complexities compelled many researchers to investigate solutions for reducing uncertainties.

Various approaches to collision avoidance were introduced over the past several decades. As the technology improved and, most importantly, as the noise of sensors reduced, more dynamic methodologies were developed. Some of the popular methods in the past comprised of edge-detection, certainty grids, or potential fields [Koren and Borenstein, 1991; Borenstein and Koren, 1991; Khatib, 1986; Kuc and Barshan, 1989; Moravec, 1988; Holenstein and Badreddin, 1991]. The edge-detection model uses edges of the obstacle as boundary line representations to avoid collision. Significant disadvantage of this method is that robot needs to stop, evaluate the surrounding and algorithmically determine the existence of edges. With certainty grid approach, representation of the environment is a two-dimensional grid of cells. Each cell has a probabilistic measurement of existing obstacles. This technique also requires periodical stops of the robot and is therefore not suitable for the commercial collision avoidance. Finally, the potential field method uses predefined environment and measures repulsive forces exert by the static obstacles while creating attractive forces towards the goal state. This method has been used largely in robotics after the seminal paper by Khatib [Khatib, 1986]. The potential fields method is known as very successful collision avoidance technique in the static and predefined environment. However, transportation occurs in highly dynamic environments with both predictable and static objects, but also moving targets that behave stochastically.

1.3.1 Mathematical models

Mathematical models are commonly used to define sea surface vehicles' dynamics and surrounding environment. In order to apply mathematical modeling to collision avoidance problem, researchers need to develop mathematical algorithms based on exact definitions and delineated solvers. This approach requires strict definition of all possibilities in advance (offline planning) and presents significant computational challenge, as large amount of data has to be processed before the optimal solution is delivered to the operator.

Newton's second law of motion defines constraints of sea surface vehicle's motion. Bound by the Newton's law, sea surface vehicles have six degrees of freedom [Browning, 1991]. Various solvers [Lisowski and Smierzchalski, 1995] are used to incorporate six degrees of freedom in real-time collision-avoidance maneuvering algorithms that are based on either static, dynamic, kinetic or matrix mathematical models [Lisowski, 1985]. However, it is significant to note that most of the mathematical methods are developed with assumptions of open sea (no static objects or draught restrictions), no change in velocity or trajectory, and that intruders are uncooperative.

Collision avoidance assessment based on starboard maneuvers in close-quarter situation, keeping the sightline always turning counterclockwise, was one of the first methods developed in the academia [Calvert, 1960]. Similar approaches were taken by other authors mainly debating feasible maneuvers to avoid collision [Morrel, 1961; Wylie, 1962]. Mitrofanov [1968] developed an electro-mechanical analogue computer that computed evasive actions and served as an anti-collision indicator. The system largely depended on user, as data had to be manually entered for mathematical model to deliver suggestive maneuvers. The user would get the advisory on the screen as non-shaded area and performed evasive maneuvers accordingly. Similarly, Jones [1974] developed maneuver diagram that helped navigators to determine areas of highest collision risk but did not offer evasive maneuver advisories. Another study that used raw radar data to determine collision risk was the trigonometric model developed by Merz and Karmakar [1976]. Cannell [1981] proposed an one stage cooperative game, where the main goal is maximizing safety through course alteration. The algorithm searched for non-conflicting actions in the matrix of possible outcomes in encounters of two vehicles. At the same time, Degre and Lefevre [1981] developed a navigation advisory model of collision avoidance based on the room-to-maneuver principle. This geometrical model generated danger zones with velocity vectors and closest passing distances. As long as the vehicle was kept outside of the shaded area (visual presentation on the collision avoidance device), collision was unlikely. The system did not consider path selection or optimization.

Ijima and Hagiwara [1991] developed one of the first deterministic autonomous collision avoidance models. It was based on the knowledge-based expert system, and it could determine collision risk, decision-making and maneuvering autonomously. The system searched the collision free path branches, after which it would select the most optimal one by determining collision danger (based on the circular domain), shortest track, least rudder angle movement and COLREGs compliance. It did not consider environmental conditions.

Lisowski and Smierzchalski [1995] proposed an optimal trajectory method, which mandates ownship to take series of precise evasive maneuvers to avoid collisions. The optimal trajectory method is focused on nonlinear computing that incorporates kinetics of the ownship to model safe course deviations based on the nonlinear admittance restrictions. Another course optimization model for definite marine environments is proposed by Skjong and Mjelde [1982] and it is based on point-mass models for ship motion. Graczyk et al. [1995] studied single change of course and/or the speed of the ownship and proposed a Potential Collision Threat Area (PCTA) with resulting safe path that is not always the optimal track. The rigid-body dynamic model for vehicle motion proposed by Yavin et al. [1994, 1995] is another model for collision avoidance maneuvers in confined waters and it is based on stochastic optimum control proposed by Lewis [1986].

One of the pure mathematical approaches was study done by Churkin and Zhukov [1998] in which authors employed both continuous (linear programming) and discrete methods. Linear programming was used to minimize the cost function of the rate of change of yaw, while discrete method of branch-and-bound was used to discretize the course and evaluate the trajectory optimally at each vertex. Linear programming is often deemed computationally expensive, while discrete method requires complex mathematical models to describe scenarios.

An abridged marine collision avoidance system was developed by Miele et al. [1999]. This team considered avoidance maneuvers as Chebyshev problems of optimal control and solved it with sequential gradient restoration algorithm. The system took data about state and control and maximized the minimum time of the distance between two vehicles; however, it did not consider environmental factors and could not be used in multi-vehicle scenarios.

Hong et al. [1999] reported a recursive algorithm for collision avoidance and path selection. The algorithm considered analytical geometry and convex set theory to generate recursive waypoints through heading command sequence, located within the neighboring area of own-ship position. This approach may not be practical for the ocean-going navigation, as it does not mimic human decision-making process and cannot comprehend complexities of marine navigation.

Some of the mathematical models had different approach. For example, Wilson et al. [2003] used the idea of missile proportional navigation to solve collision issue between ownship and intruder. The main goal of the algorithm, called Line of Sight Counteraction Navigation (LOSCAN), is to generate acceleration commands in order to increase the misalignment between the vehicle's relative velocity and the line-of-sight. Burns et al. [1988] proposed broader approach by modeling the ownship and its immediate environment, while Yavin et al. [1997] proposed a tanker realistic model for avoiding intruders and other common obstacles at sea.

1.3.2 Fuzzy set theory and fuzzy inference

Ability of machines to select various degrees of truth and partial truth opened the door into research of computational thinking. A fuzzy concept implies gradations of meaning and is applied to a certain degree or with a certain magnitude of likelihood. In the following text, several prominent studies of fuzzy logic with application to the marine collision avoidance problem are delivered.

Even though there were earlier attempts to define many-valued sets, Iranian computer scientist Lofti Zadeh is the first person to formally define fuzzy concept in his pivotal 1965 paper on fuzzy sets [Zadeh, 1965]. Logicians and philosophers sometimes call such an approach "degree-theoretic semantics", but the more usual term is fuzzy logic or many-valued logic [Cook, 2009]. By specifying a range of conditions, categorizing, identifying operational rules, examining how probable conditions are, or assigning some scale of measurements it is possible to decrease the amount of fuzziness. This reduction of fuzziness is commonly known as defuzzification and it is defined as process of logical portrayal of fuzzy concepts by fuzzy sets [Williamson, 1996]. In the predefined set of rules, fuzzy logic is effective in interpolation between those rules. Bearing in mind that human experts commonly set rules using common sense, it is legit to expect that predefined rules are not incisive. Fuzzy logic can be used to make expert decisions based on fuzzy sets and unclear rules or approximate and uncertain data [Jamshidi et al, 1993].

Japanese scientists were the first to apply concepts of fuzzy logic in practice. The first notable application was on the high-speed train in Sendai, in which fuzzy logic was able to improve the efficiency, comfort, and precision of the ride [Kosko, 1994]. Concurrently, in 1994 group of computer scientists at University of New Mexico developed fuzzy logic collision avoidance for a mobile robot [Martinez et al., 1994]. Martinez et al. successfully managed to achieve collision free sensor-based motion control of a mobile robot. A fuzzy logic based intelligent control was used to computationally handle uncertainties inherent in the collision avoidance problem. Main advantages of fuzzy logic approach to collision avoidance is the ability to model obstacle recognition using the linguistic terminology and that the computational load is considerably lighter than those of edge-detection, certainty grids, or potential fields. There is no pre-defined route used in this study, but just simple addition of higher-level path planning heuristics that would allow robot to follow route and avoid obstacles throughout the navigation. In the study by Martinez et al., researchers determined that 16 rules were enough to effectively avoid collision of mobile operated robot as if the human expert performed controlling tasks remotely.

James [1986] adopted fuzzy set theory to categorize collision avoidance decisions based on distance and passing side. Hasegawa [1987] was one of the first scholars to offer automatic collision avoidance system for ships based on fuzzy logic. Rommelfanger [1998] investigated human decision-making process and categorized it with multicriteria fuzzy logic.

Hwang's study of fuzzy collision avoidance for sea surface vehicles developed an expert system of fuzzy interface, an inference engine to simulate expert's decision and robust autopilot system to guide vehicles towards safe waters [Hwang, 2002]. The system allows for either navigator or autopilot to actually steer a vehicle, after the fuzzy logic resolved potential danger and proposed a safe route.

Lee and Rhee [2001] used a fuzzy reasoning method, namely TCPA and DCPA, to determine and resolve collision risks. Authors decided to engage layering approach where first algorithm would browse the action space (an expert knowledge pool of COLREGs) and then second A* search algorithm would determine real-time safe actions of minimal cost (collision risk x the required travelling distance). Even though the model was proven feasible, it did not take environmental influences into account and assumed constant speed of ownship.

Another COLREGs incorporated collision avoidance system was developed by Lee and Kim [2004]. The system uses polar histograms, developed by Moravec and Elfes [1985] for mobile robots' navigation, to represent computational risk of collision around ownship. Histograms would present valleys (areas of no data) that are all potential safe sectors for a vehicle to navigate to. Fuzzy relational product was used to evaluate the most optimal safe zone. All potential safe zones were then checked for COLREGs compliance and resultant navigational path selected. The major drawback of this system was its incapability to deal with multiple encounters.

A study by Kao, Lee, Chang and Ko [2007] investigated efficient fuzzy alerting system for vessel traffic services (VTS). Automatic Identification System's (AIS) data was integrated in the Marine Geographic Information System (MGIS) to propose a platform upon which the collision alerting will be delivered to the VTS personnel. The study used calculations of sea surface vehicle domain with vehicle inertial force to generate models of a guarding ring and develop danger indexes. The research team used a marine GIS spatial analyst module to predict collision time and position aiding the VTS operator in early decision making.

L. P. Perera and his team from the Technical University of Lisbon focused on intelligent decision making for collision avoidance based on the fuzzy logic [Perera et al., 2009]. Resulted system positively satisfied simulation testing where vehicles successfully avoided collisions bound by the COLREGs. The study took in consideration target vehicle's position and velocity as main contributing factors in determining collision risk and potential, as well as the size and shape of the vehicle domain, together with the area bounded for the water dynamics, as other causal elements of collision risk assessment. The main advantage of this study is amalgamation of helmsman's expertise and expert knowledge of ocean navigation with the collision avoidance algorithm. This allowed the fuzzy inference to act realistically as if the human expert was making decisions and navigating a sea surface vehicle. This research resulted in effective computational detection of the collision risk; however, it is assumed that more complex collision conditions in multi-vehicle situations can occur, and that uncertainties regarding the target vehicle remain to be resolved.

Another analysis of fuzzy logic collision avoidance system includes vessel traffic service (VTS) collision alerting. Su, Chang and Cheng [2012] developed a knowledge base of COLREGs and incorporated it in the fuzzy monitoring system by proposing a novel collision danger domain that forbids entering of give-way vehicle. Special attention was given to the optimal rudder control to avoid large deviations from the planned course. Engine movements were not taken into account. Researchers developed a system that provides navigational warnings with collision danger levels and aids VTS officials to recommend the optimal rudder steering advice to navigators in the surveillance area.

Fuzzy logic can be used to reduce fuzziness within decision-making, however, if not defined adequately, fuzzy logic can fail to resolve uncertainty. Considering this notion, fuzzy logic alone is sometimes considered inadequate to cope with the complexity of real time collision avoidance, therefore hybrid expert and neuro-fuzzy systems are proposed by the academia.

1.3.3 Artificial Neural Networks and Hybrid models

Considering that fuzzy logic is generally insufficient to cope with dynamics of the collision avoidance problem and that mathematical models offer only deterministic algorithms, focus of the academia has shifted to hybrid models of collision avoidance. Sea surface vehicles and environments they navigate in are complex non-linear systems that require heuristic approach.

Neural networks are a collection of statistical learning models used for estimation of functions that are based on a great number of indefinite inputs. Neural networks used in machine learning are based on biological neural networks where a series of interconnected neurons interact with each other. In machine learning, neurons have adaptive quantified weights with ability of learning. McCulloch and Pitts [1943] developed computational foundation for neural network in 1943, but after the pivoting paper by Minsky and Papert [1969], academia realized that processing strength of current machines was a limiting factor to effectively handle neural network's computational demands. This discovery slowed the research of neural networks until computers were able to handle higher processing demands. The major advantage of neural networks is the ability to approximate functions they learn from observed data.

Neural networks' ability to learn [Anderson, 1995] persuaded authors to apply this methodology [Patterson, 1996] to the collision avoidance problem and navigation in general. Xianyi [1999] was one of the first scholars applying the neural networks framework to real time robot collision avoidance problem. Fuzzy Neural Network (FNN) based solvers were developed by Hiraga et al. [1995] in order to exploit fuzzy rules to quantify static and dynamic danger levels and develop decision-making charts for collision avoidance. Zhu et al. [2001] used artificial neural networks to calculate intruder's domain based on visibility, CPA direction, and maneuvering characteristics as deterministic input factors. By applying artificial neural networks to train fuzzy inference system parameters in an intelligent decision-making support system, Zhuo and Tang [2008] developed fuzzy logic system with the goal of solving the anti-collision problem in multi-vehicle encounter situation. Harris et al. [1999] proposed another neuro-fuzzy system with the main goal of developing intelligent guidance system with generic multi-step predictor for obstacle avoidance.

Benjamin and Curcio [2004] exploited interval programming when solving collision avoidance problem. The focus of the study was to develop expert knowledge system with predefined COLREGs compliant rules, compare it with the external conditions and propose collision-free conforming path. This requires a large and detailed database, which can present computational challenges in the real time collision avoidance.

By developing more complex and multifaceted hybrid anti-collision system based on fuzzy relational products [Bandler and Kohout, 1980], Lee and Kim [2004] addressed multi-intruder challenges of COLREGs. Collision regulations are easy to follow in two-vehicle encounter situations but are much more complex to apply in multi-vehicle collision avoidance and that is where Lee and Kim's system contributed the most.

Another collision avoidance system based on fuzzy set theory and neural networks arose from the study conducted by Liu and Shi [2005]. The system as a whole consisted of three subnet neural networks. The first one determined encounter type based on DCPA, course and distance, and offered resolution avoidance action maneuvers (maneuver to starboard, to port, or act as a stand-on vehicle). The second one managed speed ratios between ownship and target and delivered fuzzy type output of small, equal or large. Finally, the third subnet was controlling alteration action through fuzzy set of magnitude and duration. The system was developed as one-on-one encounter, selected target with the highest risk of collision and ignored other traffic in the area.

Szlapcynski [2006] investigated Chang's et al [2003] raster grid method for path selection and collision avoidance. Szlapcynski added turn penalties, time-dependent forbidden zones and ownship speed reduction ability. The speed reduction was modeled as a linear function of distance to the forbidden zone with the help of binary search algorithm that determined the minimal necessary speed reduction in order to avoid collision. Speed reduction was a last resort in case that change of course is deemed as not possible by the system.

The most common problem with the artificial neural networks is that machines need a large pool of training scenarios prior taking the real-world challenges. There is an ubiquitous threat of undertraining (focusing on only few training experiences) or overtraining (learning the vehicle to always turn to port can lead to machine learning of constant turns to port). Neural networks suffer from the same learning sufferings as do the human brain. Unlike human brain, which stores fractions of experiences and connects it with present state to express memory, machines store full knowledge and require large storing capacities to do so. This is the reason it is necessary to investigate usage of latent Q-states that share fragments similarly like a human brain.

1.3.4 Evolutionary and genetic heuristics

Vonk et al. [1997] developed framework of evolutionary computation as collection of stochastic optimization algorithms design to mimic evolutionary theory of Charles Darwin as “survival of the fittest”. Evolutionary algorithms are most valuable when optimizing search strategies of infinitely large search spaces and solving real world complex problems [Back, 1996, Zeng, 2003]. In collision avoidance, evolutionary techniques search for the fittest solution among the pool of possible outcomes (safe path selection). However, it is necessary to bear in mind that evolutionary algorithms are not strictly defined, but rather heuristic, which indicates that there is no definite algorithmic solution to a problem.

Ito et al. [1999] employed genetic algorithm to compute collision avoidance navigational paths. As many authors did before, Ito et al. used vehicle’s domain to define danger zone, after which the feasible passing points were randomly generated. The genetic algorithm was then utilized to optimize passing points into selected route. Distance, energy loss, danger level and straightness were factors contributing to the optimization algorithm. COLREGs compliance and environmental factors were not part of the scope of this study. Zeng [2003] had a similar approach engaging genetic algorithm with consideration of environmental conditions in open sea conditions. However, it omitted COLREGs as a part of computed safe navigation path.

Smierzchalski [1999] developed a sea surface vehicle trajectory planning system using evolutionary algorithm with a possibility of speed alterations through gene mutations in specific sections of the navigational trajectory. This study used polygon-shaped domains to determine danger zone, then generated feasible navigational trajectories, which were afterwards optimized based on cost function of spatial, time and trajectory's smoothness requirements (maximum turning angle between particular trajectory sections in turning points).

Smierzchalski and Michalewicz [2000] developed a vehicle encounter free navigation based on evolutionary planner navigator (EP/N) study developed by Jing et al. [1997]. Considering that genetic algorithms were successfully used in mobile robot navigation [Lin et al., 1994], Smierzchalski and Michalewicz adapted existing techniques to ensure evasive steering and path generation in predefined environments under the real-time constraints. The system has a component of time and allows the variation of the ownship velocity. This study uses evolutionary theory to generate chromosomes with variable-length sequence of genes. Each gene contains information such as the vehicle coordinates. Finally, each gene stipulates turning point coordinates, safe trajectories among them, and speed of the vehicle required.

Several difficulties exist with evolutionary computation. One of the main issues is complexity of the problem we want to resolve. Namely, finding the optimal solution for highly complex multimodal issues require costly fitness function evaluations that can take hours, or even days to simulate. That is why evolutionary algorithms find good fitness in approximation. There are other issues with complexity, such as gradation. Whenever problem consists of large number of genes that can mutate, there is often an exponential increase in search space size. That is why evolutionary algorithms can hardly cope with the holistic approach to solving. Problem has to be dissected into small issues with designated solvers. The challenge remains in connecting all entities into one general solution. Considering that the mutation considers only previous stage of evolution, the stop criterion is not always clear and there is a constant risk of destructive mutation. In many cases, where holistic problem is complex, operator will separate the problem into small entities and assign individual algorithms to each of the issues, which will result in local optima issue where the system suffers from absence of long-term fitness. Bearing in mind that the collision avoidance problem consists of dynamic data set, evolutionary algorithms suffer from early solvers convergence and can give advisories without considering the full data set and can easily miss the vital information necessary for the safest route selection. Evolutionary algorithms are not a good fit for binary problems such is decision-making (assessment of collision risk existence), but rather optimization of the path selection.

It is evident from numerous attempts to solve the collision avoidance problem that academia offers various solvers and that most probable real-life system will consist of hybrid methods connected into one holistic arrangement. However, not many authors considered issues with sensor modalities, stochastic behavior of intruders, oceanographic and meteorological challenges, and cooperative collision avoidance in dynamic environments. This study bridges the gap towards the holistic collision avoidance system that can assist navigators in decision-making.

1.4 Human element as a root cause of marine incidents

Human activity exists in all manufacturing and service sectors. Faulty design or programming code can cause accidents even after many successful years of exploitation. Initially, cause can be attributed to system or technology, but after deeper investigation, root causes often point towards human error and negligence. Maritime industry is not an exemption and even though 19^{th} century brought staggering improvement in safety standards, maritime fleet expansions and new technologies brought new challenges in maintaining incident-free operations.

International Maritime Organization still defines shipping as highly dangerous industry [IMO, 2004], as the number of accidents is still relatively high in comparison with other industries. Human error is still predominately one of the major causes of the marine accidents (75 – 96 %) [Allianz, 2017]; nevertheless, shorter contracts and increased number of crew onboard can keep this number to a lower level. However, latency of many factors that influence cognitive ability, such is mental fatigue, is still not studied in its entirety, mainly because incident investigations track quantitative data to determine root causes of incidents and complacencies, but only deeper and further analysis of human behavior within organizations can reveal some of the potential root causes.

Some of the recent accidents (MV Wakashio, USS Fitzgerald, MT Sanchi, MV Gulf Livestock 1, Costa Concordia) prompted general public to ask the question: Why ship collisions and accidents still happen with so much of technology advancements? The recent incident investigation about USS Fitzgerald collision with ACX Crystal, which resulted in 7 fatalities concluded that "the course change proved to be a critical error, and investigators were unable to determine the reason for it" [NTSB, 2020]. It is evident that crucial links to find root causes and appropriate preventive actions are still missing.

Even though it brought significant improvements in daily management of the shipboard operations, IMO's [2003] focus has been restricted to risk management, safe operations and environment protection. While these focuses are still relevant, the human element contains additional layers that require careful research in order to properly manage risks. State of the art research delivered various definitions of the human element and there is no clear consensus of defining the term, even though most of the authors agree on the term human error being an incorrect decision, unsafe act, or failure to react. Table 1.1 delivers usual taxonomy of the human element.

As it is depicted in Table 1.1, Human element is comprised of both safety and Human resource categories. While Standards of Training, Certification and Watchkeeping of Seafarers Convention (STCW) and International Safety Management (ISM) focus on improving safety standards and risk management for individuals, teams and organizations, MLC is focused in utilizing known processes of organizational behavior to enhance quality of everyday life of seafarers.

Table 1.1 – Components of the Human Element

	Individual	Team	Organization
Safety	*Accident Causation and Human Error* Crew resource management (CRM) and crisis/emergency management Risk Management: System reliability and resilience engineering Training and assessment of competence	*Maritime Ergonomics* User-centered design concepts Human–computer interaction (HCI) Habitability issues: for example, noise and vibration	*Safety Culture* Organizational culture Safety maturity models
Human resource	*Health, Well-being, and Working Conditions* Performance influencing factors (PIFs), for example, fatigue, alcohol, and drugs Performance measurement	*HRM Practices* Recruitment, selection, and retention	*Corporate Social Responsibility* National cultural differences

Source: Barnett and Pekcan, 2017

Barnett [2005] has done extensive research on the origins and classification of the human error with four main categories: slips, lapses, mistakes, and violations. While lapses and slips are usually unintentional, mistakes and violations often contain intent. However, the real question remains if it would be necessary to search for root causes in active failures of competence and complacent violation of rules, or concentration should be on latent and intangible failures of various organizational, individual and group undertakings. Baker and McCafferty [2005] have determined that the total number of accidents is actually declining, but the human error remains a significant factor in 80-85 % of marine incidents. They have also concluded that fatigue and task omissions play significant role in failures of situational awareness.

Maritime ergonomics plays a significant role when teams are interacting in stressful situations. When the process calls for an immediate decision and inputs are easily derived from computer interfaces, decision-making process will be shorter and information noise reduced. In December of 2000, IMO [2000] developed framework for ergonomic assurance of bridge equipment and bridge layouts that ensure all new vehicles are built and delivered as per the ergonomic standards.

United Kingdom's Marine Accident Investigation Branch has determined that fatigue is one of the major contributing factors in collisions and groundings [MAIB, 2004]. Even though fatigue prevention is a major focus of many authorities and hours of work and rest are audited with scrutiny, unmeasurable quality of sleep and rest is not considered in these audits, so this is one of the areas of improvement and yet another potentially fatal latent risk factor.

Recruitment, selection, and retention is another human resource category that requires careful approach especially if the organization has established safety culture. Shipping organizations are known for their high level of diversity. Diversity can be great for building a sharing platform of experiences, views, and ideas; however, diversity can also create a closed down interaction environment where work force feels detached and voiceless. Considering that there are no human resources representatives onboard vessels, managing competence and organizational behavior is not a trivial task.

These categories can have both positive and negative effects on how seafarers are handling stress and risk onboard ships. Depending on the corporate strategy and culture, human element can be managed, and the best traits of people could be utilized to create additional value. However, it is also evident from many maritime incidents that toxic environment and poor human resource management can lead to catastrophic consequences. Each company will have to decide how much efforts and resources should be utilized to manage latent risks.

Batalden and Sydnes [2017] conducted an extensive study about causality of very serious maritime accidents. They discovered that root causes of very serious maritime incidents (total loss of a ship, loss of life or severe pollution) are predominately latent within the higher level of organization with faulty resource processes, organizational processes and oversights, while for the serious and less serious incidents root causes could be found among unsafe conditions and unsafe acts onboard vessels. The study concludes that shore-ship managerial links have to be strong, open communication promoted, and that a balance between organizational efficiency and thoroughness has to be found in order to safeguard shipboard operations against latent perils. It is important to note that human errors rarely occur because of wrongdoing of a single person, but rather because of multi-level failures within organizations. Proper education and competence development play a significant role, but it is equally important to have appropriate working environment and safety culture that will safeguard against complacency and trivial mistakes that can lead to significant consequences.

Wagengaar and Groeneweg [1987] conducted a study about chain of errors that is still relevant today. They have determined that a human error is rarely a single operator's fault and that it usually takes several layers of errors until it results with incidents. This study determined that out of 100 surveyed maritime incidents, each incident had from 7 until 58 chain errors that resulted in an incident. The main focus of maritime safety analysis and risk management in the past 30 years was to build effective barriers that will prevent rollover of mistakes from one instance to another. Regardless of the efforts and improvements noted, maritime incidents due to chain of errors still occur, so additional protection is required. In the domain of vehicle navigation, an augmented automated system could aid in determining risk potentials and give overview to personnel when action or consideration is needed.

When considering consequences of human errors onboard commercial sea surface vehicles, incidents caused by improper handling of equipment and tools, faulty maintenance or no maintenance at all, or failure to follow procedures and regulations are identified. MAIB [2004] conducted a survey of 1647 collisions, groundings, and reported near misses, and they determined that there are three major causes of maritime incidents:

- 1/3 of incident were related to fatigue;
- 1/3 were caused by the lack or loss of situational awareness;
- while 1/3 of the maritime incidents were caused during nighttime with a single navigator occupying navigational bridge.

Further on, the MAIB's survey noted that 55 % of maritime incidents are collisions, 31 % are grounding, while 14 % belongs to the rest of the categorized incidents. It is interesting to note that 67 % of maritime incidents happen in good weather conditions with good visibility.

Maritime incident investigation reports and statistics are usually done by a flag state of a vehicle, or by local investigative branches of the closest land. Insurance companies also track statistics of claims for the vehicles they cover. Therefore, aggregated information studies are scarce. European Maritime Safety Agency (EMSA) collects data about marine incidents for all EU and UK flagged vessels, as well as for all incidents that happen in EU waters. Protection and Indemnity insurance clubs (P&I clubs) insure maritime organizations. P&I clubs also make annual reports for their covered sea surface vehicles.

In their latest Annual overview of marine casualties and incidents for the year 2019, EMSA [2019] outlined key statistics for the period from 2011 until 2018. In that period there were 23,073 casualties and incidents with 25,614 vehicles involved, out of which 230 vehicles were declared a total loss. As depicted in Figure 1.1, more than a half of the surveyed incidents (54.2 %) were navigational errors leading to contacts with fixed objects (15.3 %), grounding (12.9 %) and collisions (26.2 %). 78 % of incidents occurred within territorial waters. It is discovered that 65.8 % of incidents were attributed to the human error, out of which 65 % are related to the shipboard operations, while the rest is related to shore management.

In figures 1.2, 1.3, and 1.4 it is noticeable that from a total of 4104 investigated marine incidents from 2011 until 2018, 65.8 % were caused by a human error, out of which 65 % were attributed to shipboard operations and 22.2 % to shore management. Under the category of shipboard management (SO), safety awareness was identified as the most contributing factor, while under the category of shore management (SM), inadequate procedures were considered as the most contributory factor.

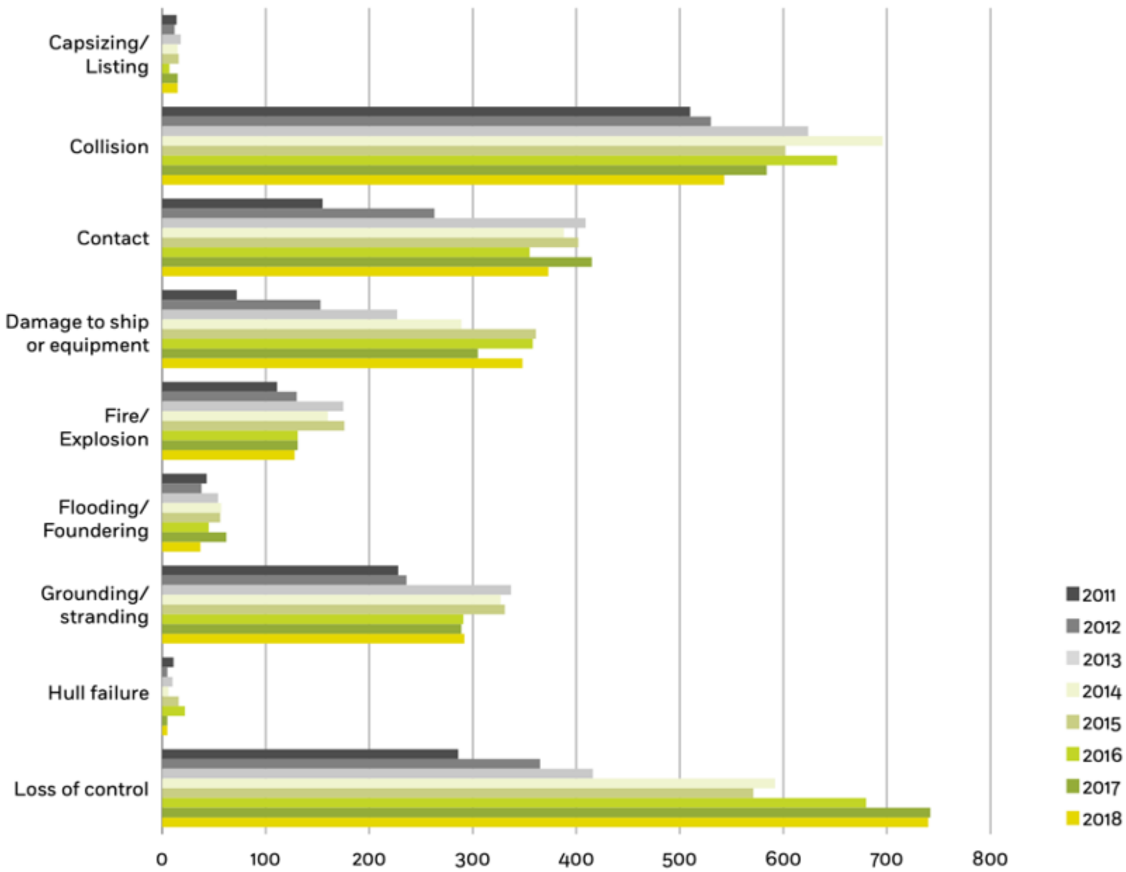
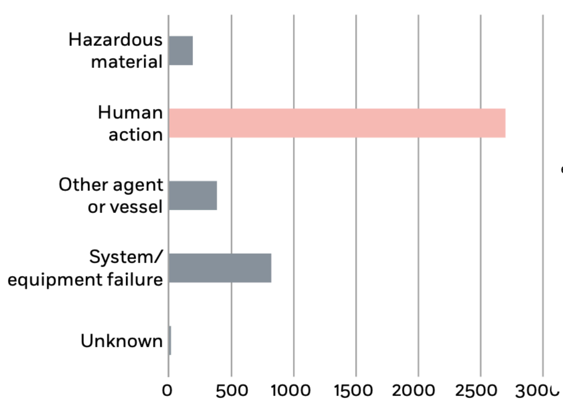
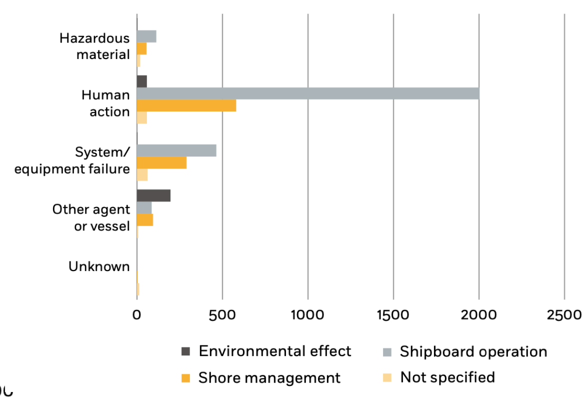


Figure 1.1 – Casualty events involving a sea surface vehicle (Source: EMSA, 2019)



(a)



(b)

Figure 1.2 – (a) Accident causes 2011-2018; (b) Main contributing factors 2011-2018 (Source: EMSA, 2019)

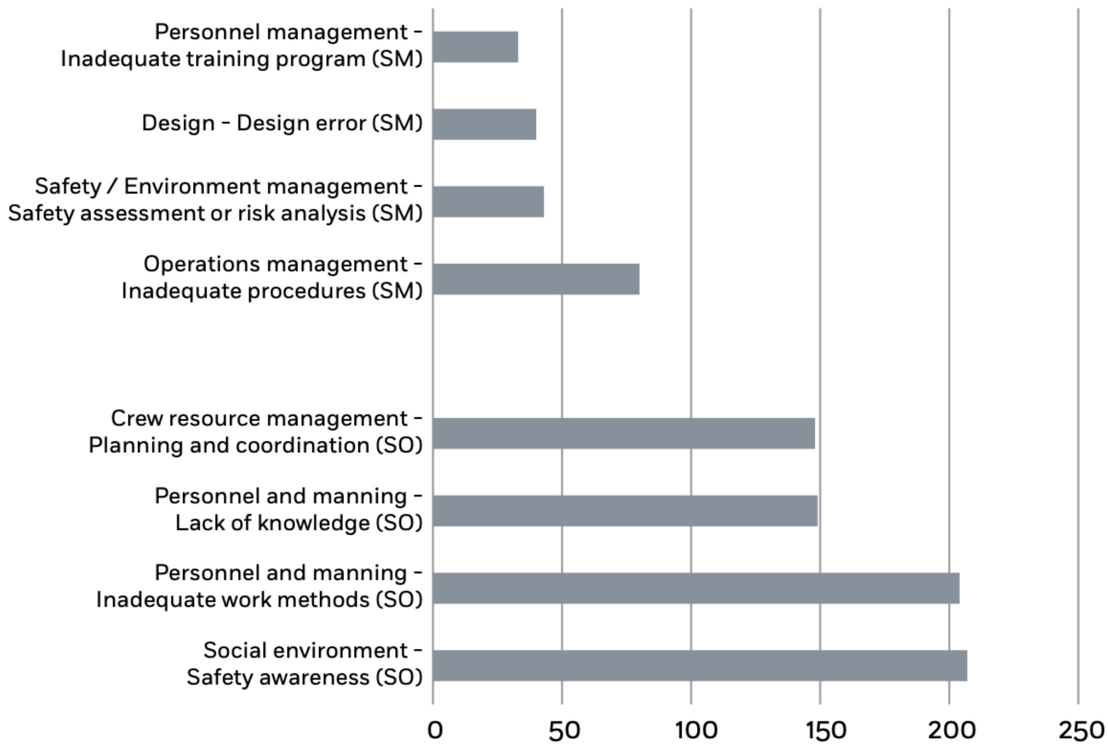


Figure 1.3 – Contributing factors related to the Human element (Source: EMSA, 2019)

In the continuation of this chapter, loss factors from the perspective of insurance are presented, but when looking through the prism of human life, figures 1.4 and 1.5 show that fatalities and injuries predominately happen during collision, capsizing, flooding, and grounding. It is, therefore, desired to reduce these numbers by implementing additional supporting systems that would aid navigators in making decisions to protect life, environment, property, and cargo.

The Swedish Club, one of the prominent Protection and Indemnity clubs, releases incident statistics every year. Their publication Claims at a Glance [2019] delivers interesting casualties' statistics from the perspective of financial loss. It takes into account only insured vehicles, but it has sufficient vehicle distribution to represent an industry as a whole.

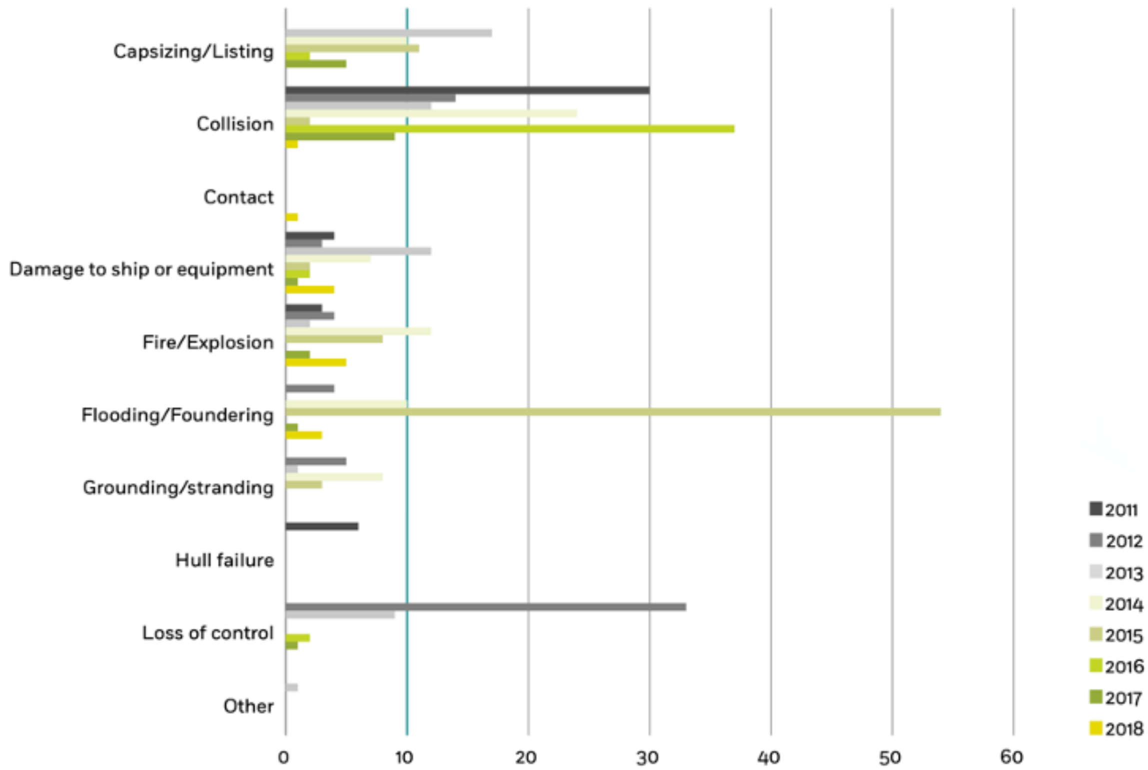


Figure 1.4 – Casualty of fatalities 2011 – 2018 (Source: EMSA, 2019)

"There is often no guarantee that a different decision would have given a different result" [Swedish Club, 2019]. With this statement Swedish club wanted to emphasize notion that assigning blame is not worthy process and that learning from previous incidents is worthwhile. It is true that it is not possible to guarantee different outcome if the decision was different, especially when dealing with complex situations. However, there are computational methods that can aid us in quantifying decision making, which can result in designing decision support systems that can reduce number of incidents, but because humans are still part of the decision-making process, it is not possible to state that decision support systems would eliminate incidents completely.

Even though cargo, illness, and injury claims have the highest frequency, collision, other P&I, and pollution have the highest average cost, but their frequency is low with 1 % of insured vehicles colliding per year [Swedish Club, 2019].

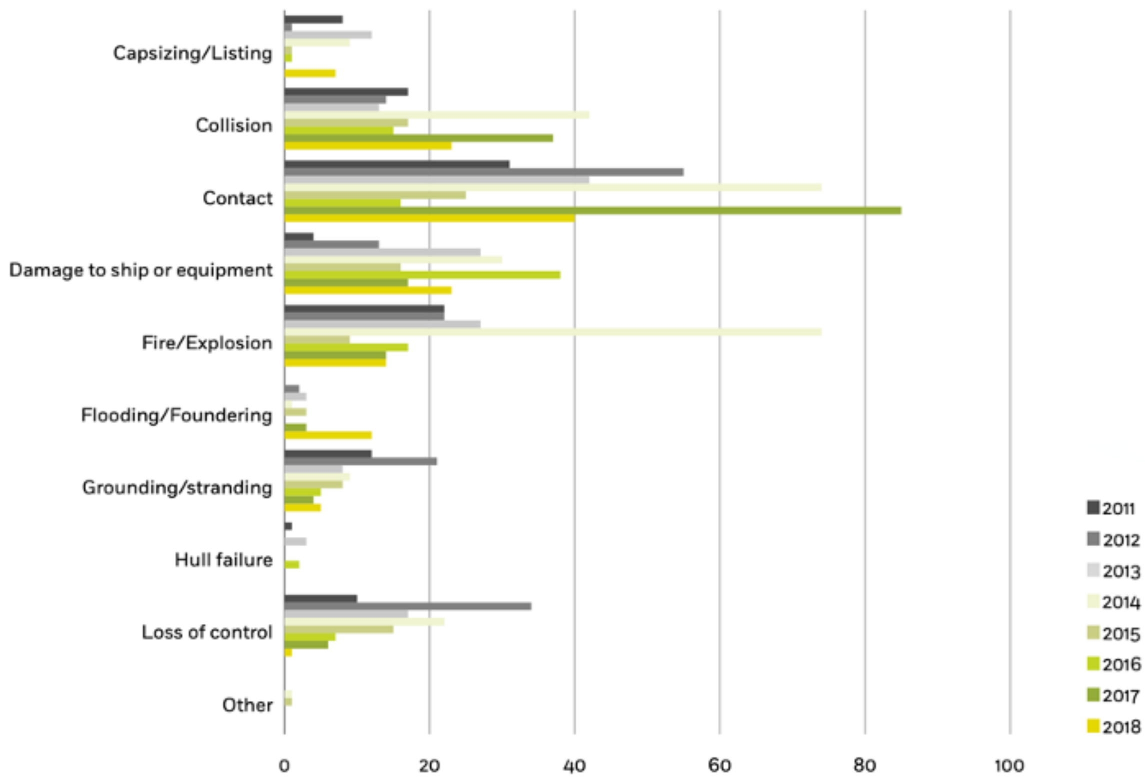


Figure 1.5 – Casualty of injuries 2011 – 2018 (Source: EMSA, 2019)

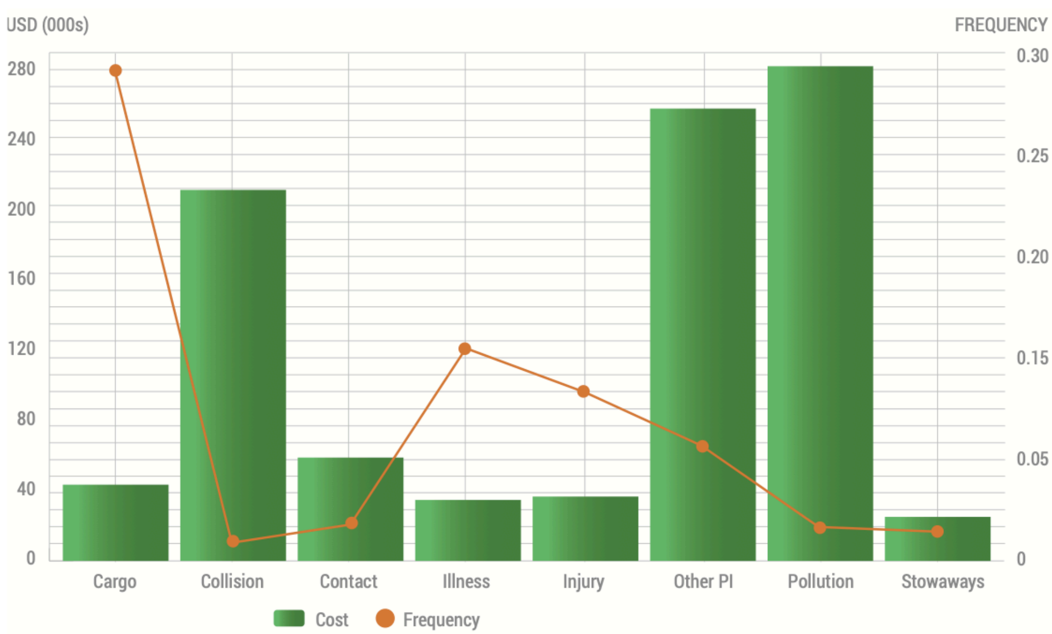


Figure 1.6 – Overview of costs and frequency of maritime incidents (Source: Swedish Club, 2019)

In the following figure, the most common causes of navigational claims are presented, as well as the costliest causes.

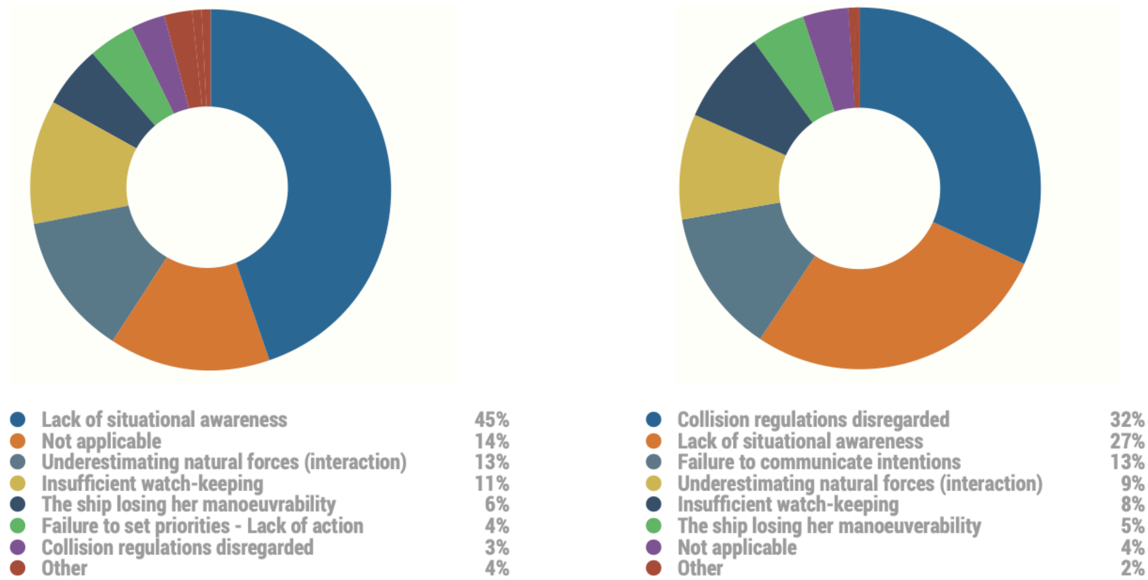


Figure 1.7 – Causality of navigational claims (Source: Swedish Club, 2019)

Other researchers studied causes of marine incidents. In his comprehensive study, Hwang [2002] determined that there is an issue with qualitative parts of Collision Regulations (COLREGs) when navigators try to apply them in real-world situations. Acar et al. [2008] with their study of marine incidents confirmed above statistics and reported that 85 % of all accidents were caused by some interpolation of human error. Macrae [2009] is another author that determined that situational awareness and failure to comprehend and apply COLREGs are main causes of collisions. As situational awareness and misunderstanding of collision regulations are interconnected, a group of researchers at Faculty of Maritime Studies [Mohovic et al., 2016] conducted a study to determine which collision regulation Rules are hardest to comprehend and which are most likely to be violated at sea. The study was conducted between January and March of 2014 with 1538 participants from 68 different countries. The study surveyed professional seafarers, instructors, and students. The results showed that there is a general lack of understanding and difficulty in applying COLREGs among surveyed practitioners. In Figure 1.8 percentages of Rules which are most difficult for students to understand is presented.

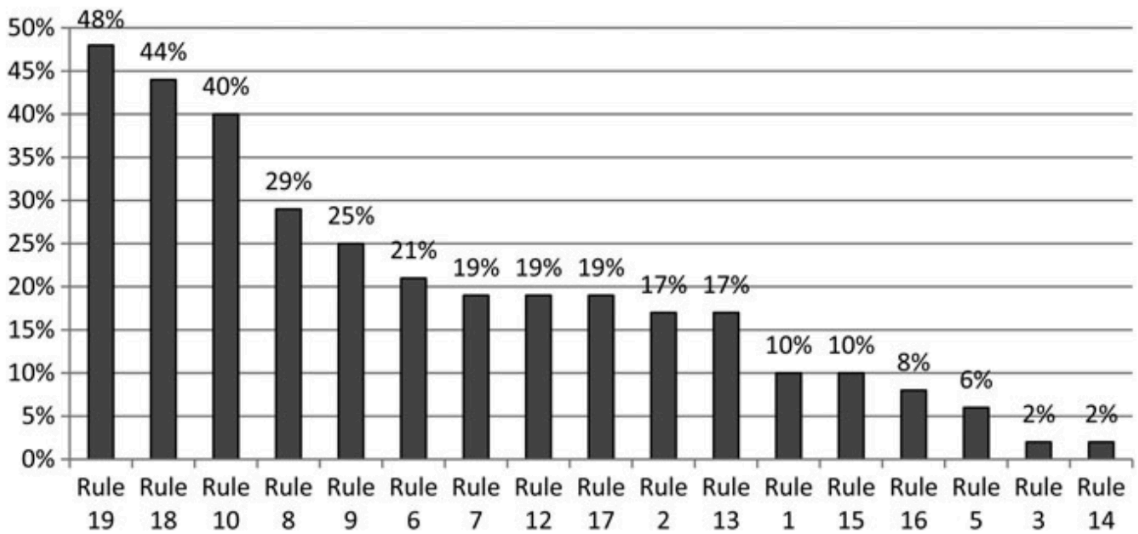


Figure 1.8 – Most difficult Rules for students to understand (Source: Mohovic et al., 2016.)

Zekic et al. [2015] conducted a similar study researching the level of knowledge and understanding of COLREGs. One of the interesting aspects of this study is the survey of participant's personal feeling about distance between vessels and when the avoiding should be initiated. Within narrow channels and straits sea surface vehicles are sailing in proximity, so it is interesting to see in Figure 1.9 (a) and (b) what would be an acceptable proximity with other vehicles, and when maneuvering action is required. Quantified approach to determining acceptable levels of proximity is delivered later in this thesis.

Loss of situational awareness and violation of collision regulations is identified as the most contributory factors of the navigational claims. Loss of situational awareness is a broad category, and it includes navigating officer's inability to recognize risk and comprehend information. From the investigative point of view, it is crucial to find the reason why someone made a certain decision in a particular navigational situation. If the answer is negligence, complacency, or lack of knowledge, then it is easier to pinpoint the corrective actions. However, investigation results and statistics point out that there are deeper chains of failures that lead to faulty decision making, so it is important to discover those root causes in order to prevent reoccurrences. Psarafits et al. [2000] have shown that technology advances have contributed to reduce the number of marine incidents, as technology allows for more informed and easier decision-making. In this study goal is to design a decision-support system that recognizes and evaluate collision, allision and grounding risks, determines appropriate collision regulations, safe speed, and offer optimal trajectories for the given situation. As it would be adaptable in the time domain, the proposed system is dynamic.

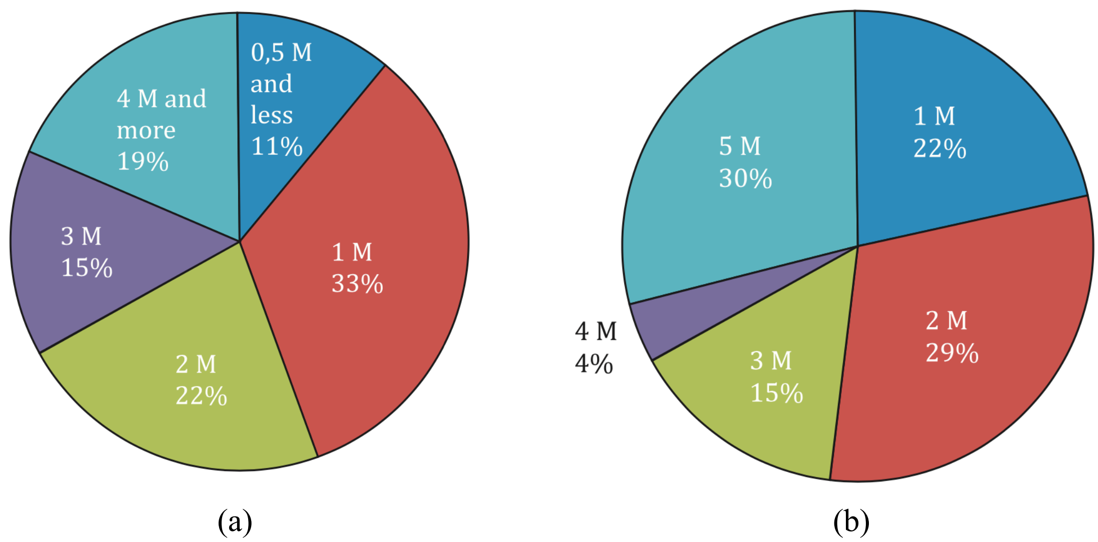


Figure 1.9 – Participants' survey on safe passing distances (a), and appropriate distance to initiate collision avoidance (b). (Source: Zekic et al., 2015.)

Focus of this research remains on decision-making support, rather than on autonomous vehicles. The main reason for this is that for many reasons, which are presented and discussed in following chapters, commercial sea surface industry is not ready for safe autonomous operation yet. The biggest barrier for autonomous navigation is regulation. Numerous articles have been published calling for revision of COLREGs, but administrations are very slow on reacting. How to assign responsibility in autonomous and human-operated interactions resulting in incidents? Another barrier is abysmal quality of sensory equipment installed onboard sea surface vehicles. Most of the alarms that human operators get on the vessels is due to sensor faults and inability of sensory equipment to detect if the signal is received for a real situation or there is a slight crossing of a threshold. Therefore, even though it is technologically possible to introduce autonomous commercial sea surface vehicles, their cost would be higher than keeping crew onboard, so until the autonomous vehicles become more economical and efficient, we will not see larger number of autonomous vehicles at sea. On the other hand, utilizing technology to aid human operators in order to reduce significant incidents is fairly inexpensive and can be done with present equipment onboard with the aid of signal fusing and error reduction.

As it will be visible from the following chapters, human element should not be eliminated from the decision supporting system, but rather utilized by extracting positive aspects of the human experience. Human operators have flexibility and creativity, so intention is to utilize human experiences as inputs that would aid algorithms to find optimal solutions and deliver guidance adjusted for a human operator. The human operator's ability to adapt to exceptional situations has to be incorporated in decision support systems in order to be viable in complex situations.

1.5 Research methodology

In order to effectively examine influence of dynamic environments on collision avoidance and path selection, various research methods are utilized. Classification method was used to split the research subjects into subcategories and then analyzed as distinct research components. Multimodality of the research required use of methods such are: methods of analysis and synthesis, methods of deduction and induction, description, classification and comparative methods, discussion, modeling methods, mathematical and statistical methods, method of compilation, generalization and specialization techniques, as well as methods of simulation and computation.

All of the before mentioned research methods aided in determination of all influential functions that were used to resolve complexities of the dynamic collision avoidance research.

1.6 Contributions

Foundation of the scientific research is systematic approach to data collection, parameter design, model development and numerical testing in order to achieve viable results. A novel hazard alerting model is proposed based on probabilistic HMM framework and offers viable resolutions for mixed-equipage environments. With such developed models, aim is to contribute towards safer navigation through faster conflict resolution, optimal path selection, informed decision-making, and reduced uncertainty for maritime transportation stakeholders.

Overall contributions of the scientific research could be outlined as follows:

- Optimizing model for hazard awareness and conflict resolution that will improve safety of navigation of sea surface vehicles.
- Development of methodology that will ensure effective peer-to-peer intent communication in mixed-equipage environments.
- Determination and analytical processing of relevant parameters that affect optimal and safe path selection taking COLREGs in consideration.
- Partial observability reduction of latent states by introducing an early collision detection method.
- Systematic review of all previous and current methods relevant to the dynamic collision avoidance of sea surface vehicles.
- Formation of lean and effective algorithms that could be programmed to various existing solutions in order to achieve holistic solution for dynamic collision avoidance system.

Results of the scientific research are compiled and presented in the concluding part of the dissertation, which confirms fundamental hypothesis without rejecting it. Considering the ubiquitous trend of applying solutions based on artificial intelligence or machine learning modelling at various types of transport technology, the main application of research results of the proposed dissertation would be at software developing entities to aid improvement of decision support technology already existing or being developed on the market. Research solutions could also be used to advance knowledge about support decision making solutions, to aid further research, to be used for developing a dynamic collision avoidance system, or to broaden the critical thinking about utilizing technology to advance maritime safety.

1.7 Organization of the thesis

The First chapter is introductory and delivers problem statement, describes research problems and challenges, as well as defines fundamental and supplementary hypotheses that are confirmed in the thesis. Related work and preliminary research are presented and discussed. An overview of the human element and its role in modern shipboard operations is given. Research methodology is selected, while original contributions described.

Chapter 2 delivers overview of Hidden Markov Model and probabilistic modeling in general, as well as the reasoning why Hidden Markov Model has been selected for modeling the hazard avoidance system proposed in the thesis. As a special type of HMMs, Markov Decision Processes, Partially Observable Markov Decision processes, and Reinforcement learning, specifically Q-state learning, have been described and defined.

Even though collision avoidance and hazard alerting algorithms are based on a model-free reinforcement learning method, Chapter 3 delivers important on-model solutions for the Maritime Hazard Alerting and Collision Avoidance System (MCAS). Considering the cost of commercial sea surface vehicles, it would not be optimal to commence voyages with model-free approach and without previous knowledge. Therefore, on-model motion control method is presented that allows for initial knowledge base, that will be updated during exploitation with model-free solutions. The chapter begins with the proposed Foraging Particle Filter that is used to filter sensory information before it is used in the collision avoidance model. This chapter also covers the benefits of fusing sensing information to increase observability of the state space. The chapter concludes with a detailed model-free motion control system that will allow for the pilotage of sea surface vehicles with appropriate responses to environmental and weather conditions, as well as allow for motion prediction, which is a crucial part of the collision avoidance algorithm.

Chapter 4 is the central work of this thesis. It delivers dynamic collision avoidance algorithm. This chapter also covers target behavior uncertainties, as it is very hard to predict if the target vehicle will follow the COLREGs or not. COLREGs classification algorithm is also a part of this chapter, and it is used to quantify relevant Rules that would help in collision avoidance automation and autonomy.

Chapter 5 depicts challenges of mixed equipage situations and what are the options to share intent with other vehicles. Sharing intent is one of the best and easiest methods to reduce target behavior uncertainty. Mixed equipage situations are difficult because some of the targets are cooperative, while others are uncooperative. Uncooperative targets bear higher risk and different approach is required when making decisions about collision avoidance.

Chapter 6 covers challenges of hazard alerting. At the present moment, commercial vehicles have user-dependent alarm thresholds and in this chapter, algorithmic solution to hazard alerting that would allow for reduction of nuisance alerts and higher dependability is developed. Navigators are often agitated by the number of alerts received in critical situations. Therefore, approach is to allow for intelligent alerting practices.

In Chapter 7 main conclusions and contributions are summarized.

Chapter 2

Hidden Markov Models and decision processes

Within this chapter rational decision making is explored. In the real world it is rare for agents to have full information about environment or other interactive agents. Often, agents don't even have full self-awareness, as some of the parameters could be hidden or partially observable. It is assumed that all professionals have intention of acting rationally, but barriers against irrationality are proposed as well. Uncertainties are essential part of modeling the world, so the approach in this thesis is to reduce uncertainties at the input stage of the signal processing in order to reduce computational burden of collision avoidance processes.

2.1 Decision making and probability

Decision making under uncertainty, such is collision avoidance of sea surface vehicles, requires a complex task of accounting for all sources of uncertainty and find optimal solutions given these uncertainties. Dynamic processes are challenging; and sometimes optimal solutions are not available, so it is necessary to search for sufficiency.

The term "agent" in this thesis is used to describe someone or something that is acting based on an observation taken from a relevant environment. Own agent is a nonphysical entity that chooses optimal or sufficient actions based on various sensory and regulatory inputs it observes. As it is shown in Figure 2.1, agent interacts with its environment in a sequence of discrete time steps t. At each time step t, the agent receives a representation of environment's state $s_t \in S$, where S represents a set of feasible states. Once a state from the environment is received, the agent selects an action $a_t \in A(s_t)$ based on sensory information $o_t \in O$, where $A(s_t)$ represents set of actions available in a state s_t, while O represents a set of all sensory information that an agent can observe. In the case of anti-collision, observations are all sensory information that an agent receives from various sensors installed on a sea surface vehicle, while action can be alerting or suggested control action to maneuver vehicle outside of a danger zone. It is important to note that observations are usually noisy and erroneous, so it is necessary to address these uncertainties when developing a decision support models, as otherwise agents could deliver faulty action suggestions. The focus of this thesis is to ensure agent is finding model-free optimal solutions. Objectives and reward shaping is utilized to achieve optimal behavior.

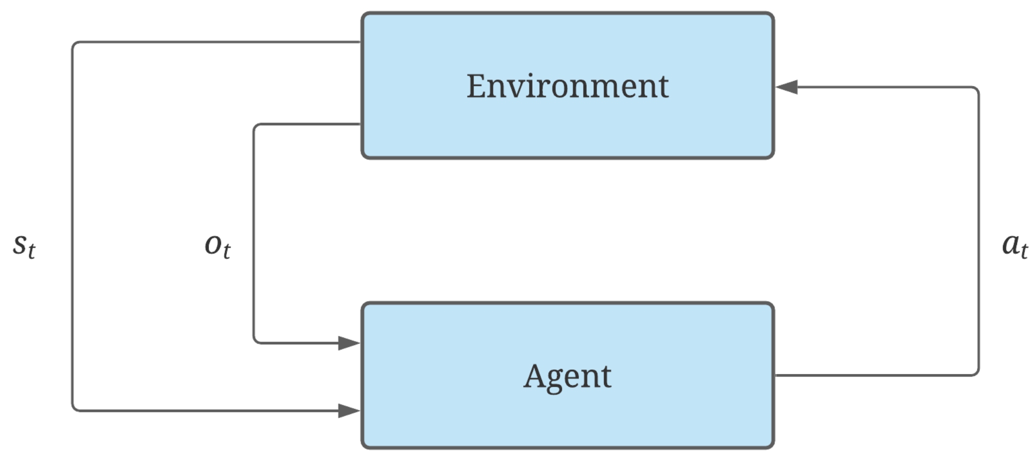


Figure 2.1 – Decision making – an agent interacting with environment

If a problem that an agent is trying to solve is simple enough, it is possible to utilize direct programming to instruct an agent how to resolve a problem. This is possible when an agent operates in deterministic world. For example, instructing an agent to cross a street when it detects a green light only within the zebra pattern, instructions are simple. However, if uncertainties of traffic light failure, or error of drivers of cars approaching the crossroad are incorporated, then the deterministic world is getting more complex, and then it is necessary to utilize different programming techniques to find optimal solutions. If we are experts of a certain process and have extensive overview of all possibilities an agent can face, we can utilize supervised learning to teach an agent how to behave optimally. This requires laying out all possibilities that an agent can face and allow it to learn before letting it operate in a world. Even though this kind of approach can be very practical, when there is a high-dimensional state space, writing down a pathway for all possible situations an agent can meet would be computationally very expensive and timely. If allow an agent to cross any crosswalk in the World by teaching it to cross only few crosswalks in one country, it would not be successful as there are many factors influencing crossing a street in different countries. Similar to supervised learning is strategic optimization, where instead of laying out all possible state space combinations, strategies that an agent can consider for a problem it faces, based on observations it receives, are developed. It is still fitting only to the lower-dimensional problems, as higher-dimensional problems would require a lot of computational power to find global maximum. Finding global maximum is not trivial, so process knowledge or problem dynamics knowledge should be utilized to guide search for optimal solutions. An agent is permitted to learn decision-making strategy while interacting with the environment and other agents.

If situation is considered in which a navigator on a commercial sea surface vehicle notices a RADAR search and rescue transponder (SART) signal on his/her RADAR screen, there are two possible reasons for the signal. Either there is a real emergency that requires an immediate action, or one of the navigators of neighboring vehicles is testing their SART equipment and their signal on own RADAR is received. These two possibilities are not considered equally by navigators. An experienced navigator will develop a belief that it is a testing signal, rather than a real incident, as his previous experience sailing the seas has developed an intuitive probability distribution between these two options. If T would represent an option of a test signal and R would represent a real incident situation, then $T \succ R$, which means that a navigator believes that the possibility of a real incident is lower than the possibility of a test signal that was mistakenly released to other vehicles. If the belief that T and R are equally plausible exists, then it is possible to write $T \sim R$.

Navigator's assumptions could be correct, but because of the consequences of not aiding a vehicle in a real incident situation, navigators should always act as it is a real incident signal. There is a way of rebalancing of the belief possibilities by utilizing additional information. If a navigator would at the same time receive another mayday signal from a radio station, AIS or satellite receiver, the belief would immediately change, and navigator would strongly believe that it is a real incident. By providing additional information it is possible to reduce uncertainties. If I would represent additional distress information, then it is possible to state that given the additional information navigator now believes that the real incident is more probable than the testing signal broadcasted to the environment $(T \mid I) \prec (R \mid I)$.

It is necessary to make some assumptions about the relationships introduced above by the operators $\prec$ and $\sim$ [Kochenderfer, 2015]. *Universal comparability* assumption requires one of the following to hold: $(T \mid I) \succ (R \mid I)$, $(T \mid I) \sim (R \mid I)$, $(T \mid I) \prec (R \mid I)$. The assumption of *transitivity* entails that if $(T \mid I) \succcurlyeq (R \mid I)$ and $(R \mid I) \succcurlyeq (F \mid I)$, where F can stand for a third option that can be defined as failure signal, then $(T \mid I) \succcurlyeq (F \mid I)$. Transitivity and universal comparability relations [Howard, 1960] allow us to represent degrees of belief by a real-valued function, so it is possible to utilize a function P with the following properties:

$$P(T \mid I) < P(R \mid I) \text{ if and only if } (T \mid I) \prec (R \mid I)$$
$$P(T \mid I) = P(R \mid I) \text{ if and only if } (T \mid I) \sim (R \mid I).$$

Furthermore, a set of additional assumptions about the real-valued function P could be developed and used to show that P should satisfy basic axioms of probability (Kochenderfer, 2015). Therefore, $0 \leq P(T \mid I) \leq 1$. In case we are certain about the $(T \mid I)$, then we can state that $P(T \mid I) = 1$, but when we believe that $(T \mid I)$ is impossible, then $P(T \mid I) = 0$. The uncertainty in the truth of the $(T \mid I)$ is then represented by values between the two extrema [Kochenderfer, 2015]. If axioms of probability are further utilized, then it is possible to represent own beliefs as conditional probability:

$$P(T \mid R) = \frac{P(T, R)}{P(R)}, \tag{2.1}$$

where $P(T, R)$ stands for the probability that both T and R are true. Conditional probability is very important for this approach; however, if the law of total probability [Howard, 1960] is considered, which requires that Z is a set of mutually exclusive and exhaustive belief possibilities, then

$$P(T \mid I) = \sum_{\zeta \in \mathrm{Z}} P(T \mid R, I) P(R \mid I). \tag{2.2}$$

Taking into consideration (2.2), we are now able to lay out the Bayes' rule [Howard, 1960], which plays an integral role in modelling decision making under uncertainty:

$$P(T \mid R) = \frac{P(R \mid T) P(T)}{P(R)}. \tag{2.3}$$

2.2 Markov chains and Markov property

If a passage of a commercial vehicle from some port in Europe to some port in the far East is considered, operators are obliged by various regulations to plan this passage thoroughly and include as much as possible information in order to avoid hazards. Even if operators performed a perfect passage planning, several factors, such are weather and the behavior of other vehicles operators meet on their passage, are random, which means that when operators try to solve collision avoidance problems, they are dealing with stochastic processes. Vehicle motion control and navigation consist of various random processes, such are steering inputs, engine control inputs, or collision avoidance decision making. Even when a vehicle is navigating same waters consistently, environmental loads are going to be different with every new passage; therefore, it will be necessary to apply different steering inputs to accomplish voyages. Similarly, even though this vehicle would do the same voyage consistently, targets met would regularly be different and their behavior would vary depending on decision making of navigating officers onboard those vehicles. Considering this example, it is possible to note that previous experience of sailing the same route would not help us in reducing the randomness of the collision avoidance process. Motion control of vehicles in collision avoidance situation would be independent of the previous experiences, as many external factors would influence both steering inputs and decision making. Stochastic process defined this way would have Markov property, as the conditional probability distribution of future decisions and states depends only on the present state. In this research it is not assumed that the collision avoidance does not depend on the past completely (especially when thinking of experience of navigators), but rather that there is a conditional independence on the previous states and dependence on the present state. Therefore, if S_t would be considered as a state of proposed collision avoidance system, then Markov property would be defined as:

$$P(S_{t+1} = j|S_t = i) \tag{2.4}$$

Therefore, Markov Property would indicate that past states and future states are conditionally independent given the present state.

A system's state space can be considered as a set of all possible configurations of a system [Puterman, 1994] (for example GPS position, heading and Speed Over Ground of a vehicle) which can either be discrete or continuous. It is important to emphasize that the state space does not have to contain only scalars, but can also consist of vectors, which will be further discussed later in this thesis. Considering collision avoidance problem of this research, Markov property is utilized to predict future states given the present state and find out what is an appropriate action in the present state given the predicted future states. This requires temporal modeling, and one of the simple temporal representations is a Markov chain [Howard, R., 1960]. The conditional distribution mentioned in expression (2.4) describes transition probability for state space. As will be seen in the following sections, initial distribution $P(S_0)$ can be defined by a convention (for example, initial state space has a heading of 360º, and steering wheel is at midships), or it is defined by system dynamics (for example without reading a speed sensor, sequence is initiated with a speed of 0 kt), or some other type of distribution such is multivariate Gaussian distribution.

Markov chains could be utilized to develop Markov models that are used to model systems of our interest. Depending on the observability of the system's present and sequential states, as well as the adjustability of the system after observations are received, 4 main categories of Markov models are recognized:

Table 2.1 – Markov state observability

	Fully observable state	Partially observable state
Independent	Markov chain	Hidden Markov model
Adjustable after observation	Markov decision process	Partially observable Markov decision process

Source: Puterman, 1994

2.3 Hidden Markov Model

Markov chain on its own would not provide a clear assistance in resolving challenges of sequential decision making under uncertainty, so it is necessary to broaden the Markov chain to a Markov model by introducing observations. Observations are information, usually from some sensing equipment, that will aid in understanding the underlying stochastic process of the system. Observations are necessary when system states are not observable, so sensing information is utilized to learn about the system being modeled. In case of discrete state variables, a Hidden Markov model (HMM) is defined, while in a case of continuous state variables, we consider dynamical system. As seen from the upcoming chapters, discretization of the state space for marine collision avoidance systems is rather necessary to reduce approximations and computing loads. Unlike the aeronautical sector, marine sector enjoys relative abundance of time and can allow for slower sampling rates. This is the reason it is possible to utilize HMMs without loss of generalization and downgrade of the number of variables in the state space while maintaining the lean approach to computing.

Markov models are powerful as they allow us to uncover the underlying distribution of hidden variables while observing visible variables. In the field of temporal models, four common inference tasks are available [Howard, 1960]:

- Filtering: $P(S_t|Y_{0:t})$
- Prediction: $P(S_{t'}|Y_{0:t})$, where $t' > t$

- Smoothing: $P(S_{t'}|Y_{0:t})$, where $t' < t$
- Most likely explanation: $\mathrm{argmax}_{X_{0:t}}\, P(S_{0:t}|Y_{0:t})$

One example of the Hidden Markov model inference is depicted in Figure 2.2. In this example a head-on situation is depicted where ownship is meeting a target vehicle. Target vehicle's state space variables are not fully observable by ownship, so it is necessary to utilize sensing equipment to observe the behavior of the target vehicle and ownship tries to determine if the target vehicle will follow Collision Regulations (COLREGs) or not. Other Rules are disregarded, so focus is maintained on the head-on situation and the Rule 14. Therefore, described HMM looks as follows:

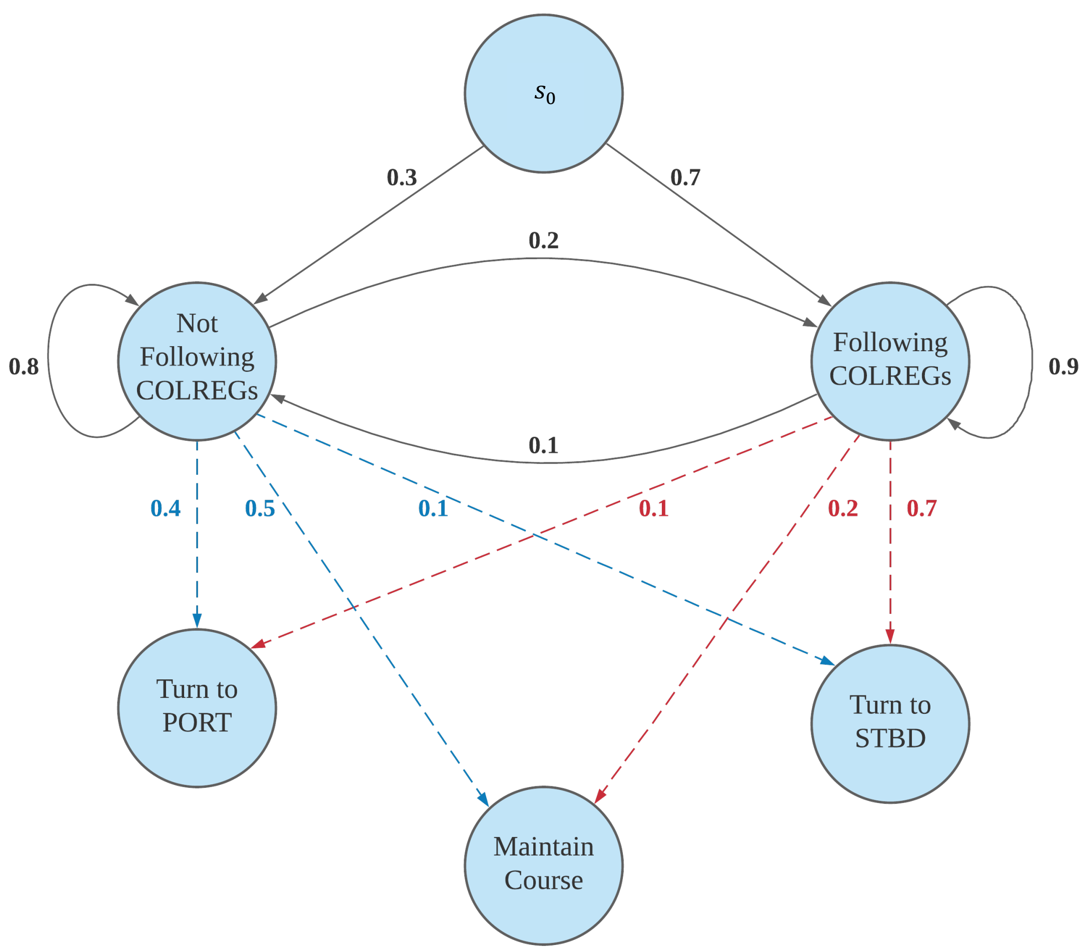


Figure 2.2 – Hidden Markov Model of COLREGs Rule 14

Two sea surface vehicles in a head-on situation are considered. To avoid collision, common practice as per the Rule 14 is that both vehicles turn their course to starboard. In this overly simplified example, we are navigating one vehicle and other vehicle is navigated by an unknown navigating officer, so the common uncertainty in these situations is action of the target vehicle. Even though Rule 14 mandates action, which should be early and substantial, there is a probability that the target vehicle will not act as per the Rule 14. We do not have definite information if the target vehicle will follow or not follow COLREGs. This means that we are dealing with an underlying stochastic process that is hidden and not directly observable, so another stochastic process is required that will produce sequence of observations. We do have a general knowledge and intuition about target behavior that would define our behavior in this situation. Ownship navigator can either be risk averse, or accept more risk depending on navigator's own tolerance and level of complacency. For the sake of simplicity, actions of own vehicle are disregarded, while actions of the target vehicle are solely considered. Target vehicle can either turn to starboard, turn to port, or maintain its own course. Based on the behavior selected by the target vehicle, speculation is made about target vehicle's COLREGs compliance, but as we do not see the state of an approaching vehicle, this behavior selection is hidden from us. Therefore, ownship assumes that the selection of following the COLREGs or not is a representation of a discrete Markov chain.

The initial distribution is based on a belief that in 70 % of cases navigators will act as per COLREGs and in 30 % they will not follow the required Rules. This distribution is different for each individual, but there is also a different way of selecting the initial distribution. It can be an equilibrium distribution but can also be an industry standard or a result of an extensive research. The transition distribution describes an actual selection of the navigator of a target vehicle with an 80 % chance that the vehicle will not follow COLREGs if it already is not following COLREGs and 20 % chance that the navigator will change their mind and follow COLREGs. On the other hand, if a navigator is following COLREGs, there is 90 % chance that it will continue to follow the COLREGs and only 10 % chance that the navigator will change mind and decide not to follow COLREGs anymore. Finally, the outcome distribution represents how likely would a navigator of a target vessel select a certain motion control, so ownship navigator can see that if the navigator of a target vehicle is not following COLREGs, there is a 40 % chance that the navigator would turn to port, 50 % chance of maintaining the same course, and only 10 % chance that the navigator would turn to starboard. In case that the navigator decides to follow the COLREGs, then there is 70 % chance that the navigator would turn to starboard, 20 % chance of maintaining the same course and 10 % chance of turning to port.

With H representing an unobservable state space, another stochastic process whose behavior is dependent on H is required. This is the reason why O represents an observation space and O is utilized to learn about H. Therefore, it is possible to state:

$P(o_t|h_t)$ – observation probability

$P(h_{t+1}|h_t)$ – transition probability,

where $o \in O$, and $h \in H$.

Hence, when observation and transition probabilities are combined together, it is possible to obtain joint probability distribution over all states and observations available within the model:

$$P(H,O) = p\left(h_{t_0}\right) \prod_{t=1}^{T-1} p(h_{t+1}|h_t) \prod_{t=1}^{T} p(o_t|h_t) \tag{2.5}$$

As the state space is hidden, it is necessary to infer the latent state probabilities. Inference is necessary, as the goal is to determine the distribution over unobserved variables taking into consideration values related to a set of observed variables [Kochenderfer, 2015].

Even though finding $P(H,O)$ is of interest, inference of $P(h_t,O)$ at a current time t given all observations available is of greater interest; both joint probability with historic observations up to the present time stamp $p(h_t, o_0, o_1, \dots, o_{t-1}, o_t)$, and conditional probability with predicted future observations given the present state $p(o_{t+1}, \dots, o_T|h_t)$. After combining joint and conditional probabilities together, the following stands:

$$p(h_t, o_0, o_1, \dots, o_{t-1}, o_t)p(o_{t+1}, \dots, o_T|h_t) = P(h_t, O) \propto P(h_t|O), \tag{2.6}$$

Where the symbol "∝" is used to denote that left-hand side is "proportional to" the right-hand side.

When presented with a model with latent state space values, main goal is to find the optimal state transition sets and outcome probabilities. The common approach is to exploit various learning techniques that would derive the maximum likelihood estimates of the HMM parameters, such are forward-backward, or Baum-Welch algorithms [Puterman, 1994]. If model requires time series prediction, then Markov chain Monte Carlo (MCMC) algorithm would be fitting better, however it is necessary to keep in mind that MCMC algorithm is computationally expensive [Howard, 1960]; therefore, a certain degree of approximation could be required. HMMs are used to this day for speech recognition, analysis of DNA sequences, as well as in the field of bioinformatics [Puterman, 1994].

Even though depicted example is simplistic, it highlights the benefit of using HMMs for solving sequential parameter problems. However, HMMs are not designed to allow for control after observations are made. As some of models proposed in this thesis require control and optimal actions, it is necessary to explore Markov Decision Processes (MDP) and Partially Observable Markov Decision Processes (POMDP).

2.4 Markov Decision Processes

In order to consider a process Markovian, it has to be independent of the previous state. In other words, Markov Decision Process is stochastic process where the state of the system changes probabilistically based on the current state and according to the defined action. For example, vessel can pass certain route many times, but it doesn't guarantee that the next time passing the same route will be collision free. Collision potential depends only on the current situation and availability of intruders at the observed moment.

Markov Decision Processes (MDPs), also known as stochastic dynamic programming, are commonly used to model decision-making where outcomes are partially under the control of the decision maker, but also random due to the stochastic nature of the process. Howard [1960] was one of the first authors to research optimization problems in his book *Dynamic Programming and Markov Processes* and developed mathematical framework for modeling decision-making. MDPs have been applied to many fields, such as robotics, economics, signal processing, artificial intelligence, communications, automated control, stochastic scheduling, and automated planning.

Markov Decision Process is fully observable stochastic control process that can move from one eminent state to any other possible state. Similarly as with Markov chains, MDPs allow decision maker to take an action available from the pool of pre-defined actions for that state. MDPs are considered continuance to Markov chains with the main modification being motivation and choice through rewards and allowable actions. MDPs are defined as sextuple:

$$(S, A, P(s_{t+1}|s_t, a_t), R_a, \gamma, \mathrm{H}), \tag{2.7}$$

The system can be described with state space S, observing at discrete time periods (t = 0, 1, 2, ...). When the system is observed to be at state $s_t \in S$, an action a_t from the action set A will be chosen. Then the system will receive a real-valued reward R_a, and eventually transfer to state s_{t+1} at the next period with state transition function $P(s_{t+1}|s_t, a_t)$, defined as probability that the system will evolve to state s_{t+1}. In other words, at a certain moment t, the process is in some state s_t, and the decision maker has an option of choosing available action a_t from the set of possible decisions A. Transition probability $P(s_{t+1}|s_t, a_t)$ is then affected by the action a_t, and the process responds by moving to a new state s_{t+1} with the corresponding reward, be it gain or loss, R_a. Since subsequent reward is associated with the transition probability, which is affected by the chosen action, discount factor γ is introduced to account for immediate and subsequent reward as an aid in finding an optimal policy. The discount factor γ, $(0 < \gamma \leq 1)$, is used to determine if agent should consider only immediate rewards (discount factor that is closer to 0), or agent should prefer distant accumulated reward (discount factor that is closer to 1). H represents horizon to limit and discretize time space.

Within the MDP framework, state space is fully observable and can be computed. In the real sector it is difficult to achieve full observability, but there are some remedies to reduce uncertainties. The goal of the MDP stochastic control process is to find a policy π that will specify an optimal action a_t that a decision maker should take when in a state s_t. A policy π with optimal actions allows for MDP to behave like a Markov chain, as transition probabilities now resemble Markov transition matrices. Therefore, it is necessary to find an optimal policy π^* of actions that maximizes expected (E) sum of rewards for applicable states:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} E\left(\sum_{t=0}^{\mathrm{H}} R_{a_t}(s_t, a_t, s_{t+1})|\pi\right) \tag{2.8}$$

It is necessary to emphasize that unlike in deterministic systems where the solution is a planned sequence of all actions from the beginning until the end, MDP policy provides an overview of the optimal action in each planned state by maximizing expected sum of rewards. Therefore, the benefit of utilizing MDPs is modeling of noise and uncertainty.

In the example below, both deterministic and stochastic environments for an abstract agent (decision-maker) are depicted. On the left side of Figure 2.3, there is a machine operating a steering wheel, while on the right side of Figure 2.3 a human operates the steering wheel. Assumption is made that there is no noise influence on robotic helmsman and that it acts as a perfect actuator. Human helmsman, on the other hand, has inherent stochastic noise and can wrongly execute requested command according to a known probability distribution (distribution is assumed in the example, but otherwise has to be empirically determined). In the deterministic world, decision maker requests from the robotic helmsman to turn the wheel to starboard by 10°, and the robotic helmsman executes that command with 100 % probability. On the other hand, in the stochastic world when a decision maker requests human helmsman to turn 10° to starboard, there is 80 % chance that the helmsman will turn the wheel to starboard, but there is also 10 % chance that the wheel will remain midships and 10 % chance that the helmsman will turn the wheel to port.

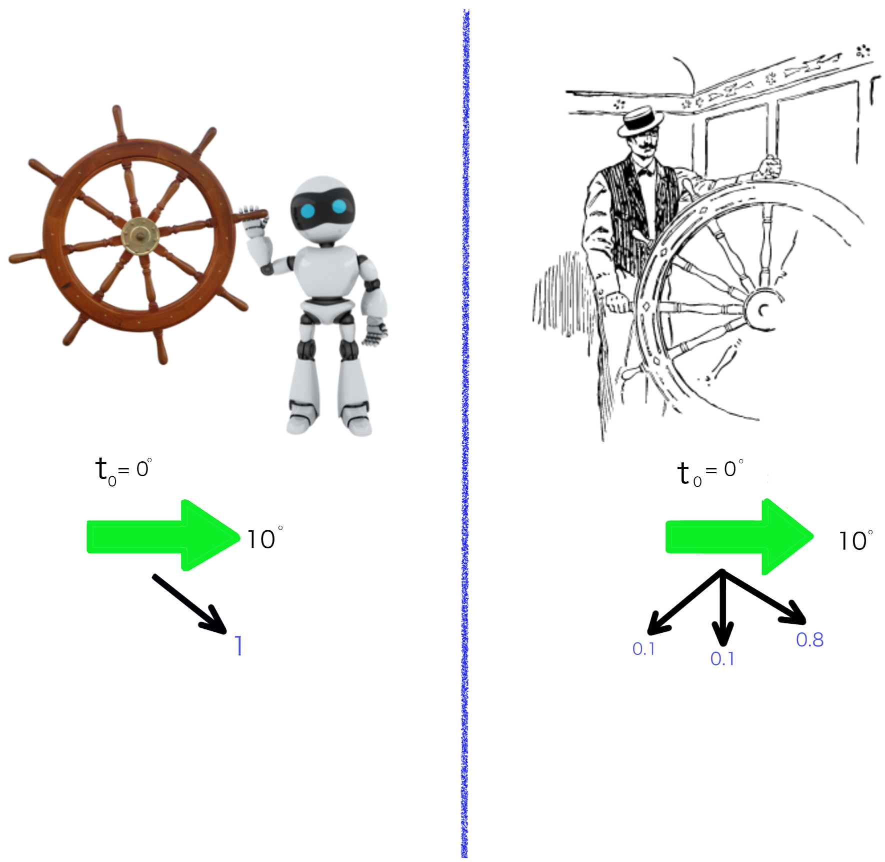


Figure 2.3 – An example of deterministic noiseless environment vs stochastic environment

Another example is a situation where a sea surface vehicle is passing a narrow channel or straits. There are three possible states: collision free, near-miss and collided. This time horizon is infinite unless a vehicle collides, after which the modeled system terminates. Commercially, every second saved is important, so higher transit speed of a vehicle is rewarded with double rewards, so the vehicle can either take a safer speed of 10 kt, or faster speed of 18 kt.

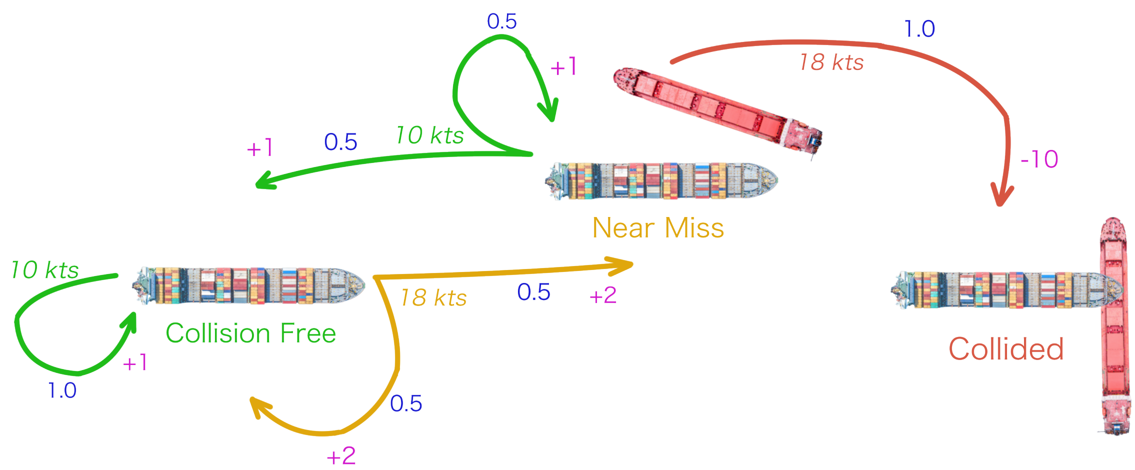


Figure 2.4 – Markov Decision Process – example

It is assumed that the container vehicle in the example is autonomous and making its own decisions based on the model depicted in Figure 2.4. Vehicle commences narrow channel transit in collision-free state. In this example, an infinite horizon is presented, so the transit can last forever, unless the vehicle reaches a terminal state of collided. If the vehicle proceeds with a safe speed of 10 kt, then there is 100 % chance that it stays collision free. If the vehicle proceeds with more rewarding faster speed of 18 kt, there is 50 % chance that it will stay collision free and 50 % chance that it will transit into a near-miss state. For this example, near miss is acceptable risk. After transition to the near-miss state, vehicle can either proceed with the safe speed of 10 kt, for which there is 50 % chance that it returns to collision-free state, or there is 50 % chance that it remains in the near-miss state. However, if the vehicle decides to proceed with the higher speed of 18 kt, then there is 100 % chance that it will collide.

Even though this example is significantly simplified, it shows us the mechanics of an MDP system. However, due to the infinite horizon, it is necessary to find a way to calculate sum of rewards. To showcase how this is done, a search tree of discussed MDP example system can be utilized.

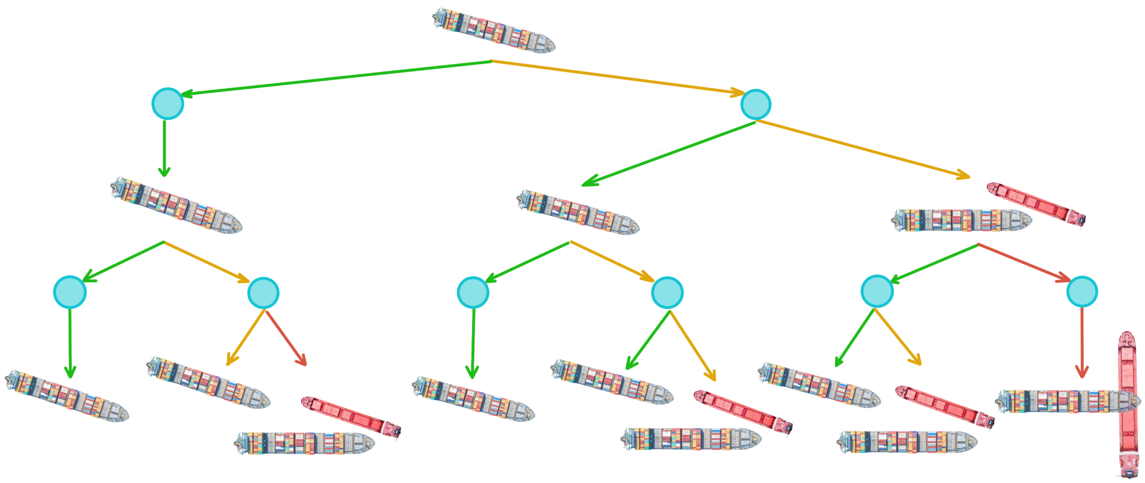


Figure 2.5 – Search tree for MDP example

The search tree discovers two important aspects of the MDP modeling. In the case of infinite horizon MDP model, search tree can be infinitely deep, and this can present significant computational challenges, especially when dealing with large state spaces. Noticeably, there is a lot of repetition and every time the sea surface vehicle is in collision-free state, the tree branches out in an identical way. Before transition from space to space happens, there is an interlink step that can be named a chance nod, or a q-state. Recalling the beginning of the chapter 2.2, transition matrix was identified as Q-matrix, so the chance nod or q-state is a transition state that contains information about rewards and actions available, before the decision about optimal action is made. A q-state can be considered as an abstract holding pattern where the agent ends up after choosing an action from an available action set before the stochastic transition to a new state is completed. As it will be seen in the later chapters of this thesis, q-states are powerful, as they can aid to learn about state space and optimal actions in an efficient way.

Another very important part of the MDP framework is summing the rewards. How to ensure convergence of optimal policy calculation when facing an infinite horizon? The answer is within the rewards discounting [Howard, 1960]. The discount factor γ is used to instruct the abstract agent to either appreciate immediate rewards or to prefer later accumulation of rewards. If the discount factor is closer to 0, then the agent will appreciate immediate rewards, but if the discount factor is closer to 1 then the agent will prefer more distant rewards. By setting the discount factor appropriately, convergence of optimal policy computation can be ensured. In the case of collision avoidance, immediate rewards are of interest, as the most immediate danger has preference than any other hazards along the planned passage. While future rewards are not omitted, immediate rewards are preferred.

Even though there is possibility of stochastic policies, optimal policies are of greater interest, as by discounting the rewards function, the actual utility of a future state will become small enough to be disregarded. Therefore, it is possible to utilize a horizon and define the maximum timesteps we look in the future or utilize discounting and find a point where utility of a future state becomes so small, that every further state is almost equal in value. It is necessary to find several optimal quantities [Puterman, 1994]: utility of a state, utility of a q-state, and an optimal policy. Convention of denoting optimal values with a superscript * is followed. The utility of a state s is denoted as $V^*(s)$ and it is an expected value of being in a state s and then acting optimally. The utility of a q-state is denoted as $Q^*(s,a)$ and it is an expected value of being in a q-state (s,a) after an agent takes an action a and then acts optimally. The optimal policy π^* lays out an optimal action from the state s. In case agent is able to learn optimal utility of a q-state, then (2.8) can take the following form:

$$\pi^* = \operatorname*{argmax}_a Q^*(s,a) \tag{2.9}$$

There are multiple ways that an agent can learn the optimal q-state utility, which will be presented in following sections; however, at this moment only the following formulation [Puterman, 1994] is considered:

$$Q^*(s,a) = \sum_{s'} P(s'|s,a)\,[R_a(s,a,s') + \gamma V^*(s')] \tag{2.10}$$

In order to simplify the formulation, s' is used to describe the state in the next timestep. The expression (2.10) still requires that the optimal utility of a state from the next timestep s' is known, so further exploration is needed in calculating the optimal utility of a state and optimal policy. However, if the optimal utility of q-state is known, the optimal policy can be directly derived by simply iterating to the furthest state s of the selected or discounted horizon.

If there is no practical way to learn optimal utility of q-states, we can find the optimal value with the simple substitution represented in the following expression, which is called the Bellman equation [Puterman, 1994]:

$$V^*(s) = \max_a \sum_{s'} P(s'|s,a)\,[R_a(s,a,s') + \gamma V^*(s')] \tag{2.11}$$

As it is possible to limit the depth of a search tree by fixing a horizon or discounting the rewards of the future states, the bottoms-up approach is taken, where a zero value to the deepest state in the tree is assigned and then work back to the top. This approach is known as value iteration [Howard, R., 1960] and the algorithm for the finite horizon or discounted horizon MDP system could be:

- $V_0(s) = 0$ – zero value is assigned to the state with no time-steps left (the finite state). This holds because the expected sum of rewards from this point on will be zero, so there is no point of going deeper in the tree.
- Then for each state expectimax is utilized as per the following:

$$V_{n+1}(s) \leftarrow \max_a \sum_{s'} P(s'|s,a)\,[R_a(s,a,s') + \gamma V_n(s')] \tag{2.12}$$

In this way it is possible to find convergence with a complexity of each iteration of $O(S^2A)$, which is computationally expensive and unfavorable, so attempts should be taken to find algorithmic solution to reduce complexity. The theorem and proofs of convergence are presented in Puterman [1994].

Depending on the MDP model, there is computationally more effective way to solve MDPs. If instead of searching for state values by iteration, policies are evaluated. The major difference is skipping search for the best action, while randomly choosing an action to go to the associated q-state, and then evaluating that policy from the randomly selected q-state. This process is known as policy evaluation [Howard, 1960].

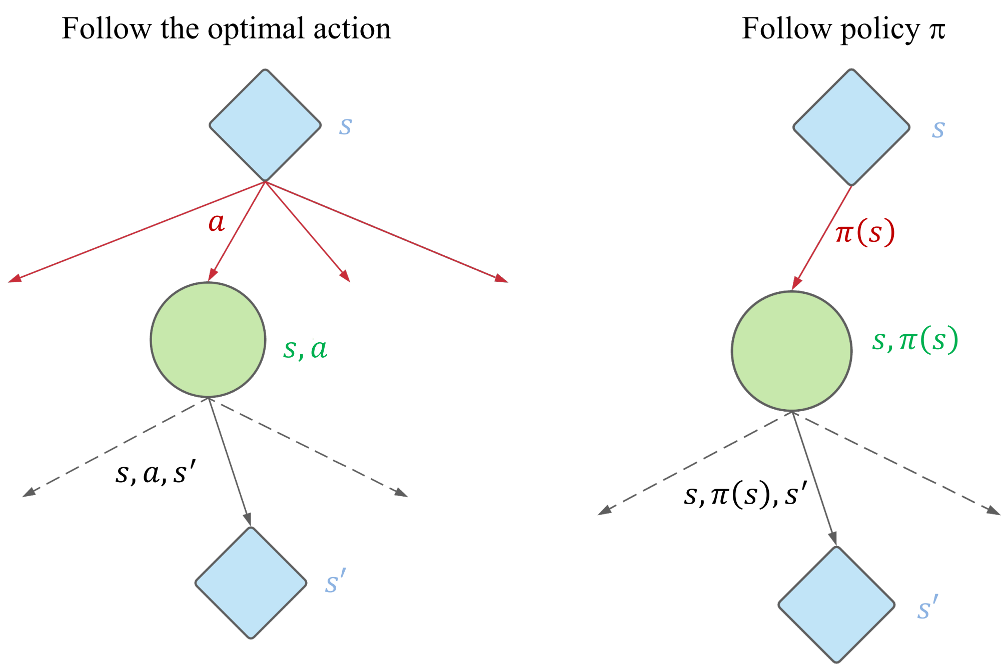


Figure 2.6 – Optimal action vs optimal policy

Policy evaluation starts by choosing a random fixed policy and then evaluating the tree. The fixed policy is a table that maps states and actions, and the agent simply follows the policy without a concern if the actions taken are optimal or not. It is difficult to recognize if this is the optimal policy, so it is necessary to compare it with other available policies. As an agent does not choose actions anymore, max operator from the state utility computation can be removed, and this is what makes policy evaluation computationally more efficient. Hence, if $V^{\pi}(s)$ is defined as an expected sum of all discounted rewards when following a fixed policy π and starting in the state s, then recursive Bellman equation for the policy evaluation is obtained [Puterman, 1994]:

$$V^{\pi}(s) = \sum_{s'} P\big(s'|s,\pi(s)\big)\,[R_a(s,\pi(s),s') + \gamma V^{\pi}(s')] \tag{2.13}$$

The policy evaluation is approached in a similar way as value iteration by initializing with $V_0^{\pi}(s) = 0$, selecting the deepest state of the tree, either by setting a horizon or discounting:

$$V_{n+1}^{\pi}(s) \leftarrow \sum_{s'} P\big(s'|s,\pi(s)\big)\,[R_a(s,\pi(s),s') + \gamma V_n^{\pi}(s')], \tag{2.14}$$

where n is the iteration number for the policy evaluation.

Efficiency for the policy evaluation is $O(S^2)$ for each iteration, as the policy is fixed, and there is no need to search for optimal action. Another convenient factor is that without searching for the optimal action, resulting system is linear and can be solved with any linear equation solvers by inversing matrices without the necessity of iterating (2.14).

Finally, there is also a policy iteration approach [Puterman, 1994], in which search for optimal values is done by firstly evaluating policies with the expression (2.14) and then iterated until values converge. Then policy improvement is done by extracting values from each evaluated policy following the one-step look-ahead Bellman equation again:

$$\pi_{k+1}(s) = \underset{a}{\operatorname{argmax}} \sum_{s'} P(s'|s,a)\,[R_a(s,a,s') + \gamma V^{\pi_k}(s')], \tag{2.15}$$

where k is iteration number for policy improvement.

With all iterative algorithms approach, starting point has to be selected. If there is previous knowledge available, it can aid convergence and shorten time of convergence significantly. Otherwise, the process starts with a random policy (for example all east), or all values zero, and then iterates until optimal policy is found.

If algorithms for policy iterations are designed carefully, optimal policy information could be obtained faster than other methods. For this, some heuristic approach would be required in order to find the best performing policy without going through full iteration steps. As it is visible from following chapters of this thesis, we utilize foraging heuristics to reduce convergence time.

2.5 Partially Observable Markov Decision Processes

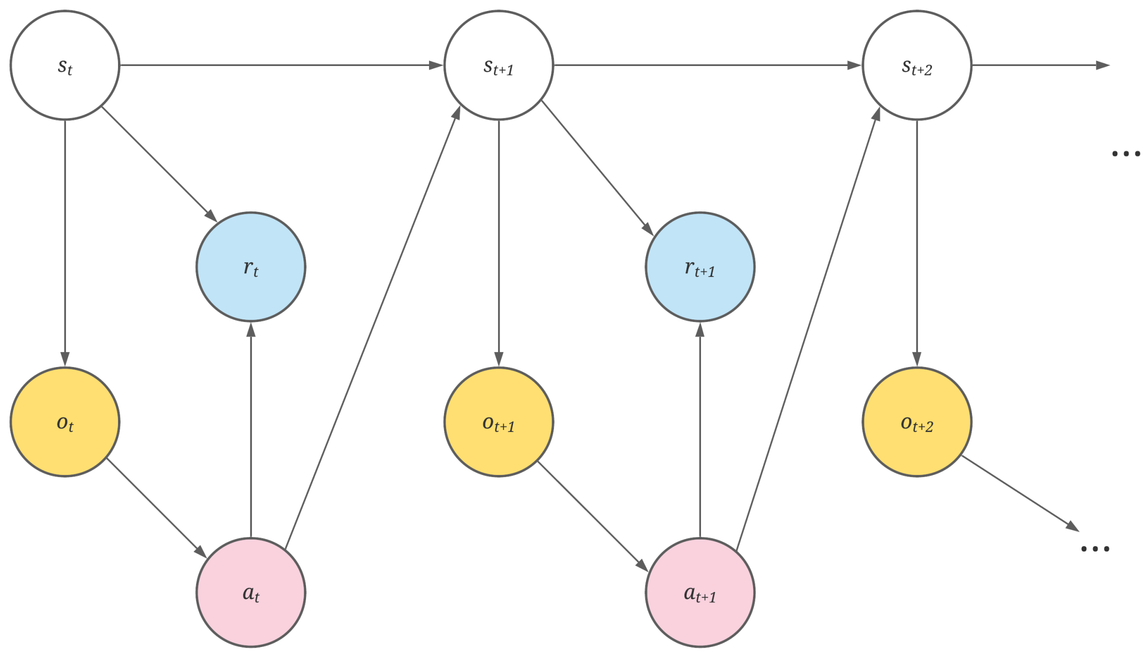


Figure 2.7 – Graphical model of Partially Observable Markov Decision Process

In the Partially Observable Markov Decision Process (POMDP) it is assumed that system dynamics are governed by an MDP, but it is not possible to observe the dynamics. MDPs are great generalization of Hidden Markov Models; however, it is hard to find real-world process that is characterized by a full state observability. Usually there is always some part of the state space that is not directly observable. This is the case with the collision avoidance problem as well. Therefore, when there is an inherent state uncertainty, sequential decision problems are modeled as POMDPs. As there are state uncertainties involved, POMDPs introduce observations, which are used to gain some insights about the state space.

POMDPs model state as a stochastic observation with the main goal of finding policies that take into account uncertainties about the current state, and future states in which the system will evolve, while maximizing predefined rewards. This is usually achieved by finding optimal actions for each belief state, which are defined as probability distributions over the state space, but as state space is unobservable, distribution based on observations taken are created. In that sense, there is a belief that the state space "looks" the same as initially believed. This allows us to solve similarly as MDPs.

POMPDs are designed to deal with incomplete information and noisy environments. There are many applications of POMDPs in the field of computer science and robotics, such as robust mobile robot navigation [Simmons and Koenig, 1995], robot control [Pineau and Thrun, 2002], machine vision [Bandera et al., 1996; Darrell and Pentland, 1996], autonomous helicopter control [Bagnell and Schneider, 2001; Ng et al., 2003], as well as in medical diagnosis [Hauskrecht, 1997] and machine maintenance [Puterman, 1994].

Similarly as with MDPs, Partially Observed Markov Decision Process can be defined as tuple:

$$(S, A, P(s_{t+1}|s_t, a_t), R_a, \Omega, O, \gamma, H) \tag{2.16}$$

In addition to the previously described parameters of this tuple POMDPs include set of possible observation the system can receive O, and the observation function Ω defined as the probability that the system, after taking an action a_t and evolving to a state s_{t+1}, will receive observation o that depends on the new state s_{t+1}, where $o \in O$.

The major difference in handling POMDPs when comparing with MDPs is that beliefs have to be updated. The belief state mimics the underlying state, so it is inherently Markovian, therefore, only knowledge of the previous belief state is needed, the action that is taken and the current observation. The main issue with the POMDP is that the belief state is defined on a continuous state space, even when the underlying state space is finite. This is because probability distributions over states are infinite [Howard, 1960].

When the transition from the state s to state s' occurs, the agent observes $o \in O$, where conditional probability is defined by $\Omega(o|s', a)$. If b is probability distribution over the state space S, then $b(s)$ represents probability that the environment is in the state s, therefore [Howard, 1960]:

$$b'(s') = \eta\Omega(o|s', a) \sum_{s \in S} P(s'|s, a) b(s) \tag{2.17}$$

However, expression (2.17) has to be normalized with normalizing factor η as per the following:

$$\eta = \frac{1}{P(o|b, a)} , \tag{2.18}$$

where $P(o|b, a) = \sum_{s' \in S} \Omega(o|s', a) \sum_{s \in S} P(s'|s, a) b(s)$.

Similarly as with MDPs, the goal is for an agent at each time stamp to choose actions that would maximize expected future discounted reward. Generally, expected return for sequence of states s_t and actions a_t could be formulated as follows:

$$R_a = \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \tag{2.19}$$

where $\gamma \in [0,1)$ is a discount factor and r represents the reward function.

It is now necessary to find belief state transition function τ when set of belief states B over the POMDP states is defined, with action space same as in the original POMDP, reward function defined for the belief states $R: B \text{ x } A \rightarrow \mathbb{R}$, and discount factor γ that is same as the underlying POMDP [Howard, 1960]:

$$\tau(b, a, b') = \sum_{o \in O} P(b'|b, a, o)\, P(o|b, a), \tag{2.20}$$

where:

$$P(b'|b,a,o) = \begin{cases} 1 & \text{if the belief update with arguments } b, a, o \text{ returns } b' \\ 0 & \text{otherwise} \end{cases} \tag{2.21}$$

Commonly when process operates in a noisy stochastic environment, initial state is unknown. This uncertainty of the initial state is represented by the probability of starting in state s, denoted as $b_0(s)$, which is used to initialize belief state update, while the update is done as per Bayes' rule as depicted in (2.17). As belief state update is the only possibility, considering that underlying POMDP state is unobservable, the state update is actually state estimation.

In line with MDP procedure, solution to a POMDP is a policy π that takes into account belief-state when determining which action maximizes the expected discounted reward.

When the state is known [Wolf and Kochenderfer, 2011], the following value function depicts the expected discounted return from a given state s:

$$V_\pi(s) = R\big(s, \pi(s)\big) + \gamma \sum_{s' \in S} P(\pi(s)|s, s') \sum_{o \in O(s)} \Omega(\pi(s)|s', o) V_\pi(s') \tag{2.22}$$

In this case the optimal policy π^* maximizes the expected discounted return from every state and has the following value function:

$$\pi^*(s) = \underset{a \in A}{\operatorname{argmax}} \left[R(s,a) + \gamma \sum_{s' \in S} P(\pi^*(s)|s, s') \sum_{o \in O(s)} \Omega(\pi^*(s)|s', o) V^*(s') \right] \tag{2.23}$$

$$V^*(s) = \max_{a \in A} \left[R(s,a) + \gamma \sum_{s' \in S} P(\pi^*(s)|s, s') \sum_{o \in O(s)} \Omega(\pi^*(s)|s', o) V^*(s') \right] \tag{2.24}$$

However, as POMDP functions in environments where systems do not know underlying states exactly, reward function and value function are evaluated over a belief-state. For the particular belief-state, reward function can be defined as:

$$R(b,a) = \sum_{s \in S} b(s) R(s,a) \tag{2.25}$$

Now, value of a belief-state can be expressed as follows:

$$V_\pi(b) = R\big(b, \pi(b)\big) + \gamma \sum_{o \in O_b} P\big(o \big| b, \pi(b)\big)\, V_\pi(b'), \tag{2.26}$$

where b' represents a future belief-state based upon the observation o. A POMDP policy, $\pi(b)$, specifies an action to take while in a particular belief-state b. The solution to a POMDP is the optimal policy, π^*, which chooses the action that maximizes the value function in each belief state [Wolf and Kochenderfer, 2011]:

$$\pi^*(b) = \underset{a \in A}{\operatorname{argmax}} \left[R(b,a) + \gamma \sum_{o \in O(s)} P(o|b,a) V^*(b') \right] \tag{2.27}$$

$$V^*(b) = \max_{a \in A} \left[R(b,a) + \gamma \sum_{o \in O(s)} P(o|b,a) V^*(b') \right]. \tag{2.28}$$

Finding an exact solution for π^* is often unfeasible as it requires iterating through all possible combination of actions, future belief-states, and observations until a certain finite moment, after which is necessary to determine optimal policy based on all rewards and probabilities. This process is computationally demanding even for moderately sized POMDPs; therefore, approximation methods were introduced to solve large POMDPs, as they generally scale much better.

2.6 Solution methods for large POMDPs

As described earlier, exact solutions for moderately sized POMDPs are impractical due to computational constraints. This is the reason why approximation methods were developed to deal with POMDPs that contain large number of states. There are two main approaches to solving POMDPs: offline that approximate the optimal policy for every possible belief-state, and online that solve only for the subspaces reachable from the current belief-state [Wolf and Kochenderfer, 2011]. Online solutions minimize problem and require much less time to compute than offline methods.

Offline POMDP solution techniques are discrete methods that require finite number of action and state spaces. Various discretization approaches, such as Point-based Value Iteration by Pineau et al. [2003], Heuristic Search Value Iteration by Smith and Simmons [2004], or Successive Approximation of the Reachable Space under Optimal Policies by Kurniawati et al. [2008], require accounting for each unique state and observation. Once an approximation of the value function is attained, the system can choose the action that maximizes the function defined by the optimal policy for the current belief-state.

On the other hand, online approaches reduce computational time of approximation as the search for the optimal action is done only from the current belief-state, rather than to compute the optimal policy for all belief-states. Discrete solving methods do not scale well with the large POMDPs as state space increases exponentially with the state variables. However, online POMDP solution methods compute series of different actions to find the largest discounted returns, which makes the compact representation of the belief-state in large state spaces and real-time approximation of solutions difficult to execute. Paquet proposed a Real-Time Belief Space Search (RTBSS) as one of the online POMDP approximation methods for large POMDPs [Paquet et al., 2005; Ross et al., 2008]. RTBSS method reduces computation time by factoring representation of the belief-state. Because of the factored representation, it is possible to represent state variables as independent entities and assign probability to each possible value of each belief-state variable. This allows for subspaces identification with probabilities of zero, which the system will not explore when searching belief-space for optimal actions.

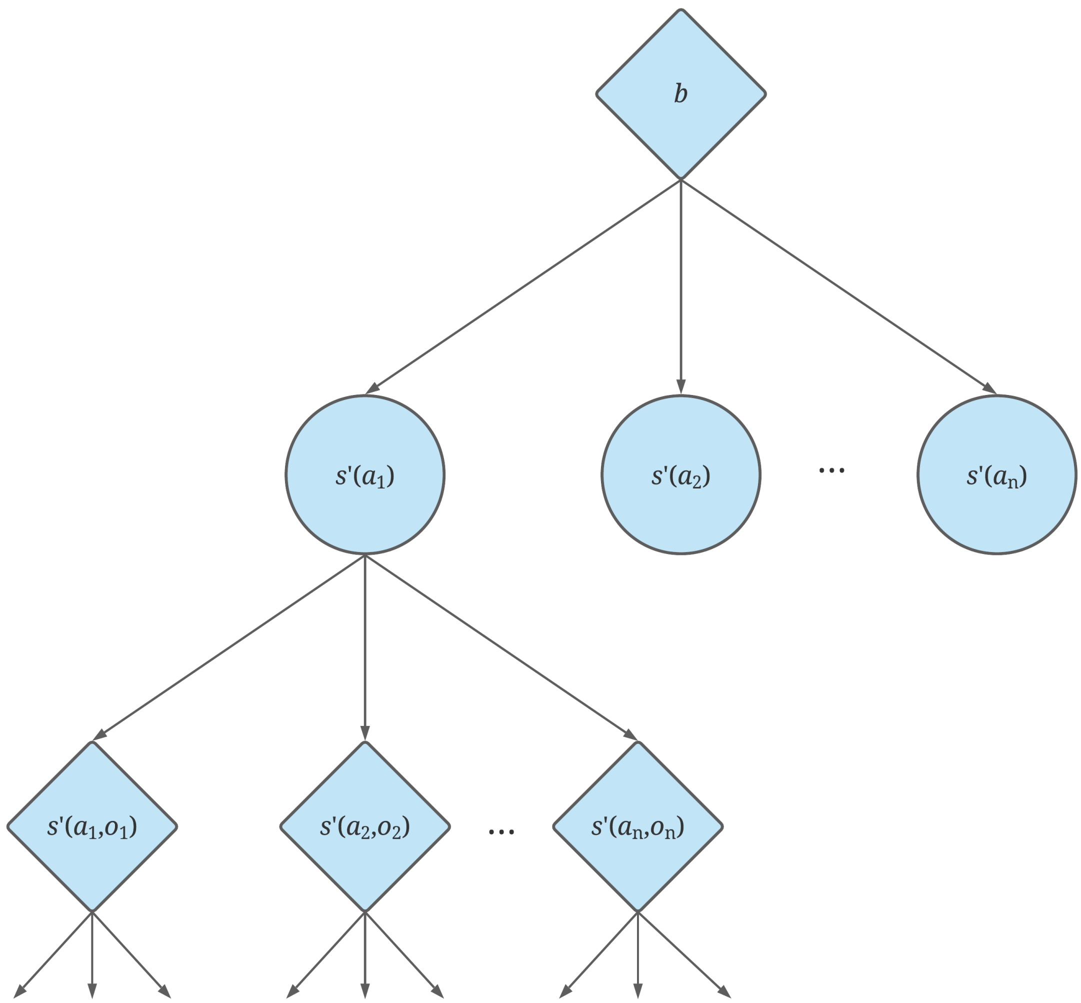


Figure 2.8 – Example of POMDP search tree

It is possible to represent a belief-state as a tree, similarly like in Figure 2.8, by starting at the current belief-state b. The branches correspond to possible actions a_n and observations o_n. Each combination of actions and observations will result in a new belief state at the next depth level of the tree. The process is repeated for each node until reaching some maximum depth, while the optimal policy is determined for the current belief-state. At each depth level, RTBSS will explore each possible belief-state given the apparent observations at that level. At this time, the algorithm determines belief-state variable value probabilities using transition and observation models with a goal of selecting the action with the highest value. In order to reduce computation times, RTBSS uses branch and bound methodology to reduce and trim sub-trees. The RTBSS algorithm solves the issue of approximation, but the problem of adequate belief-state representation remains. Thrun [2000] developed a feasible solution for the belief-state representation problem with his sample-based belief-state representation model that allows for non-linear transitions. Thrun's particle projection algorithm is especially useful for predicting future belief-states.

2.7 Reinforcement learning

Up to this point assumption was made that transition probabilities and reward functions are known; however, it is rare to find a system that offers this data to an abstract agent in the real sector. Considering collision avoidance problem, transition probabilities for going from one state to another are not intuitively known by the decision maker. Most of the navigators will have a belief about being in a certain state, but the risk thresholds would vary significantly.

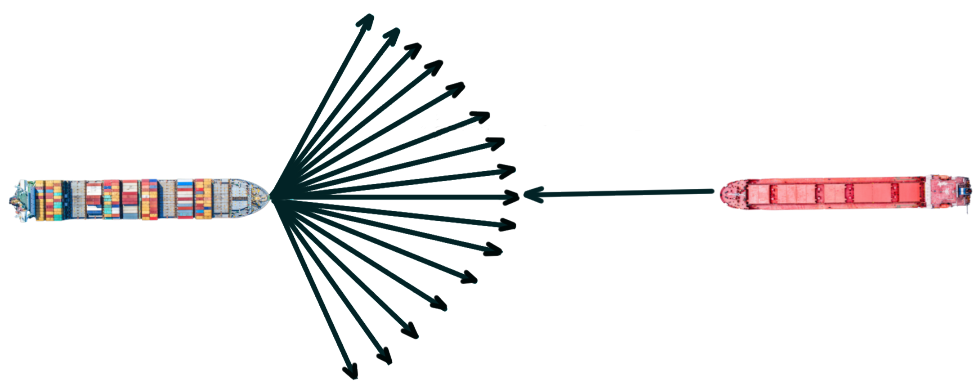


Figure 2.9 – Head-on evasive course selection

In Figure 2.9, an example of head-on situation is depicted where own vehicle has an option of selecting various courses in order to avoid collision with the head-on target. What is not known is which of these courses is the most rewarding course and what is the transition probability associated with each of the courses. Let's disregard Collision Regulations and personal intuition about the navigation for now. In case that the ownship agent is an autonomous agent that is first time set in this environment and in this situation. Agent is instructed not to collide and not to run aground with clearly stated rewards and costs for being collision free or collided. The goal of this study is model-free learning, and this is where reinforcement learning will aid to achieve a goal without necessity of fully modeling the world. For the sake of clarity, in this section it is assumed that the underlying Markov process is fully observable (MDP), but it would be extended to POMDPs in the future sections of this thesis.

As the model is unknown ahead of time, offline planning is not an option anymore. Alternatively, the abstract agent needs to learn the model online while exploring the environment. Optimized behavior is not designed, so the agent has to interact with the environment, pick actions and learn if these actions are good or bad. In addition to complexity of designing rewards and carefully discretizing action space, a proper balance of exploitation of the knowledge the agent has from previous experience with exploration to gain the knowledge is a challenge that needs to be confronted in the design stage of the collision avoidance and motion control models. It is also worth mentioning that reinforcement learning is often used when dealing with high dimensionality and complexity of known models as well [Howard, 1960]. It is immediately noticeable that allowing for an expensive commercial sea surface vehicle with dangerous cargo to explore the world on its own without knowledge of states and rewards would not be a good idea, so a certain level of knowledge is required before allowing vehicles to explore the world on their own. As the aim of this research is not full autonomy, but rather a decision-making support, human knowledge, experience, intuition, and uncertainty to learning models are integrated and endeavors made to find optimal policies and associated actions.

Reinforcement learning is not a revolutionary idea, as the premise is taken from the fields of biology and psychology where human and animal behavior is studied, specifically behavior after receiving rewards. The system is similar to the MDP and POMDP systems described earlier. The only difference is that the agent does not know that the MDP is there, but the system still behaves as an MDP. In the case of an MDP, agent takes an action, transitions in another state (which is observable to the agent), gets a reward and collects knowledge of the transition experience. After this, an agent again selects an action and transitions to another state collecting more knowledge about the environment. One successful example of utilizing reinforcement learning is the snake robot from the Stanford's AI Lab [http://ai.stanford.edu/~ang/originalHomepage.html]. The task given to the snake robot was to learn to climb a step without modeling the environment and without modeling forces on the robot. The only thing used as an input was a reward that the robot will get if it climbs a step. This approach is allowing the robot to learn on its own how to climb a step, but another approach could be to utilize knowledge of how real snakes climb steps and program that knowledge into the model so that this knowledge is available to the agent.

Figure 2.10 – States without transition probabilities

As depicted in Figure 2.10, it is possible to observe states and know possible actions, but it is not possible to see reward function or transition probabilities. When the agent faces this environment for the first time, it does not know which of the states: collision-free, near-miss, or collided is good or bad. This is the reason commercial vehicles are not allowed to explore the world, collide, collect negative rewards, and then learn that collided is a bad state. Initial knowledge is achievable offline with models or simulation and then transferred to an agent that is living in the real world. This is the reason why approach in this research includes both model-based and model-free learning. Even though for different reasons, model-based learning is actually performed for many years before the new build vessel is allowed to sail out from a shipyard. A series of simulations with mathematical models is done in the box, which is then confirmed under controlled conditions during sea trials. This collected data is then used to prime the navigators with knowledge about the new vehicle, so that proper decisions could be made about the motion control of the vehicle. Even though priming is important, models and simulations can't capture full dynamics of the real world. Motion control is complex and high-dimensional, so a fair amount of approximation is needed to get the representation of the environment. This is the reason why exploration is of crucial importance for dynamic environments.

There are many algorithms and approaches in the field of reinforcement learning [Sutton and Barto, 1998], but for this three-fold problem of motion control, collision avoidance, and hazard-alerting, Q-learning and SARSA algorithms fit well.

The idea behind the Q-learning is to utilize similar process as value iteration, but with Q-values. So, when looking at (2.12), a way has to be found to iterate value of Q-states Q by replacing state values of Bellman equations:

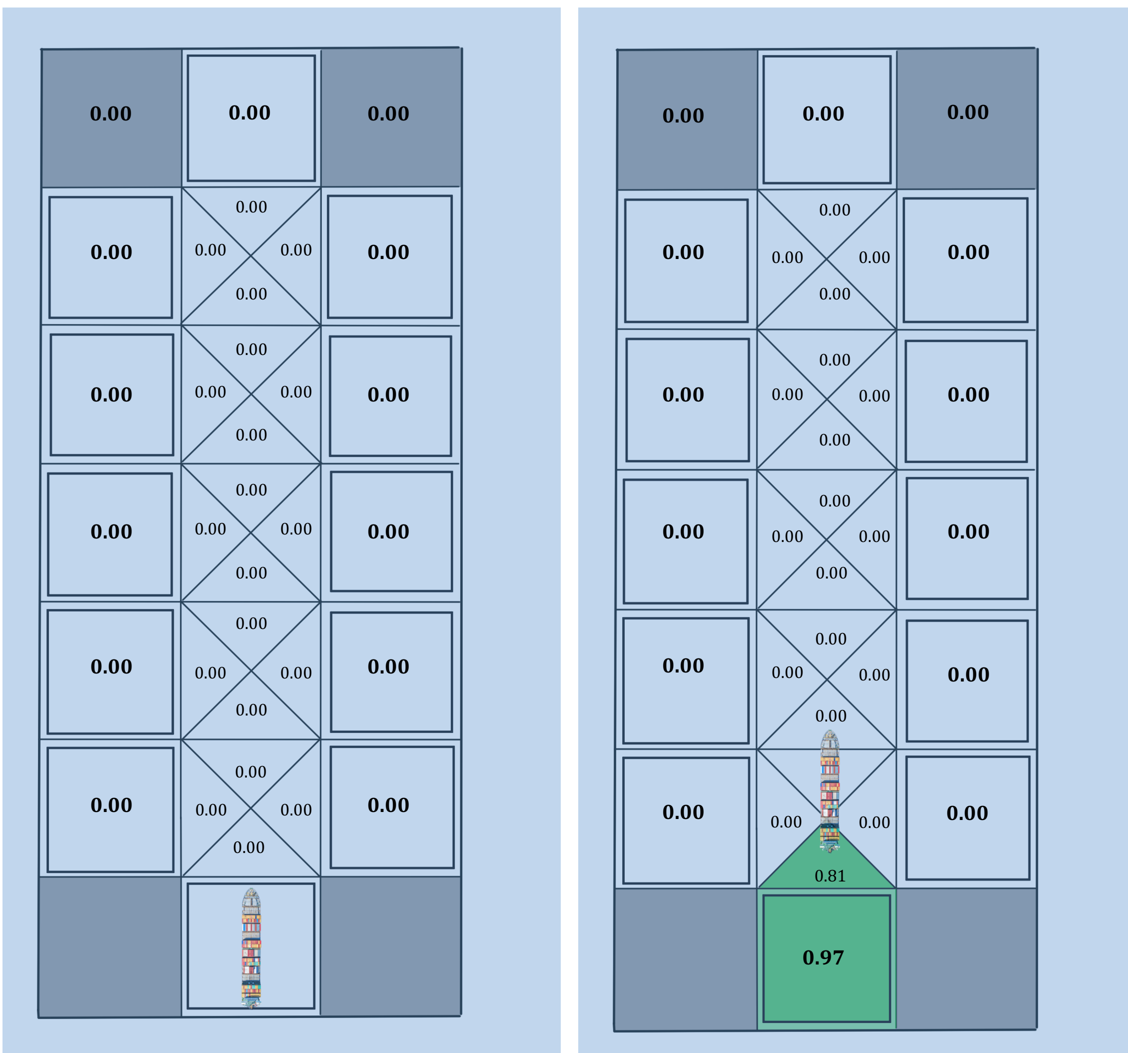
$$Q_{n+1}(s,a) \leftarrow \sum_{s'} P(s'|s,a)\left[R_a(s,a,s') + \gamma \max_{a'} Q(s',a')\right] \qquad (2.29)$$

As transition probabilities and/or reward function values are unknown to the abstract agent, sample estimation with Q-values is utilized and Q-state values incrementally updated:

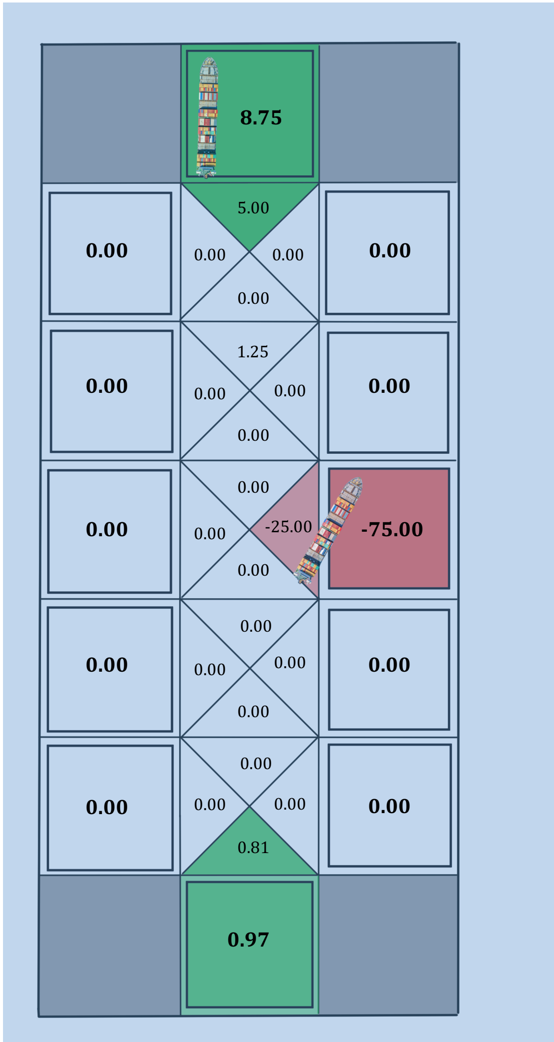
$$Q_{n+1}(s,a) \leftarrow Q_n(s,a) + \alpha\left(R(s,s') + \gamma \max_{a'} Q(s',a') - Q_n(s,a)\right), \qquad (2.30)$$

where $R(s,s')$ is the reward that agent receives when moving from the state s to state s', and α is learning rate ($0 < \alpha \leq 1$).

If finite horizon is not utilized, or if discounting factor close to 1 is assigned, then convergence could be difficult, so in the problem of motion control, collision avoidance and hazard alerting, it is necessary to limit the horizon, as more immediate situations and rewards are of greater interest. The learning factor is crucial when designing a model. Learning factor closer to 0 will incentivize agent to rely on prior knowledge and disregard exploration, while learning factor closer to 1 will use new experience to override previous knowledge and allow for more exploration than exploitation. Presented collision avoidance system has to be risk averse, so when in online stage, low learning rate would be assigned in order to discourage exploration, while higher learning rates could be assigned in offline learning models (simulation and mathematical modelling) which will provide enough knowledge for agent to exploit real-world environments. Certain amount of exploration should always be encouraged, as otherwise learning could lead to overfitting and that the agent stays within the known comfort and safety. The following example shows the relationship between exploration and exploitation.



(a) (b)



(c)

(d)

Figure 2.11 – Narrow channel navigation – RL approach

In this example, a situation where sea surface vehicle is sailing in a narrow channel is depicted. Vehicle has to stay within the channel, or otherwise it runs aground. Vessel receives rewards if it reaches WP 1 or WP 2. However, none of the information is known to the agent. It has to fully explore waters in order to learn which states are good and which states are bad. Figure 2.11 a) shows the beginning of exploration. Figure 2.11 b) shows several first iterations and it is possible to notice that the vehicle found a rewarding state. Now, it is crucial to state that if this system was designed to prefer exploitation and avoid exploration (learning factor α is closer to 0), then the vehicle would simply stay within these two states and collect rewards. It would never move forward. By allowing exploration (learning factor α is closer to 1), an agent is encouraged to search for other rewarding states and so figures 2.11 c) and 2.11 d) show how an agent is visiting states that have large negative rewards in order to learn which states are good and which states are bad.

This kind of approach in the real-world would not be acceptable. This is the reason why some level of previous knowledge is required. Also, integration with electronic charts (ECDIS), RADARs and AIS would allow for vehicle to have necessary information about available states. In this way, the agent would exploit previous knowledge, but to allow for the vehicle to reach next waypoint, a certain degree of exploration is required. Finding the appropriate learning rate is something that is done on trial basis, so it will depend on the situation, model and goals.

With some assumptions, SARSA (s, a, R, s', a') algorithm is a good alternative to Q-learning, as it uses an actual action rather than to maximize over all possible actions. The Bellman update than looks like (Sutton and Barto, 1998):

$$Q_{n+1}(s,a) \leftarrow Q_n(s,a) + \alpha\big(R(s,s') + \gamma Q(s',a') - Q_n(s,a)\big). \qquad (2.31)$$

As long as there is an appropriate exploration approach, SARSA will converge to the same result as Q-learning, but with less computational expense. More on these approaches in later chapters of this thesis.

Chapter 3

Noisy sensing, data fusion and motion control

In this chapter the first stage of the dynamic collision avoidance problem is presented, which is motion control and sensing. Motion control is successful as long as sensing of the environment is within acceptable levels of noise. Various manufacturers deliver different levels of quality within sensing; therefore, it is valuable to investigate uncertainties of the sensing equipment. False targets or nuisance readings will inevitably result in reduced trust in the collision avoidance system by the operators.

To model collision avoidance problem, focus is maintained on reducing uncertainty as much as possible. In order to determine risk of collision, it is important to collect information about ownship and intruders' heading, course, speed, relative distance and angle, etc. All this information is gathered by utilizing various sensors. Even though a case of collision avoidance with perfect sensor is covered, this is done to benchmark performance of the model in the real-world scenarios, as sensors are always bounded by noise, interruption, and field of view restrictions. One of the challenges involving shipboard sensors is the inconsistency of noise experienced by different manufacturers of sensing equipment and modeling their uncertainty. Also, environmental impact on sea surface vehicle's motion control is of great importance when executing collision avoidance maneuvers. Therefore, this chapter starts by introducing Foraging Particle Filter (FPF) that will be used to reduce uncertainties caused by sensor noise.

3.1 Foraging Particle Filter (FPF)

When estimating a correct output, various process and measurement noises should be considered. If a system of interest is observed holistically, system's states provide the current status of control, condition, and observability of the process. To ensure precise controls, system state updating within the sampling time frame is of crucial importance. However, due to noisy sensing, estimating internal states becomes a difficult task that usually leads to approximation of hidden states. State estimation, and more general filtering, plays a significant role in various domains, such as target tracking [Chen, 2012; Dias and Bruno, 2013], robot navigation [Hiremath et al., 2014; Atia et al., 2010], computer vision and robotics [Dellaert, et al., 1999; Isard and Blake, 1998], process management [Gao and Ho, 2007; Abdullah and Zribi, 2013; Gao and Ho, 2006], etc.

For linear systems with Gaussian noise, industry and academia has commonly turned to Kalman filter solutions for state estimation [Fossen and Perez, 2009; Gustafsson et al., 2000; Kailath et al., 2000; Chen, 2012; Roshany-Yamchi et al., 2013] and grid-based methods for dynamical systems with finite states [Ristic et al., 2004; Arulampalam et al., 2002]. Nonlinear systems required a different approach, as state estimation is not that straightforward and includes hidden states. Extended Kalman Filter (EKF), Cubature Kalman Filter (CKF), and Unscented Kalman Filter (UKF) have been developed for this purpose, but due to their poor state estimation recursive Monte Carlo signal processing solutions have gained in popularity [Sunahara, 1969; Julier and Uhlmann, 1997; Arasaratnam and Haykon, 2009]. Particle filters thrive well in nonlinear systems with non-Gaussian measurement noises and outlier robustness. The main objective of particle filtering is approximation of the posterior distribution by drawing a number of particles from the approximated distribution when some of the state variables are only partially observed or unknown.

In this chapter several known issues of particle filters are studied. Considering the approximate nature of posterior estimation and particle representation, particle degradation and impoverishment problem arises. Resampling process can help particle degradation, but the sample impoverishment is more complex to tackle. If left untreated, particle filters could fail to correctly estimate the dynamic state. Particle filters estimate states of partially observable Markov chains with high computational burden, degree of which depends largely on algorithmic solutions for a posterior estimation. The intention is to address issues of particle degradation and impoverishment by generating particle scions and designing a lean algorithmic solution with higher computational efficiency.

This approach focuses on likelihood determination and resampling procedure that follows foraging principles from nature. Defined in this way, the approach belongs to metaheuristics methods that are incorporated in the classical MDP filter in order to enhance its performance. Literature review offers us several intelligent and optimization approaches to creating intelligent particle filters: Evolutionary Algorithms (EA) [Uosaki et al., 2004; Uosaki and Hatanaka, 2007], Genetic Strategy (GS) [Higuchi, 1997; Kwok et al., 2005; Park et al., 2009, Yin et al., 2016], Ant Colony Optimization (ACO) [Xu et al., 2009; Zhu et al., 2010; Heris and Khaloozadeh, 2013], Particle Swarm Optimization (PSO) [Tong et al., 2006; Zheng and Meng, 2008], Artificial Fish Swarm (AFS) [Xiaolong et al., 2008], Markov Chain Monte Carlo (MCMC) PSO [Jing and Vadakkepat, 2010], etc. Proposed approach is based on the similar philosophy of intelligent optimization taking advantage of foraging behavior of wildlife. In the proposed algorithm sampling phase is enhanced by utilizing intelligent foraging in order to attract fittest particles of the state space to most probable locations. In the FPO, foraging is used as an adaptive fitness generator that aids estimation performance and improves solutions of the filtering problem.

3.1.1 Nonlinear filtering model

The general particle filter estimates states of dynamical systems. To estimate posterior, particle filters require data and probabilistic generative model of the system. Data will depend on the dynamic system for which the particle filter is used and can contain sensor measurement data and control data. In the continuation of this chapter and to follow a common notation from control theory, subscript t is used to refer to the occurrence in time, while superscript ^{t} is be used for all events that led up to that time t. System dynamics are described as:

$$x_t = f_x\left(x_{t-1}, \upsilon_{x_t}\right) \tag{3.1}$$

$$z_t = f_z\left(x_t, \upsilon_{z_t}\right) \tag{3.2}$$

$$u_t = f_u\left(x_t, \upsilon_{u_t}\right) \tag{3.3}$$

where $f_x(\cdot)$, $f_z(\cdot)$, and $f_u(\cdot)$ depict process, measurement, and control functions respectively. $x_t \in \mathbb{R}^{n_x}$ is the state to be estimated, $z_t \in \mathbb{R}^{n_z}$ is the measurement and $u_t \in \mathbb{R}^{n_u}$ is the control in the system dynamics. However:

$$z^t = z_1, z_2, \ldots, z_t \qquad \text{and} \qquad u^t = u_1, u_2, \ldots, u_t$$

Therefore, measurement at the time t is denoted z_t, while u_t denotes the control within the time period $(t-1, t]$. The dimensions of x_t, z_t, and u_t depend on the particular system dynamics and are denoted as n_x, n_z, and n_u respectively. $\upsilon_{x_t} \in \mathbb{R}^{n_x}$, $\upsilon_{z_t} \in \mathbb{R}^{n_z}$, and $\upsilon_{u_t} \in \mathbb{R}^{n_u}$ are noises with known probability density.

Statistical model of the system dynamics described above could be defined as:

$$p(x_0) \tag{3.4}$$

$$p(x_t|x_{t-1}) \tag{3.5}$$

$$p(z_t, u_t|x_t) \tag{3.6}$$

where $p(x_0)$ is an initial distribution that is used to build future state estimations, $p(x_t|x_{t-1})$ could be defined as transition probability distribution that corresponds to process equation (3.1), while $p(z_t, u_t|x_t)$ is described as likelihood distribution that is built upon measurement and control distributions.

Like other members of Bayesian filters, particle filters use data $p(x_t|z^t, u^t)$ to estimate the posterior distribution of the dynamical system following the recursive formula:

$$p(x_t|z^{t-1},u^{t-1}) = \int_0^t p(x_t,x_{t-1})p(x_{t-1}|z^{t-1},u^{t-1})dx_{t-1} \tag{3.7}$$

$$p(x_t|z^t,u^t) = \frac{p(z_t,u_t|x_t)p(x_t|z^{t-1},u^{t-1})}{\int_0^t p(z_t,u_t|x_t)p(x_t|z^{t-1},u^{t-1})dx_t} \tag{3.8}$$

However, $p(x_t|z^{t-1},u^{t-1})$ and $p(x_t|z^t,u^t)$ contain complex probability integral, and as a non-linear system is described, it is difficult to obtain analytic solution for the posterior distribution. Considering the continuing nature of states x, controls u, and measurements z, it is difficult to calculate the entire posterior in reasonable time, even when applications are discrete. Instead of analytical calculations, particle filter approximates the posterior with a mass of particles $x_t^{[j]}(j = 1,2,\dots,PN)$, where PN is the number of particles. Initial particles $x_0^{[j]}$ are drawn from $p(x_0)$. One of the particle filters benefit is that the horizon is finite once number of particles has been selected. The challenge remains to find the number of particles to be generated while maintaining computational efficiency in line with the sampling frequency.

Sequential Importance Sampling (SIS) [Rubin, 1988] is introduced to deal with difficulties of sampling from the posterior distribution:

$$q(x_t|z^t,u^t) = q(x_{t-1}|z^{t-1},u^{t-1})q(x_t|x_{t-1},z^t u^t), \tag{3.9}$$

where q is importance distribution and p continues to represent nominal distribution.

To approximate the posterior in an efficient way, weights are assigned to each of the generated particles. The weight of each particle could be determined by a recursive formula combining expressions (3.8) and (3.9):

$$w_t^{[j]} = \frac{p(x_t|z^t,u^t)}{q(x_t|z^t,u^t)} \tag{3.10}$$

$$w_t^{[j]} \propto w_{t-1}^{[j]} \frac{p\left(z_t,u_t|x_t^{[j]}\right)p\left(x_t^{[j]}|x_{t-1}^{[j]}\right)}{q\left(x_t^{[j]}|x_{t-1}^{[j]},z^t,u^t\right)} \tag{3.11}$$

$w_t^{[j]}$ is the importance weight, while $\tilde{w}_t^{[j]}$ denotes the normalized importance weight. Therefore, by utilizing Monte Carlo method, the posterior distribution $p(x_t|z^t, u^t)$ can be approximated as follows:

$$\hat{p}(x_t|z^t, u^t) \approx \sum_{j=1}^{NP} \tilde{w}_t^{[j]} \delta\left(x_t - x_t^{[j]}\right) \tag{3.12}$$

where $\delta(\cdot)$ is the Dirac delta measure. Therefore, taking into account Sequential Importance Sampling:

$$E(x_t) = \int_0^t x_t \frac{p(x_t|z^t, u^t)}{q(x_t|z^t, u^t)} q(x_t|z^t, u^t) dx_t \approx \frac{1}{NP} \sum_{j=1}^{NP} w_t^{[j]} x_t^{[j]} \tag{3.13}$$

It is necessary to emphasize that sequential importance sampling suffers from particle degeneracy and after several iterations all, but one particle, will have small weights which can be misleading, especially if there is a long tail in the posterior distribution and higher probability is located in that tail. Doucet et al. [2000] have shown that particle degeneracy is unavoidable in sequential importance sampling. Effective sample size NP_{eff} is a tool developed by Arulampalam et al. [2002], which can be used to determine particle degeneracy degree:

$$NP_{eff} = \frac{1}{\sum_{j=1}^{NP} \left(\tilde{w}_t^{[j]}\right)^2} \tag{3.14}$$

If the number of effective particles falls below a certain pre-defined threshold, resampling step is engaged. Various approaches to resampling are commonly used in practice with a common goal of avoiding particle degradation. However, resampling carries a sample impoverishment side effect, which can cause erroneous state estimation due to lack of particle diversity in the state space. Particle degeneracy is especially evident outside the three-sigma range of the posterior distribution. A general solution is resampling with a goal of removing weaker particles (particles with smaller weights) while retaining heavier particles (particles with larger weights). As denoted before, the main issue after resampling is the impoverishment problem where iterations increase focus on one sampling position, while the number of unique particles decreases steadily reducing the accuracy of posterior estimation. In Figure 3.1 the effects of particle degeneracy and sample impoverishment can be seen.

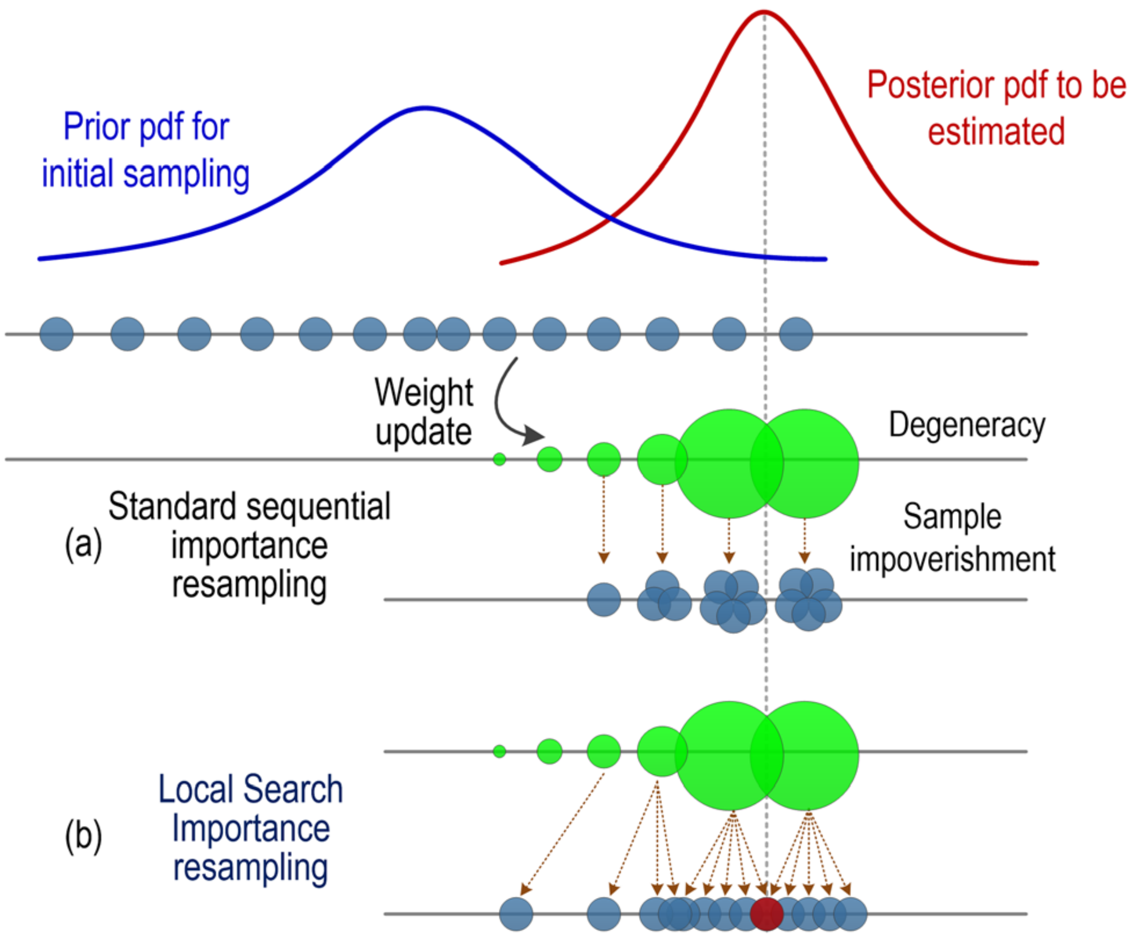


Figure 3.1 – Particle weight update with (a) standard sequential importance resampling and (b) local search importance resampling (Source: Wang and Gao, 2016)

3.1.2 Foraging Particle Filter optimization

Resampling solves the issue of particle degeneracy, but before proceeding with resampling, it is necessary to tackle the impoverishment issue after which it is possible to proceed with generation of the particle scions that would approximate posterior distribution. Proposal in this thesis is to use nature's foraging process when designing algorithm that will solve the issue of impoverishment. This solution is similar to genetic, swarm and evolutionary approaches, where intelligent optimization is used to ensure computational efficiency. Modified particle filter that exploits fitness function is presented, which can be defined as the food source with the highest energy nutrients.

Foraging process applies to all living beings that have to use their sensing parts to search for the food. Mobility of foraging animals is driven by the food source. In this case, the food source represents any possible solution. Therefore, initially all generated particles in the first step are considered a possible solution / food source. Once the particle filter has generated the particles, foraging algorithm is employed to find particles with the fittest solution. If no previous observation / measurement is available, all generated particles will have the same probability. Then by utilizing the measurement available and adjusted for the measurement noise, food source is updated via recursive fitness probability formula. Number of recursive cycles has to be decided in advance, depending on the process and dynamic system that is controlled. The solution space is updated until the satisfactory threshold is reached, which again depends on the system dynamics. In this way sample impoverishment are managed before taking a resampling step.

Once the initial set of particles is generated by drawing $x_0^{[j]}$ from $p(x_0)$, foraging process commences by determining the food position with the highest nutrient and energy values:

$$p\left(x_t^{[j]} | fit_x^{[j]}\right) = \frac{fit_x^{[j]}}{\sum_{n=1}^{PN} fit_x^{[n]}} \tag{3.15}$$

where $p\left(x_t^{[j]} | fit_x^{[j]}\right)$ is used to update the initially generated particle positions with the fitness information, and *PN* is the particle number generated by the particle filter. In this case assumption is made that foraging is done in herd and that the number of herd members corresponds to the particle number. This will be used in the next step when the fitness of particles is compared with neighboring particle, after which the particle with lower fitness value is eliminated. The following expression is used to determine fitness function:

$$fit_x^{[j]} = exp\left[-\frac{1}{2\upsilon_k}\left(z_t - \tilde{z}_{(t|t-1)}^{[j]}\right)^2\right] \tag{3.16}$$

where $\upsilon_k$is the measurement noise covariance (this will depend on the system dynamics and if more than one observation source is available), z_t is the newest observation, while $\tilde{z}_{(t|t-1)}^{[i]}$ is the predicted value of z_t given the measurements up to time $(t-1)$.

The next step could be characterized as greedy update of particle positions with a goal of attracting particles towards the higher likelihood of the posterior distribution:

$$x_{t+1}^{[j]} = x_t^{[j]} + \varpi^{[j]}\left(x_t^{[j]} - x_t^{[l]}\right) \tag{3.17}$$

where $\varpi^{[i]}$ is a random number in the interval $[-1,1]$ and l is to differentiate between a monitored and neighboring particle, so it is a random selected index that will be different from i.

If the fitness function shows that $x_{t+1}^{[i]}$ has a better fitness result than $x_t^{[i]}$, then the position of the food source $x_t^{[i]}$ is changed to $x_{t+1}^{[i]}$, otherwise it remains the same. In this way all particles are attracted to the region of higher fitness value and impoverishment is avoided. Iteration is repeated until pre-defined iteration counter is done, iteration time is satisfied, when fitness value reaches certain threshold, or some optimization/constraint rule is satisfied, which depends on the system dynamics and design of the particle filter.

After the foraging process is completed, the particle filter resumes with sequential importance sampling, assigning weights to particles as described in Chapter 3.1.1. NP_{eff} is used to determine if resampling is required. Resampling is unavoidable, as after a while all but one particle will have small weights. If NP_{eff} is equal to NP, all particles have same weights and that is the lowest degree of particle degeneration. On the other hand, if the $NP_{eff} = 1$ only one particle would accumulate all the weight, which is not desirable and carrier the highest degree of degeneracy. For example, threshold could be set to $NP_{th} = \frac{2}{3}NP$.

In order to resolve degeneracy issue, resampling is done as per the Multinomial Resampling technique [Doug and Cappé, 2005]. For NP times the following steps are repeated:

1) Generate a random number u_j from the uniform distribution $(0,1]$;
2) Search the variable $i \in \{1, \ldots, NP\}$ which satisfies:

$$\sum_{m=1}^{i-1} q_m < u_j \leq \sum_{m=1}^{i} q_m \tag{3.18}$$

3) Store the $x_t^{[i]}$ as a scion particle.

Once the multinomial resampling is completed, particles with small weights are eliminated and scion particles are created for the particles with larger weights. Therefore, the posterior $\hat{p}(x_t|z^t, u^t)$ distribution can be approximated by the scions as:

$$\hat{p}(x_t|z^t, u^t) \approx \frac{1}{NP}\sum_{j=1}^{NP} NP_t^{[j]}\delta\left(x_t - x_t^{[j]}\right) \tag{3.19}$$

where $NP_t^{[j]}$ is the number of scions for the parent particle $x_t^{[j]}$.

Foraging Particle Filter algorithm is as follows (in this example time limit is set for an iteration):

Algorithm 1 - FPF Algorithm

Input: particle number NP; initial sample set $x_0^{[j]}$

Output: State estimation result $\hat{x}_t$

Initialization: $t = 0$

for $j = 1, 2, \ldots, NP$ do:

    Draw initial particles $x_0^{[j]}$ from $p(x_0)$ and set the initial weights as $1/_{NP}$

    Sample from the distribution $p(x_t|x_{t-1})$ to obtain particles $x_t^{[j]}$

end for

Set iteration counter, $t \leftarrow 1$;

while $t \leq T$ **do**

    Foraging sample optimization:;

    for $j = 1, 2, \ldots, NP$ do:

        Calculate fitness value for all particles $x_t^{[j]}$ with (3.16)

        Update the particles with (4.15) to receive particle estimation with fitness value $x_t^{[j]}$

        Update the particle positions with greedy selection (3.17)

        Return updated $x_t^{[j]}$

        Increment iteration counter, $t \leftarrow t + 1$;

    end for

end while

Calculate the normalized weights for all $x_t^{[j]}$ particles using equations (3.10) and (3.11) and denote particles as $\left\{x_t^{[j]}, \widetilde{w}_t^{[j]}\right\}$

Use the expression (3.14) to evaluate sample effectiveness

if $NP_{eff} < NP_{th}$ **then**

    Multinomial resampling:;

    for $j = 1, 2, \ldots, NP$ do:

        Generate a random number u_j from the uniform distribution (0,1]

Search the variable $i \in \{1, \ldots, NP\}$ which satisfies (3.18)
Store the $x_t^{[i]}$ as a scion particles
end for
Estimate and return the hidden state $\hat{x}_t$ using equation (3.19)
else
Estimate and return the hidden state $\hat{x}_t$ using equations (3.12) and (3.13)
end if
end

3.1.3 Experimental results

In order to benchmark performance of the Foraging Particle Filter, a scalar growth model experiment has been conducted. In this example a well-known econometrics model that is commonly used to evaluate nonlinear filters [Arulampalam et al., 2002; Park et al., 2009; Kalami Heris and Khaloozadeh, 2014] is used. The scalar growth model is defined by:

$$x_{t+1} = \gamma_1 x_t + \frac{\gamma_2 x_t}{1 + x_t^2} + \gamma_3 \cos(1.2t) + d_t \tag{3.20}$$

$$y_t = \frac{1}{20} x_t^2 + e_t \tag{3.21}$$

where d_t and e_t are independent zero-mean Gaussian noises with variances σ_d^2 and σ_e^2 respectively. The constants of the model are defined as: $\gamma_1 = 1, \gamma_2 = 12, \gamma_3 = 7, \sigma_d^2 = 4,$ and $\sigma_e^2 = 4$. The initial state of the dynamic system is set as $x_0 = 0.1$.

Add extrinsic input u_t to (3.20):

$$x_{t+1} = \gamma_1 x_t + \frac{\gamma_2 x_t}{1 + x_t^2} + \gamma_3 \cos(1.2t) + u_t + d_t \tag{3.22}$$

It is assumed that u_t is unknown and that model receives measurements as per (3.21). Foraging Particle Filter (FPF) is compared with basic particle filter (PF), Sequential Importance Resampling (SIR) filter, Auxiliary Particle Filter (APF), Regularized Particle Filter (RPF) and Ant Colony Estimator (ACE). All these filters are used in simulation to estimate the state in (3.20) and (3.21). In the simulation it is assumed that:

$$u_t = 70\delta(t-40) = \begin{cases} 70, & t = 40 \\ 0 & \text{otherwise} \end{cases} \tag{3.23}$$

where $\delta(\cdot)$ denotes Dirac's delta function. The algorithms are not allowed to know the value of u_t and will have to use expression (3.20) instead of (3.22) for state estimation.

One thousand simulations have been conducted with each algorithm. The Particle Number for all algorithms has been set to 500, while PF and RPF used resampling threshold of 100. FPF algorithm is trialed with various number of particles and discovery was made that with as low as 75 particles it is possible to get comparable and even better performing results.

In Table 3.1 the minimum RMSE (the best), the maximum RMSE (the worst) and the mean of the Root-Mean-Squared Error (RMSE) values of all algorithms are compared. Also, box plots showing statistical representation of the RMSE is given in the Figure 3.2. Algorithms were not permitted knowledge of the input signal as defined in (3.22) as this signal creates sample impoverishment problem. Comparative performance of FPF was done as well.

As depicted in Figure 3.2, it is evident that FPF and ACE statistically outperform other algorithms. FPF requires more steps due to weighing requirements, but it still makes lower computational burden than PF, SIR, APF or RPF algorithms. Even though FPF and ACE outperform other algorithms in all aspects of RMSE, low standard deviation shows us that these methods are reliable and sustainable solutions.

Table 3.1 – Root-Mean-Squared Error comparison for observed algorithms

Algorithm	Mean	Min.	Max.	STD
PF	45.6038	14.6842	75.2844	7.6385
SIR	45.8362	14.8038	74.8728	7.4245
APF	45.6277	14.4604	75.5792	7.7584
RPF	32.1084	1.0400	68.3889	19.1499
ACE	3.4547	0.6171	19.3974	3.4932
FPF	3.0119	0.2539	21.0613	2.6266

By utilizing trial and error method it was determined that increasing particles number beyond a certain level will only increase computational expense without any significant improvements in estimate accuracy. This is numerically presented by Table 3.2. As visible from Figure 3.3, STD falls significantly after 50 iterations. Therefore, it is possible to increase the number of iterations beyond the 100 iterations until it is noticed that the computational expense is higher than the benefit of an extra iteration. Of course, more powerful machines will allow for a higher number of iterations, but as long as the machine can handle 100 iterations, results are acceptable from the accuracy point of view. Similarly, when considering figures 3.4, 3.5, and 3.6, it is possible to notice a similar trend, even though for a Minimum RMSE, difference is not as prominent as it is in the case of Standard Deviation. If computing power allows for 500 iterations, an optimal resulting Scalar Growth Model can be achieved, even though 100 iterations would still allow for good results as well, albeit suboptimal.

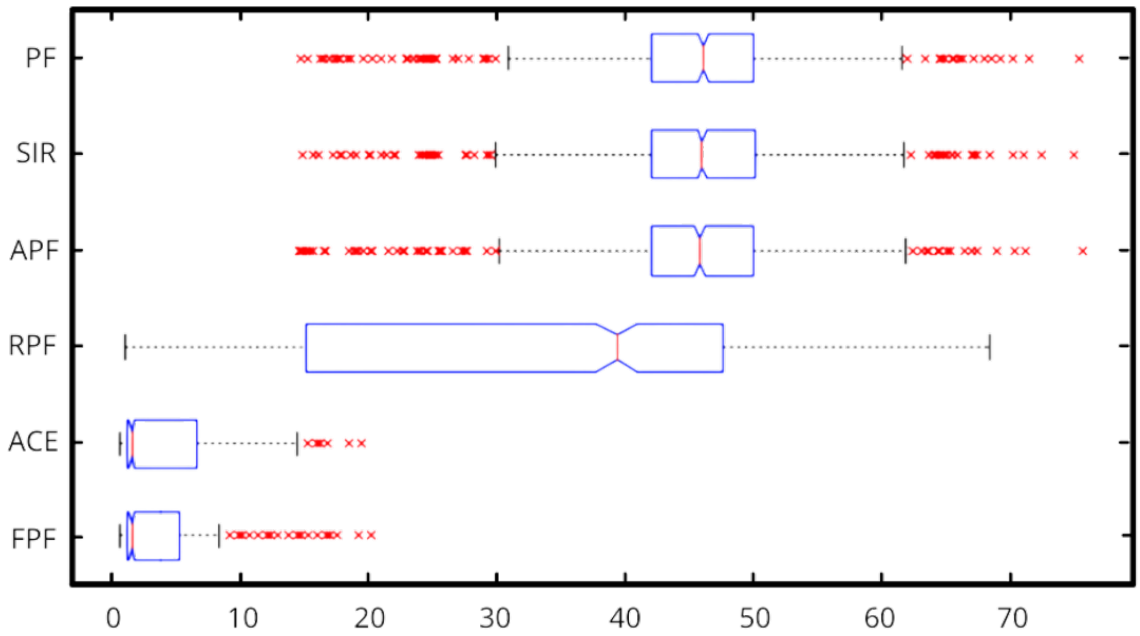


Figure 3.2 – Box plots depicting statistical RMSE information of state estimation with 1000 iterations

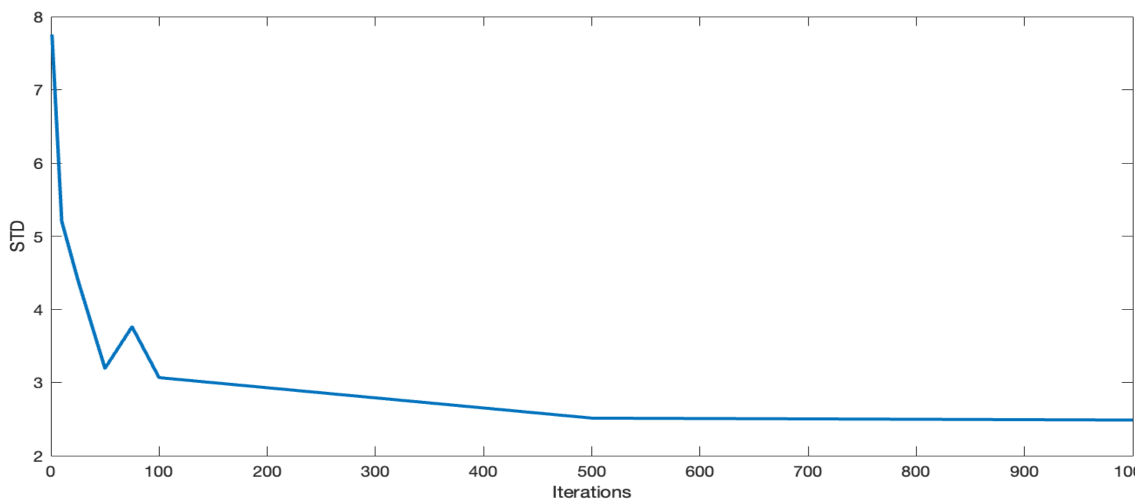


Figure 3.3 – Standard Deviation for Scalar Growth Model

Table 3.2 – Influence of Particles Number on RMSE

Particles	Mean	Min.	Max.	STD
1	10.5434	0.0019	27.8876	7.7554
5	8.0041	0.0012	27.313	6.6277
10	5.9845	0.0018	27.4369	5.2042
25	5.014	0.0013	24.3715	4.3996
50	3.6203	0.0041	28.0541	3.1935
75	4.279	0.0006	25.6167	3.7616
100	3.5148	0.0033	23.955	3.0667
500	2.9119	0.0019	19.9184	2.5136
1000	2.8664	0.0004	27.0695	2.4866

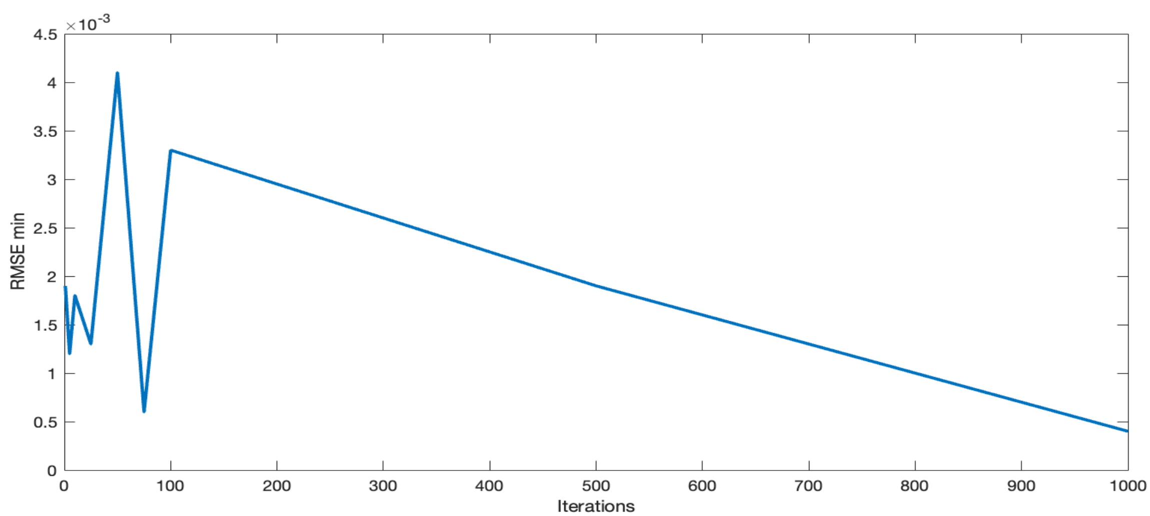


Figure 3.4 – Minimum Root-Mean-Squared Error for Scalar Growth Model

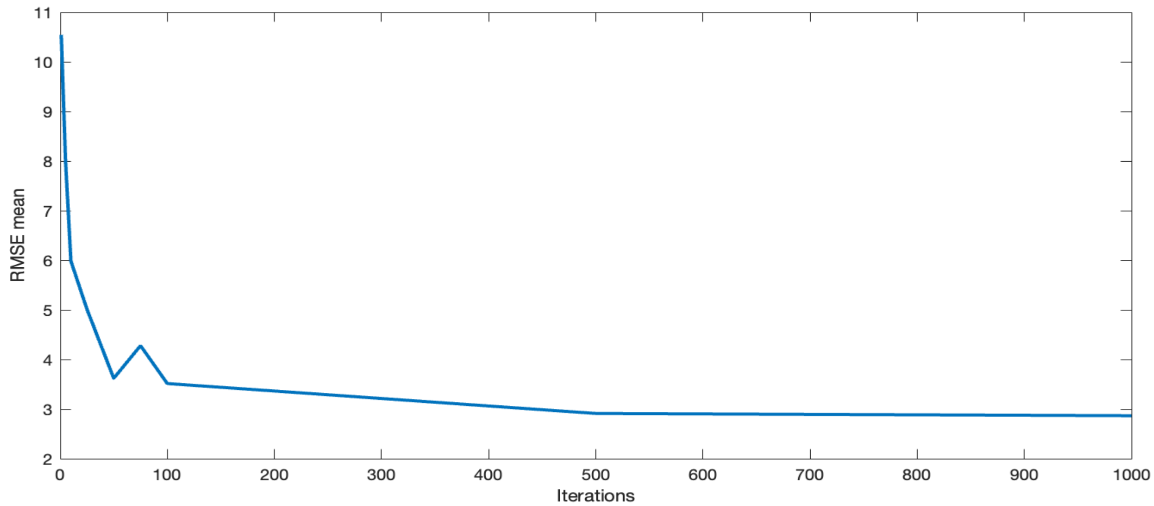


Figure 3.5 – Mean Root-Mean-Squared Error for Scalar Growth Model

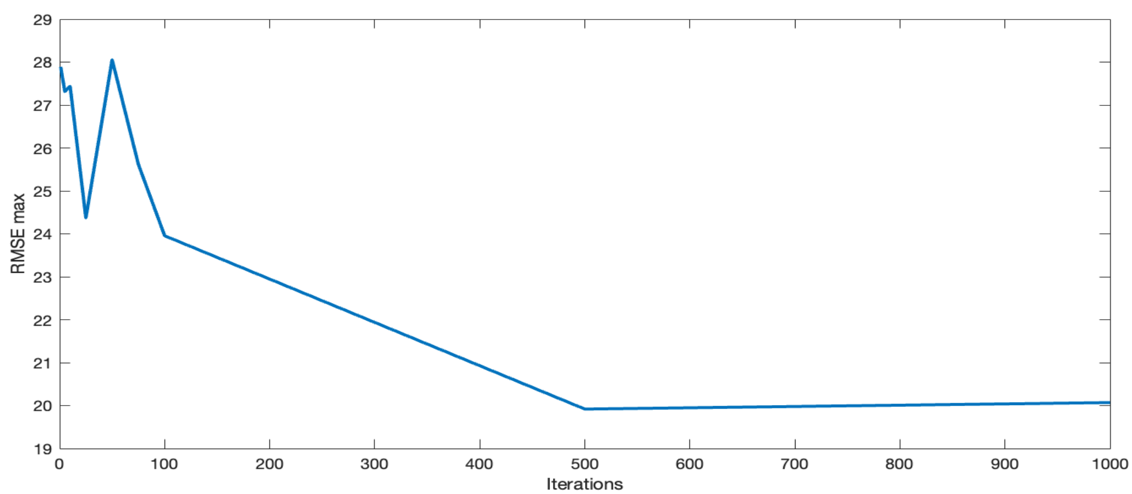


Figure 3.6 – Maximum Root-Mean-Squared Error for Scalar Growth Model

Figures 3.7 and 3.8 depict success rate of FPF in estimating scalar values, where it is visible that proposed Foraging Particle Filter manages to estimate scalars effectively with minimal errors. In order to showcase scalar estimation accuracy, iterations number has been reduced to 100 to declutter the overview.

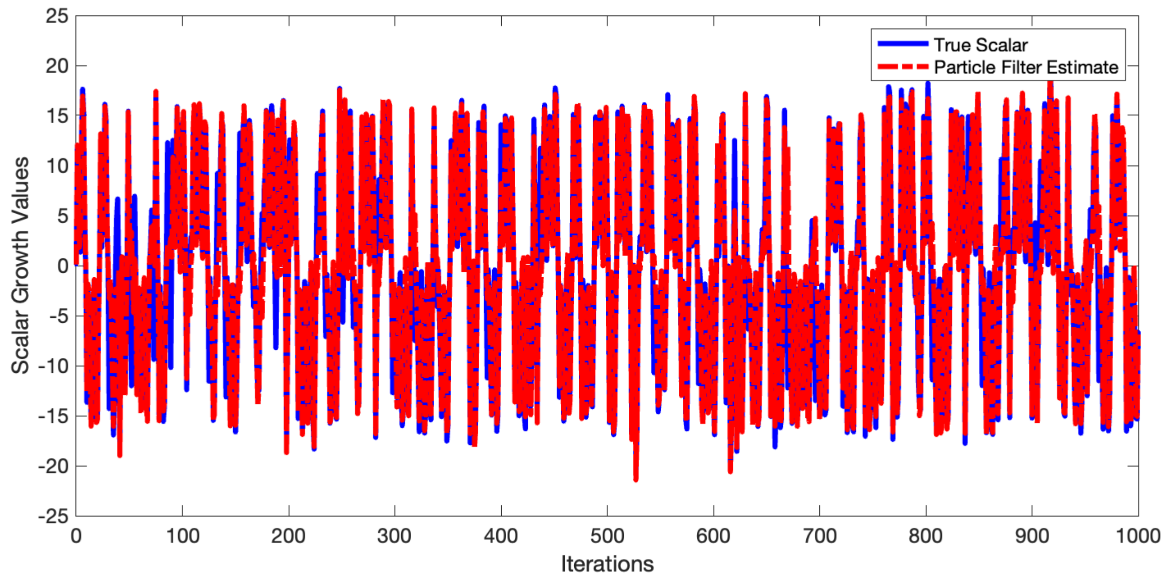


Figure 3.7 – Foraging Particle Filter scalar estimation with 1000 iterations

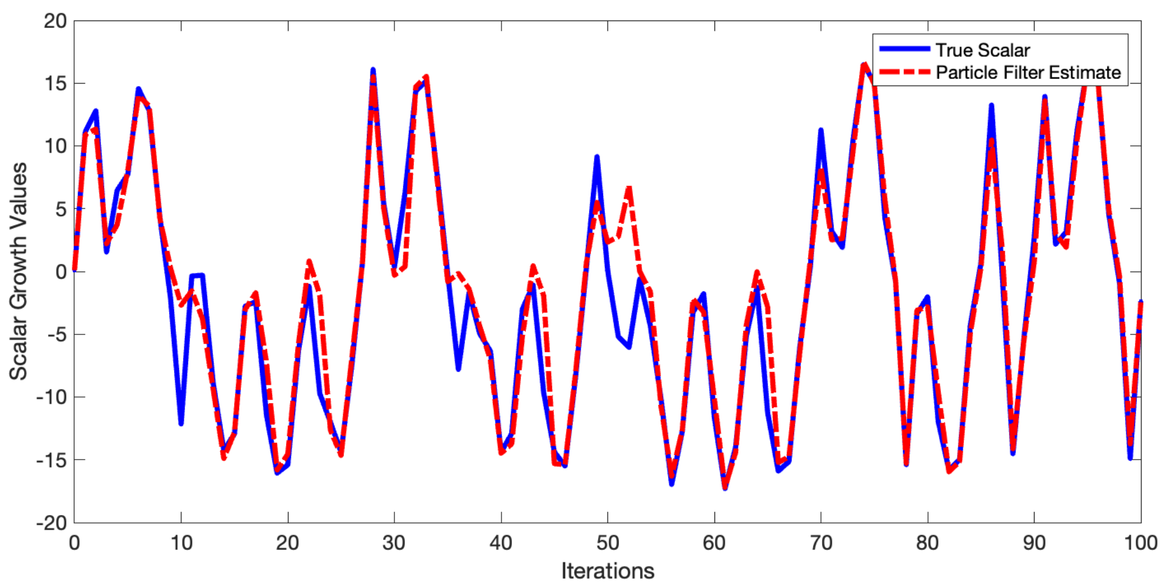


Figure 3.8 – Foraging Particle Filter scalar estimation with 100 iterations

3.2 Nonlinear filtering for motion control of sea surface vehicles

The main goal of designing Foraging Particle Filter is to reduce uncertainty and improve state estimation in collision avoidance system dynamics. Together with position fixing, sensor errors and human uncertainties, heading control is one of the parameters that have significant impact on accurate representation and collision avoidance decision-making. In the proposed model, sea surface vehicles are assumed to be ocean going vessels of various size that deliver cargo and passengers; therefore, dynamic positioning is not explicitly covered in this thesis, but this work can be extended in the domain of the offshore industry with minor adjustments. One could argue that the present setup onboard the vessels is sufficient for effective decision-making, however in lieu with the development of decision support system, dynamics with tighter motion control that take in consideration uncertainties and sensor noises are required to effectively deliver accurate state estimations and appropriate control measures.

3.2.1 Ship motion control overview

In the last 40 years the demand for accurate ship handling and reliability of the motion control has been increasing. The level of demand correlates with the operational demands, so there is higher demand in the offshore, autonomous, and precision sailing industry, while the classic navigating vessels have relaxed standards due to the ocean-operating zones where classic autopilot with lower level of precise filtering and actuator control is acceptable. However, modern vessels already have a sophisticated equipment that allow for implementation of various software solutions that could significantly improve motion control while the cost of implementation would remain acceptable. Position and heading control with trajectory tracking is of interest of this research, but other control objectives, such is wave-induced motion reduction, could be included in the model.

Kalman filters have been used extensively in the previous years, as the level of discretization and linearization was acceptable for the motion control. However, given the stochastic nature of the external forces (wind, waves, and currents), aim is to use nonlinear state estimation that can avoid linearization, can accept other than Gaussian noises and can avoid computing Jacobi matrices. When the system dynamic is strongly nonlinear and has a serous noise levels, particle filters offer a better fit for the filtering task.

As mentioned earlier, the level of precision in sea surface vehicles' motion control depends primarily on the operational demands, while being influenced significantly by the environmental forces. Figure 4.9 delivers a general overview of the modern motion control system, which consists primarily of the navigation and signal processing module, guidance module and control module [Fossen, 1994; Fossen, 2002]. The trajectory generation largely depends on the operator's requirement. In this study, a decision support system for the last-minute collision avoidance at sea is of interest. Therefore, vessels are assumingly following their passage plans and the guidance system will engage only after the collision avoidance sequence is engaged. Following the collision alerts and taking in consideration other parameters of the system dynamics, the guidance systems needs to generate trajectory that will safely avoid collision, grounding, allusion with minimum deviation from the planned route taking ship's actual motion dynamics that will allow operators to execute proposed maneuvers. The controlling module can deal with heading control, position control and even pitch and roll motion damping. In the proposed model, classic vehicles with propeller and rudder control are covered; therefore, main focus is heading control, while for positioning a classic position fixing method is used, as there are no actuators for automatic position fixing assumed. However, even though positioning and pitch and roll damping is not a part of the proposed model, these could be easily integrated for sea surface vehicles requiring dynamic positioning. Finally, the navigation model takes the focus of navigator's attention as it provides reliable measurements of position and heading by collecting data from GNSS (GPS/DGPS, GLONASS, GALLILEO, etc.), speed logs, RADAR and ARPA, gyrocompasses, echo sounders and accelerometers. Not many ocean-going sea surface vehicles carry accelerometers onboard, but considering their fairly low price, integrating the inertial navigational system in the existing bridge layout is not a difficult or expensive task.

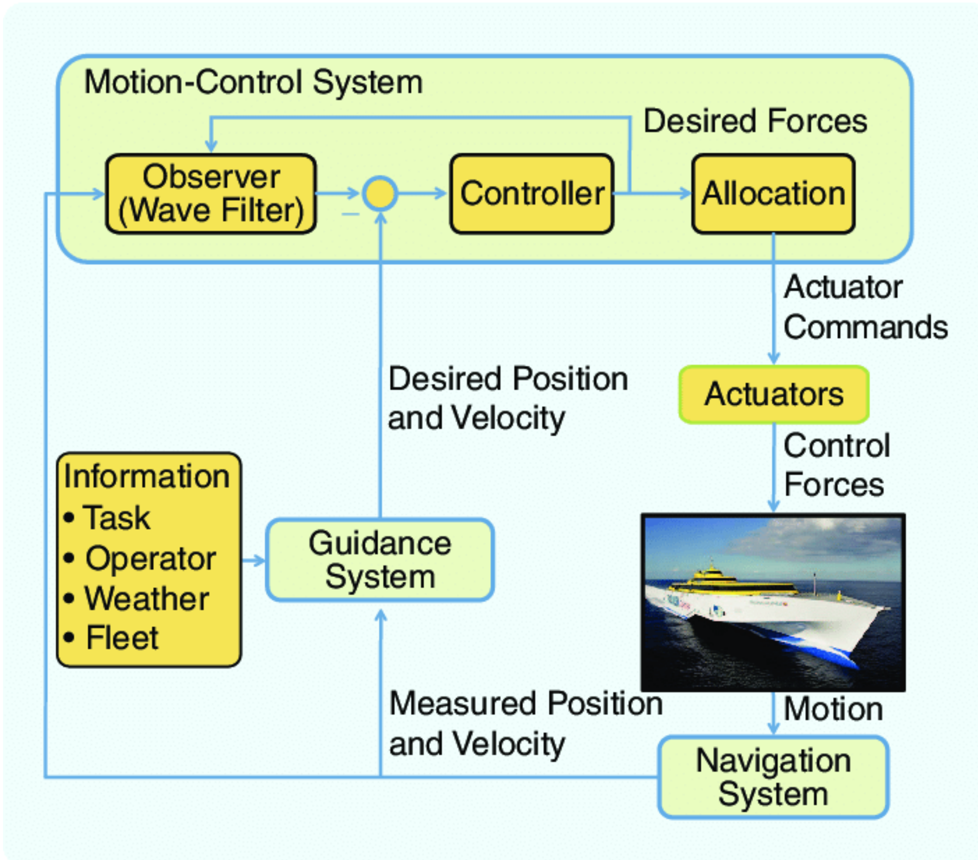


Figure 3.9 – Basic components of a modern ship motion-control system (Source: Fossen, 2002)

Wind, waves, and currents are considered stochastic environmental forces disturbing the motion control system, which are separated in wave- and low-frequency components [Fossen, 2002]. Once reaching the hull of the sea surface vehicle, waves create pressure changes on the hull surface that will result in pressure-induced forces. Waves that depend linearly on the wave elevation have the same frequency as the waves and are called wave-frequency forces [Fossen, 2002]. However, there are also nonlinear wave forces that exist due to quadratic dependence of the pressure on the fluid-particle velocity and have both higher and lower frequencies than the wave frequencies [Newman, 1977; Faltinsen, 1990]. The high frequency wave forces (especially at the sum of all wave frequencies) are usually too high to be considered for the motion control, so it is necessary to filter out these frequencies in order to ensure stability of the motion control system, preserve actuators and reduce environmental emissions.

Similarly like wave induced forces, current and wind induce forces on the ship's hull. Wind forces consist of mean forces and random components due to gusts. Only mean forces are considered, as gusts are not compensated for due to higher frequencies [Fossen, 2002]. Current induced forces are considered low frequency with changes when the current speed and/or direction varies; however, these changes are easier to predict than waves and winds. Therefore, low frequency forces such are wind, currents, and nonlinear waves are considered in proposed motion control model, as ride control (pitch and roll control) is out of scope for collision avoidance algorithm.

The task of maintaining the proper heading, or ultimately course over ground largely depends on the sea state. Even the most experienced human helmsmen will struggle with determining what is the current heading of the vessel when the sea state is above medium and high. The gyro output will produce high nonlinear oscillations and the helmsman will have difficulties in determining how much to compensate with the rudder to keep the correct heading. The human approach is to linearize the process and try to find the mean value and see if there is any lead in the force on either port or starboard side in order to apply more counter-movement rudder. However, this is a difficult endeavor and usually results with overcorrecting, as it is hard to estimate behavior without filtering. The aim is to aid this process with designing a particle filter that separates the high frequency wave motion from the low frequency motions and gives a corrected input to the controller. The particle filter is taking input from multiple sensors and provides an estimate of velocities in the applicable degrees of freedom. The advent of GNSS and gyrocompass allowed us to design support systems that rely on position measurement and heading information. These measurements are used in the motion control system to function in three degrees of freedom; surge, sway, and yaw.

3.2.2 Modeling of sea surface vehicle dynamics

3.2.2.1 Kinematic motion model

In order to develop full mathematical model of vessels' motion, it is necessary to start with the kinematic description of vessels' motion by utilizing two reference frames: an Earth-fixed frame and a body-fixed frame. If the earth-fixed north and east position (n, e) and yaw (heading) ψ relative to the north is expressed in the vector form $\boldsymbol{\eta} = [n, e, \psi]^{\mathrm{T}}$, and if surge u and sway v velocities, together with the yaw rate r that are intrinsic to the body-fixed frame, are represented in the vector form $\boldsymbol{\nu} = [u, v, r]^{\mathrm{T}}$, it is possible to express the transformation between earth-fixed and body-fixed vectors as (Fossen, 2002):

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{3.24}$$

In this research, model of a conventional ocean-going vehicle that is not equipped with actuators correcting roll and pitch is developed; therefore, pitch and roll modes in the proposed controlling framework are omitted. Assumption is made that own sea surface vehicle is stable and has positive metacentric height, so there will be no permanent roll or pitch moments, but only oscillatory ones. This will allow for expressing the whole kinematic equation of motion as a rotation matrix of yaw:

$$\boldsymbol{R}(\psi) = \begin{bmatrix} cos\,\psi & -sin\,\psi & 0 \\ sin\,\psi & cos\,\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.25}$$

where $\boldsymbol{R}^{-1}(\psi) = \boldsymbol{R}^{\mathrm{T}}(\psi)$.

It is imperative to mention that surge u and sway v velocities are time derivatives of the position of the origin of the body-fixed frame relative to the origin of the earth-fixed frame expressed in the body-fixed frame, while the yaw rate r is a component of the angular velocity of the body-fixed frame with respect to the earth-fixed frame, also expressed in the body-fixed frame [Fossen, 2002]. Figure 3.10 depicts all degrees of freedom for a sea surface vehicle.

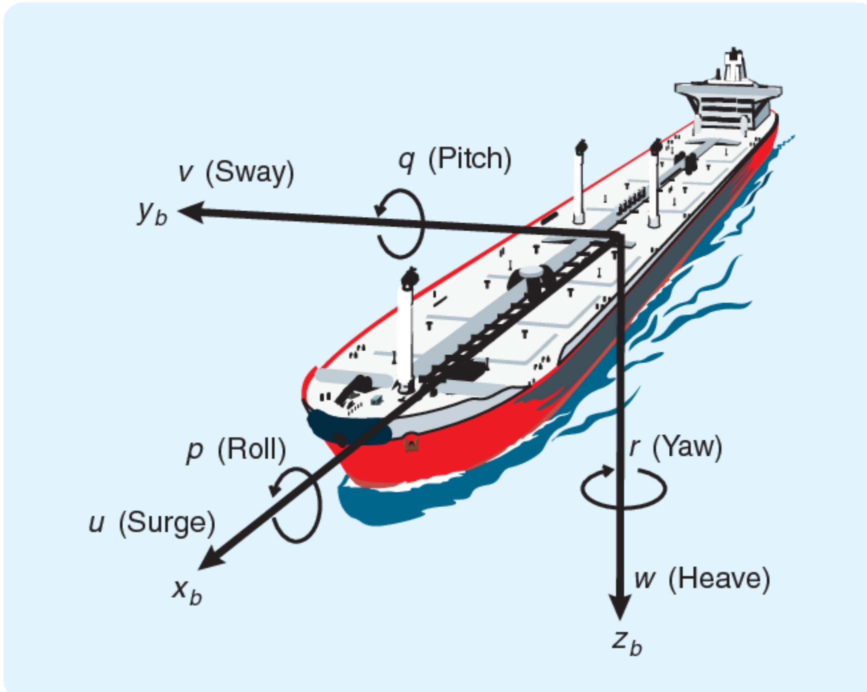


Figure 3.10 – Motion variables for a marine vessel (Source: Fossen, 2002)

3.2.2.2 Low-frequency sea surface vehicle model

Considering the kinematic expressions (3.24) and (3.25), the sea surface vehicles' dynamics could be described as [Fossen, 1994]:

$$(\boldsymbol{M}_{RB} + \boldsymbol{M}_A)\dot{\boldsymbol{v}} + \boldsymbol{C}_{RB}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{d}(V_{rc}, \gamma_c) = \boldsymbol{\tau}_{control} + \boldsymbol{\tau}_{wind} + \boldsymbol{\tau}_{waves} \tag{3.26}$$

On the right side of the expression (3.26) vectors of forces due to control, wind and waves are presented, which belong to the body-fixed frame. $\boldsymbol{\tau} = [X, Y, N]^{\mathrm{T}}$, where X represents the surge force, the Y represents the sway force, while N is the yaw moment. The $\boldsymbol{M}_{RB}$ represents rigid-body mass matrix. $\boldsymbol{C}_{RB}(\boldsymbol{v})$ stands for skew-symmetric Coriolis-centripetal matrix, which is a consequence of expressing the motion equations in body-fixed frame. The $\boldsymbol{M}_A$ is the positive-definite hydrodynamic added mass matrix that appears as vessel moves in the water and causes the pressure on the hull to be proportional to the velocities and accelerations of the sea surface vehicle relative to the fluid. Therefore $\boldsymbol{M}_A$

contains forces that describe change in momentum in the fluid caused by the vessel accelerations. These are defined by:

$$\boldsymbol{M}_{RB} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_g \\ 0 & mx_g & I_z \end{bmatrix}$$

$$\boldsymbol{C}_{RB}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix}$$

$$\boldsymbol{M}_A = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix} \tag{3.27}$$

where x_g represents the longitudinal center of gravity relative to the body-fixed frame, while added-mass coefficients $X_{\dot{u}}$, $Y_{\dot{v}}$, $Y_{\dot{r}}$, $N_{\dot{v}}$ and $N_{\dot{r}}$ depend on the hull shape.

Remaining term on the left side of the equation (3.26) denotes the current and damping factors $\boldsymbol{d}(V_{rc}, \gamma_c)$, which reflect the transfer of energy from the sea surface vehicle to the fluid. These factors depend on the speed and direction of the current relative to the vessel and they are calculated using the following expressions [Fossen, 2002]:

$$V_{rc} = \sqrt{u_{rc}^2 + v_{rc}^2} = \sqrt{(u - u_c)^2 + (v - v_c)^2} \tag{3.28}$$

$$\gamma_{rc} = -\mathrm{atan2}(v_{rc}, u_{rc}) \tag{3.29}$$

where u_c and v_c are defined as components of the current velocity in the body-fixed frame, while γ_{rc} represents the angle of the current relative to the bow of the sea surface vehicle.

It is possible to express surge, sway, and yaw current functions as nondimensional coefficients $C_{X_c}(\gamma_{rc})$, $C_{Y_c}(\gamma_{rc})$, $C_{N_c}(\gamma_{rc})$ and get [Fossen, 1994]:

$$\boldsymbol{d}(V_{rc},\gamma_{rc})=\frac{1}{2}\rho V_{rc}^{2}\begin{bmatrix}A_{F_c}C_{X_c}(\gamma_{rc})\\A_{L_c}C_{Y_c}(\gamma_{rc})\\A_{L_c}\,LOA\;C_{N_c}(\gamma_{rc})\end{bmatrix}\tag{3.30}$$

where LOA is Length Over All, A_{F_c} and A_{L_c} are frontal and lateral projected areas of the submerged part of the hull, and ρ is density of the water. The current coefficients could be determined experimentally using scale models or by utilizing fluid dynamics principles [Fossen, 2002]. However, unless the sea surface vehicles is subjected to extensive hydrodynamic analysis, it will be very hard to determine current coefficients accurately; therefore simplification of the model using linear damping term and bias term [Fossen and Strand, 1999] is usually utilized to develop the approximation:

$$\boldsymbol{d}(V_{rc},\gamma_{rc})\approx\boldsymbol{D}\boldsymbol{v}-\boldsymbol{R}^{\mathrm{T}}(\psi)\boldsymbol{b}\tag{3.31}$$

where

$$\boldsymbol{D}=\boldsymbol{D}^{T}=\begin{bmatrix}D_{11}&0&0\\0&D_{22}&D_{23}\\0&D_{32}&D_{33}\end{bmatrix}\qquad\boldsymbol{b}=\begin{bmatrix}b_1\\b_2\\b_3\end{bmatrix}\tag{3.32}$$

Taking in consideration linear damping and slowly varying bias, simplified sea surface vehicle model is:

$$\boldsymbol{M}\dot{\boldsymbol{v}}+\boldsymbol{D}\boldsymbol{v}=\boldsymbol{\tau}+\boldsymbol{R}^{\mathrm{T}}(\psi)\boldsymbol{b}\tag{3.33}$$

$$\boldsymbol{\tau}=\boldsymbol{B}_u\boldsymbol{u}\tag{3.34}$$

where $\boldsymbol{\tau}=[X\ Y\ Z]^{\mathrm{T}}\in\mathbb{R}^{\mathbf{3}}$ now denotes a vector that contains the control forces and moments generated by the propulsion system (in this case a propeller), while $\boldsymbol{b}\in\mathbb{R}^{\mathbf{3}}$ represents a vector of non-modeled external forces and moments caused by the effects of wind, waves and currents related to the earth-fixed frame. $\boldsymbol{B}_u\in\mathbb{R}^{3xr}$ represents a constant matrix that gives a transmission between the input and the thrust or in other words, describes actuator configuration, while $\boldsymbol{u}\in\mathbb{R}^{\boldsymbol{r}}$ $(r\leq 3)$ denotes the control inputs. Considering the development of (3.28), $\boldsymbol{M}=\boldsymbol{M}_{RB}+\boldsymbol{M}_{A}$; however, a small Froude number is assumed, the inertia matrix $\boldsymbol{M} \in \mathbb{R}^{3x3}$, which contains the mass of the sea surface vehicle and additional hydrodynamic inertia is given by (Fossen, 1994):

$$\boldsymbol{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - N_{\dot{v}} & I_z - N_{\dot{r}} \end{bmatrix} \tag{3.35}$$

where m is the mass of sea surface vehicle, while I_z represents the moment of inertia about the z-axis of the body-fixed frame. Considering the control model where low-frequency motions are dominant, it is possible to assume that frequency is independent of the added inertia. This leads to $\dot{\boldsymbol{M}} = 0$.

For a straight-line stable sea surface vehicle, it is possible to assume that $\boldsymbol{D} \in \mathbb{R}^{3x3}$ is a strictly positive damping matrix because of a linear wave damping and laminar skin friction; therefore, the linear damping matrix is expressed as [Fossen, 1994]:

$$\boldsymbol{D} = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & mu_0 - Y_r \\ 0 & -N_v & mx_G u_o - N_r \end{bmatrix} \tag{3.36}$$

where the assumption is made that the speed of the sea surface vehicle is $u_o > 0$ when moving forward. At this moment, DP vessels are not considered and moving astern for the scope of collision avoidance is not of interest; however, by introducing the auto-telegraph later in text, proposed model can easily be adapted to include astern movements for collision avoidance as well. Naturally, damping forces are nonlinear, but linear damping is a good assumption if we consider a constant speed at cruising and within the sampling interval.

3.2.2.3 Bias modeling

Similarly as with current forces, the wind forces could be modeled by nondimensional force coefficients [Fossen, 2002]:

$$\boldsymbol{\tau}_{wind} = \frac{1}{2}\rho_a V_{rw}^2 \begin{bmatrix} A_{F_w} C_{X_w}(\gamma_{rw}) \\ A_{L_w} C_{Y_w}(\gamma_{rw}) \\ A_{L_w}\, LOA\; C_{N_w}(\gamma_{rw}) \end{bmatrix} \tag{3.37}$$

where ρ_a represents the density of air, A_{F_w} and A_{L_w} denote frontal and lateral projected wind areas, LOA remains the Length Over All, while V_{rw} (wind speed) and relative wind direction γ_{rw} are calculated utilizing following expressions:

$$V_{rw} = \sqrt{u_{rw}^2 + v_{rw}^2} \tag{3.38}$$

$$\gamma_{rw} = -\mathrm{atan2}(v_{rw}, u_{rw}) \tag{3.39}$$

together with:

$$u_{rw} = u - \underbrace{V_w \cos\beta_w}_{u_w} \tag{3.40}$$

$$v_{rw} = v - \underbrace{V_w \sin\beta_w}_{v_w} \tag{3.41}$$

where V_w and β_w represent speed and direction of the wind relative to the earth-fixed frame. Similarly as with currents, the wind coefficients could be obtained either through computational fluid dynamics or model tests. However, for motion control purpose, wind and current speed and directions measurements are used for approximate feedforward compensation coupled with the feedback low level control, while errors associated with this compensation are expressed as bias.

Assuming that bias forces in sway and surge, as well as the yaw moment are all slowly varying, environmental bias can be modelled as Markov process of the first order:

$$\dot{\boldsymbol{b}} = -\boldsymbol{T}^{-1}\boldsymbol{b} + \boldsymbol{\phi}\boldsymbol{\omega} \tag{3.42}$$

where $\boldsymbol{b} \in \mathbb{R}^3$ is a vector of bias forces, $\boldsymbol{T} \in \mathbb{R}^{3x3}$ is a diagonal matrix of positive bias time constants, $\boldsymbol{\phi} \in \mathbb{R}^{3x3}$ is a diagonal matrix scaling the amplitude of $\boldsymbol{\omega}$, while $\boldsymbol{\omega} \in \mathbb{R}^3$ is a vector of zero-mean Gaussian white noise. This model can describe all slow varying environmental forces caused by 2nd order waves, currents, and winds (omitting gusts).

3.2.2.4 High-frequency sea surface vehicle model

In this thesis second order wave model is used to describe the ship motion caused by the first order wave force. As mentioned earlier, wave forces present a sum of linear and nonlinear components. The linear component is oscillatory in nature and has the same frequency as the wave elevation. The nonlinear part, however, has both lower and higher frequency than the wave elevation. Only the lower frequency nonlinear and linear waves are considered in the motion model. While lower frequency nonlinear waves are modelled by a bias term, linear wave forces are usually transformed into an equivalent output disturbance. In order to present the second order model in state-space framework, linear wave frequency is presented in the following form:

$$\dot{\boldsymbol{\xi}} = \boldsymbol{\Omega}\boldsymbol{\xi} + \boldsymbol{\Sigma}\boldsymbol{\omega} \tag{3.43}$$

$$\boldsymbol{\eta}_{\omega} = \boldsymbol{\Gamma}\boldsymbol{\xi} \tag{3.44}$$

where $\boldsymbol{\xi} \in \mathbb{R}^3$ is the wave force state vector, $\boldsymbol{\omega} \in \mathbb{R}^3$ represents zero-mean Gaussian white noise of the wave force model excitations, while $\boldsymbol{\Omega}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{\Gamma}$ denote constant matrices of appropriate dimensions. The first order wave induced motion $\boldsymbol{\eta}_{\omega} = [n_{\omega}, e_{\omega}, \psi_{w}]^{\mathrm{T}}$ is added to the low frequency motion components of the sea surface vehicle making it the sum of the low frequency motion components and the wave frequency motion components, which is evident in Figure 3.11.

In order to advance accuracy of the model, appropriate approximation of the wave spectrum should be used. To sustain computational agility, 2nd order wave model is used for the 1st order wave induced motion, which was originally proposed by Balchen et al. [1976] and improved by Sælid et al. [1983]. Based on their findings, this model can be expressed for each degree of freedom as:

$$h_w^i(s) = \frac{K_{wi}s}{s^2 + 2\zeta_i w_{0i} s + w_{0i}^2} \tag{3.45}$$

where ζ_i $(i = 1...3)$ is the relative damping ratio, w_{0i} $(i = 1...3)$ is the dominating wave frequency, while K_{wi} $(1...3)$ presents a parameter related to the wave intensity. Taking into account 3 degrees of freedom, a state-space expression of (4.45) would be:

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_1 \\ \dot{\boldsymbol{\xi}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3x3} & \boldsymbol{I}_{3x3} \\ \boldsymbol{\Omega}_{21} & \boldsymbol{\Omega}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3x3} \\ \boldsymbol{\Sigma}_{3x3} \end{bmatrix} \boldsymbol{\omega}_w \tag{3.46}$$

$$\boldsymbol{\eta}_w = [\mathbf{0}_{3x3} \quad \boldsymbol{I}_{3x3}] \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix}$$

where, considering that proposed model has three degrees of freedom, resulting second-order noise filter approximation contains state vectors with six components $\boldsymbol{\xi}_1 = [\int n_1 dt, \int e_1 dt, \int \psi_1 dt, n_1, e_1, \psi_1]^{\mathrm{T}}$ and $\boldsymbol{\xi}_2 = [\int n_2 dt, \int e_2 dt, \int \psi_2 dt, n_2, e_2, \psi_2]^{\mathrm{T}}$. $\boldsymbol{\Omega}_{21} = -\mathrm{diag}\{w_{01}^2, w_{02}^2, w_{03}^2\}$, $\boldsymbol{\Omega}_{22} = -\mathrm{diag}\{2\xi_1 w_{01}, 2\xi_2 w_{02}, 2\xi_3 w_{03}^2\}$, $\boldsymbol{\Sigma}_{3x3} = diag\{K_{w1}, K_{w2}, K_{w3}\}$, while $\boldsymbol{\omega}_w = [\omega_n, \omega_e, \omega_\psi]^{\mathrm{T}}$ is the zero-mean Gaussian white noise.

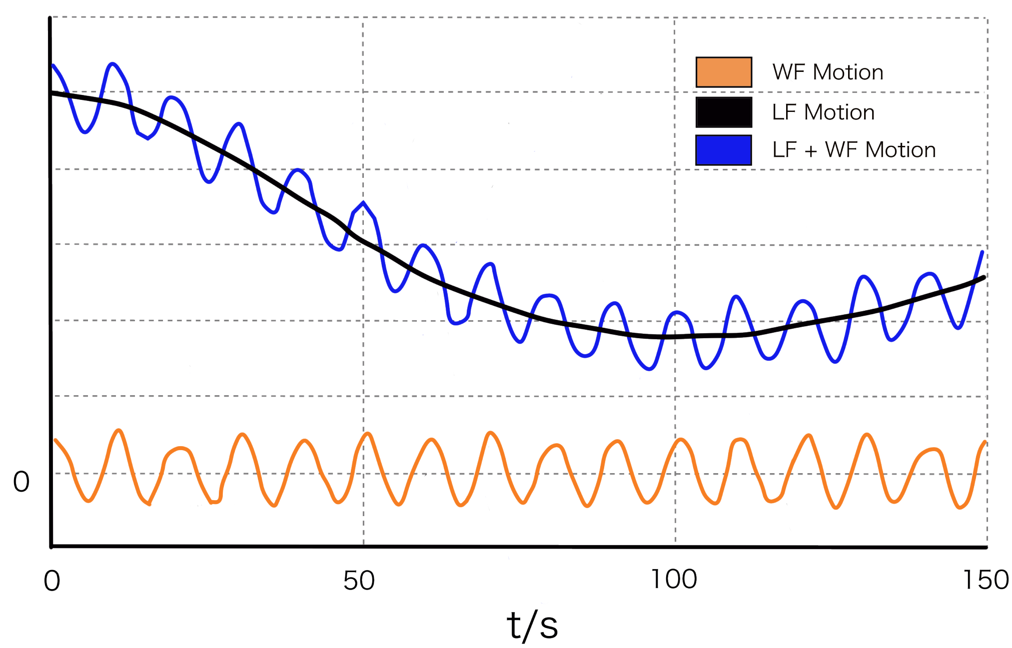


Figure 3.11 – Sum of wave frequency and low frequency motion components

3.2.2.5 Measurement model

A conventional and commercial surface vehicle will only have position and heading measurements available. Some of the vehicles will also have rate of turn measurement available and other will also have Inertial Measurement Units. However, in this thesis focus remains on the under-actuated vehicles that usually achieve thrust by propeller and execute turns by rudder. Considering that the dynamic positioning is out of scope of this research, control of sway and surge could be disregarded, but as it can be seen in the following chapters, improvement of control and reducing uncertainties by using IMUs is considered, so proposed measurement and control model could still include everything required to dynamically control motion of vehicles. Majority of the vehicles use GPS / Differential GPS to estimate position and gyrocompasses to measure heading. Error rate and uncertainty will depend on several factors; these are going to be discussed later in text. Measurement level can, therefore, be defined as:

$$\boldsymbol{y} = \boldsymbol{\eta} + \boldsymbol{\eta}_{\omega} + \boldsymbol{\nu} \tag{3.47}$$

where $\boldsymbol{\eta}_{\omega}$ is the vehicle's wave frequency motion due to the first-order wave induced disturbance, while $\boldsymbol{\nu} \in \mathbb{R}^3$ is a zero-mean Gaussian white measurement noise. Additionally, to ensure the ship observer can compute control forces in sway, surge and moments of yaw, actuator measurements $\boldsymbol{u}$ are required**,** where assumption is made that $\boldsymbol{B}_u$ matrix is known:

$$\boldsymbol{\tau} = \boldsymbol{B}_u \boldsymbol{u} \tag{3.48}$$

For under-actuated surface vehicles $\boldsymbol{\tau}$ is certainly going to be simplified. State estimator is used to distinguish the low frequency motion components $\boldsymbol{\eta}$ from the noisy measurements. It is imperative that the wave frequency caused by the first-order wave-induced disturbance does not enter in the feedback loop, as it will increase fuel consumption, cause deterioration of the actuators and negatively impact the environment.

3.2.2.6 Resulting system model

Taking into account all expressions so far, total system can be presented as follows:

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_1 \\ \dot{\boldsymbol{\xi}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3x3} & \boldsymbol{I}_{3x3} \\ \boldsymbol{\Omega}_{21} & \boldsymbol{\Omega}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3x3} \\ \boldsymbol{\Sigma}_{3x3} \end{bmatrix} \boldsymbol{\omega}_w$$

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{D}\boldsymbol{\nu} = \boldsymbol{\tau} + \boldsymbol{R}^{\mathrm{T}}(\psi)\boldsymbol{b} + \boldsymbol{\omega}_\nu \tag{3.49}$$

$$\dot{\boldsymbol{b}} = -\boldsymbol{T}^{-1}\boldsymbol{b} + \boldsymbol{\phi}\boldsymbol{\omega}_b$$

$$\boldsymbol{y} = \boldsymbol{\eta} + \boldsymbol{\eta}_w + \boldsymbol{\nu}$$

The resulting 15th order state-space model for an observer design that includes dynamic positioning and heading control is:

$$\dot{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{E}\boldsymbol{\omega} \tag{3.50}$$

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\nu}$$

where $x = [\boldsymbol{\xi}^{\mathrm{T}}, \boldsymbol{\eta}^{\mathrm{T}}, \boldsymbol{b}^{\mathrm{T}}, \boldsymbol{\nu}^{\mathrm{T}}] \in \mathbb{R}^{15}$ is state vector, $\boldsymbol{u} = [n_p \quad e_p \quad \psi_p]^{\mathrm{T}} \in \mathbb{R}^p$ $(p \geq 3)$ is considered a controllable vector, while $\boldsymbol{\omega} = [\omega_1^{\mathrm{T}}, \omega_2^{\mathrm{T}}, \omega_3^{\mathrm{T}}]^{\mathrm{T}}$ denotes vector of process noise. $\boldsymbol{y} \in \mathbb{R}^3$ is the measurement vector that denotes DGPS and gyrocompass measurement mixed with measurement noise where $\boldsymbol{x} \in \mathbb{R}^3$ denotes the state vector of the system. $\boldsymbol{\nu} \sim N(\mathbf{0}, \boldsymbol{R})$ is the measurement noise vector with covariance matrix $\boldsymbol{R}$. State (system) matrix $\boldsymbol{A}$, input matrix $\boldsymbol{B}$, process noise amplitude matrix $\boldsymbol{E}$, and output matrix $\boldsymbol{H}$ are defined as (Fossen, 2002):

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_w & \mathbf{0}_{6x3} & \mathbf{0}_{6x3} & \mathbf{0}_{6x3} \\ \mathbf{0}_{3x6} & \boldsymbol{I}_{3x3} & \mathbf{0}_{3x3} & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x6} & \mathbf{0}_{3x3} & \mathbf{0}_{3x3} & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x6} & -\boldsymbol{M}^{-1}\boldsymbol{D} & \boldsymbol{M}^{-1} & \mathbf{0}_{3x3} \end{bmatrix} \tag{3.51}$$

$$\boldsymbol{B} = \begin{bmatrix} \mathbf{0}_{6x3} \\ \mathbf{0}_{3x3} \\ \mathbf{0}_{3x3} \\ \boldsymbol{M}^{-1}\boldsymbol{B}_u \end{bmatrix} \tag{3.52}$$

$$\boldsymbol{E} = \begin{bmatrix} \boldsymbol{E}_w \\ \mathbf{0}_{3x3} \\ \boldsymbol{I}_{3x3} \\ \boldsymbol{M}^{-1} \end{bmatrix} \tag{3.53}$$

$$\boldsymbol{H} = [\boldsymbol{C}_w \quad \boldsymbol{I}_{3x3} \quad \mathbf{0}_{3x3} \quad \mathbf{0}_{3x3}] \tag{3.54}$$

$$A_w = \begin{bmatrix} \mathbf{0}_{3x3} & \boldsymbol{I}_{3x3} \\ \boldsymbol{\Omega}_{21} & \boldsymbol{\Omega}_{22} \end{bmatrix} \qquad \boldsymbol{E}_w = \begin{bmatrix} \mathbf{0}_{3x3} \\ \boldsymbol{\Sigma}_{3x3} \end{bmatrix} \qquad \boldsymbol{C}_w = [\mathbf{0}_{3x3} \quad \boldsymbol{I}_{3x3}] \tag{3.55}$$

Finally, in order to utilize the above-defined observer on a computer, discretization of the following form is required:

$$\boldsymbol{x}(k+1) = \boldsymbol{\Phi}\boldsymbol{x}(k) + \boldsymbol{\Lambda}\boldsymbol{u}(k) + \boldsymbol{\Theta}\omega(k) \tag{3.56}$$

$$\boldsymbol{y}(k) = \boldsymbol{H}\boldsymbol{x}(k) + \nu(k) \tag{3.57}$$

where

$$\boldsymbol{\Phi} = \exp(\boldsymbol{A}h) \tag{3.58}$$

$$\boldsymbol{\Lambda} = \boldsymbol{A}^{-1}(\boldsymbol{\Phi} - \mathbf{I})\boldsymbol{B} \tag{3.59}$$

$$\boldsymbol{\Theta} = \boldsymbol{A}^{-1}(\boldsymbol{\Phi} - \mathbf{I})\boldsymbol{E} \tag{3.60}$$

with h denoting the sample time. Sampling time depends on the vessel dynamics. In the proposed model, where focus is maintained on yaw rate only, sampling rate will depend on the fastest rate of turn achieved when looking through the prism of force to velocity response.

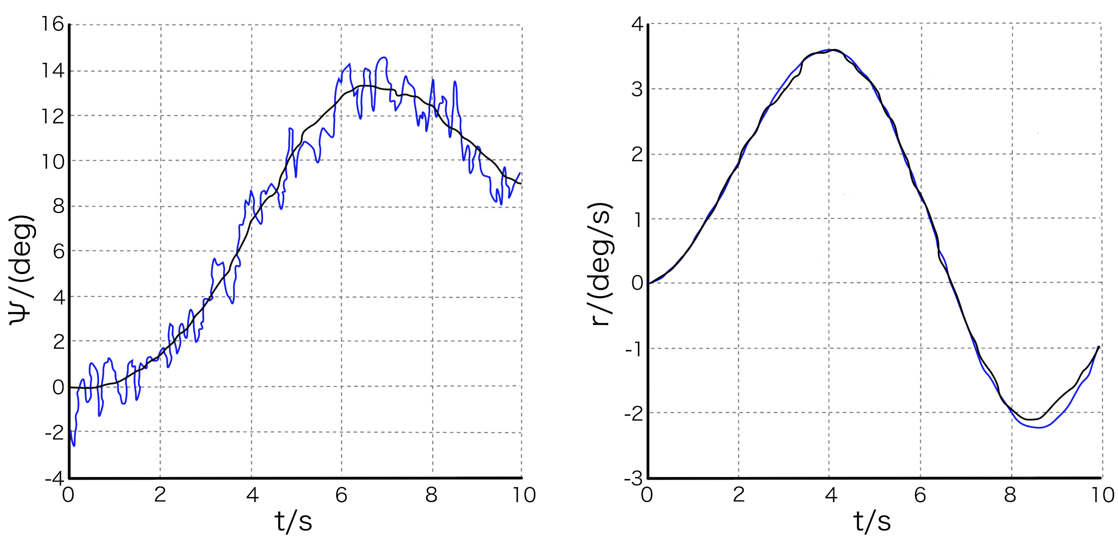
3.2.3 Simulation of nonlinear observer design

In this research intention was to use data for the own vehicle used in collision avoidance simulations, however due to unavailability of all required data from the owner, different scale-down model is used to determine influence of environmental forces on sway, surge and yaw [Fossen, 2002]. To assess the fitness of the designed filter, simulations were done with the following parameters: LOA=1.19 m, the mass m = 17.6 kg, and:

$$\boldsymbol{M} = \begin{bmatrix} 25.8 & 0 & 0 \\ 0 & 33.8 & 1.0115 \\ 0 & 1.0115 & 2.76 \end{bmatrix}, \quad \boldsymbol{D} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 7 & 0.1 \\ 0 & 0.1 & 0.5 \end{bmatrix}$$

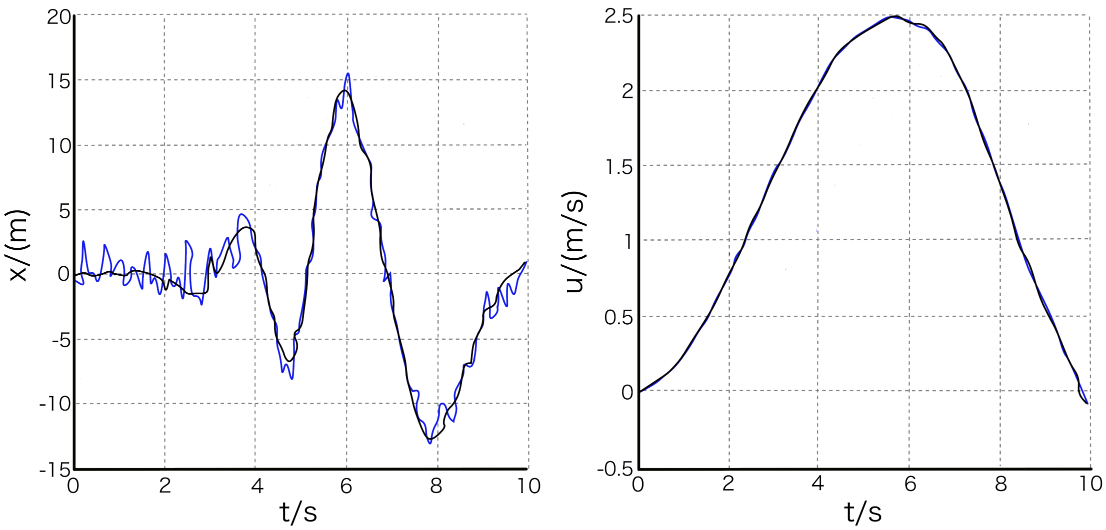
The bias time constants were set at $T_b = diag(100{,}100{,}100)$. The wave model parameters were set as $\varsigma = diag(0.1{,}0.1{,}0.1), w_0 = diag(0.8{,}0.8{,}0.8)$. The control inputs were selected as: $\tau = [20sin(0.02t), \quad 20sin(0.03t), \quad 5sin(0.04t)]^T$.

The simulation results are shown in the figures below. Efficacious tracking of the position and velocity is obtained even though the measurements are highly noise corrupted.



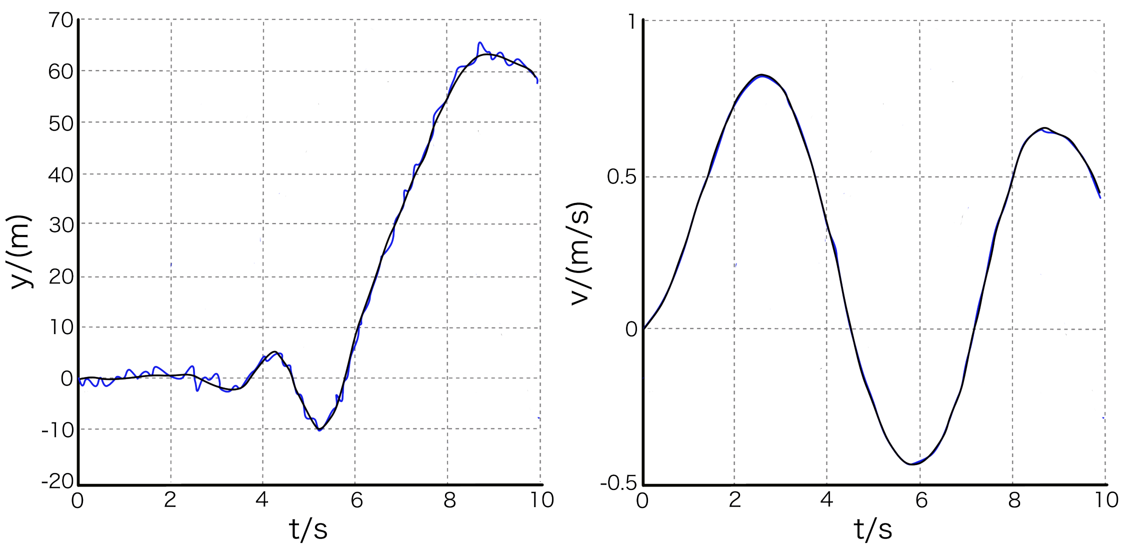
(a) (b)

Figure 3.12 – Simulation results: (a) actual heading angle Ψ in blue color and predicted in black, (b) actual yaw rate r in blue and predicted in black color



(a) (b)

Figure 3.13 – Simulation results: (a) actual position x in blue and estimated in black, (b) actual velocity in blue u and estimated in black



(a) (b)

Figure 3.14 – Simulation results: (a) actual position y in blue and estimated in black, (b) actual velocity in blue v and estimated in black

Even though this is a nonlinear filter for dynamic positioning, only yaw moment can be derived and used for steering control of the vehicle. Both velocity and low-frequency position are successfully computed from noisy position measurements. Nonlinear particle filter can avoid computing Jacobi matrix during the linearization process of Kalman filter designs.

In this chapter a nonlinear passive observer based on foraging particle filter has been designed. Simulation studies have shown that low-frequency position and velocities of the surface vehicle, as well as the environmental disturbances, could be computed from noisy position measurements. First order wave-induced disturbances are successfully filtered. The simulation has shown that all estimation errors converged exponentially to zero.

The main advantage of the nonlinear approach is that the kinematic equations of motion do not require linearization about a set of predefined constant yaw angles (36 operating points with 10 degrees each), while this is required in Kalman Filtering. Also, particle filters do not require tuning, unlike Kalman filters that require tuning of weighting matrices with dimensions 15 x 15 for estimation error and 3 x 3 for control input vectors.

The observer is applicable to both dynamic positioning and tracking control of sea surface vehicles.

3.3 Intelligent autopilot design with Learning from Demonstration

As mentioned earlier, this thesis is focused on underactuated surface vehicles; therefore, focus will be on controlling mainly rudder and telegraph inputs. Most ocean-going vehicles will have rudder limits of 35° to port and starboard side, but the telegraph input varies sparsely. Ownship that is used in collision avoidance examples in later chapters is utilized, but with small adjustments this model is applicable to any surface vehicle. In this study approach is to utilize reinforced learning technique that will allow us that all training is happening offline. As a first step, simulations are done in order to have control actions optimized for the set of inputs. The learning set is updated with observations that are collected during exploitation with a goal of adaptive training and solution sets. As an additional step, supervised training with human helmsman and pilots is included, so that the system dynamic can mimic responses from the human operators, especially when speed is low and larger control inputs are required to stabilize the vehicle in desired heading and/or course.

3.3.1 Primary quantitative data collection

Primary quantitative data collection was conducted utilizing approved and accredited simulator. Dynamics of commercial sea surface vehicles were developed in the Wärtsilä Navi-Trainer Pro NTPRO 4000 (Transas) simulator. Considering that model-free solution is developed, substantial data is required to allow abstract agent's learning of vehicles' dynamics. For this chapter relevant data are environmental loads, such are wind, waves, swell and current, while for other chapters interaction data has been used to construct collision avoidance situations. Simulator was utilized to reconstruct some of the investigated maritime incidents.

After extracting data from the Wärtsilä simulator, numerical and computational tasks were conducted in MATLAB 2021a programming language and numerical computing environment, running on a MacBook Pro with Intel 2.8 GHz Quad-Core Inter Core i7 processor, NVIDIA GeForce GT 750M 2 GB graphics card, 16 GB RAM and macOS Catalina 10.15.4. Each of the presented learning algorithms have specifically tuned parameters for learning, but all of them used Adam Optimizer [Kingma and Ba, 2014] as a stochastic first-order gradient optimization of objective functions.

Each reinforcement learning problem is approached by extracting required data from available sources, then designing simulation spaces in MATLAB, allowing agents to learn, and then learned agents utilized in simulated scenarios. Simulations are crucial part of reinforcement learning, as it would be expensive and unsafe to explore the real-world with commercial sea surface vehicles. That is why agents learn in simulated environments, after which they are utilized in models, or sea trials of the commercial sea surface vehicles, before they are allowed to aid decision making. As simulators have their limitations, agent still needs exposure to the real-world before it is deemed ready for commercial use. Training results are used as an agent's previous knowledge data base in order to expedite convergence of stochastic optimization during simulation and real-world exploitation. Once exploring and exploiting in the real world, the proposed system continues to learn and adapts the learning buffer with new experiences in pursuit of an optimal behavior.

To evaluate motion control solutions proposed in the following chapters of this thesis, 720,214 data particles were extracted describing environmental loads on sea surface vehicles. 202 hours of recorded data were analyzed and incorporated in learning models for selected sea surface vehicles. In order to extrapolate lateral and longitudinal velocities, wind, currents, swell and wind waves interactions with yaw, surge, and sway motions were investigated. Each data extrapolation from the Wärtsilä simulator for each sea surface vehicle was sampled with 1 HZ frequency for a minimum period of 20 minutes. This process was repeated for each of the 594 data collecting points. In addition, various maneuvers were made to capture human steering performance with maintaining headings and courses, as well as to perform zig-zag maneuvers to benchmark steering performance of expert helmsmen that would aid learning and precise turning of proposed auto-pilot model.

A standard 16 compass rose directions were used to extract data and determine lateral and longitudinal velocities of sea surface vehicles in different wind, current, swell and wind wave conditions. The extracted data is used to predict future states of interacting vehicles and to make informed decisions.

As within this research interest remains in motion control of underactuated vehicles, yaw data under the influence of applicable environmental loads were extracted. In order to get in-depth overview of longitudinal and lateral velocities, all vehicles of interest were simulated, and motion effects under environmental loads recorded. Velocities were collected and transferred in lookup tables to simulate chain of observations in reinforcement learning environments. In order to avoid cluttering, the following figures depict samples of collected data for the own vehicle.

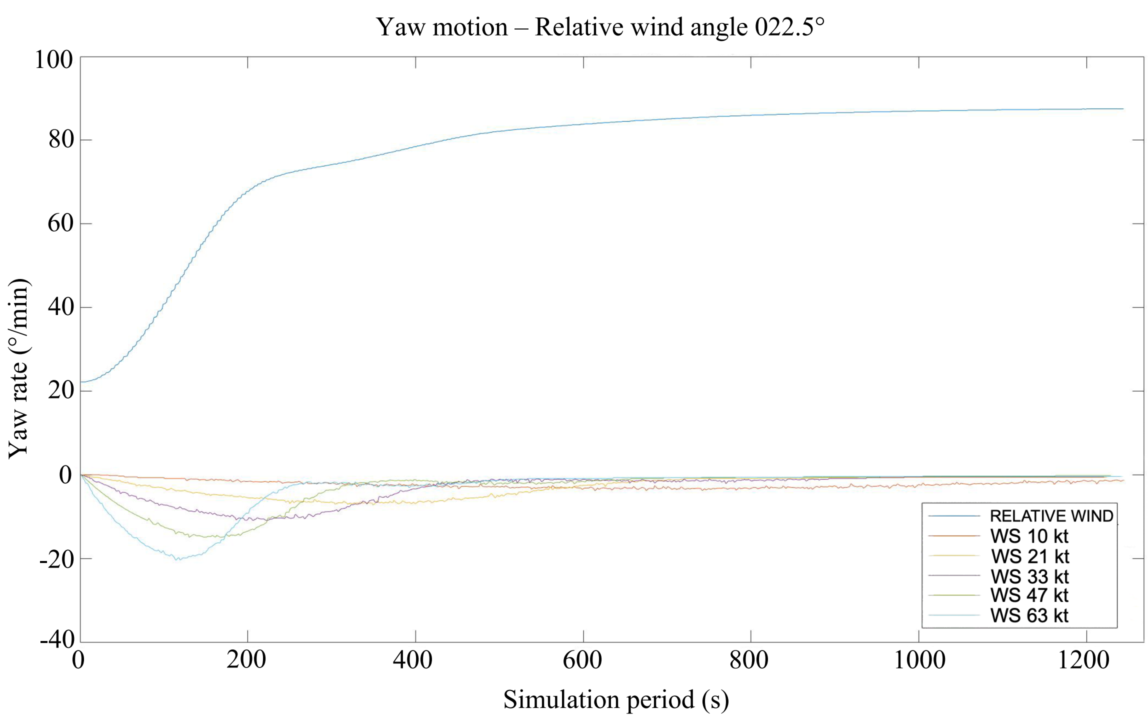


Figure 3.15 – External disturbances sampling (example 1)

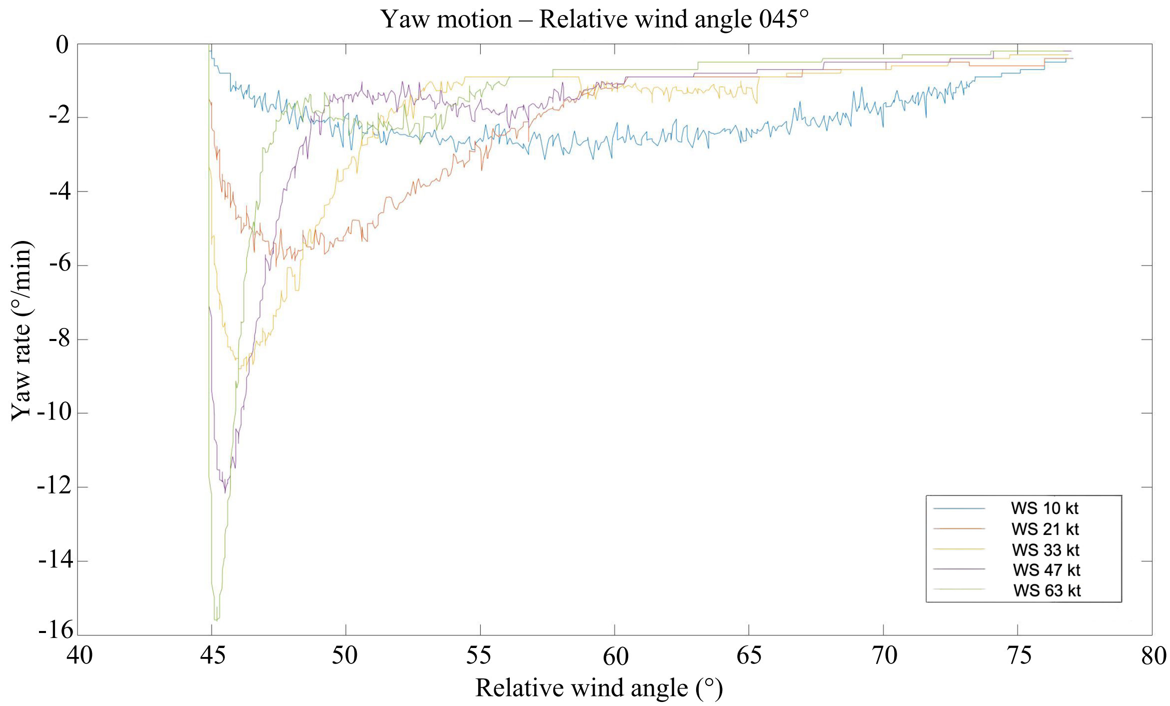


Figure 3.16 – External disturbances sampling (example 2)

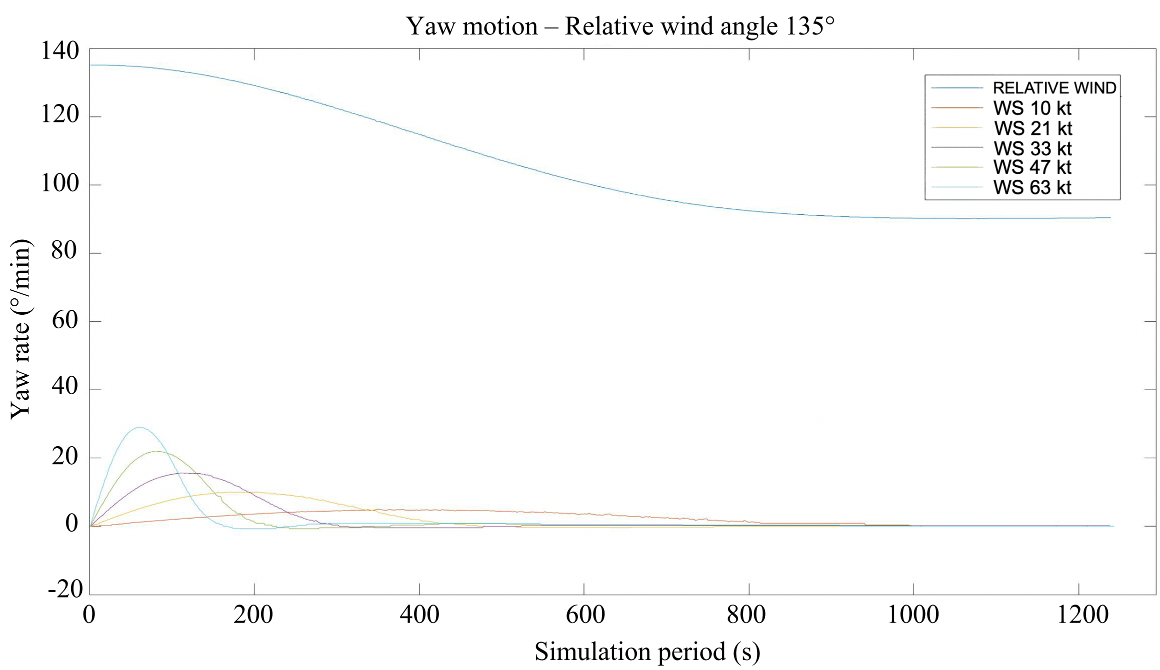


Figure 3.17 – External disturbances sampling (example 3)

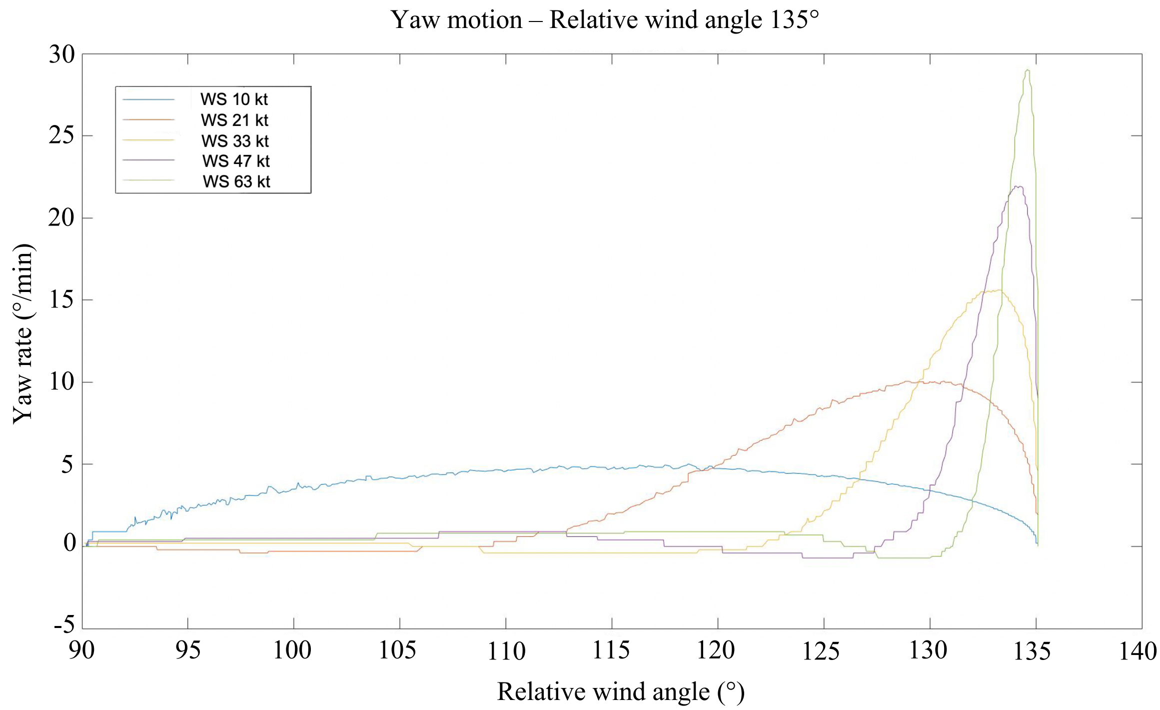


Figure 3.18 – External disturbances sampling (example 4)

Once the data of external disturbances has been evaluated and processed, focus shifted on selecting the best approach to learning.

3.3.2 Reinforcement Learning approach

Automatic piloting is not a novelty; however, considering the various speeds and external disturbances, linearity of the usual control setup with PID controllers and filters can struggle in adapting to dynamic situational circumstances. When designing a controller that can mimic human control inputs and use the wide knowledge specter of the human expert, it is necessary to consider advanced learning techniques during design. The central issue of this approach is to design a sequential decision making when the system dynamics are stochastic in nature. A sea surface vehicle is susceptible to various dynamic and unpredictable forces, where behavior of a surface vehicle is stochastic as well. Even though the under-actuated sea surface vehicle is limited to port-starboard movement of the rudder, it is still a continuous sequential decision-making problem as the agent (human or artificial operator) has to determine how much and in which direction the rudder has to be forced in order to compensate for various disturbances and maintain the desired heading, course or track. The reason why it is necessary to consider this problem as sequential decision making is because a bad decision at one time step can be of low importance to the safety of the considered vehicle but can in the future have a fatal consequence. Aim of this study is the design a combination of model predictive and model-free control that will use filtered inputs to get the best possible information and reduce partial observability of a sea surface vehicle's state, which includes its position and heading in dynamic environment.

Proposed sea surface vehicle control problem is modeled as Markov decision process, for which a reinforcement learning [Sutton and Barto, 1998] is fitting well. Reinforced learning is inspired by biological learning, where agent interacts with a model that is updated in discrete time steps through observing the accessible space of that model and decides on taking the best action available by evaluating all options dependent on both the immediate and cumulative reward this agent receives. The solution is, therefore, not a single action, but a sequence of actions, which is called a policy. Finding the optimal policy can be a challenge in highly dimensional spaces. While designing proposed training algorithm, quality of data should be of great importance, model representation and training fitness. Initial training set based on simulator data is developed, while the agent gets updates in training through exploration during real-world exploitation.

If agent was allowed to explore environment without prior knowledge and without any expert's supervision, convergence of optimization would be difficult. Depending on problem that needs to be solved, several approaches exist. In this case it is possible and desirable to utilize human experts to guide learning by demonstrating appropriate behavior. Human helmsman expert does not have intrinsic knowledge of vessel's dynamics, but solely based on gyro heading and rate of turn indicator manages to perform heading controls and turns efficiently and safely. It is, therefore, safe to assume that building an initial database with human demonstrations is a good way to aid optimization convergence when utilizing autopilot in daily endeavors through rewards shaping. Continuing to the briefing of the RL in the previous chapter, an overview of shaping rewards is delivered.

3.3.3 Rewards shaping in Reinforcement Learning

A carefully defined reward is of crucial importance in sequential decision making. To learn consequence of an immediate action, it would be required to look further in the future. In this thesis it is argued that sequential decision making is beneficial not only for the virtual agents, but also for everyday life. For example, a driver that actively thinks of consequences regularly in his or her life would be driving down a road and approaching bend of that road calculate risk of taking that bend. This means that he would anticipate that there is a possibility of road work or accident he cannot currently see with his eyes, so he would either take the bend slowly, proceed with care and his foot released from accelerator and ready for breaking, or take any other action with care. With properly defined reward function and domain modeling, it is possible to find optimal policy that describes transition from states to states by taking optimal actions. Properly defined reward function is not only a mean to find optimal policy, but also to reduce computational burden for optimization task.

Sometimes it is easy to think of a main reward for a certain task. For example, an agent can be instructed to cross the street and get a large reward for crossing the street. Additional rewards and penalties could be also included so that an agent always uses a crosswalk, to wait for the green lights, to check left and right before crossing, etc. These smaller rewards and penalties can be considered a reward shaping as it aids an agent to learn faster. However, caution should be taken when shaping rewards as inappropriate shaping can induce poor learning outcomes. Learning from mistakes is a powerful learning technique as it narrows the learning path to optimized goal. Therefore, the goal of rewards shaping would be to aid learning agent by utilizing previous knowledge and accelerate convergence to an optimal policy.

Reward shaping is finding its roots in behavioral psychology [Skinner, 1938] and has been used successfully in various domains [Brys, et al., 2015; Dorigo & Colombeti, 1994; Mataric, 1994; Ng, et al., 1999; Randløv and Alstrøm, 1998; Saksida, et al., 1997]. There were also negative examples of rewards shaping [Randløv and Alstrøm, 1998] where the optimized goal was never reached. If an agent that tries to learn maritime collision avoidance is envisioned and rewards are shaped in a way that anytime an agent is in the collision free state gets a reward but does not get penalties for getting away from the goal state, agent would turn the surface vehicle 180 degrees away from any danger, going the opposite way from the goal state. It is important to have a good understanding of the system dynamics to avoid undesired behavior of an agent. This is where learning from demonstration can aid us to avoid specifying a reward explicitly and avoid convergence in local optima or suboptimal convergence. Therefore, it is important to ensure that rewards shaping is not affecting underlying policy, so optimal policy has to be invariant to reward changes.

When shaping a reward function is too difficult and does not converge to optimal policy, Ng and Russell [2000] offer a solution through inverse reinforcement learning. This approach requires extracting features and linear constraints that define optimal policy, which will be used to derive a reward function. Deep reinforcement learning [Minh et al, 2015] uses deep neural networks to deal with optimality issues of reinforcement learning with extensive application in gaming. The real-world applications are still scarce, and to the best of our knowledge this approach is one of the first attempts to avoid policy invariance by combining rewards shaping, deep reinforcement learning and model predictive control to achieve optimal control of underactuated sea surface vehicles.

If reward function is defined in such a way that almost perfectly resembles learning goal, there would be no need for reward shaping. However, it is hard to expect that rewards design would be non-trivial in complex real-world environments. The basic idea of reward shaping is to learn a policy for some $MDP = (S, A, \mathcal{T}, \gamma, R)$ with a possibility to reduce the search space and bound it with shaping reward function $F : S \times A \times S \longmapsto \mathbb{R}$. In order to simplify expressions, transition probability is denoted as $\mathcal{T}$. However, to achieve smooth convergence, it is necessary to search for the optimal policy by utilizing transformed version of the original MDP by applying the shaping reward function and amend the reward function to $R' = R + F$. Therefore, this research is now interested in investigating transformed version of the original MDP, $MDP' = (S, A, \mathcal{T}, \gamma, \mathrm{R}')$. However, when defining reward shaping this way it is necessary to ensure that policy optimization in MDP' is equal as in MDP, which requires equality of $R_{MDP} = (R + F)_{MDP'}$ for the same transition from state s to state s'.

Even though this approach is developed to resolve sea surface vehicles' motion control challenges, as well as to have a viable solution for the collision avoidance, aim is no loss of generality. When modeling real world problems, situations arise when transition probabilities $\mathcal{T}(\cdot)$ and reward functions $R(s, a, s')$ are not known. Even in this setup, objective is to find an optimal policy $\pi^*_{MDP'}$ in transformed MDP that will be equally optimal in the native MDP. Ng et al. [1999] discovered that shaping rewards using difference of potentials is the approach that can be used for all problems that could be modeled as MDP. In this way difference of potentials approach is viable tool for giving advice to a reinforcement learner, but as it does not include action space, it can only provide indication if any of the visited states are good or bad. Therefore, extending on Ng et al. [1999] to include actions to allow for more comprehensive reinforcement learning that considers both states and actions:

Theorem 1 *Assuming S, A, γ, and a shaping reward function F are given, F is a potential-based shaping function if there exists a real-valued function $\Phi : S \times A \longmapsto \mathbb{R}$ such that for all $s \in S - \{s_0\}, a \in A - \{a_0\}, s' \in S, a' \in A$,*

$$F(s, a, s', a') = \gamma\Phi(s', a') - \Phi(s, a), \tag{3.61}$$

where $S - \{s_0\} = S$ and $A - \{a_0\} = A$ if $\gamma < 1$, with requirement that potential-based shaping reward function F contains necessity and sufficiency logic conditions to assure optimal policy consistency weather learning from MDP' or native MDP. Sufficiency is of particular interest as it states that if F is a potential-based shaping function then any optimal policy in MDP' will be optimal in MDP and vice versa. Necessity provides insight where if F is not a potential-based shaping function then there is no optimal policy in MDP' that will be optimal in MDP.

Sufficiency proof: Considering F in the form (3.61) in undiscounted form $\gamma = 1$, then it is possible to replace $\Phi(s, a)$ with $\Phi'(s, a) = \Phi(s, a) - k$ for any constant k, which would not change shaping rewards F. It is, therefore, possible to assume that the Φ used to express F in the form (3.61) satisfies $\Phi(s_0, a_0) = 0$ when $\Phi(s, a)$ is replaced with

$\Phi(s,a) - \Phi(s_0,a_0)$, where s_0 represents zero-reward absorbing state.

As stipulated in [Sutton and Barto, 1998], the optimal Q-function of the native MDP Q^*_{MDP} will satisfy the Bellman Equation as follows:

$$Q^*_{MDP}(s,a) = \mathrm{E}_{s' \sim \mathcal{T}(\cdot)}\left[R(s,a,s',a') + \gamma \max_{a' \in A} Q^*_{MDP}(s',a')\right] \tag{3.62}$$

Introducing the shaping function notations and taking some algebraic treatment, we get:

$$Q^*_{MDP}(s,a) - \Phi(s,a) = \mathrm{E}_{s'}\left[R(s,a,s',a') + \gamma\Phi(s',a') - \Phi(s) + \right.$$
$$\left. \gamma \max_{a' \in A}\left(Q^*_{MDP}(s',a') - \Phi(s')\right)\right] \tag{3.63}$$

Further, it is possible to define $\hat{Q}_{MDP'}(s,a) \triangleq Q^*_{MDP}(s,a) - \Phi(s,a)$ while substituting this and the $F(s,a,s',a') = \gamma\Phi(s',a') - \Phi(s,a)$ expression into the (3.62) to get:

$$\hat{Q}_{MDP'}(s,a) = \mathrm{E}_{s'}\left[R(s,a,s',a') + F(s,a,s',a') + \gamma \max_{a' \in A} \hat{Q}_{MDP'}(s',a')\right] \tag{3.64}$$

$$= \mathrm{E}_{s'}\left[R'(s,a,s',a') + \gamma \max_{a' \in A} \hat{Q}_{MDP'}(s',a')\right] \tag{3.65}$$

which is basically the Bellman equation for MDP'. If considering the undiscounted case, $\hat{Q}_{MDP'}(s_0,a) = Q^*_{MDP}(s_0,a) - \Phi(s_0) = 0 - 0 = 0$. Therefore, it is evident that $\hat{Q}_{MDP'}(s,a)$ satisfies the Bellman equations for MDP' being the uniquely optimal Q-function. Considering that $Q^*_{MDP'}(s,a) = \hat{Q}_{MDP'}(s,a) = Q^*_{MDP}(s,a) - \Phi(s)$, the optimal policy for MDP' satisfies

$$\pi^*_{MDP'} \quad \in \quad \operatorname*{argmax}_{a \in A} Q^*_{MDP'}(s,a) \tag{3.66}$$

$$= \operatorname*{argmax}_{a \in A} Q^*_{MDP}(s,a) - \Phi(s,a) \tag{3.67}$$

$$= \operatorname*{argmax}_{a \in A} Q^*_{MDP}(s,a) \tag{3.68}$$

which also makes it optimal in MDP. By simply reversing the roles of MDP and MDP', the same proof can be utilized and show that every optimal policy in MDP is also optimal in MDP', which completes the proof. □

The proof of necessity has been thoroughly covered in [Ng et al., 1999] and is therefore omitted in this thesis. Empirical benefits and challenges of reward shaping and engineering have been covered by many authors [Mataric, 1994; Ng et al., 1999; Randløv and Alstrøm, 1998].

Before continuing with the experiment, it is necessary to find the appropriate shaping function. Based on the Theorem 1 and deliveries of [Ng et al., 1999], we consider the following Lemma:

Lemma 1 *Considering Theorem 1 and assuming that F takes the form* $F(s,a,s',a') = \gamma\Phi(s',a') - \Phi(s,a)$*, where* $\Phi(s_0) = 0$ *when* $\gamma = 1$*, then for all* $s \in S$*,* $a \in A$*,*

$$Q^*_{MDP'}(s,a) = Q^*_{MDP}(s,a) - \Phi(s,a), \tag{3.69}$$

$$V^*_{MDP'}(s) = V^*_{MDP}(s) - \Phi(s). \tag{3.70}$$

Proof: Theorem 1 covered the proof of the (3.69), while (3.70) is directly derived from the relationship $V^*(s) = \max_{a\in A} Q^*(s,a)$. □

Ng et al. [1999] show that reward shaping holds for arbitrary policies, not only the optimal policy, which makes reward shaping robust for near-optimal policies as well, while maintaining indifference on policy selection. The main task, therefore, remains a selection of Φ, which is based on a collection of expert knowledge about the domain, either through experience, peer reviewed knowledge, or various sensor input that can be used to derive the knowledge. In depicted experiment, an undiscounted case with $\Phi(s_0) = 0$ and $\Phi(s) = V^*_{MDP}(s)$ is presented, which allows usage of (3.70). This tells us that the value function in MDP' is $V^*_{MDP'}(s) \equiv 0$. It is necessary to keep in mind that this is not the only approach in choosing reward shaping function, but it is easy and functional method.

In regards of reinforcement learning, research hypothesis is that reward shaping performs better than no shaping, so at this time SARSA algorithm is used [Sutton and Barto, 1998], which is reward shaping adapted. Unlike the Q-learning, SARSA (Figure 3.19) is an on-policy learning algorithm which is based on a repetitive cycle of agent being in a state $s \in S$, taking an action $a \in A$, for which it receives a reward R, after which it ends up in a new state $s_1 \in S$, and takes action $a_1 \in A$. With presented selections we get the tuple (s, a, R, s_1, a_1) from which the name SARSA is derived. When differentiating on-policy and off-policy approaches, focus is mainly on whether the update of the policy is based on actions taken, or not. In this case, the action taken drives the update of the policy. This is also the case in collision avoidance, where action selection becomes one of the central challenges to solve.

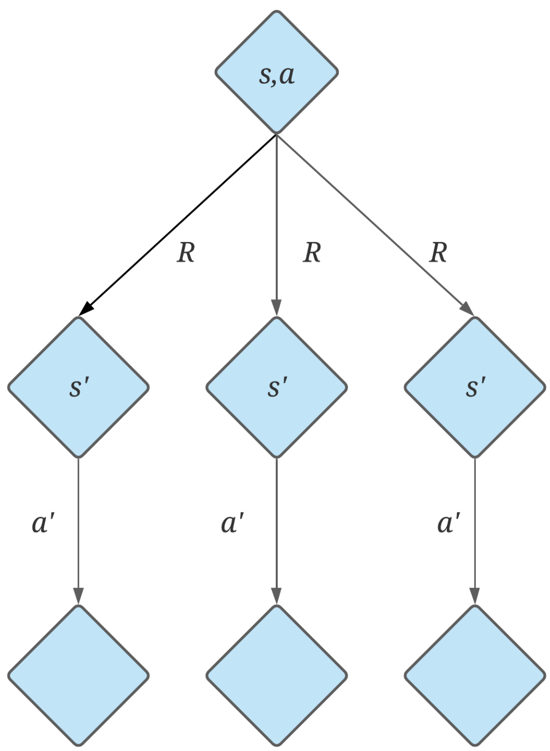


Figure 3.19 – SARSA Algorithm

SARSA has a larger emphasis on action selection that follows the current policy, after which the Q-values are updated, rather than Q-learning where greedy action, which offers the maximum Q-value, is selected. SARSA allows for corrections in optimization through exploration. Finding a proper ε-greedy hyperparameter is often a complex tuning endeavor. However, exploration will ensure that the optimization is conservative, and that solution avoids pitfalls of large negative rewards. Therefore, in Q-learning the algorithm is sourcing the highest valued action in the next state, while in SARSA it is the value of the action that was taken according to the current policy in force.

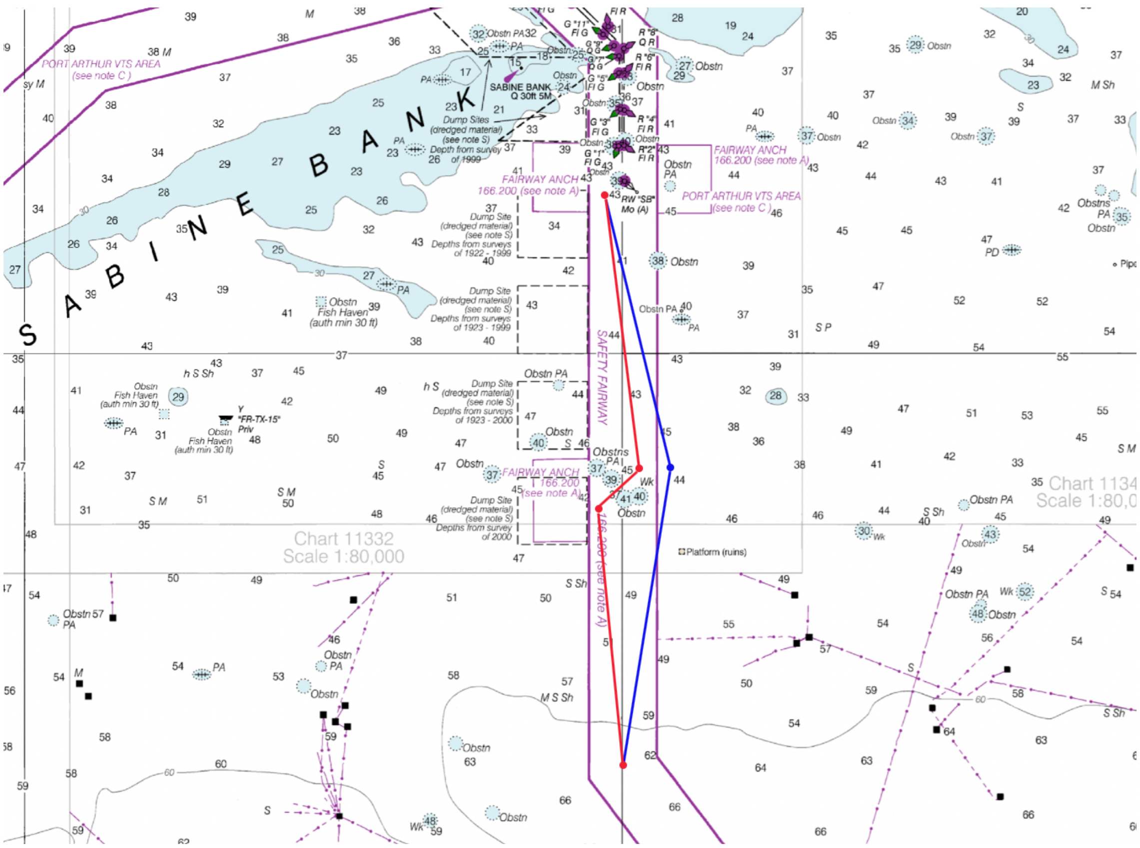


Figure 3.20 – Q-value and SARSA comparison Source: author on NOAA chart 11332 (https://charts.noaa.gov/PDFs/11330.pdf visited on 10-May-2021)

Figure 3.20 depicts the comparison of Q-value and SARSA algorithms. Represented by the red color is the Q-value algorithm that took more conventional path and successfully avoids the wrecks present on the way. In this scenario a sea surface vehicle leaving the port of Sabine Pass is depicted, and the vehicle is taking a southernly route to follow recommended fairway. In the blue color SARSA recommended path is shown and it is obvious to see that there are differences. SARSA algorithm takes into consideration risk factors and experience of either previous passes or human navigators that already passed in that area and know that passing between these wrecks can be challenging and often there are targets present at anchorages near the wrecks. Keeping all other factors aside, the Q-value algorithm has converged to an optimal policy that takes us safely to the next waypoint, but the SARSA algorithm takes more information into consideration and therefore is more risk averse. In this situation, SARSA algorithm steps outside the recommended fairway, but according to the navigation rules of the area, that is not forbidden and is considered safer for the situation agent faced. No targets were considered for this scenario, but experience of sailing in that area has been utilized and implemented it in SARSA algorithm, which gave us recommended path that was optimal when taking all risk factors into the reward space.

Algorithm 2 – SARSA Algorithm with ε-greedy exploration

Input: States S, Actions A, Reward function $R : S \text{ x } A \rightarrow \mathbb{R}$, Learning rate $\alpha = 0.02$,

Transition Function $\mathcal{T} : S \text{ x } A \rightarrow S$, Discounting $\gamma = 1$, ε-greedy factor 0.10

Output: Q

Initialization: $Q(s, a), \forall s \in S, a \in A(s)$, arbitrarily, and $Q(terminal\ state, \cdot) = 0$, or by initializing with $Q(s, a) = 0$ as there is no prior knowledge and state value

for each episode (time horizon or distance step horizon) do:

Initialize s

Chose a from s using policy derived from Q (ε-greedy)

for each step of episode do:

Take action a, observe R, s'

Choose a' from s' using policy derived from Q (ε-greedy)

Determine shaping functions $\Phi(s, a) = \Phi(s) + \Phi(a)$

Calculate $F(s, a, s', a') = \gamma\Phi(s', a') - \Phi(s, a)$,

where $\gamma = 0.1$ (3.71)

Calculate $R_F(s, a, s', a') = R(s, a, s') + F(s, a, s', a')$ (3.72)

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R_F + \gamma Q(s',a') - Q(s,a)] \tag{3.73}$$

$s \leftarrow s'; a \leftarrow a'$

end for when s is terminal

end for and **return** Q

end

As mentioned earlier, ε-greedy methodology is taken to promote exploration and avoid pitfalls of strictly following the optimized route. This is because the real world is stochastic in nature, and it is necessary to take into account that at some random moment during the exploitation a catastrophic failure is possible. If agents would remain in simulated spaces, running aground, allision or collision with another object is easily repairable with a simple restart of the simulation process, but if exploitation without consequences is allowed in the real world, potentially devastating outcomes could arise for human life, property, and environment.

At each time step, ε-greedy approach allows agents to select a random action with a pre-defined probability in the range of $0 \leq \varepsilon \leq 1$, in order to avoid consistent selection of the learned optimal action based on learned Q value. In this case ε-greedy probability is selected to be 0.1. More generally:

$$\pi(s) = \begin{cases} \text{random action from } A & \text{if } \beta < \varepsilon \\ \underset{a \in A}{\operatorname{argmax}}\, Q(s,a) & \text{otherwise,} \end{cases} \tag{3.74}$$

where $0 \leq \beta \leq 1$ is a random number selected at each time step.

The results of the experiment depicted in the Appendix A show that shaping rewards outperforms learning without shaping rewards, which is conclusion we needed in order to use this approach in developing an autopilot based on reinforcement learning and to use the similar approach when developing collision avoidance algorithm. We can also see in the previous example that because initialization was done with random selection of parameters, agent struggles to find optimal solution at the beginning. In this protected virtual environment, we can allow for longer learning processes, but in the real world, we want to limit the initial exploration and get the optimal solution faster, for which the learning from demonstration fits well.

As stability of the autopilot in the dynamic environments is in focus, we foster the efforts of downgrading the randomness of handling processes. We achieve better observability by downgrading hidden and partial observable states and actions, so rewards shaping is an integral part of achieving safer waterways.

3.3.4 Tuning hyperparameters for the optimal training

As important it is to accurately design data flow and efficient signal processing, it is equally important to develop training strategy that would warrant convergence to the global maxima or minima.

Within this research, three hyperparameters have a significant influence over convergence rate during training sessions, which are learning rate (α), discount factor (γ), and epsilon greedy action selection factor (ε). It is important to state that different tuning of hyperparameters is required for portfolio of challenges faced within this work. To determine the best tuning, an experiment has been conducted in which one of the proposed algorithms (heading-economy) has been utilized and tested different learning behavior when hyperparameters are tuned differently.

Within the heading-economy environment, three scenarios are selected to test various tuning of the hyperparameters. The first scenario is the best guess where rewards and penalties are selected for the three most prominent rewards of the algorithm: 1. heading desired equals heading filtered, 2. heading filtered is within the 1° from the desired heading, and the 3. where heading filtered is within the 2° from the desired heading. Therefore, it has been arranged that for each instance reinforcement learning agent manages to keep the heading filtered equal to the heading desired, the agent will receive 10 points of reward, but if it fails, the penalty will be 0, as this is economy mode where certain allowance of error is given in order to preserve actuators of a steering gear. In the case where heading filtered is $\pm 1°$ from the desired heading, 5 reward points are collected, and if the difference is larger than 1°, 10 penalty points are received. Finally, if the heading filtered is within the $\pm 2°$ difference, the agent receives 1 reward point, but when the heading difference is larger than 2°, the agent receives 50 penalty points. Therefore, $Reward_1$ has a 10→0 reward-penalty setup, $Reward_2$ has a 5→(-10) setup, while $Reward_3$ has a 1→(-50) reward-penalty setup.

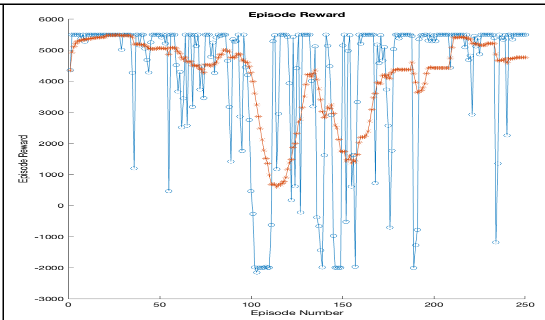
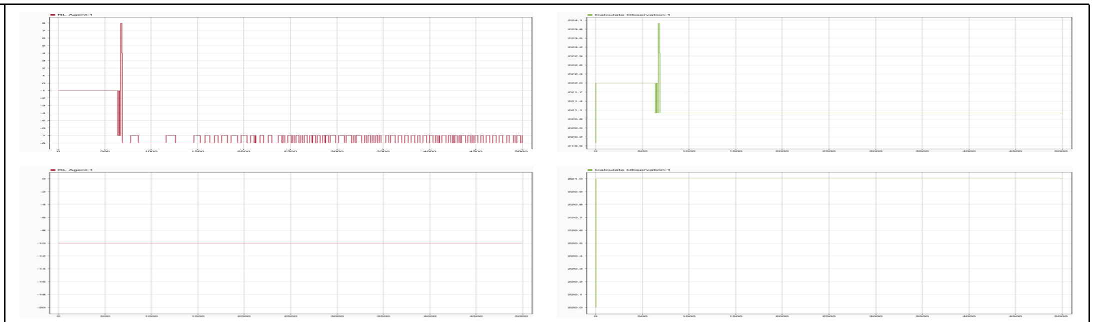
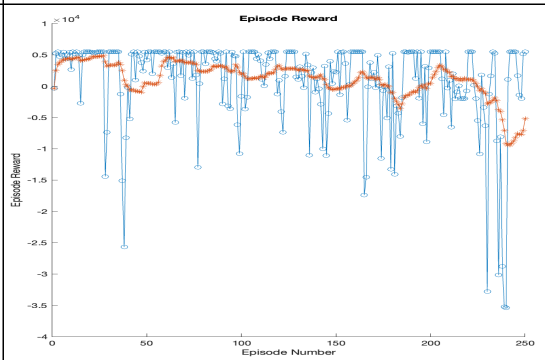
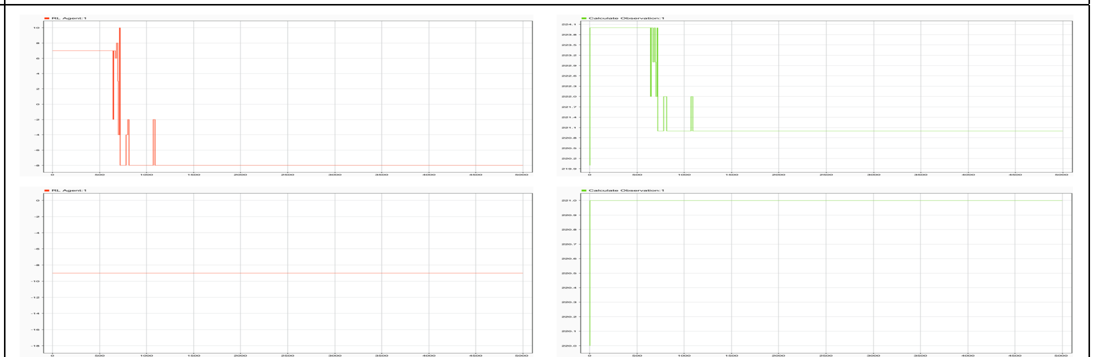
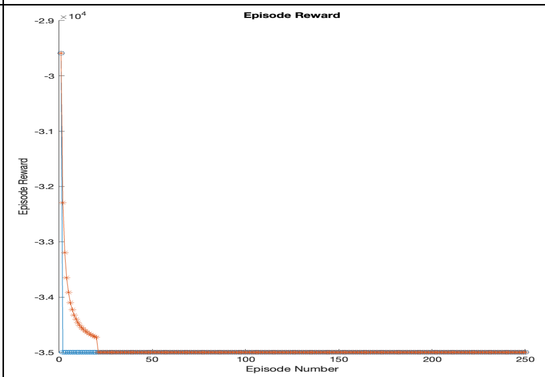
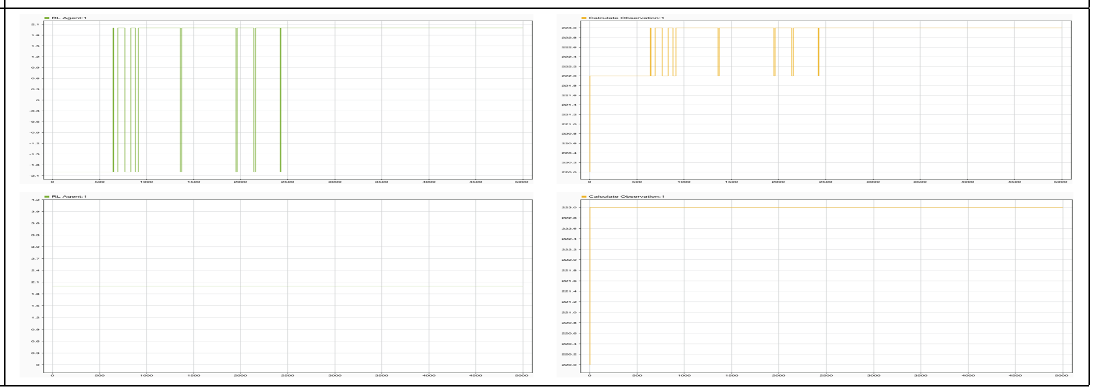
For all selected scenarios high learning rate and made a combination of gamma and epsilon to form nine different possibilities in order to find the best combination for all trainings, but also to have a deeper understanding of each of the hyperparameters available to us. As mentioned earlier, learning rate (α) is a real number ranging from 0 to 1. Setting α to 0 means that reinforcement learning agent learns nothing from new actions, while agent with α equaling to 1 completly disregards previous knowledge and notices only the latest information. Setting discount factor γ to 0 forces agent to disregards future rewards and only focuses on immediate rewards, while setting γ to 1 would encourage agent to look for higher rewards far in the future. Finally, the epsilon greedy action selection is vital for the randomness of the algorithm and is required to avoid convergence to a local maxima or minima. If ε was set to 0, agent would not explore, but rather only exploit the knowledge it already has, while the agent with ε 1 would completely ignore previous knowledge and focus only on the random actions. Setting any of these hyperparameters to either 0 or 1 would not deliver desired results and learning would be compromised. Finding a good balance is not easy, so trials are continued to find out the optimal balance for handling of underactuated sea surface vehicles. Fairly high learning rate parameter has been selected, as learning from new actions is encouraged, giving that researched problem is defined deterministically. Discount and epsilon greedy parameters are defined as described in Table 3.3.

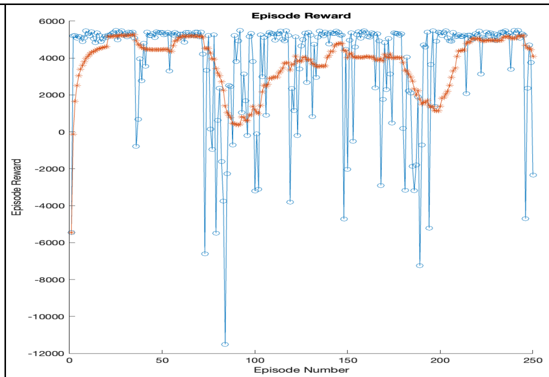
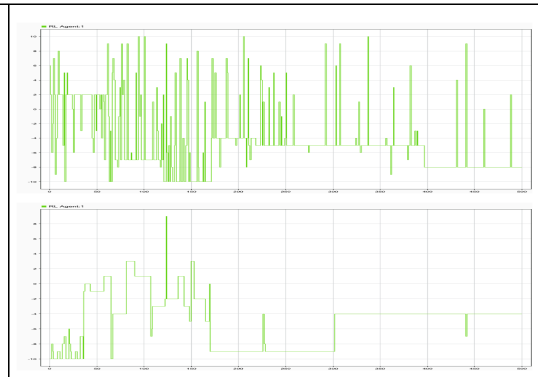
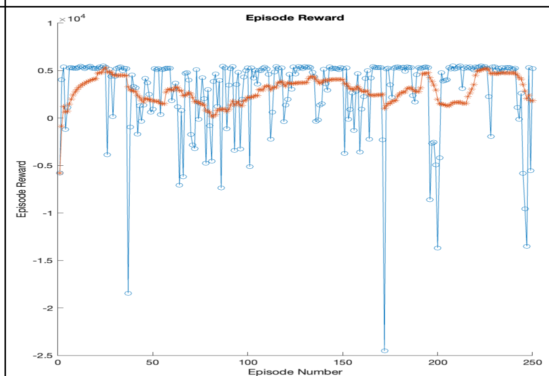
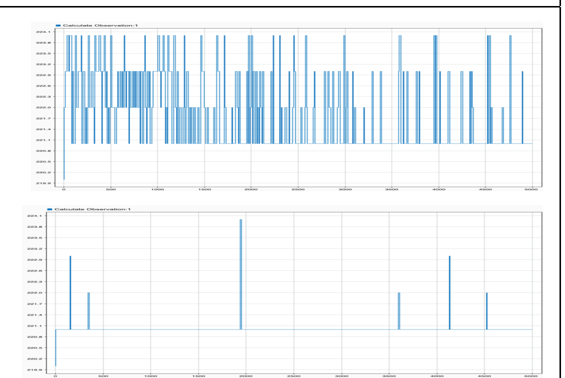
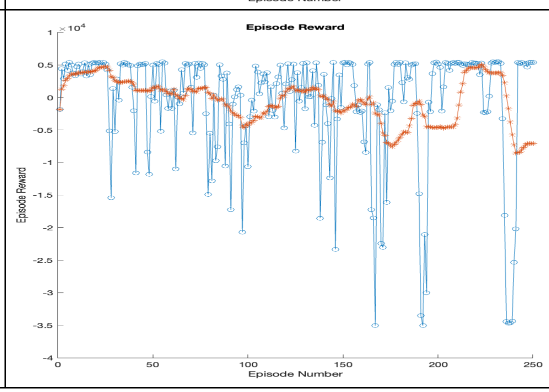
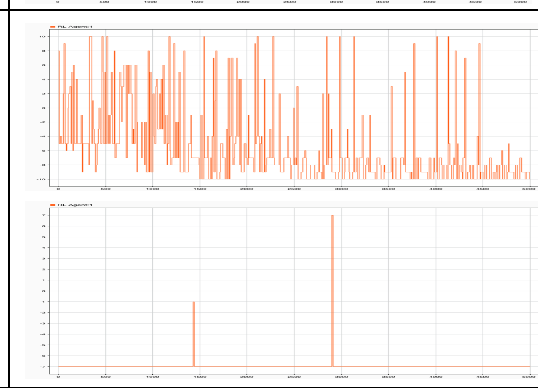
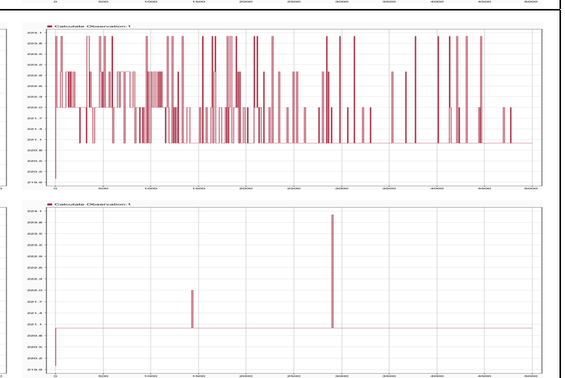
Table 3.3 – Hyperparameters tuning combinations

OPTION	ALPHA - α	GAMMA - γ	EPSILON - ε
A	0.9	0	0
B	0.9	0.5	0
C	0.9	1	0
D	0.9	0	0.5
E	0.9	0	1
F	0.9	0.5	0.5
G	0.9	0.5	1
H	0.9	1	1
I	0.9	1	0.5

For each of the first scenario options 250 training episodes were made with 500 steps per episodes, making 125,000 steps per training in total. Each training is represented by a graph, where blue line and associated spots are related to each episode accumulated rewards, while the orange line represents movement of an average rewards throughout the training session. The training was conducted with economy mode of the heading autopilot, where one steering pump was used, action space was limited to a discrete set [-10, 10] of rudder movement. Wind strength was set to 21 knots with relative direction of 67.5°, current speed of 2 knots and relative direction of 74°, corresponding wind wave of force 6 and relative direction of 45°, and swell height of 4 with relative direction of 67.5°.

Table 3.4 – First training session

A		
B		
C		

D	 Episode Reward Episode Reward Episode Number		
E	 Episode Reward Episode Reward Episode Number		
F	 Episode Reward Episode Reward Episode Number		

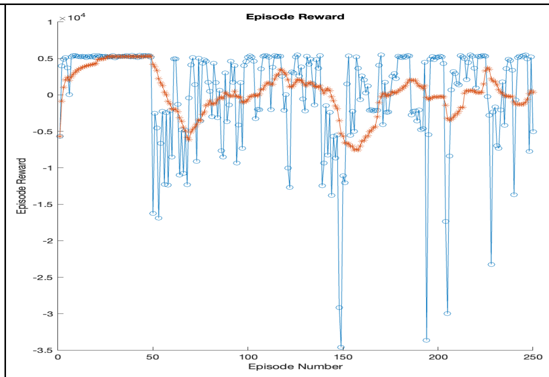
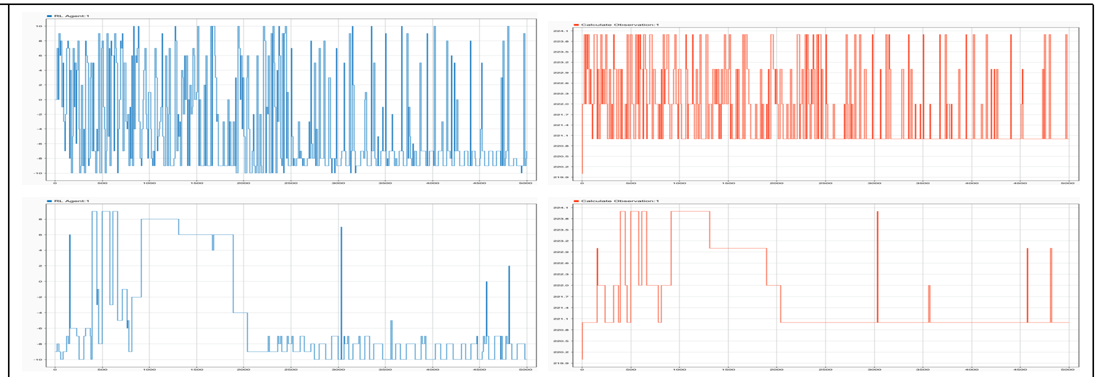
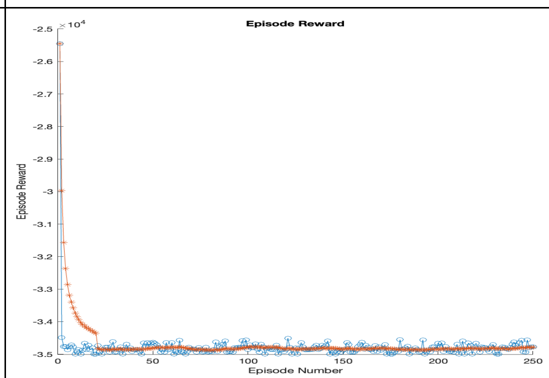
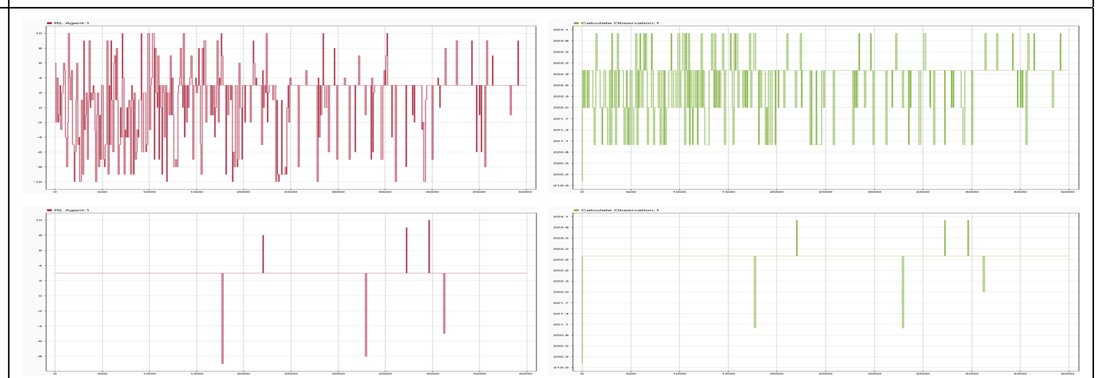
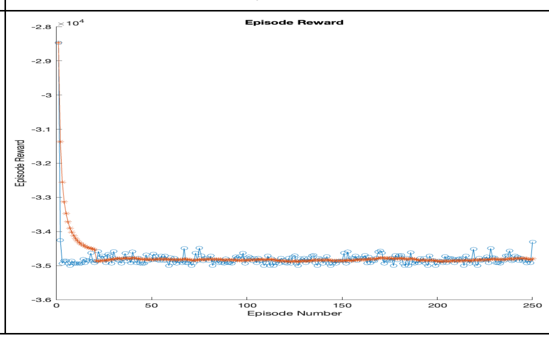
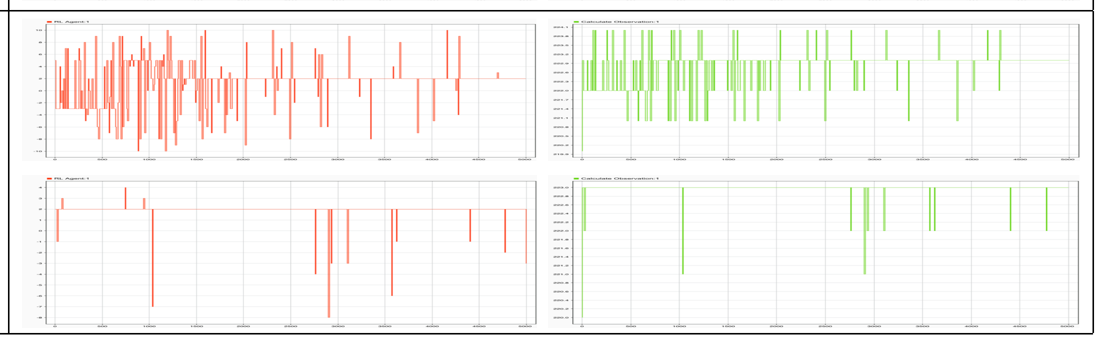
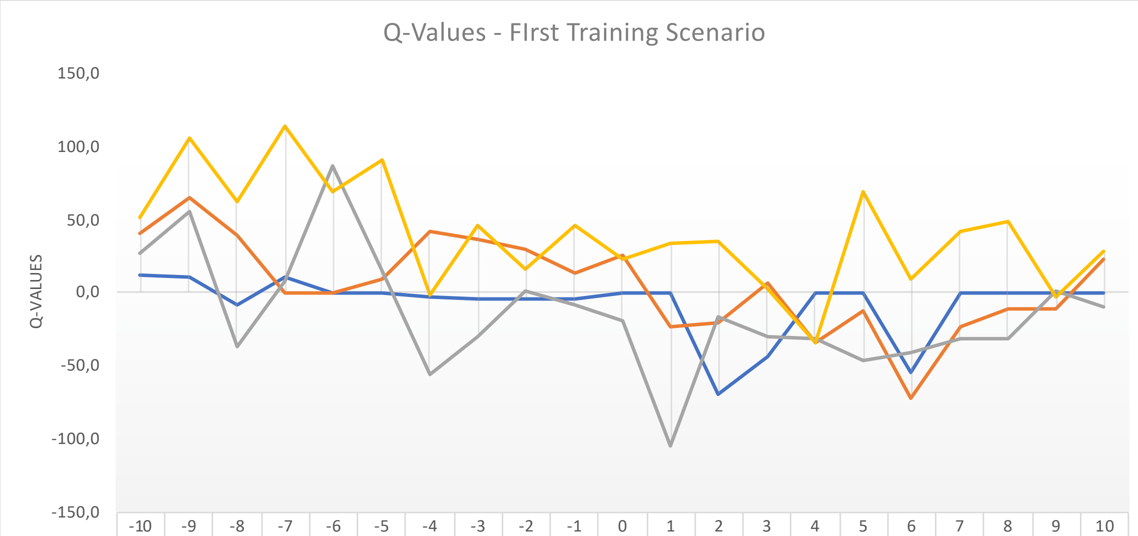
G	Episode Reward Episode Number	
H	Episode Reward Episode Number	
I	Episode Reward Episode Number	

Table 3.4 depicts training progress. In the second training progress graph is depicted where training statistics is shown for each episode. On the *x* axis we see the Episode Number is presented, while on the *y* axis is the Episode Reward, be it positive or negative (penalty). After conducting the reinforcement learning tests it is noticeable that whenever the discount factor was set to one (options C, H, and I), learning failed. When the gamma was set to 1, agent only focused on the long-term rewards and failed to learn anything from the more immediate experiences. Therefore, it is evident that the discount factor should be set closer to 0, rather closer to 1. On the other hand, when the exploration greedy parameter has been set to 1 (options E and G, while H is discarded due to gamma), an increased exploration activity is visible even after 250 training sessions, but agent did learn the accurate action for the training scenario. Viable options are A, B, D and F, but it is not visible from the training session data which of these options are optimal approach to training. This is mainly due to a high learning rate that values learning from the latest steps and will inadvertently support exploration, which is evident on graphs where for a steady period of time high rewards are visible and then exploration kicks in and causes negative spikes of episode rewards. Exploration is valued as it aids to avoid local maxima or minima convergence. To find the optimal option, it is necessary to look at the Q-table data and compare graphs of Q-values.

By taking a closer look at Figure 3.21 with graphical representation of Q-values for each of the available discrete actions, it is noticeable that option A does not provide a distinctive action for the training scenario and Q-values appear to be flat. This is mainly to the fact that the exploration was completely discouraged. With the options B and D, better behavior is noticed; however, training converged to a local maximum, so only the option F provided the accurate convergence. Fluctuations of the Q-values are still noticeable, but as it will be discovered in the following pages, this is due to rewards space design that warrant some rewards shaping in order to get a clear distinction between global and local maxima. Therefore, it is evident that positive number for both gamma and epsilon is required, so the focus remains on the option F to make further investigations.

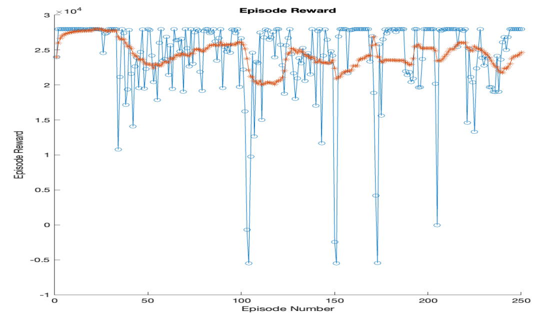
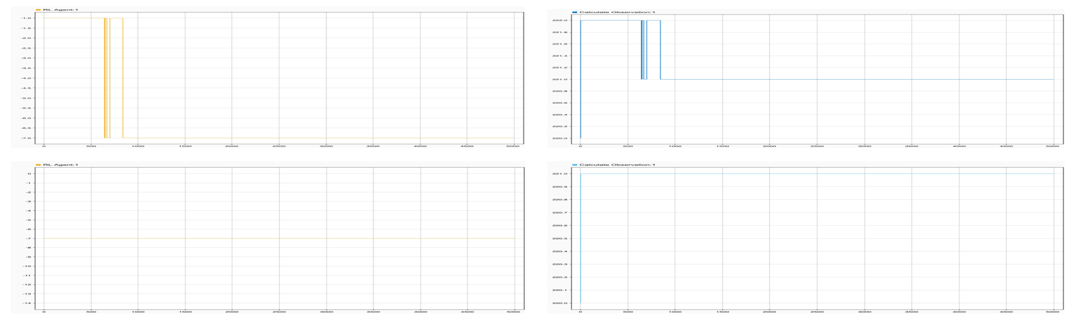
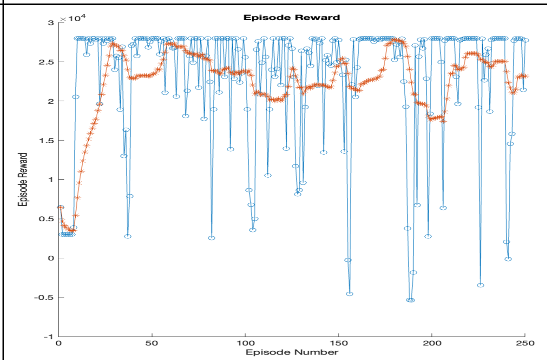
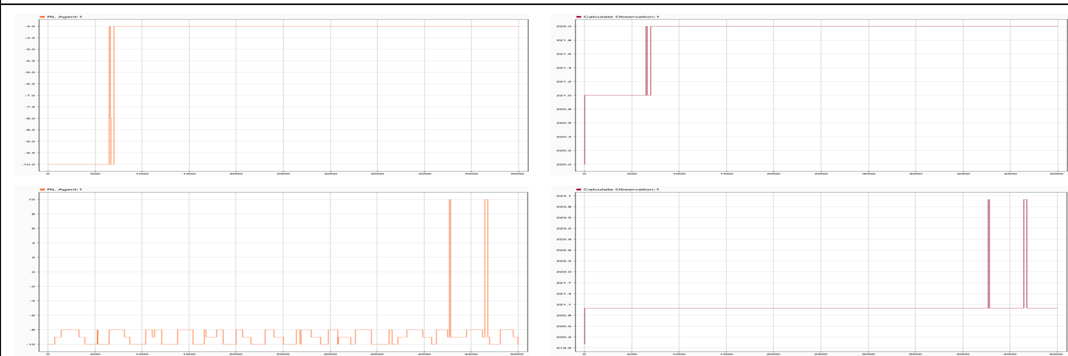
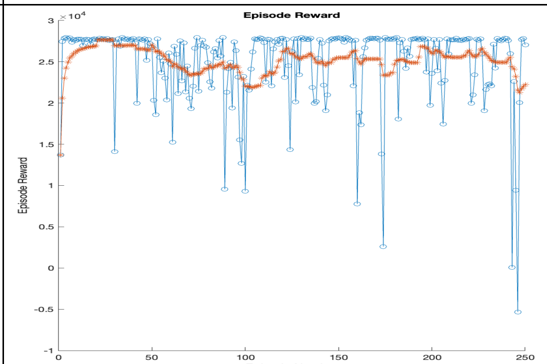


	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
A	11,5	11,0	-8,9	9,8	0,0	0,0	-3,5	-3,9	-4,0	-4,0	0,0	0,0	-69,1	-43,8	0,0	0,0	-54,4	0,0	0,0	0,0	0,0
B	40,6	64,5	39,3	0,0	0,0	8,6	41,1	36,5	28,8	13,7	25,8	-23,1	-21,1	7,0	-34,8	-12,8	-72,6	-23,0	-11,1	-11,6	22,0
D	27,3	54,9	-37,7	7,4	86,5	14,4	-56,2	-30,1	0,5	-8,2	-19,6	-104,	-16,7	-30,2	-31,5	-46,5	-41,5	-31,7	-31,0	0,6	-10,1
F	50,5	105,2	61,4	113,6	68,4	91,0	-1,8	46,1	15,2	46,1	23,1	33,1	34,9	1,7	-34,5	68,5	8,8	41,8	48,1	-3,7	27,6

Figure 3.21 – Q-values comparison for the First Training Scenario

With the second and third scenario (figures 3.22 and 3.23), confirmation of findings is sought for the above-mentioned options, but at the same time different rewards schemes are investigated. The second scenario has a reward space setup as follows: $Reward_1$ has a 100→0 reward-penalty setup, $Reward_2$ has a 50→0 setup, while $Reward_3$ has a 1→0 reward-penalty setup. So, behavior of agent when only rewards are available is tested. In the case of the third scenario, reward space set up consists of large penalties: $Reward_1$ has a 10→(-10) reward-penalty setup, $Reward_2$ has a 5→(-50) setup, while $Reward_3$ has a 1→(-100) reward-penalty setup. As depicted in the tables 3.5 and 3.6, in both second and third sessions, similar behavior is noticed as it was in the case of the first session. Upon a further investigation of the Q-values, it was noticed that larger rewards offer a better distinction of the global maxima, while larger penalties promote steadier learning and reduce chance of convergence near local maxima. It is noticeable that the third session offered convergence of almost all options at the global maximum (action -7) and that leads to important conclusion that stick is better than a carrot, so we have smaller allowance to err on the correct selection of penalties, than to make an incorrect selection of rewards.

Table 3.5 – Second training session

A		
B		
D		

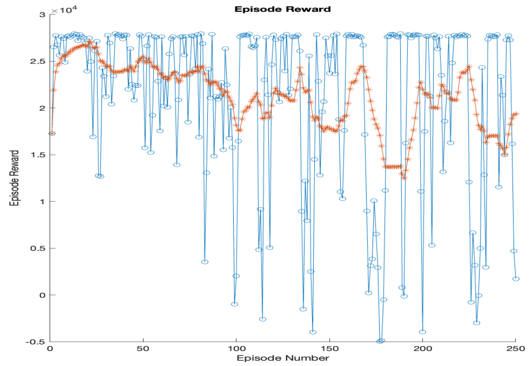
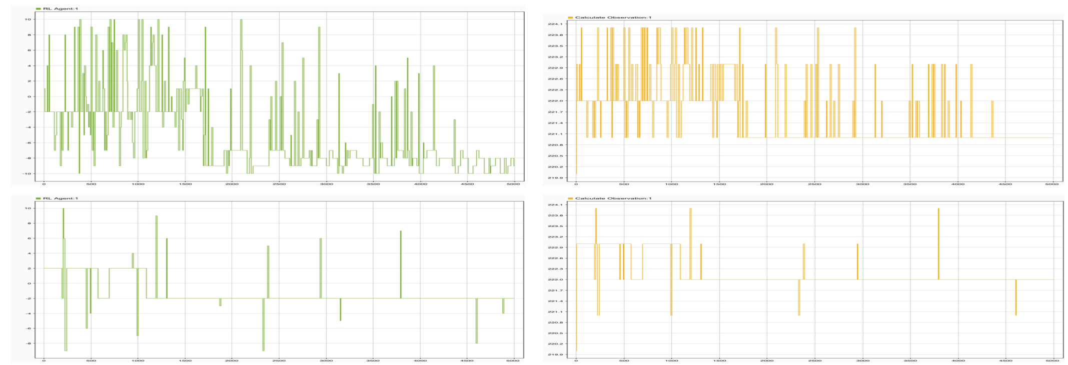
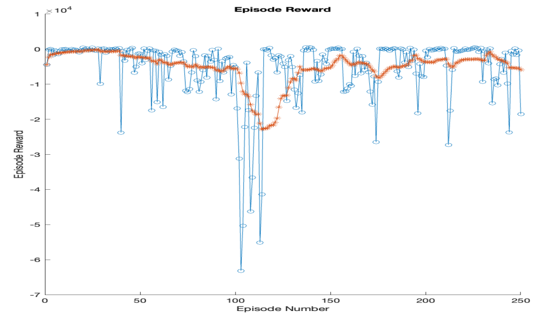
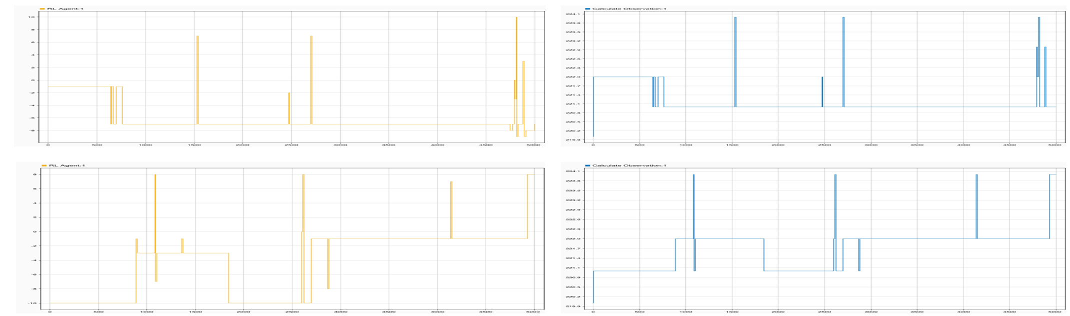
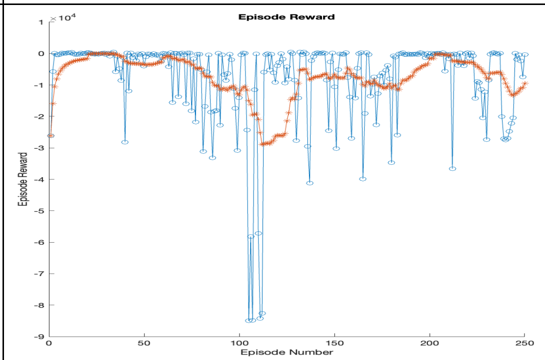
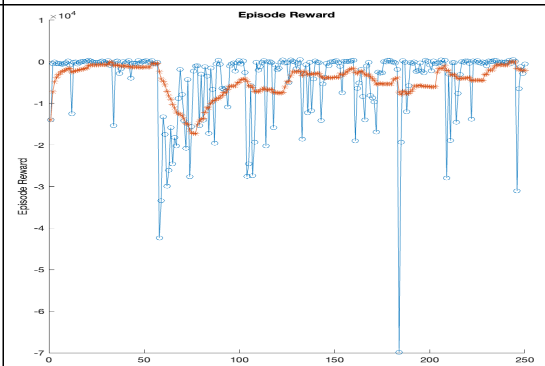
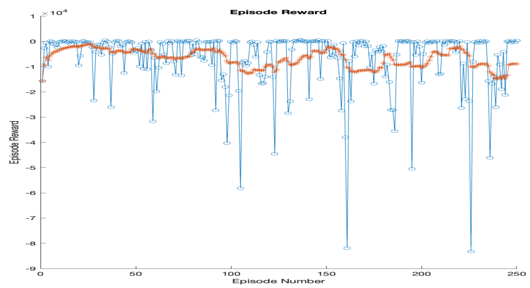
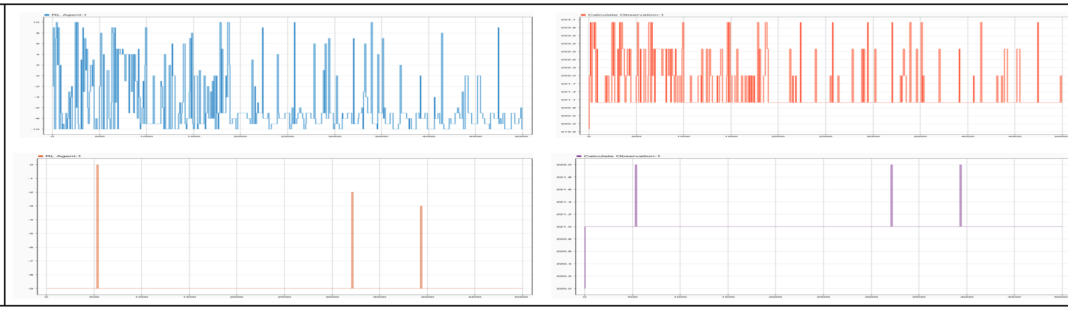
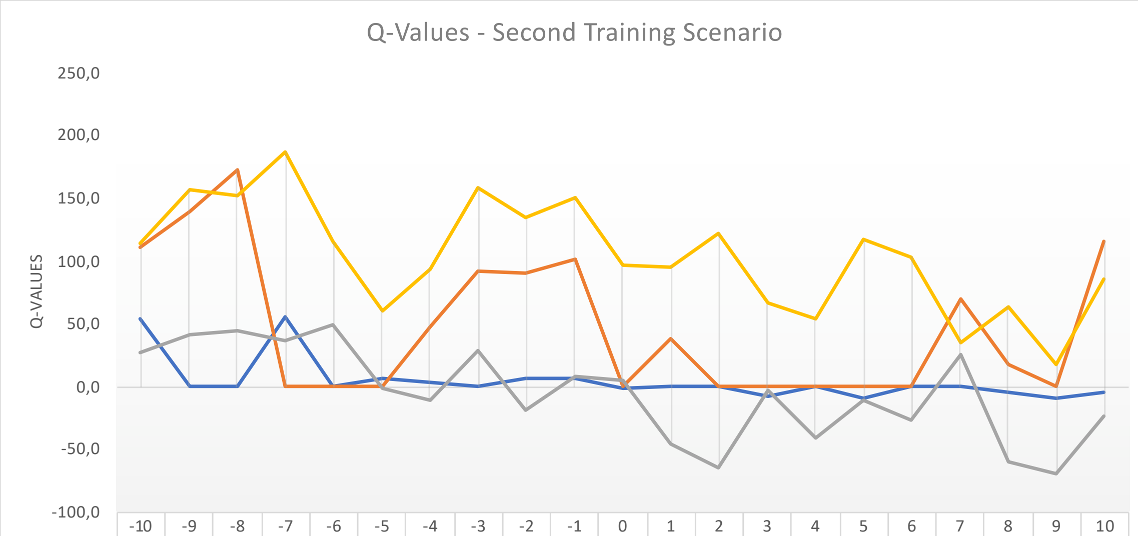
F	 Episode Reward Episode Reward Episode Number × 10^4 3 2.5 2 1.5 1 0.5 0 -0.5 0 50 100 150 200 250	

Table 3.6 – Third training session

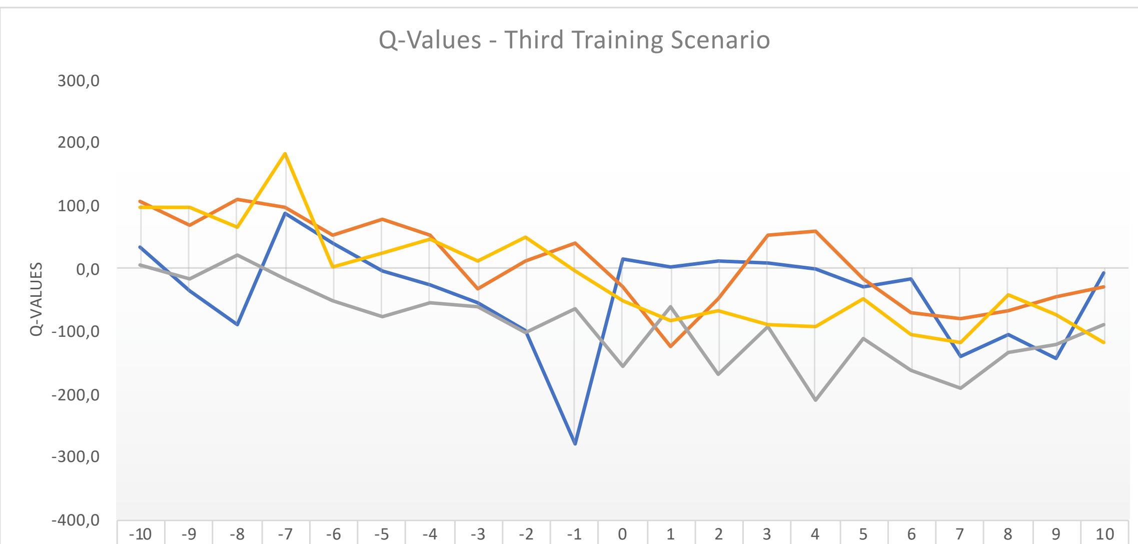
A		
B		
D		

F		



	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
A	53,9	0,0	0,0	55,9	0,0	5,9	3,9	0,0	7,0	6,5	-0,7	0,0	0,0	-8,1	0,0	-10,0	0,0	0,0	-5,3	-8,6	-4,7
B	110,5	139,4	173,2	0,0	0,0	0,0	47,5	92,4	91,1	101,2	0,0	37,5	0,0	0,0	0,0	0,0	0,0	69,5	17,9	0,0	116,4
D	27,3	41,6	44,7	35,8	49,0	-2,0	-11,0	27,9	-19,5	7,9	5,2	-45,8	-64,1	-2,9	-41,1	-11,1	-26,8	26,0	-59,6	-70,3	-23,0
F	113,9	157,3	152,5	187,6	114,9	61,0	94,0	158,5	133,9	151,2	96,3	95,6	122,7	65,9	54,6	116,5	103,0	34,5	63,9	17,5	85,3

Figure 3.22 – Q-values comparison for the Second Training Scenario

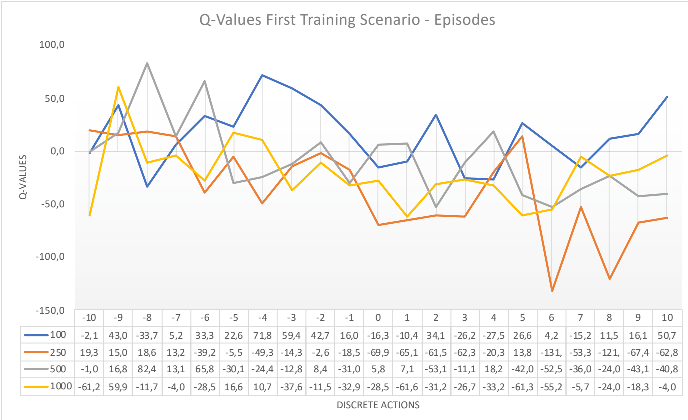


	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
A	34,9	-35,6	-89,4	89,5	39,4	-2,2	-25,5	-54,4	-98,9	-279,	16,7	2,3	13,2	10,2	-1,7	-30,2	-17,6	-140,	-105,	-143,	-5,2
B	108,2	69,6	111,6	97,9	54,4	77,4	54,9	-32,0	11,3	40,9	-28,2	-122,	-47,9	52,1	61,0	-16,9	-70,0	-79,0	-68,1	-43,6	-27,6
D	6,5	-17,1	20,8	-15,7	-49,9	-77,9	-54,9	-61,6	-100,	-64,8	-155,	-61,9	-167,	-90,8	-210,	-112,	-163,	-190,	-134,	-121,	-90,4
F	96,3	98,3	66,8	183,6	2,6	25,6	45,8	12,4	51,1	-2,3	-51,8	-81,3	-66,2	-90,2	-91,4	-49,2	-104,	-116,	-42,8	-72,2	-117,

Figure 3.23 – Q-values comparison for the Third Training Scenario

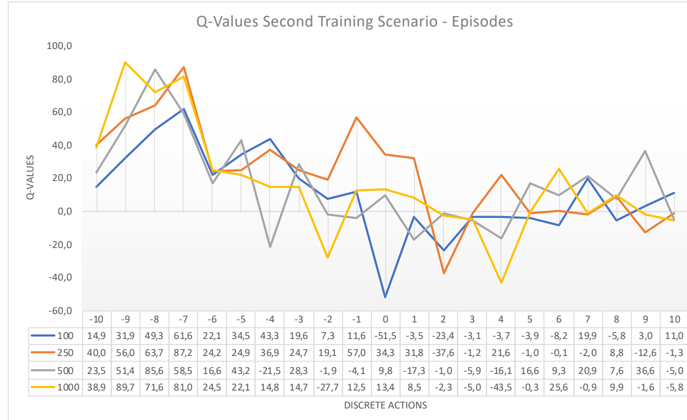
Even though hyperparameter settings were investigated and received important insights about tuning, fundamental questions remain open: What is the optimal length of training? How do we ascertain that training session was successful and converged to a global maximum? In order to answer those questions, additional tests had to be conducted by utilizing option F by alternating the allowed training episodes between 100, 250, 500 and 1000 episodes. In Figure 3.24 Q-values of the First Training Scenario is depicted, and it is visible that 100 episodes is not enough for a successful training. 250 episodes shows a better performance, but there are some incorrect spikes of convergence in the positive action space, while the correct convergence is in the negative action space. This also points to necessity of prolonging the training, so it is noticeable that results are more realistic when the training lasts for 500 or 1000 episodes. Having more steps is computationally more expensive and time consuming, so the benefit of prolonging the training has to be evident to make it viable. In presented case, 1000 episodes do not show any measurable improvement in training, so 500 steps is the viable training option. Figures 3.25 and 3.26 depict the Second and Third Scenario with identical conclusions. In the Second Scenario it is also visible that there is much better distinction of where the global maximum is, as all options clearly show preference of action space between -10 and -7. On the other hand, the Third Scenario Figure shows utilization of larger penalties to drive positive side of the action space (0 to 10 degrees rudder deflection) away from the global maxima and ensure that no convergence can happen in that space. This is why the Forth Scenario is introduced where reward space is defined as follows: $Reward_1$ has a 50→-5 reward-penalty setup, $Reward_2$ has a 10→-10 setup, while $Reward_3$ has a 5→-50 reward-penalty setup. Therefore, larger rewards and larger penalties are combined in order to get a better distinction of the global maxima, so Figure 3.27 depicts the Forth Scenario and confirms that 500 episodes is the optimal approach to training.

Finally, as there are many Q-values spikes when training with 0.9 learning rate, training is continued with learning rate at 0.1 and 0.4 and Figure 3.28 shows that the best performing learning rate is 0.4, so 0.4 learning rate fits best for handling of underactuated vehicles training tasks.



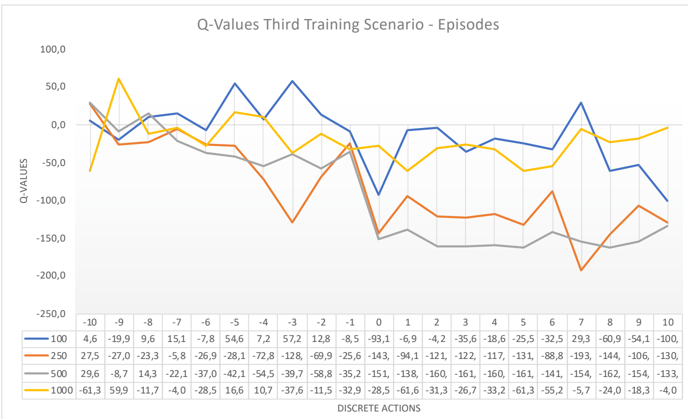
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
100	-2,1	43,0	-33,7	5,2	33,3	22,6	71,8	59,4	42,7	16,0	-16,3	-10,4	34,1	-26,2	-27,5	26,6	4,2	-15,2	11,5	16,1	50,7
250	19,3	15,0	18,6	13,2	-39,2	-5,5	-49,3	-14,3	-2,6	-18,5	-69,9	-65,1	-61,5	-62,3	-20,3	13,8	-131,	-53,3	-121,	-67,4	-62,8
500	-1,0	16,8	82,4	13,1	65,8	-30,1	-24,4	-12,8	8,4	-31,0	5,8	7,1	-53,1	-11,1	18,2	-42,0	-52,5	-36,0	-24,0	-43,1	-40,8
1000	-61,2	59,9	-11,7	-4,0	-28,5	16,6	10,7	-37,6	-11,5	-32,9	-28,5	-61,6	-31,2	-26,7	-33,2	-61,3	-55,2	-5,7	-24,0	-18,3	-4,0

Figure 3.24 – First Training Scenario (Q-Values)



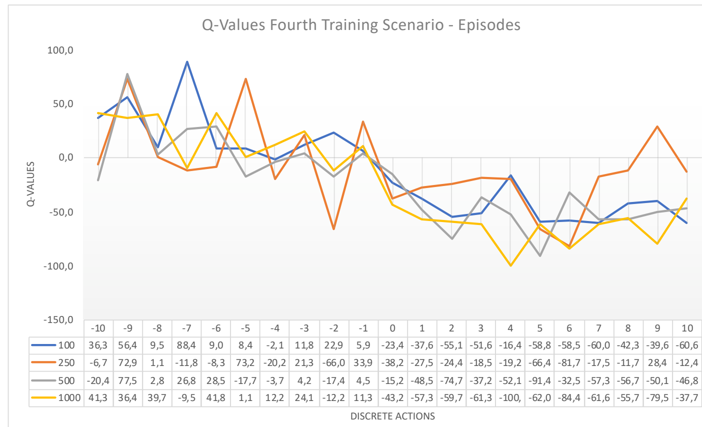
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
100	14,9	31,9	49,3	61,6	22,1	34,5	43,3	19,6	7,3	11,6	-51,5	-3,5	-23,4	-3,1	-3,7	-3,9	-8,2	19,9	-5,8	3,0	11,0
250	40,0	56,0	63,7	87,2	24,2	24,9	36,9	24,7	19,1	57,0	34,3	31,8	-37,6	-1,2	21,6	-1,0	-0,1	-2,0	8,8	-12,6	-1,3
500	23,5	51,4	85,6	58,5	16,6	43,2	-21,5	28,3	-1,9	-4,1	9,8	-17,3	-1,0	-5,9	-16,1	16,6	9,3	20,9	7,6	36,6	-5,0
1000	38,9	89,7	71,6	81,0	24,5	22,1	14,8	14,7	-27,7	12,5	13,4	8,5	-2,3	-5,0	-43,5	-0,3	25,6	-0,9	9,9	-1,6	-5,8

Figure 3.25 – Second Training Scenario (Q-Values)



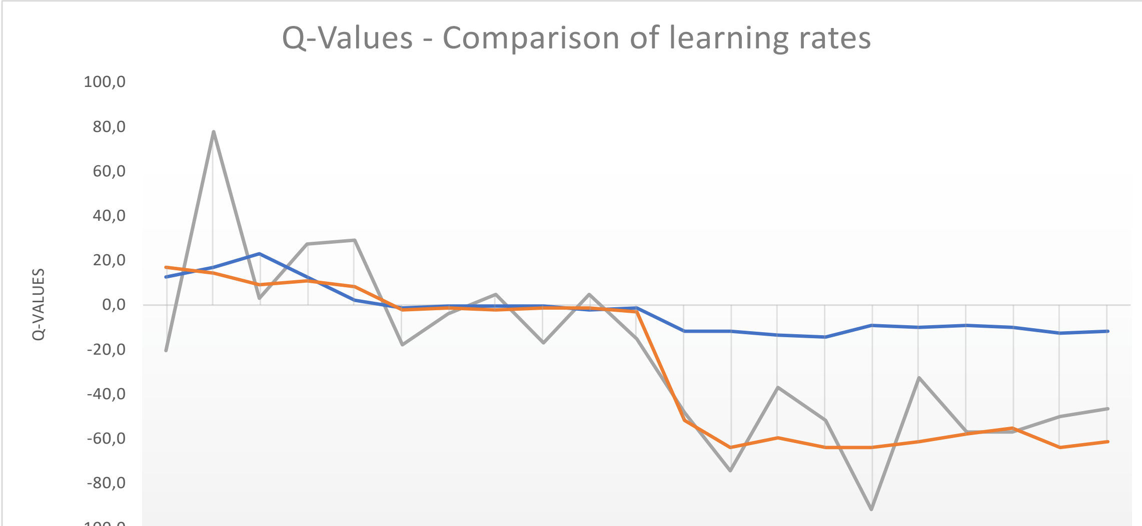
	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
100	4,6	-19,9	9,6	15,1	-7,8	54,6	7,2	57,2	12,8	-8,5	-93,1	-6,9	-4,2	-35,6	-18,6	-25,5	-32,5	29,3	-60,9	-54,1	-100,
250	27,5	-27,0	-23,3	-5,8	-26,9	-28,1	-72,8	-128,	-69,9	-25,6	-143,	-94,1	-121,	-122,	-117,	-131,	-88,8	-193,	-144,	-106,	-130,
500	29,6	-8,7	14,3	-22,1	-37,0	-42,1	-54,5	-39,7	-58,8	-35,2	-151,	-138,	-160,	-161,	-160,	-161,	-141,	-154,	-162,	-154,	-133,
1000	-61,3	59,9	-11,7	-4,0	-28,5	16,6	10,7	-37,6	-11,5	-32,9	-28,5	-61,6	-31,3	-26,7	-33,2	-61,3	-55,2	-5,7	-24,0	-18,3	-4,0

Figure 3.26 – Third Training Scenario (Q-Values)



	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
100	36,3	56,4	9,5	88,4	9,0	8,4	-2,1	11,8	22,9	5,9	-23,4	-37,6	-55,1	-51,6	-16,4	-58,8	-58,5	-60,0	-42,3	-39,6	-60,6
250	-6,7	72,9	1,1	-11,8	-8,3	73,2	-20,2	21,3	-66,0	33,9	-38,2	-27,5	-24,4	-18,5	-19,2	-66,4	-81,7	-17,5	-11,7	28,4	-12,4
500	-20,4	77,5	2,8	26,8	28,5	-17,7	-3,7	4,2	-17,4	4,5	-15,2	-48,5	-74,7	-37,2	-52,1	-91,4	-32,5	-57,3	-56,7	-50,1	-46,8
1000	41,3	36,4	39,7	-9,5	41,8	1,1	12,2	24,1	-12,2	11,3	-43,2	-57,3	-59,7	-61,3	-100,	-62,0	-84,4	-61,6	-55,7	-79,5	-37,7

Figure 3.27 – Forth Training Scenario (Q-Values)



	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
500_09	-20,4	77,5	2,8	26,8	28,5	-17,7	-3,7	4,2	-17,4	4,5	-15,2	-48,5	-74,7	-37,2	-52,1	-91,4	-32,5	-57,3	-56,7	-50,1	-46,8
500_01	12,6	17,1	22,5	12,7	2,0	-1,6	-0,3	-0,3	-0,9	-2,7	-1,5	-11,5	-11,6	-13,8	-14,3	-9,3	-10,5	-9,3	-10,3	-12,8	-12,2
500_04	16,8	14,4	8,6	10,7	8,1	-2,6	-1,5	-2,5	-1,2	-1,8	-2,9	-51,4	-63,6	-59,5	-64,3	-64,3	-61,7	-57,9	-55,1	-63,5	-61,1

Figure 3.28 – Comparison of learning rates (Q-values)

It is now possible to conclude that the best approach to training is by tuning the hyperparameters that assure convergence to a global maximum, and for the handling of underactuated sea surface vehicles all hyperparameters should be above 0 and below 0.5. By investigating further, for depicted case scenario, the best performance is achieved if the learning rate is 0.4, discount factor 0.1, and epsilon-greedy parameter 0.1. Also, it is necessary to state that all experiments in the continuation of this thesis would be designed so that training is completed with 95 % of confidence, and that the training stops if an average reward received throughout the most recent 10 consecutive episodes is above of 95 % of the maximum possible reward for a single episode. For example, if the maximum achievable reward for each step is 10 and there are 500 steps per episode, the maximum reward is 5,000. Therefore, the agent would have to receive 4,750 reward points for 10 consecutive episodes for the training to stop. If this is not achieved throughout the training session, mainly due to exploration, the training would stop after 500 episodes.

3.3.5 Building the initial database - Learning from Demonstration

Looking at the previous chapter's example, experiment commenced as if there was no previous knowledge or experience and stated simply that all states in the state space domain are equal to zero $Q(s, a) = 0$, which shows that all states have equal value, and that agent can select any of the actions available to start exploring and exploiting. Even though simple, this approach comes at a high price of convergence time. Any approach with guidance to the agent in good faith, even if not very precise, will aim to shorten the time of convergence, which is the reason why learning from demonstration is introduced.

If human operator is navigating a sea surface vehicle, regulations mandate that navigators have to study navigation, pass various practical tests, undergo a year time of cadetship and pass a rigorous qualification exam before they are allowed to navigate a commercial vessel. It will take some time for them to gain confidence, but their first navigational decisions are not without prior knowledge and understanding of the material. Therefore, aim of this chapter is to provide a set of learned parameters that will guide learning algorithms to commence exploring, select optimal paths and learn from mistakes during exploitation.

When designing a model based on MDP, the reward function plays a vital role. Even a small change in constants within a reward function can make a significant change in accuracy of the solution and sequential decision making. That is why reward function design usually requires a lot of trial-and-error attempts guided by experts' overview in order to achieve the desired output. Considering the extent of desiderata in the motion control problem of sea surface vehicles, writing down an explicit reward function is not a straight-forward task, as not only movements by following trajectories have to be included, but also limitations of the considered vehicle and external factors taken into consideration. On one hand focus is on reward on keeping the desired heading while on the other hand penalties for undesired movements have to be designed to allow for nonlinear inputs and outputs. This issue can be approached by either inferring the reward function through utilization of inverse reinforcement learning where we learn the reward function, or by shaping the reward function to achieve sustainable results. In this research a model-free approach using reward shaping is used in order to achieve risk averse and collision free path generation, starting with an auto-pilot model design. As the Foraging Particle Filter filters external influences, sensor errors and uncertainties of proposed control model, in continuation design of the autopilot by utilizing MDP framework is presented.

Navigating a sea surface vehicle with many external forces that change unpredictably is highly dynamic effort, where considerable number of desiderata have to be monitored in order to control the heading safely. If making a simple turn repetitively is contemplated, but with different external influences and speeds, different inputs to the rudder for each of these turns are obtained. Also, when thinking of turning rates, an expert piloting the sea surface vehicle would determine what should the speed of turning be for that turn, so inputs to rudder would be nonlinear in the dataset for each change in environmental dynamics, as well as for each individual pilot expert. In order to depict this phenomenon, an experiment is consolidated where two piloting experts are doing one section of navigation in a channel and then we let one inexperienced agent to learn steering from the experienced experts.

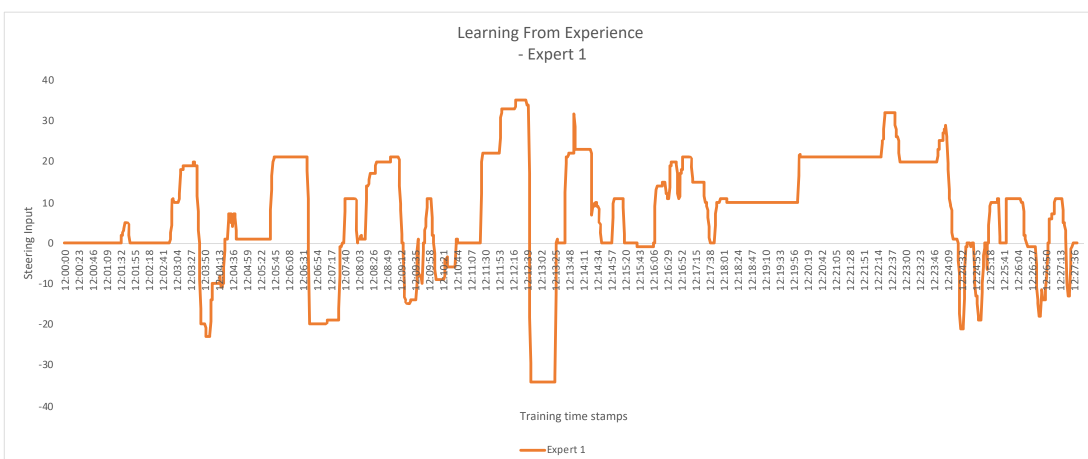


Figure 3.29 – Learning from experience – Expert 1

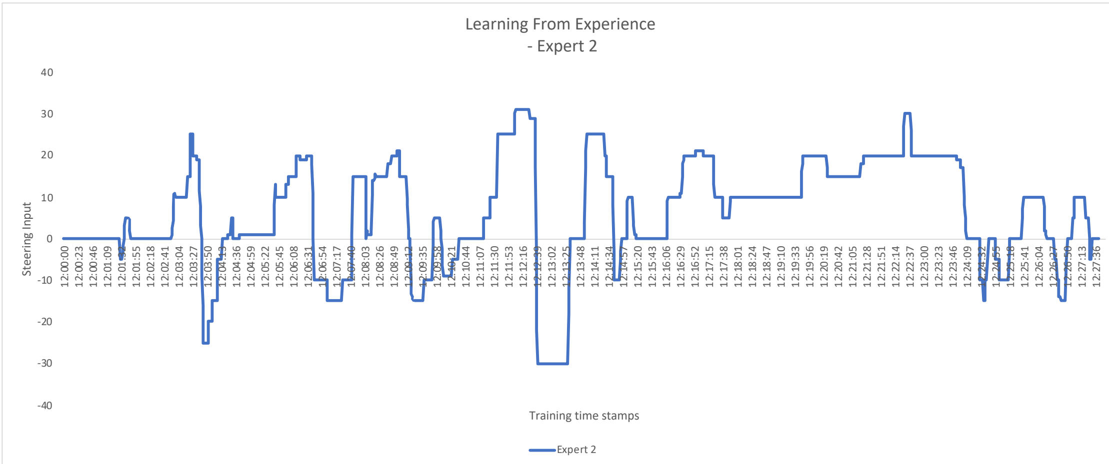


Figure 3.30 – Learning from experience – Expert 2

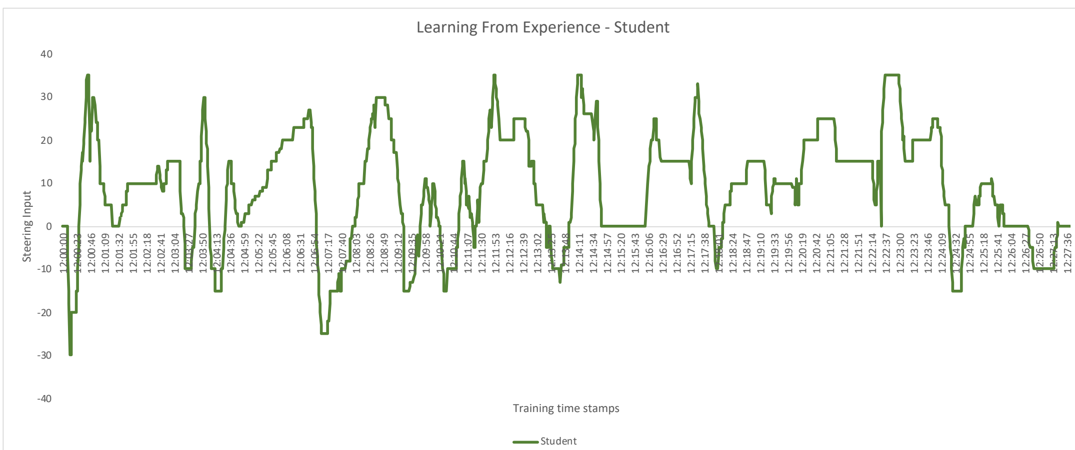


Figure 3.31– Learning from experience – Student

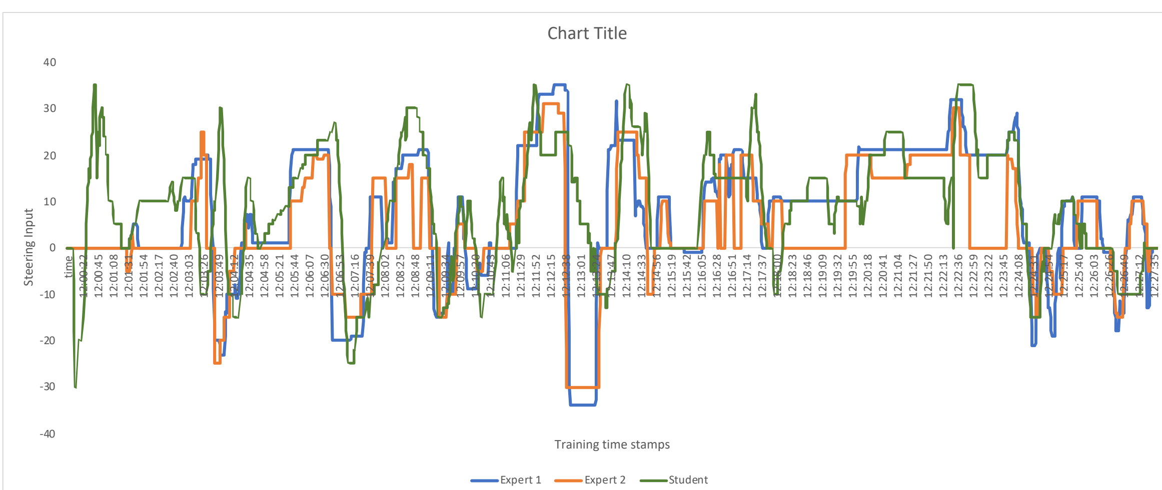


Figure 3.32 – Learning from experience – Comparison

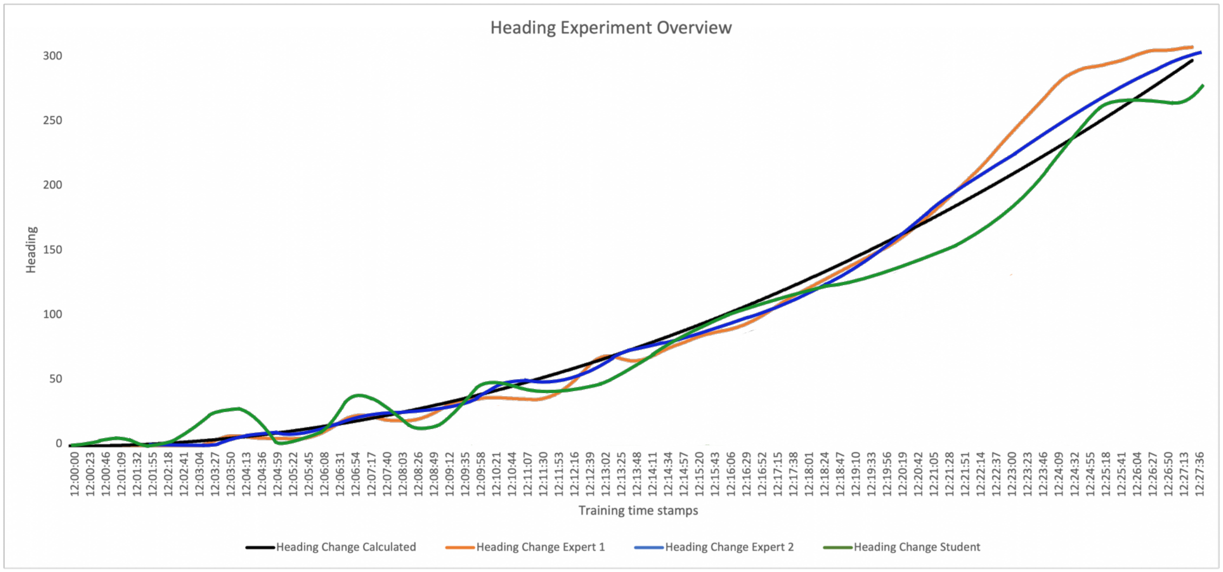


Figure 3.33 – Heading experiment overview – scenario 1

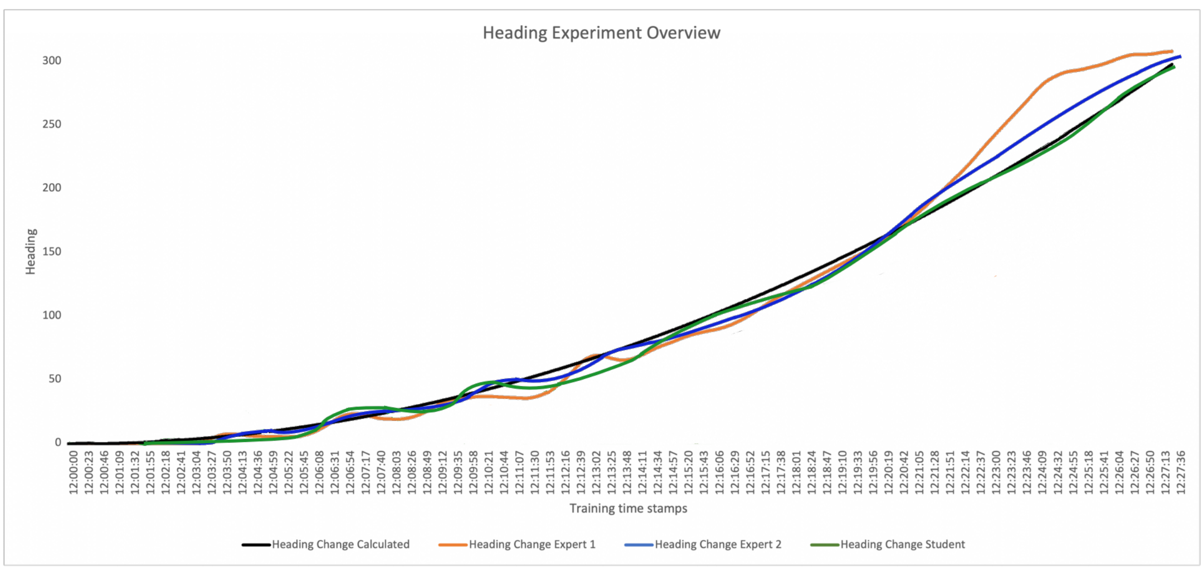


Figure 3.34 – Heading experiment overview – scenario 2

Fresh system without any previous experience can be considered as model of inexperienced student depicted by the experiment. There is no special reward function for the student, but rather they are left to learn from demonstrations, specifying what is expected from him. As visible from results, imitation learning was successful. The average-able expert is not defined in this work but making a larger study with many experts can reveal what would be the best handling by averaging all approaches. If the goal was to simply mimic all of the expert's behavior, supervised learning could be utilized [Abbeel and Ng, 2004], however in presented system dynamics external factors, speed and traffic flows are different, so reinforcement learning is the best fitting machine learning method for heading control of sea surface vehicles.

The ability to maintain heading and/or course in dynamic environments is crucial for safe and collision-free navigation of sea surface vehicles. When maintaining heading and/or course, it is not only important to do it with as little deviation from the intended heading and/or course, but also to do it in a manner that will respect the limitation of the actuators installed onboard vehicles, while keeping the environmental influences at minimum by allowing for actuators to act economically or precise depending on the selected mode of operation. The goal of this section is to present a self-learning autopilot that successfully copes with environmental disturbances when maintaining heading or courses derived from following desired trajectories (passage plans designed by navigating officers).

An autopilot is modelled where navigator still plans routes and supervises the execution. However, this model can be easily tweaked so that it can be used in autonomous surface vehicles. Even though model development is commenced by learning from human operators, intention is not to cover all possibilities, environments, and scenarios to obtain good control policies, but rather to aid convergence of the reinforcement learning algorithm. After the initial demonstration learning and rewards shaping, reinforced learning is used to get the sequential control series that are infused in the model predictive controller (MPC). MPC actuates vehicle's rudder and, in case of autonomous vehicles, thrust. For fully actuated vehicles, as it is case in dynamic positioning, MPC could control thrust power for all propulsion units. In this thesis decision support system is considered, so thrust control can be integrated as suggestive alert, rather than actual direct control. In case of commercial surface vehicles, increasing speed and slowing down requires efforts from marine engineering department onboard to engage additional generators, adjusting cooling parameters, changing certain modes of the main engine, and similar, therefore, it is not possible to design an automatic controller, but rather have a suggestive action from the decision support system.

Literature shows that reinforced learning is successfully used on ground [Williams et al., 2017] and air [Tran et al., 2015] traffic systems, while sea surface vehicles have limited attention [Liu et al., 2016] mainly due to signal noise, dynamic environments, sensor hysteresis, but also quality of sensor equipment. In order to control cost, and because of perceived lower risk levels by administrations, lower quality sensor equipment is used onboard the commercial sea surface vehicles, which is one of the main factors why maritime industry is still not ready for full autonomy retrofit. Another difficulty that has to be considered is the cost of running experiments and exploring with full size vehicles. Most of the approaches in the area of marine control have been focused on proportional integral derivative (PID) controllers [Naeem, et al., 2012], while additionally contributions in linear quadratic controllers (LQR) [Lefeber, et al., 2003] and neural networks [Peng, et al., 2013] are presented, but limited literature in model predictive controllers (MPC) [Annamalai et al., 2015; Cui et al., 2019]. However, to the best of our knowledge, this is one of the first approaches that utilizes probabilistic sequential decision making with foraging optimization to design an intelligent marine auto pilot. Therefore, main contributions of this work is to propose an inverse RL method to develop initial database of weights and control sequences, after which RL solution is designed for online data collection, feature extraction and update of learning in the offline setting, and finally to evaluate performance of designed controller in simulated environment.

In this work the following framework is developed: firstly, features are extracted from human experts, this is then followed by the reward shaping, while the last part of the framework consists of execution and exploration part. It is necessary to keep in mind that the scope of this thesis is decision support system and not autonomy; however, slight adjustment is required to fit this model on surface vehicles without human supervision. Before further exploring the framework of the system dynamics, it is imperative to mention that most marine autopilots operate in several modes. Modes would vary between manufacturers, but the general idea is that there is an economy mode and at least one precision mode. Speed of surface vehicles is another important parameter, as speed largely influences control inputs to get the same outcome at the higher speed (larger rudder deflection is required when speeds are low). When these modes and limitations are taken into consideration, it is possible to see that the shape and dimension of the features relevant to proposed model will be slightly different, so reward functions depend on the mode and limitation imposed by the navigator. Speed of a sea surface vehicle could be a determining factor for modes and limitations selection by the autonomous agent in case of unhabituated vehicles. In this chapter the focus is autopilot design, so static and dynamic obstacle avoiding is omitted, which would be analyzed in the oncoming chapters.

The feature extraction part is used to extract state features such as referent position, true heading (after filtration), desired heading (either as a direct input from navigating officer, or calculated as per the desired course), and deviation from the desired track or trajectory. Various imperfect sensors observe these features; however, uncertainty of sensors is included in the filtering and collision avoidance sections, so it is assumed that the input is filtered before entering this model. To further improve accuracy of sensing input, extensive study covering measurement noise of particular manufacturer's sensor can be done and results implemented in this model. After feature extraction, learning from demonstration is used to learn desired rudder control from expert demonstrations. Demonstrations are used only to allow for a faster and more accurate algorithm convergence. After this step, reward shaping is used to tune the rewards that fit the desired task and to compute the optimal sequence of control. Finally, control is applied to rudder via MPC, while at the same time exploration and knowledge transfer is used to collect new data and to update the results of the simulation and in this way proposed autopilot is self-adapting and constantly learning. It is important to state that all learning is conducted offline, while exploration and rudder control happens online. In this way, computing power and seamless sampling efficiency is preserved.

3.3.6 Heading control

As mentioned earlier, an autopilot usually operates in either heading control mode or course control mode. When autopilot is used only to maintain certain heading, the vital information needed is the yaw rate. After the yaw rate is filtered utilizing FPF, filtered information is fed to the learning model. Unlike course control where the aim is to keep the vessel at certain course and adjust the heading to achieve precision, in heading control mode, autopilot tries to maintain a desired heading and allow for drift. In order to preserve actuators, it is necessary to compensate only for low frequency motion, while filtering out the high frequency yaw rate.

Solving a motion control of sea surface vehicles is not trivial; therefore, goal is to find a solution that is sustainable and effective. As author is a master mariner with 15 years of sea service experience, informal interviews with other navigators were made in regards of preferences when interacting with autopilots. Two features were commonly noted: 1) autopilot should resemble human-like control in order to maintain heading and course effectively without unnecessary overshooting when completing a turn, and 2) having a visual trajectory reference on RADAR and ECDIS is the clearest way of understanding future maneuvers. It is, therefore, prefered to model course and heading control as reinforced learning problem with exploration in the real-time environment, but with a prior knowledge of steering control developed through reward shaping based on human demonstration. When shaping reward is deemed too complex, inverse reinforcement learning techniques could be used to learn reward functions by extracting features of the model, but in this case, forming a scalable and efficient reward function is possible, so classic reinforcement learning is used. Shaping reward function takes additional efforts in the design stage, but once the final reward function is tuned correctly, model has lower dimension than what would be case with feature extraction. Higher dimensionality often leads to higher computational complexity and in order to preserve computing power, reward shaping through demonstrations approach is selected, as it requires fewer feature vectors.

Heading control is defined as an ability for the autopilot to maintain a heading selected by a navigator. That means that the aim of the heading control is to control the rudder deflection sufficiently to maintain the heading as close to the desired heading. In order to maintain desired heading, a progressive penalties approach is taken, where, depending on the economy or precision modes, an agent is allowed to stay within the certain parameters off the heading before initiating off-heading alarm. In presented example, a common setup is considered, where two steering pumps could be engaged. At open seas, the sea surface vehicle would use one steering pump, while in confined waters and during port approaches, selection would switch to both pumps running consecutively to ensure stability, redundancy and increase speed of response. Another selection is between ECONOMY and PRECISE modes of operations, where economy mode allows for a higher deviation from the desired heading in order to preserve actuators, while in precision mode, rudder deflects more frequently in order to preserve precision. Finally, rate of turn selection is possible in order to limit the speed of turn. In this example it is possible to select rate of turn of 5, 10, 15, 20, or 30 degrees per minute. The most challenging part of this approach remains how to implement the dynamic and kinematic abilities of the considered sea surface vehicle. This is the reason why artificial agents extract knowledge from sea trials and demonstrations data.

Most of the commercial sea surface vehicles are designed to have two separate steering systems, as steering is considered a critical equipment by class societies. Therefore, to ensure redundancy, two systems and two steering pumps are commonly installed onboard. A single system is used at open sea and both system in series are used in confined waters. Some of the commercial operators will have a policy that describes when navigators should use two systems simultaneously. Having two systems working in a series improves the speed of the rudder deflection, so SOLAS requirement for the deflection speed is usually done with two systems working together. For experiments and simulations, a model of an LNG Tanker with steam propulsion and redundant steering machinery is used, which has two electric motors for each steering system. Speed of deflection was measured onboard a real LNG Tanker that is identical to the model.

Implementation of the rudder deflection speed as a constraint is straightforward due to its linear nature. In the case of large underactuated sea surface vehicles, the period of deflection is short enough that it will not cause any noticeable yaw change until it deflects fully to the desired deflection angle, so it can be disregarded as a factor of influence. This might not be the case with smaller and lighter vehicles, for which analysis needs to be made and reward shaping utilized to address any yaw acceleration during the deflection from one position to another is done. It is imperative to mention that agent is given task to maintain a certain heading and disregards turning of a sea surface vehicle, which requires an experience or exploration mined knowledge that will serve as a guide to find a global maximum.

As the goal of heading control is to maintain heading as close as the desired heading, state space is defined as:

$$S = (H_d, H_F, ROT, R_d, M_{AP}, P_n) \tag{3.75}$$

where H_d represents a desired heading that is selected by a navigator, H_F is filtered heading that is received from the gyro compass and filtered by the FPF, $ROT = \{-30°, 30°\}$ is the Rate of Turn or yaw rate indication and could be measured or taken from the system installed onboard and is scaled from -30° to 30° with negative indicating port side (30° is the dynamics and kinematics limit considered vehicle has), $R_d = \{-35°, 35°\}$ is a rudder deflection indication required to ensure correct position of the rudder has been selected, $M_{AP} = \{ECONOMY, PRECISION\}$ represents autopilot modes and in this case these are economy and precision, but this could be extended to variety of precision levels, and $P_n = \{1P, 2P\}$, which stands for Pumps number and represents a selection of one or two steering pumps engaged at the moment of making a control decision. Considering that most of the autopilots allow only for the full degree scale selection, that the change in heading is sluggish, and that the representation of the gyro heading data is in 10th-s of a degree, higher accuracy is not required, so H_d and H_F are considered as discrete variables, including the rest of the tuple $ROT, M_{AP},$ and P_n. With heading control, we are mostly interested in immediate results; therefore, discounting of rewards can be ignored and focus maintained on short term performance, which reduces computational burden.

The action space is discrete in the case of underactuated sea surface vehicle, and taking in consideration deflection limits, there are two types of action spaces available:

$$A_{ECO} = \{-10°, -9°, -8°, \dots, -1°, 0°, 1°, \dots, 8°, 9°, 10°\} \tag{3.76}$$

$$A_{PRECISION} = \{-35°, -34°, -33°, \dots, -1°, 0°, 1°, \dots, 33°, 34°, 35°\} \tag{3.77}$$

The way how the autopilot will operate largely depends on the desired behavior from the designer of the system. One model is created, but all parameters could be fine-tuned as per the requirement of the navigator and type of the sea surface vehicle. Approach in this research is mostly based on the yaw rate, where the agent wants to achieve the set ROT as soon as possible, that is why reward function is designed in a way to promote larger rudder deflection and then gradually reduce deflection as the ROT (yaw rate) is approaching the set point, while discouraging overshooting.

For all autopilot training 16 scenarios of external disturbances are utilized as per Table 4.7, which is presented in the Appendix B. After training is done, behavior has to be simulated by introducing scenarios that are different than training scenarios in order to verify that training was successful. Therefore, 5 simulation scenarios are utilized as per Table 4.8 presented in the Appendix B. In the real world, training data would be collected during sea trials and then initial database built utilizing the sea trials data, training, and simulation. During exploitation a certain amount of exploration would be allowed and then buffer of knowledge developed according to the agreed algorithm.

Economy mode. The commercial shipping operators have to take in consideration fuel, consumables and spare part costs, while reducing the impact on the environment when sailing across the seas. Naturally, safety of the crew, sea surface vehicle, and environment plays the most important role, so in a case of anti-collision maneuvers, the control of a sea surface vehicle should be as precise as possible. However, in regular exploitation goal is to preserve steering system and allow for an additional play when controlling the heading. The economy mode is usually achieved by limiting ROT and deflection of the rudder. In this approach the main setup is maximum 10° of deflection and maximum 5° of ROT. However, a user can set this up as per his/her own wish through the interface page. In line with the above, following reward space is defined:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) + \alpha_3 R_3(S_i, A_j) + \alpha_4 R_4(S_i, A_j) + \alpha_5 R_5(S_i, A_j) + \alpha_6 R_6(S_i, A_j) \tag{3.78}$$

where

$$R_1(S_i, A_j) = \begin{cases} 20, & \text{if} \quad H = H_F \\ 0, & \text{otherwise} \end{cases} \tag{3.79}$$

$$R_2(S_i, A_j) = \begin{cases} 10, & \text{if} \quad H = -1° \leq H_F \leq 1° \\ -1, & \text{otherwise} \end{cases} \tag{3.80}$$

$$R_3(S_j, A_j) = \begin{cases} 1, & \text{if} \quad H = -2° \leq H_F \leq 2° \\ -10, & \text{otherwise} \end{cases} \tag{3.81}$$

$$R_4(S_i, A_j) = \begin{cases} 10, & \text{if} \quad ROT = ROT_{DES} \\ 5, & \text{if } ROT = ROT_{DES} - 5°/\text{min} < ROT_{DESIRED} < ROT_{DES} + 5°/\text{min} \\ -10, & \text{otherwise} \end{cases} \tag{3.82}$$

$$R_5(S_i, A_j) = \begin{cases} -1, & \text{if} \quad M_{AP} = ECONOMY \text{ and } ROT > 10°/\text{min} \\ 0, & \text{otherwise} \end{cases} \tag{3.83}$$

$$R_6(S_i, A_j) = \begin{cases} -1, & \text{if} \quad P_n = 1P \text{ and } \delta_R > 0.418 \text{ seconds} \\ -1, & \text{if} \quad P_n = 2P \text{ and } \delta_R > 0.2265 \text{ seconds} \\ 0, & \text{otherwise} \end{cases} \tag{3.84}$$

In such defined reward space, all R_i's are normalized. The R_1 rewards heading output that is identical to the heading requested by a navigator. The term R_2 penalizes autopilot to exceed 1° off the desired heading, while R_3 penalizes even stronger if the autopilot heading is off by 2° from the desired heading. As agent is using economy mode, this allows that autopilot has a regular off-heading without strongly intervening with rudder control to maintain desired heading. Using this approach, it is possible to setup reward and penalty points to get the desired outcome. The R_4 rewards autopilot to reach the desired yaw rate as soon as possible and penalizes if the yaw rate is different from the desired yaw rate by more than 5°. The R_5 penalizes yaw rate higher than 10° in order to minimize impact on the speed of the sea surface vehicle and avoid unnecessary strains on propulsion in order to compensate for the loss of speed as a consequence of larger yaw rates or rudder deflections. Finally, R_6 is mainly used to have a dynamic and time dependent trajectory generation, as the rate of deflection (δ_R) is one of the main limiting factors in steering underactuated sea surface vehicles. $\alpha_i \geq 0$ are weights that allow us to raise importance of certain component of the reward function when required.

Precision mode. When there is a need for a precise movement and heading control, most of the commercial autopilots allow for at least one precise setpoint. Some of the newer commercial models allow users to choose various levels of precision mode, as well as tuning parameters as per their own preference. In the proposed model, parameters are easily tunable to the preference of a user by tweaking the reward function. One of the possible settings is showcased, having in mind that selected sea surface vehicle model is an underactuated commercial vessel. During ocean passages economy mode is suitable; however, when in coastal waters, waters with higher traffic, during collision avoidance or when approaching/leaving ports, precision mode fits better. Heading control algorithm has an option of an automatic switching from economy to precision mode when risk of a collision is detected. In precision mode, full range of action space is allowed. Therefore, reward space is similar to (3.80), but the individual rewards are different:

$$R_1(S_i, A_j) = \begin{cases} 100, & \text{if} \quad H = H_F \\ -10, & \text{otherwise} \end{cases} \tag{3.85}$$

$$R_2(S_i, A_j) = \begin{cases} 10, & \text{if} \quad H = -0.5° \leq H_F \leq 0.5° \\ -1, & \text{otherwise} \end{cases} \tag{3.86}$$

$$R_3(S_j, A_j) = \begin{cases} 1, & \text{if} \quad H = -1° \leq H_F \leq 1° \\ -10, & \text{otherwise} \end{cases} \tag{3.87}$$

$$R_4(S_i, A_j) = \begin{cases} 10, & \text{if} \quad ROT = ROT_{DES} \\ 1, & \text{if } ROT = ROT_{DES} - 2°/\text{min} < ROT_{DES} < ROT_{DES} + 2°/\text{min} \\ -10, & \text{otherwise} \end{cases} \tag{3.88}$$

$$R_5(S_i, A_j) = \begin{cases} -1, & \text{if} \quad P_n = 1P \text{ and } \delta_R > 0.418 \text{ seconds} \\ -1, & \text{if} \quad P_n = 2P \text{ and } \delta_R > 0.2265 \text{ seconds} \\ 0, & \text{otherwise} \end{cases} \tag{3.89}$$

Therefore, the sum of all rewards is slightly different than in (3.78), as there is one less category that requires consideration:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) + \alpha_3 R_3(S_i, A_j) + \alpha_4 R_4(S_i, A_j) + \alpha_5 R_5(S_i, A_j) \tag{3.90}$$

The reward space of the precision mode is similar in nature to the economy mode, except that it has one component less, as it is less restrictive for the yaw moment. The description of the reward components is similar, so R_1 is still rewarding the heading accuracy; however, in this case the reward points are higher and there is a penalty for failing to accomplish this goal. The R_2, R_3, and R_4 are slightly tuned to promote a more precise steering outcome. Finally, the R_5 remains focused on dynamic and kinematic limitation of the steering system installed onboard the modelled sea surface craft.

Algorithm 3 – HEADING Control Algorithm *ECONOMY* and *PRECISION* modes

---

Input: States S, Actions $A_{ECONOMY}$ or $A_{PRECISION}$ depending on the user selection

Reward function $R : S \text{ x } A \rightarrow \mathbb{R}$, Importance weights $\alpha_i \geq 0$, Discounting $\gamma = 0.1$,

$M_{AP} = ECONOMY$ or $PRECISION$, $P_n = 1P$ or $2P$, ε-greedy factor 0.10,

Learning rate $\alpha = 0.5$.

Output: Q

1 Initialization: $H_d = H_F$, $ROT = 0$, $Q(s, a) = 0$.

for each episode (time horizon of 5 minutes) do:

Initialize s

Chose $a = 0°$ as first action.

for each step of episode do:

Take action a, observe R, s'

Choose a' from s' using policy derived from Q (ε-greedy)

$Q(s,a) \leftarrow Q(s,a) + \alpha\left[R\left(S_i, A_j\right) + \gamma Q(s',a') - Q(s,a)\right]$

$s \leftarrow s'$; $a \leftarrow a'$

end for when s is terminal

end for and **return** Q

end

The discount factor is $\gamma = 0.1$ as focus is on immediate rewards, while fixed learning rate is set $\alpha = 0.5$ as this problem is relatively deterministic and learning from new experiences is desired. The ε-greedy policy is tuned as in (3.86). As assumption is made that there are no previous experiences, the algorithm initializes with desired heading equal to the filtered heading and yaw rate equal to zero. This basically means that agent keeps rudder at midships and wait for the disturbance to act before allowing for correction. Even though turns could be handled by the same algorithm, in order to speed up convergence and ensure accuracy, turns are introduced as separate algorithm that takes human experts' knowledge and abilities into consideration when generating trajectories.

The results of training and simulations are presented in tables B.1 and B.2 of the Appendix B, and it is evident that both training and simulations have shown feasibility of the system. Heading control show success under various environmental loads.

3.3.7 Course control

Unlike the heading control where external influences were disregarded for maintaining the desired heading and drift was allowed, when designing autopilot that can maintain a true desired course, all external influences should be considered. This issue can be approached through expansion of the heading control algorithm and by introducing deviation from the desired "perfect" course in the state description, but it can also be assumed that the vessel is equipped with any type of IMU, as the availability of these sensors is now cost effective, so installing additional equipment onboard is not making a large burden to the commercial shipowners. With IMUs it is possible to track accelerations and by having a previous experience and/or simulation data available, it would assist greatly the algorithm to generate trajectories with higher accuracies. Also, exploration is then improved, as real-life use is supported by data that can improve learning and it is possible to directly compute influences of external disturbances to sway, yaw, or surge.

When solving collision avoidance problems, the quality of predicted paths and trajectories is of great importance. If external forces are taken into account, quality of predictions has a potential to be more accurate than predictions without external influences. Therefore, it is imperative to complete sea trials as accurate as possible to get a reliable data about sea surface vehicle's behavior in still water. Simulations could be then used to derive accelerations in three degrees of freedom (even more if the computational power is available) and then used as a set of knowledge for the algorithm. In the exploitation phase, it is possible to monitor accelerations and correct simulated data with the real-world experience. This would be approach when designing a model of a course keeping autopilot for an underactuated sea surface vehicle.

State space for course control autopilot is defined:

$$S = \begin{pmatrix} n, e, C_d, COG, H_{R,} H_F, ROT, S_R, SOG, RPM, u, v, r, R_d, \\ W_S, W_D, Curr_S, Curr_D, S_{STATE}, S_D, SW_H, SW_D, M_{AP1}, M_{AP2}, P_n \end{pmatrix} \tag{3.91}$$

Immediately it is noticeable that the state space for the course control is larger and more complex. It is necessary to state that higher dimensions would increase computational complexity, so it is necessary to include only the necessary vectors in state-action couples. The state tuple includes n and e, representing the North and East position that is received from the filtered and fused GNSS receiver. C_d represents a desired course, which is information derived from a route made in ECDIS, or requested by a user, COG is an actual course over ground a sea surface vehicle is doing at a certain time-step and it is used only as an indication, H_R represents a requested heading, which is a direct link between a course requested and a control of a sea surface vehicle. H_R is not requested by a user, but rather by calculating influence of external disturbances on maintaining a desired course. This is also either given by a system installed onboard, or system calculates influence on its own. Filtered heading $H_F$is still required in order to ensure appropriate rudder control. $ROT = \{-30°, 30°\}$ remains to represent the Rate of Turn or yaw rate indication and could be measured or taken from the system installed onboard and is scaled from -30° to 30° with negative indicating port side (30° is the dynamics and kinematics limit own vehicle has), SOG as speed over ground represents a surge rate and is measured by GNSS or speed log when ground is available, while RPM is included to link the speed of surface vehicle with revolutions per minute, which is standard input unit for marine propulsion. Related to RPM is S_R and this is one of the outputs from the anti-collision algorithm in order to optimize trajectories when speed reduction is required. RPM and S_R are not necessary for the course control algorithm; however, they are mentioned as the same algorithm will be used in the collision avoidance scenarios. $u, v,$ and r are known from the previous chapter and they represent surge, sway, and yaw velocities respectively. These are the velocities that would be received from an IMU sensor and use to calculate influence on a sea surface vehicle. If IMU is not available, current influence can be taken directly form the equipment installed onboard of a certain commercial vessel, or it would be possible to take the set calculated by the ARPA or ECDIS. $R_d = \{-35°, 35°\}$ is a rudder deflection indication required to ensure correct position of the rudder has been selected. The following state members $(W_S, W_D, Curr_S, Curr_D, S_{STATE}, S_D, SW_H, SW_D)$ are wind speed in knots, wind direction, the speed of a current in knots, direction of a current, sea state according to agreed scale, which also describes height and wavelength, sea waves direction, swell height, and swell direction. All these components influence $u, v,$ and r and are therefore important indicators. $M_{AP1} = \{ECONOMY, PRECISION\}$ represents autopilot modes and in this case these are economy and precision similarly as with heading control, but in the course control there is also $M_{AP2} = \{TRACK, GO\ TO\ WAYPOINT\}$, where track mode ensures that sea surface vehicle remains on the track planned by a navigator and laid on ECDIS, or that a shortest course to waypoint mode is used. The difference of these two modes is in the way how the control is ensured. In the track mode even a small deviation from the track will cause autopilot to achieve return to track by using as much rudder is possible, so the heading change can sometimes be substantial. Unlike the track mode, go to waypoint selection will allow the system to find the optimal path to reach the next waypoint and set course towards the next waypoint position. Finally, $P_n = \{1P, 2P\}$, is still a selection of one or two steering pumps engaged at the moment of making a control decision.

The action space remains the same, so it is now possible to tune rewards for specific tasks. At the beginning, a reward function is designed and then shaping functions developed in order to assist the autopilot in finding optimal trajectories. With course control, external disturbances are important, as to make accurate trajectory predictions, external disturbances have to be considered. Once a sea surface vehicle is produced and delivered, the first step is to ensure that sea trials are done properly and according to the classification societies' guidelines. As the data and sea trial modules are restrictive, simulator modeling of a vessel in question is proposed to gain as much as possible prior knowledge. Exploration without prior knowledge is very expensive, as the risk of collisions and groundings with a commercially exploited sea surface vehicle is restrictive. That is why detailed simulation exploration is conducted and the reason why results primarily focused on external influences on selected sea surface vehicles. This allows the course keeping algorithm to have a knowledge database, which would be updated with real-world data in order to improve accuracy of predictions. The course-keeping autopilot has a goal of maintaining desired course, selected wither by a user or demanded by a route developed as a passage plan from one position to another. By planning a route, a set of waypoints is developed, after which navigators have to ensure that a sea surface vehicle maintains its required course taking external disturbances into account. Interestingly, COG is calculated, so focus is maintained on heading rather than designing a reward function that will reward or penalize course differences, except for the auto-pilot mode.

Economy mode. As the autopilot modes have been described in heading control section, it is possible to continue straight to define a reward space as follows:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) + \alpha_3 R_3(S_i, A_j) + \alpha_4 R_4(S_i, A_j) + \alpha_5 R_5(S_i, A_j) + \alpha_6 R_6(S_i, A_j) + \alpha_7 R_7(S_i, A_j) + \alpha_8 R_8(S_i, A_j)\,, \tag{3.92}$$

where

$$R_1(S_i, A_j) = \begin{cases} 20, & \text{if} \quad H_R = H_F \\ 0, & \text{otherwise} \end{cases} \tag{3.93}$$

$$R_2(S_i, A_j) = \begin{cases} 10, & \text{if} \quad H_R = -1° \le H_F \le 1° \\ -1, & \text{otherwise} \end{cases} \tag{3.94}$$

$$R_3(S_j, A_j) = \begin{cases} 1, & \text{if} \quad H_R = -2° \le H_F \le 2° \\ -10, & \text{otherwise} \end{cases} \tag{3.95}$$

$$R_4(S_i, A_j) = \begin{cases} 10, & \text{if} \quad ROT = ROT_{DES} \\ 1, & \text{if}\; ROT = ROT_{DES} - 5°/\text{min} < ROT_{DES} < ROT_{DES} + 5°/\text{min} \\ -10, & \text{otherwise} \end{cases} \tag{3.96}$$

$$R_5(S_i, A_j) = \begin{cases} -1, & \text{if} \quad M_{AP1} = ECONOMY \text{ and } ROT > 10°/\text{min} \\ 0, & \text{otherwise} \end{cases} \tag{3.97}$$

$$R_6(S_i, A_j) = \begin{cases} -10, & \text{if} \quad M_{AP2} = TRACK \text{ and } n_A, e_A \ne n_{tF}, e_{tF} \\ 20, & \text{otherwise} \end{cases} \tag{3.98}$$

$$R_7(S_i, A_j) = \begin{cases} -10, & \text{if} \quad M_{AP2} = GO\ TO\ WP \text{ and } C_d \ne COG \\ 20, & \text{otherwise} \end{cases} \tag{3.99}$$

$$R_8(S_i, A_j) = \begin{cases} -1, & \text{if} \quad P_n = 1P \text{ and } \delta_R > 0.418 \text{ seconds} \\ -1, & \text{if} \quad P_n = 2P\ and\ \delta_R > 0.2265 \text{ seconds} \\ 0, & \text{otherwise} \end{cases} \tag{3.100}$$

Reward space is defined similarly like in heading control with two additional rewards. In (3.93), (3.94), and (3.95) the only difference is using H_R, which stands for heading requested by the course keeping autopilot. (3.96) and (3.97) remain the same as with heading control. However, (3.98) and (3.99) are a new addition and they are here to manage reward and penalty for course keeping. As mentioned before, autopilots usually have track keeping and go to waypoint modes. The (3.98) reward is designed to penalize movement from the track. That is why equality compares Actual N-E position, which is determined on a chart/ECDIS and the fused and/or filtered N-E position at the time step t. The (3.99) reward is connected to the go to waypoint mode, and it penalizes situations where desired course is different than the measured (via GNSS) course over ground. So this reward is in charge to maintain desired course towards the next waypoint regardless of the originally planned track. (3.100) reward remains the same as before. $\alpha_i \geq 0$ are weights that allow us to raise importance of certain component of the reward function when required. In this case, it is necessary to ensure that the $\alpha_6 = 0$ when user selects go to waypoint mode, as well as that $\alpha_7 = 0$ when user selects track mode.

Precision mode. Precision mode was also extensively explained in the heading control section, so it is possible to design the reward function as follows:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) + \alpha_3 R_3(S_i, A_j) + \alpha_4 R_4(S_i, A_j) + \alpha_5 R_5(S_i, A_j) + \alpha_6 R_6(S_i, A_j) + \alpha_7 R_7(S_i, A_j) \tag{3.101}$$

where

$$R_1(S_i, A_j) = \begin{cases} 100, & \text{if} \quad H_R = H_F \\ -10, & \text{otherwise} \end{cases} \tag{3.102}$$

$$R_2(S_i, A_j) = \begin{cases} 10, & \text{if} \quad H_R = -0.5° \leq H_F \leq 0.5° \\ -1, & \text{otherwise} \end{cases} \tag{3.103}$$

$$R_3(S_j, A_j) = \begin{cases} 1, & \text{if} \quad H_R = -1° \leq H_F \leq 1° \\ -10, & \text{otherwise} \end{cases} \tag{3.104}$$

$$R_4(S_i, A_j) = \begin{cases} 10, & \text{if} \quad ROT = ROT_{DES} \\ 1, & \text{if } ROT = ROT_{DES} - 5°/\text{min} < ROT_{DES} < ROT_{DES} + 5°/\text{min} \\ -10, & \text{otherwise} \end{cases} \tag{3.105}$$

$$R_5(S_i, A_j) = \begin{cases} -10, & \text{if} \quad M_{AP2} = TRACK \text{ and } n_A, e_A \neq n_{tF}, e_{tF} \\ 50, & \text{otherwise} \end{cases} \tag{3.106}$$

$$R_6(S_i, A_j) = \begin{cases} -10, & \text{if} \quad M_{AP2} = GO\ TO\ WP \text{ and } C_d \neq COG \\ 50, & \text{otherwise} \end{cases} \tag{3.107}$$

$$R_7(S_i, A_j) = \begin{cases} -1, & \text{if} \quad P_n = 1P \text{ and } \delta_R > 0.418 \text{ seconds} \\ -1, & \text{if} \quad P_n = 2P \text{ and } \delta_R > 0.2265 \text{ seconds} \\ 0, & \text{otherwise} \end{cases} \tag{3.108}$$

The precision mode is again similar to the economy mode with a difference in yaw rate restriction. Similarly as in the heading control model, reward points are altered in order to ensure precision and the reward related to the economy mode is removed. Rewards (4.107) and (4.108) also have higher reward points in order to focus on precision.

Before defining an algorithm for course control, reward shaping is added. First shaping is to ensure stability and precision of trajectories, while the second shaping is in charge of infusing previous knowledge and simulated data in the algorithm in order to avoid local maxima.

If we recall how shaping was done in the simple example depicted in the previous chapter, distance shaping and action shaping is considered. In the instance of course control autopilot modeling, it is unnecessary to shape actions. Unlike the simple example where actions were directions of movement, in the autopilot design actions are rudder deflections and the main reward space takes care of performance limits. Therefore, heading shaping is considered instead of action shaping. When thinking of heading, it is envisioned that rarely there would be a situation where vessel would need to steer 180° away from the present heading. That is why progressive heading shaping reward is introduced:

$$
\Phi(H_R) = \begin{cases} -8 & \text{if } (H_R' - H_R) = [-45°, -30°] \text{ or } [30°, 45°] \\ -6 & \text{if } (H_R' - H_R) = [-90°, -45°) \text{ or } (45°, 90°] \\ -4 & \text{if } (H_R' - H_R) = [-135°, -90°) \text{ or } (90°, 135°] \\ 0 & \text{if } (H_R' - H_R) = [-180°, -135°) \text{ or } (135°, 180°] \\ -10 & \text{otherwise} \end{cases} \tag{3.109}
$$

This shaping has a goal of promoting the continuation of the same heading. It is not significantly punishing alteration of the course, but when keeping in mind that this tuning is still done for a large underactuated sea surface vehicle, altering course always takes time and to find optimal solution which is in the other direction of the current course should be made expensive. In this sense, it is necessary to avoid that agents are looking for a favorable solution in the state space which require a huge impact on performance and stability. Certainly, if there is no better solution than to turn the vessel to the opposite direction, it will be proposed by the system, but it should be one of the last resorts.

The other part of this shaping function is the distance shaping and as it was shown in Cui et al. [2019], the Euclidian distance is more effective than Mahalanobis distance for direct measurements. Next section shows how Mahalanobis distance can play a vital role when tuning reward shaping from demonstrations, but for the direct use on measurements, Euclidian distance is the approach warrantying better results. It is difficult to find viable data with which uncertainty could be modeled similarly to presented basic navigation experiment, where assumption was made that in 10 % of cases behavior of own agent or other agents will be out of ordinary. In the real world, marine autopilots rarely make mistakes that would need attention, unless when considering overshooting a desired heading or course. However, overshooting has been covered by designing a main reward function penalizing that kind of behavior. The major uncertainty is actually with the decision making of navigators in collision situations, but this will be modeled when designing an algorithm to avoid collisions. In the case of autopilot, the biggest problem an agent could face is unavailability of the steering gear due to some failure; for which a backup system could be used automatically, and the autopilot would then try to engage auxiliary systems available or give an audio warning to the operator that Non-Follow-Up

(NFU) system should be in use. In line with this, distance uncertainty is used as shaping function by measuring Euclidean distance between the predicted mean of a sea surface vehicle's position in the time-step when own agent should be at the next waypoint (n_P, e_P) and the actual position of the next waypoint (n_A, e_A), where P stands for predicted:

$$\Phi(s) = -\left(\frac{1}{2}\|(n_P, e_P) - (n_A, e_A)\|^2\right) \tag{3.110}$$

In line with the (3.72), shaping potential looks like:

$$\Phi(s, a) = \Phi(s) + \Phi(H_R) \tag{3.111}$$

or:

$$\Phi(s, a) = -\left(\frac{1}{2}\|(n_P, e_P) - (n_A, e_A)\|^2\right) + \Phi(H_R) \tag{3.112}$$

With shaping potential defined this way, reward shaping function that will be used in the final algorithm is as follows:

$$F(s, a, s', a') = \gamma\Phi(s', a') - \Phi(s, a)$$

Before the course keeping algorithm can be presented, it is necessary to learn how external disturbances affect the three degrees of freedom. Assumption is made that IMU is available onboard a sea surface vehicle. If not, algorithm can work without motion vectors, but it would require longer exploration as the predictions would have a certain degree of offset. Initially, a model of a sea surface vehicle and a certified simulator is required to get the motion behavior. However, once in exploitation, a motion data is collected, and the knowledge buffer updated with real-world data for future use.

The way how motion vectors are introduced is by vector arithmetic. There is a base motion vector when a sea surface vehicle is going through a calm water with no influence of external disturbances. Then just by adding components of surge, sway, and yaw, a new resultant vector can be calculated considering all external disturbances. In that sense, proposed algorithm can at all times predict trajectories better, even when in turns or going into another weather condition. As collected data is mostly csv and excel based, with simple trigonometric manipulation all compass-based data is transferred to cartesian equivalents, which allows to utilize this data for algebraic treatment. Once the external disturbance data has been evaluated, it is possible to design an algorithm for course control as follows:

Algorithm 4 – COURSE Control Algorithm *ECONOMY* and *PRECISION* modes

---

Input: States S, Actions $A_{ECONOMY}$ or $A_{PRECISION}$ depending on the user selection

Reward function $R : S \text{ x } A \to \mathbb{R}$, Importance weights $\alpha_i \geq 0$, Discounting $\gamma = 0.5$,

$M_{AP1} = ECONOMY$ or $PRECISION$, $P_n = 1P$ or $2P$, ε-greedy factor 0.10,

Learning rate $\alpha = 0.4$, $M_{AP2} = TRACK$ or $GO\ TO\ WAYPOINT$.

Output: Q

Initialization: $H_R = H_F$, $S_R = SOG$, $Q(s, a) = 0$.
for each episode (time horizon of 5 minutes with lookahead until next WP) do:
    Initialize s
    Read sensory information and update s with updated motion vectors
    According to motion vectors update COG and C_d
    Chose $a = 0°$ as first action.
    for each step of episode do:
        Take action a, observe R, s'
        Choose a' from s' using policy derived from Q (ε-greedy)
        Determine shaping functions $\Phi(s, a) = \Phi(s) + \Phi(H_R)$
        Calculate $F(s, a, s', a') = \gamma\Phi(s', a') - \Phi(s, a)$,
            where $\gamma = 0.1$
        Calculate $R_F(s, a, s', a') = R\left(S_i, A_j\right) + F(s, a, s', a')$
        $Q(s, a) \leftarrow Q(s, a) + \alpha[R_F + \gamma Q(s', a') - Q(s, a)]$
        $s \leftarrow s'$; $a \leftarrow a'$
    end for when s is terminal
end for and **return** Q
end

In this instance the discount factor is $\gamma = 0.5$ as higher emphasis is required on later rewards, while maintaining fixed learning rate $\alpha = 0.4$, because this problem is still deterministic, but more exploitation is necessary to maintain compactness of the predicted trajectories. The ε-greedy policy is tuned as in (3.89). Updating sensory information is one of the most important steps of the course control autopilot. Wheel amidships is still a first action, as start from the system stability is assumed. This can be tuned by allowing algorithm to follow another ε-greedy policy in the prior that will select optimal action without exploration.

Course control training was done similarly as for heading control and the results showed that both systems managed to thrive under environmental loads, which is evident in tables B.5 and B.6 of the Appendix B. Together with the turn control and auto-telegraph, course and heading controls are crucial to ensure that collision avoidance trajectories are feasible and optimal for the reward space relevant in that moment.

The course control algorithm delivers an open loop sequence of control actions. Foraging Particle Filter was used to reduce uncertainty of input signals; hence, the trajectory prediction is delivered with intrinsic uncertainty. Additional uncertainty would be behavior of users, but this will be covered in collision avoidance chapter. Any controller or actuator can be used to utilize the control action open loop sequence, which will directly move the sea surface's rudder in the required position. Utilization of the model predictive control framework is proposed. Only the framework is used, and the word "model" is disregarded in the name of the MPC. The reward function with reward shaping have delivered the optimal open-loop trajectory for the selected horizon. The next step would be to convey the open loop sequence to an implicit feedback controller. MPC framework is used, which allows us to obtain an implicit closed loop controller by updating the control sequence at each following state. The MPC framework allows that in the finite horizon open loop control sequence only the current state is used to get the sequence and then only the first control in that sequence is applied to the system. MPC is great in handling various degrees of constraints [Mayne et al., 2000], so the framework is utilized, while constraints remain in the reward space. Mayne et al. [2000] have extensively proved stability of the MPC framework mainly by using the value function of a finite horizon control sequence as Lyapunov function to enhance stability.

Course control autopilot algorithm would be sufficient for controlling a sea surface vehicle for the entirety of passage plan. However, with proposed model free approach, introduction of prior knowledge is required to aid convergence of the algorithm when there are turns at various speeds. For this reason, turns are investigated as a separate algorithm that is interconnected to both heading and course control autopilot models.

3.3.8 Yaw control

Yaw control is a challenge on its own due to the fact that changes in directions of external disturbances and the inertia of a sea surface vehicle occur rapidly and it is hard to find optimal trajectory in this way. When considering heading control, turning is usually done with a preset of maximum ROT, while the track or waypoint autopilots already have route developed by a navigator, so that change in course is known well in advance. Course control and heading control algorithms would be able to handle turns, but the learning horizon is shortened, and this is the reason why demonstration is included as a knowledge base for the algorithm to seek optimality in limited state-action space to ensure optimal time of convergence.

Before designing a reward space, it is imperative to know that for the planned voyage, all waypoint turns will be known in advance. This provides us with an opportunity to pre-calculate turn radiuses and ROT required to achieve those turns. Among other factors, the available depth is a significant factor influencing the size of a turning circle, so in practice deep water and shallow water behavior is usually considered. One of the key elements of successful passage planning is determination of the Wheel-Over Position (WOP). This is the position where deflection of rudder should commence in order to execute a turn without overshooting or turning short. Many factors influence decision on where the WOP should be, such are sea surface vehicle's maneuvering characteristics, available depth and width, while the execution is mainly affected by appropriate maneuvering timing and external disturbances. In the extensive study conducted by Ugurlu et al. [2015] lack of communication of the bridge team, improper passage planning, position fixing errors, faulty maneuvering and interpretation errors, fatigue and unfamiliarity with bridge equipment are main root causes identified for analyzed groundings.

In the digital era there is an advantage of visualizing own passage in geometric form throughout all segments of a voyage. With paper charts all turns were a sequence of consecutive straight lines, while with ECDIS there is an option of selecting only one waypoint that will allow for more natural curvature representation on the display.

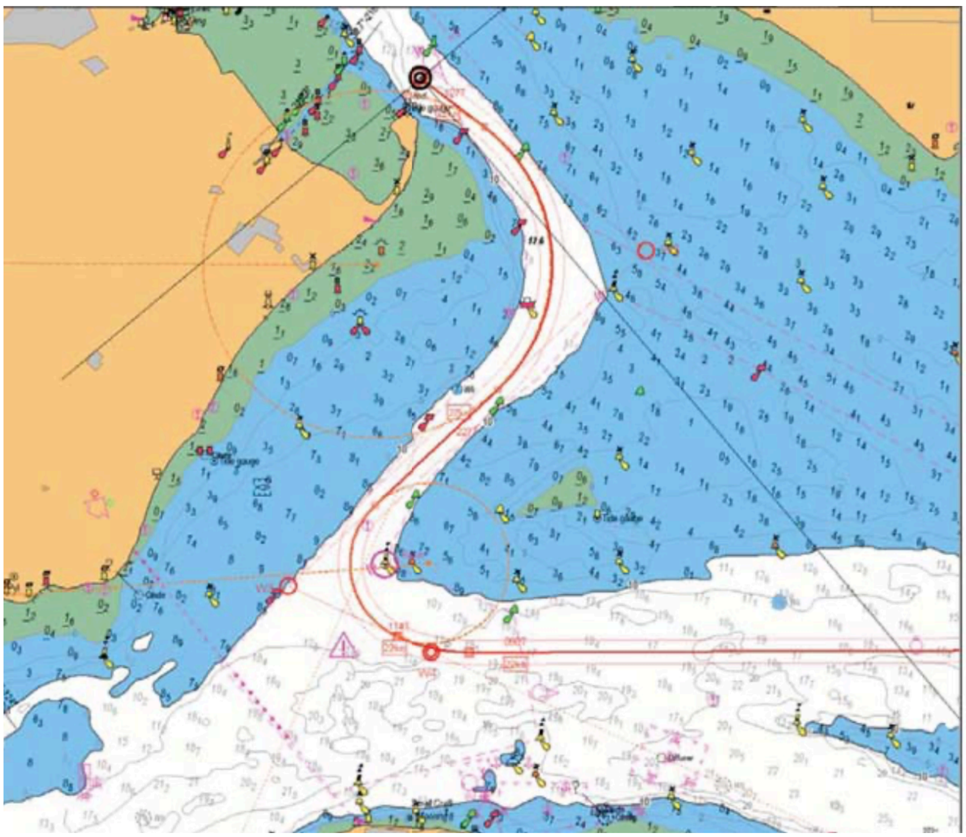


Figure 3.35 – ECDIS turn excerpt

As it is visible from above chart visualization, the actual waypoint is located in the shallow water, but the ECDIS will offer geometric representation as a curve that sea surface vehicle has to follow in order to stay within the safe waters. In the planning stage, a navigator has to adjust the radius and confirm visually on the electronic chart that the turn is safe. This radius is then taken as an input in determining what will be the actual Rate Of

Turn (ROT) to accomplish this turn with a certain sailing speed.

As there are no separate regulations for autonomous vehicles yet, all sea surface vehicles have to comply with certain chapters of the International Convention for the Safety of Life at Sea (SOLAS). Depending on the size and displacement of sea surface vehicle, different parts of the SOLAS convention apply. In this research focus is still maintained on underactuated commercial sea surface vehicles and most of them have to comply with the same minimum requirements in regard to passage planning and safety of navigation. All sea surface vehicles have to comply with the Chapter V "Safety of navigation", regardless of the size on all voyages. The Chapter V delivers requirements about how the passage plan should be made and progress monitored throughout the entire voyage. It emphasizes requirement for a navigator to foresee navigational hazards and adverse weather conditions along the way and takes risk averse approach when plotting a navigational path to ensure sea surface vehicles are clear of navigational and environmental hazards. Voyage plans have to be comprehensive and cover all movements from berth to berth. The most important requirements for this research are related to the importance of considering safe speed when planning a voyage, to ensure enough under-keel clearance at all segments of the voyage, to plot the intended route on appropriate scale charts and specifically to mark all course alteration points keeping in mind the vehicles' turning circle at planning speeds. Eternal disturbances are hard to predict at the beginning of a long voyage, apart from currents that could be predicted with fair accuracy well in advance. It is, therefore, a common practice and usually a requirement from commercial operators that wheel over positions (position where rudder deflection has to commence in order to overcome vehicle's inertia) and turn radii are calculated and included in passage plans.

After completion of a passage plan, navigator will have a list of all turns that will happen along a route. This is why with electronic charts a list of all radii of turn is maintained, as ECDIS allows for geometric display of turns. ECDIS determines the radius utilizing simple geometric computation of circle size knowing the two course lines that act as tangent to a circle and knowing the angle between two courses. When turn radius is known by a navigator, calculation of ROT is done by utilizing circle formula:

$$ROT = \frac{180v}{60\pi r} \approx \frac{SOG}{r} \ \ [°/\mathrm{min}] \tag{3.113}$$

The $\pi = 3$ approximation is good enough for the ROT determination, considering that commercial ROT indicators usually use a 1° scale. Aim is to maintain consistent ROT during a turn to ensure sea surface vehicle is turning according to predetermined radius of a turn, for which a constant vehicle's speed is required. In this case, an algorithm will read speed sensor and adjust the required ROT throughout the turn in order to ensure that any drop or gain of speed caused by a turn or change in external disturbances is compensated for and that the turn is done according to a planned path curvature.

Therefore, a navigator should ensure that the turning circle is hazard free, that the ROT is not excessive to cause damage to a cargo, sea surface vehicle, environment or people, and to ensure that turning circles of neighboring WPs do not overlap. Finding two points of tangency is not enough to safely execute a turn. Appropriate wheel-over position is required to overcome the inertia of a particular sea surface vehicle. This, again, proves that difficulty in marine control is the highly dynamic nature of sea surface vehicles' design and environmental conditions at a moment of control. This is why reinforcement learning fits well, as it allows for solving highly dynamic and high-dimensional problems with model-free approach.

Maneuvering characteristics of any sea surface vehicle will depend on the amount of water available below the keel. Lower UKC will influence turnings to be wider simply because there is no space for water to be displaced underneath a sea surface vehicle. There are many factors influencing maneuverability (Vujicic et al., 2018) and the scope of this research is not to investigate all of them in detail; developing a base for algorithmic approach that will be able to learn turns and perfect its execution during exploitation is of interest. For the proposed algorithm, it is assumed that the speed of a propeller and SOG is in the expected region excluding external disturbances. This means that speed stability is sought to ensure that agent does not face higher turning speed, which is available when the propeller's rotational speed is larger than the vehicle's speed. We can think of leaving an anchorage and that vehicle is stationary. In that situation sea surface vehicle will "happily" turn when higher revolutions are commanded from a navigator.

As depicted in Figure 3.36, sea surface vehicle's turn happens in three phases. Phase I commences once the rudder is deflected to the side where vehicle should turn. Initially, a sea surface vehicle would experience a yaw moment to the opposite side of a turn until the moment of vehicle's mass inertia is overcome. The impact of this moment will depend on the shape of a hull and draught of a sea surface vehicle. Wider vehicles with shallower draughts are more influenced by the opposite yaw moment in the first phase. The phase II commences when the bow starts to yaw to the desired direction and when the rudder force overcomes the moment of mass inertia. In this phase reduction in sea surface vehicle's speed due to strong lateral resistance is noticed. The phase III depends on the characteristics of a vehicle, and it usually commences after 100° to 120° heading alteration. Assuming there are no external disturbances, in this stage vehicle slows down and its speed, radius and ROT become constant. It is evident that the speed of a sea surface vehicle will drop during a turn and that is why ROT has to be adjusted throughout the turn to execute the turn according to the planned turn radius. In its resolution MSC.137(76) [2009], International Maritime Organization (IMO) delivered standards for testing maneuvering characteristics of new built vessels. The sea trials are made with 85 % of the engine output, on the even keel and in the deep sea. The standard is that the advance should not exceed 4.5 vehicle's length and that the tactical diameter should not exceed 5 vehicle's length when the turn is done with constant 35° rudder deflection. As per Vujicic et al. [2018], the deep sea for turns is 4xdraught and if the depth is less than this, then diameter increases but not more than 8 vehicle's length.

As mentioned earlier, a sea surface vehicle will not immediately start a turn, but rather even turn to opposite direction initially, so it is paramount to find the moment where the deflection of a rudder should happen in order to execute turn as per planned radius.

The sea trial results usually offer turning circles with 35°, 20°, and 10° rudder deflections. In the following diagram A_0 is the wheel-over position. What will be the distance from the actual yaw moment to the deflection side depends on pivot point (for practical point it is assumed that it is ¼ of the vehicle's length from the bow and it changes to 1/3 from the bow in the phase II), external disturbances, and maneuvering characteristics. Therefore, a lot of factor influence turning performance and wheel-over position. For the purpose of the proposed algorithm, approximation method is used by taking a tangent to a turning circle that corresponds to a rudder deflection. The progress on an arc of a turning circle will not be unison rudder deflection, but rather it is possible to envision approach where larger deflection is given initially to overcome inertia as soon as possible and then control the desired ROT depending on the achievable speed. On short turns, speed drop will not be relevant, but on larger turns it will be a significant factor.

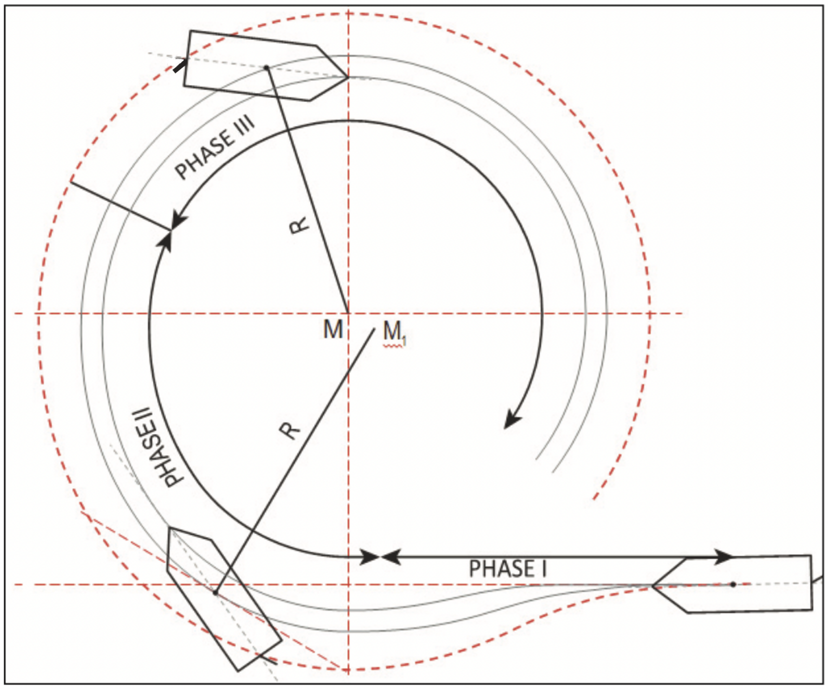


Figure 3.36 – Three phases of the ship's turn (Source: Modified from Vujicic et al., 2018)

Because of the complexity of turning a sea surface vehicle, this approach was to have a separate algorithm that handles only the turns. In this regard, whenever the change of course or heading is more than 5°, autopilot immediately turns to the turning mode and tries to complete a turn with least deviation possible until a sea surface vehicle is either on a following track, keeping desired heading after the turn, or maintains desired course towards the next waypoint. Integrated on the RADAR or ECDIS screen, there could be an option of graphically showing the turn with selected radius, so when the turning circle coincides with the planned circle, navigator would select execute and the turn is done accordingly. Therefore, autopilot can be programmed in a way that turns are done following next heading, next course, or next radius. Keeping this in mind, and ensuring the lean approach, yawing state space is defined as follows:

$$S = \begin{pmatrix} n_F,\ e_F, n_R,\ e_R, r, VRM, C_d, COG, H_{R,}\ H_F, ROT_D, ROT_A, \\ d_{LOG}, SOG, RPM, u, v, r, R_d, W_S, W_D, Curr_S, Curr_D, P_n \end{pmatrix} \tag{3.114}$$

It is necessary to emphasize again that not all of the state members are members of the SARSA calculation, but rather describe the state at every time step. Most of the state members were described previously. Added are $n_F,\ e_F$, which represent filtered and fused North-East position, $n_R,\ e_R$, which represent North-East position of the planned curve on route, while r represents planned turn radius that is extracted from a passage plan. VRM stands for Variable Range Marker and it is part of RADAR. This measurement is used to track performance in relation to the desired turn radius. d_{LOG} is a log depth and it is important for calculating the shallow water effect, as turning in shallow or deep water is different. $ROT_{D\ \&\ A} = \{-30, 30\}$ stand for rate of turn demanded and rate of turn actual. ROT is determined by the (3.126) expression and is a variable once the turn commences as the SOG is expected to drop if the turn is large enough. Success of a turn is rated by maintaining position as close as possible to the planned curve. This is accomplished by comparing the position of the planned curve of a route and actual position of a sea surface vehicle. There is no distinction between economy and precise modes, nor there is a track or go to waypoint mode, as these are irrelevant for successful completion of a turn.

The action space remains the same. As the study is focused on underactuated vehicles, rudder control is of interest. Propulsion RPM control is possible, but this will be covered in the next chapter, as it is not as straight forward as it would be with propulsion systems covering multiple degrees of freedom.

As the autopilot modes are irrelevant for turning mode, it is possible to define a reward space as follows:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) + \alpha_3 R_3(S_i, A_j) + \alpha_4 R_4(S_i, A_j) \tag{3.115}$$

where

$$R_1(S_i, A_j) = \begin{cases} -10, & \text{if} \quad n_R, e_R \neq n_F, e_F \\ 100, & \text{otherwise} \end{cases} \tag{3.116}$$

$$R_2(S_i, A_j) = \begin{cases} 50, & \text{if} \quad ROT_A = ROT_D \\ 10, & \text{if } ROT = ROT_D - 2°/\text{min} < ROT_D < ROT_D + 2°/\text{min} \\ -10, & \text{otherwise} \end{cases} \tag{3.117}$$

$$R_3(S_i, A_j) = \begin{cases} 20, & \text{if} \quad VRM\ to\ (n_r, e_r) = r \\ 10, & \text{if} \quad VRM\ to\ (n_r, e_r) = r - 0.1(r) < r < r + 0.1(r) \\ -10, & \text{otherwise} \end{cases} \tag{3.118}$$

$$R_4(S_i, A_j) = \begin{cases} -1, & \text{if} \quad P_n = 1P \text{ and } \delta_R > 0.418 \text{ seconds} \\ -1, & \text{if} \quad P_n = 2P \text{ and } \delta_R > 0.2265 \text{ seconds} \\ 0, & \text{otherwise} \end{cases} \tag{3.119}$$

In case of a yawing mode, it is noticeable that the reward space is somewhat smaller. This is because even though turns are difficult to perfect from the maneuverability point of view, the goal of successful turn is simple to describe. The reward (3.116) rewards the agent whenever the filtered position of a sea surface vehicle equals to the position of a course turn track. With this reward penalty is assigned whenever the vehicle is not on a track. Certainly, there is a GNSS position discrepancy, so filtering and fusion of GNSS signals is employed to get an accuracy to the highest level. Wherever available, a Real-Time Kinematic (RTK) system can be used. Confined, shallow and busy waterways are usually situated near coastal areas, where RTK availability could be beneficial, so the accuracy of ownship positioning can be greatly enhanced. The reward (3.117) allows for a slight slack in precision of ROT, as it is hard to get fully accurate result considering dynamic nature of sea surface vehicles' movement. However, this reward also penalizes ROT that is more than 2°/min higher or lower than the desired ROT. The (3.118) reward expression is meant to be as a backup in case that reliability of the GNSS system is not good at a certain moment. In this reward ARPA is used to determine distance from the imaginative fixed point. This point is assigned when planning a voyage, as this is the time own turn radius is set up. The selected turn radius will point towards the center of a turning circle and that position is known as radius North-East position (n_r, e_r). As turn should be executed following the preset radius, own vehicle will be turning on the arc of a turning circle, so the distance from the (n_r, e_r) should be equal throughout the turn. This gives us an opportunity to track the distance with VRM and measure performance. Therefore, the agent is rewarded for any instance the VRM range is equal to radius and rewarded slightly less if the VRM stays within the 10 % of the radius. For any other position, the agent gets punishment. The (3.121) reward is the same reward as in other modes ensuring appropriate performance constraints of a rudder system.

To kick start a reinforced learning process and narrow down the search for optimized trajectories, three stages of learning are used: learning from demonstrations part, execution part, and learning from execution part. In this sequence, ability and knowledge of experts is used to guide algorithm in finding the optimal space. Experts simulate scenarios that would be helpful in future (various degrees of turns to the port and starboard side). These demonstrations are imperfect and sometimes suboptimal; however, they aid to narrow down the search space for optimality. This is the reason why expert demonstration state-action pairs are stored in the Expert Demonstration buffer, as these state-action trajectories can later on be compared with execution state-action pairs and replaced if the execution trajectories were closer to the optimal space. This can also be done by designing Expert Demonstration buffer, Execution buffer and Best Buffer. The best trajectories are stored in the Best Buffer, which would then be used in the future as a guide to optimality.

Once all expert demonstration data has been collected, it is necessary to find a suitable way to implement expert demonstration in the optimization algorithm. Approach is to use demonstration as a shaping function in order to guide trajectory optimization by narrowing the space search. Within the framework of reinforcement learning and hidden Markov models in general, aim is to find an optimized policy that artificial agent can follow and execute a required task. In the presented case, interest lies within a part of policy that is known as trajectory. Trajectory will provide a sequence of actions to control own sea surface vehicle. The expert demonstration data is collected as a sequence of state-action pairs $[(s_0, a_0), \dots, (s_n, a_n)]$, which is then stored to the Expert Demonstration buffer B_{ED} and used when required by the autopilot algorithm as a shaping reward or bias. The central question remains how to compare expert demonstration and exploration data, and selected option is similarity metric that is derived from multi-variate Gaussian distribution and Mahalanobis distance.

What makes proposed approach somewhat easier to describe is the discrete action space. Also, considering that input data will be centralized in one single table, slowdown of normalization can be avoided, and non-normalized multivariate Gaussian distribution used to find similarity between space-action pairs. Intuitively, if two states have different actions, similarity is zero, while otherwise Gaussian similarity is used to quantify the similarity and use it for comparison of pairwise data. Considering high dimensionality of state space, multivariate normal distribution fits well as it can generalize one-dimensional Gauss distributions to higher dimensions.

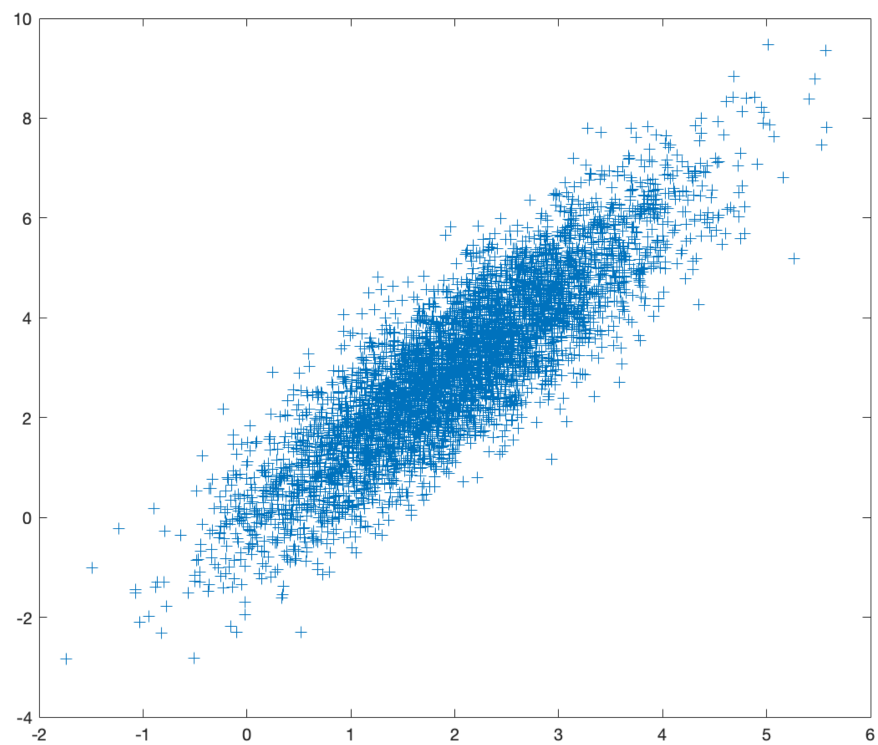


Figure 3.37 – Generation of random numbers from a multivariate normal distribution in Matlab

As it is visible from a multivariate Gaussian distribution example above, the fitting data is bounded by an elliptically shaped high probability region, so Mahalonobis distance can be utilized as the descriptive statistics representing the distance to the testing state

s_t from the expert demonstrated state s_{ED}, usually representing the mean:

$$g(s_t, s_{ED}, \Sigma) = \exp\left\{-\frac{1}{2}(s_t - s_{ED})^T \Sigma^{-1} (s_t - s_{ED})\right\} \tag{3.120}$$

In line with the (3.120) expression, when $s_t = s_{ED}$, similarity is 1, while it will be closer to zero as the dissimilarity rises. As stated earlier, it will be zero once the action is not the same in both states. In this case, both state s_t and state s_{ED} have to contain same action pair in order to be considered for similarity. The covariance Σ is responsible to measure the influence of demonstrated state-action pairs, and in order to speed up the computation, state space can be normalized to $[0,1]$ and, similarly as in Brys et al. [2015], Σ defined in the form of identity matrix multiplied by a constant σ, or $\Sigma = \sigma I$. Then, highest shaping potential is calculated by finding the expert demonstration with the highest similarity among the state-action pairs that have the same action associated with them:

$$\Phi_{ED}(s, a) = \max_{s_d, a} g(s_t, s_{ED}, \Sigma) \tag{3.121}$$

This potential function is then used to model a shaping function that will be integrated with the main reward function in a similar way as it was done in course keeping autopilot:

$$F_{ED}(s, a, s', a') = \gamma \Phi_{ED}(s', a') - \Phi_{ED}(s, a) \tag{3.122}$$

Finally, the shaping is then added to (3.117) and the complete reward function with shaping has a form as follows:

$$R_{F_{ED}} = R(S_i, A_j) + F_{ED}(s, a, s', a') \tag{3.123}$$

As it was evident in previous algorithms, assumption was made that the first action $a_0 = 0°$ was “wheel amidships”, as there was no previous knowledge. In this case, previous knowledge is available, so initialization of first action similarly like in Wiewiora et al. [2003] is possible by utilizing the expert demonstration in the initial Q-value:

$$Q_0(s, a) = \Phi_{ED}(s, a) \tag{3.124}$$

Finding action in this instance is easy and special optimization technique is not required, but rather taking an action that can be found in an expert demonstration state which is the most similar to initial state.

Similarly like with utilizing expert demonstrations for turning trajectory optimization, advantage can be taken of the collected data after exploitation. This data is stored, compared with demonstrated data, and then it is possible to select which one is better in order to speed up and enhance further trajectory optimization. Same approach can be used for other autopilot modes when collecting a real-world data. In such a way, search space size can be reduced and optimization time improved. Hence, in addition to B_{ED}, we can also envision B_{EX} as an exploitation buffer and B_{BEST} as a third buffer that will store the best of the exploitation and expert demonstration buffers.

Algorithm 5 – YAW Control Algorithm (Expert Demonstration Shaping only)

Input: States S, Action $A_{PRECISION}$

Reward function $R : S \text{ x } A \rightarrow \mathbb{R}$, Importance weights $\alpha_i \geq 0$, Discounting $\gamma = 0.5$,

$P_n = 1P$ or $2P$, ε-greedy factor 0.10, Learning rate $\alpha = 0.1$

Output: Q

1 Initialization: $H_R = H_F$, $S_R = SOG$, $Q(s,a) = 0$ or $Q_0(s,a) = \Phi_{ED}(s,a)$, Determine required radius r for the turn and plot the position of the radius center (n_r, e_r), Initialize required ROT with (3.124) using planned SOG Read depth d_{LOG}. If $d_{LOG} = N/A$, assume deep waters. If $d_{LOG} < 8(draught)$ consider shallow water data table. If $d_{LOG} \geq 4(draught)$ consider deep water data table.

2 **for** each episode (until $C_d = COG$) do:

3 Initialize s

4 Read sensory information and update s with updated motion vectors

5 According to motion vectors update COG and C_d

7 Chose a from $Q_0(s,a)$.

for each step of episode do:

Take action a, observe R, s'

Read SOG and update required ROT with (3.112)

According to motion vectors update COG and C_d

Choose a' from s' using policy derived from Q (ε-greedy)

Taking selected action into account, source the most similar state in expert demonstration and compute gaussian

$$g(s_t, s_{ED}, \Sigma) = \exp\left\{-\frac{1}{2}(s_t - s_{ED})^T \Sigma^{-1}(s_t - s_{ED})\right\}$$

Determine shaping potential $\Phi_{ED}(s, a) = \max_{s_d, a} g(s_t, s_{ED}, \Sigma)$

Calculate $F_{ED}(s, a, s', a') = \gamma\Phi_{ED}(s', a') - \Phi_{ED}(s, a)$,

where $\gamma = 0.1$

Calculate $R_{F_{ED}} = R(S_i, A_j) + F_{ED}(s, a, s', a')$

$Q(s, a) \leftarrow Q(s, a) + \alpha\left[R_{F_{ED}} + \gamma Q(s', a') - Q(s, a)\right]$

$s \leftarrow s'$; $a \leftarrow a'$

end for when s is terminal

end for and **return** Q

end

As the aim is to successfully complete a turn and be accurate when executing, discount factor is maintained at $\gamma = 0.5$, so that both immediate and later rewards are taken into account. The learning rate is set to $\alpha = 0.1$, as previous experience is available, and agents would like to exploit more than to explore. The ε-greedy policy is tuned as in (3.74). The hierarchy of the turn algorithm has been described throughout the chapter; however, the importance of sensory reading is emphasized. As long as information from sensors is reliable, progress can be tracked appropriately. This algorithm ends once the turn is completed, and as stated before this algorithm will commence if the planned turn is with higher than 5° difference in courses. Smaller turns are handled by the heading and course keeping modes. Some factors like radius, shallow or deep-water classification, required course to complete turn and position of the radius are assumed fixed for the duration of the turn. However, this is easily tunable for the various depths, if deemed necessary. Emphasis remains on dynamic ROT values due to speed drop when turning. The resulting trajectory can be feed to actuators similarly as with course keeping autopilot model.

From the zig-zag maneuvers made on the simulator necessary data for the artificial agent to learn turning of the simulated vehicle in both deep water and shallow water environments have been extracted. The simulated turn is North from the Sabine Pass LNG Terminal where Sabine and Neches river meet. The position of the turn is 29° 28’ N, 093° 52’W. The turn is from true course 019° to 279°, making it a 100° turn in a very narrow channel bounded with shallow water. The channel is dredged to allow for sea surface vehicles of maximum 13.1 m draught, while own vehicle is navigating with 10.88 m draught even keel, slow ahead engine and speed of 7 kt. A calculated baseline turn radius is used to compare the turns made by the expert and the artificial agent. The navigational chart used to depict trainings and simulations is NOAA Chart 11342 (https://charts.noaa.gov/PDFs/11342.pdf, retrieved on 24-May-2021).

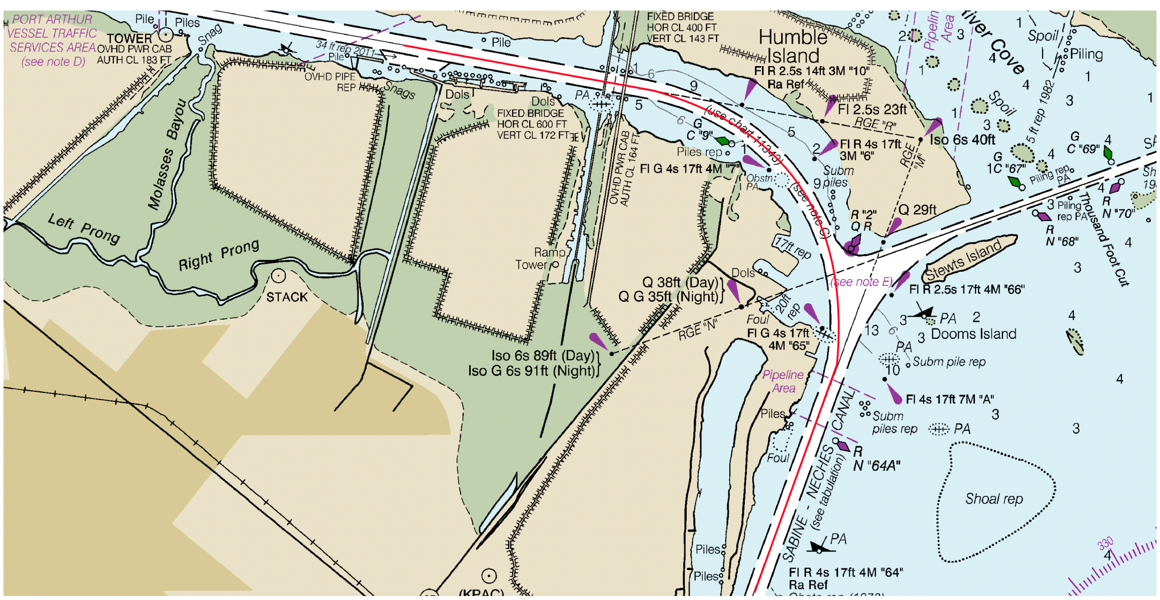


Figure 3.38 – Calculated turn as a baseline for training and simulation

In Figure 3.38 baseline turning is represented by the red line. Following the baseline would be considered as the perfect turn, even though distance from obstructions were not taken into considerations as a risk factor of the voyage planning. In the real world, agents would try to stay further away from obstructions, and this is something that will be taken into consideration in the following chapters.

Master with 7 years of experience commanding vessels of similar size and type has conducted a turning maneuver that is analyzed and presented in Figure 3.39. Standard error was calculated and finally a Root Mean Squared Error calculated in order to verify accuracy and overall success of the maneuver. It is evident that the expert completed maneuver without incidents, but with the higher margin from the obstructions due to the personal risk assessment and navigation bias.

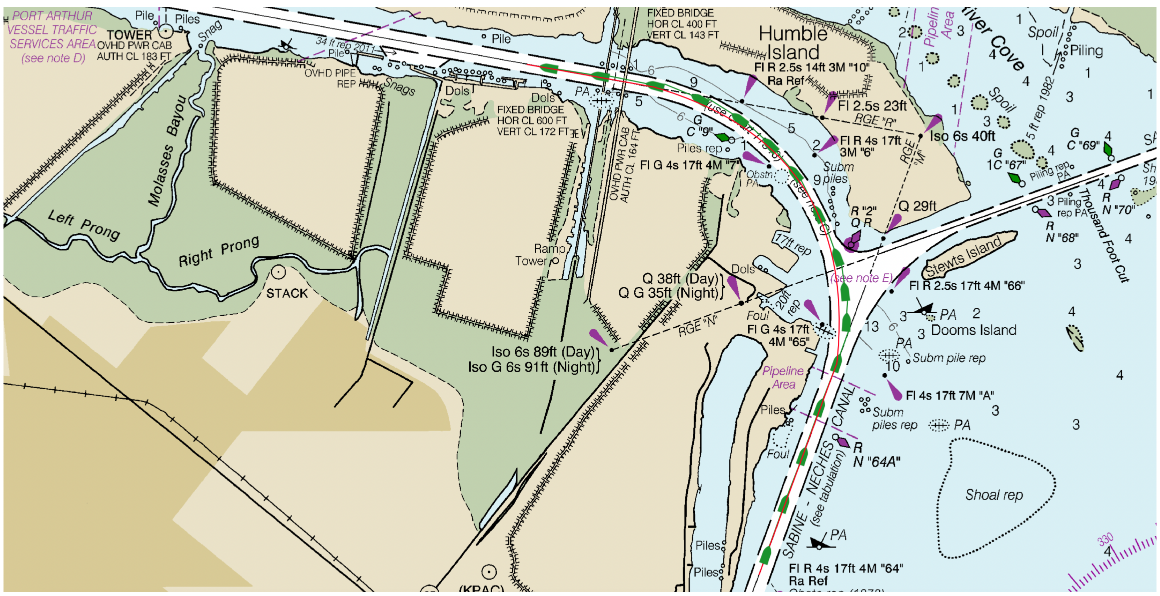


Figure 3.39 – Expert's turn maneuver

Due to the risk-averse approach by the expert, some deviation from the calculated turning line is visible, but the simulation vehicle has been handled safely throughout the turn. It is important to note that the expert's performance is used as a learning input data to the artificial agent in order to reduce the convergence time. As depicted by Figure 3.40, between simulations 12 and 20, as well as 22 and 32, increase in the deviation can be seen, which is caused by the expert's steering influenced by personal experiences. The numeric values are positive, as the distance is of interest, rather than the orientation of the expert's vehicle. RMSE is 209.52 m for the expert's performance.

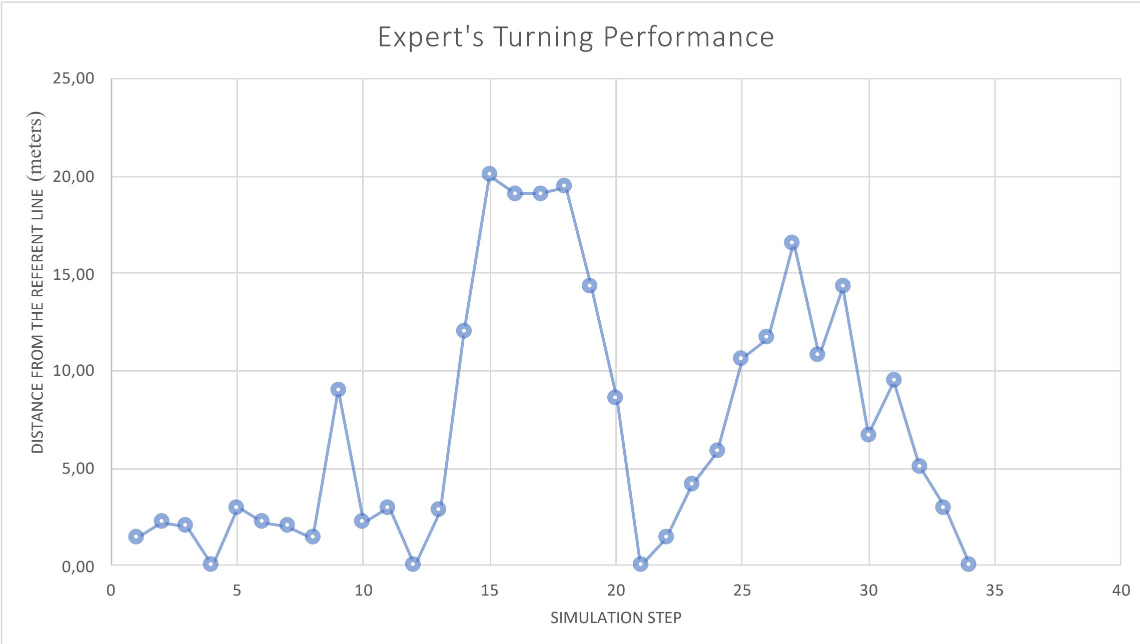


Figure 3.40 – Expert’s turning performance with deviation from the referent line

With the aid of the learning from expert, artificial agent performed well when predicting its simulation steps, so the deviation is smaller along the simulation steps. However, it is noticeable that the artificial agent has been navigating closer to the obstructions and in the real world this could be considered a near-miss incident, so it is important to introduce risk-averse behavior by appropriately tuning the rewards space. In Figure 3.41, artificial agent performed a tighter turn with less deviation, so the system performs as expected. Due to the rewards space design, the system is trying to emulate the calculated turn, but finds possible solution space within the expert’s inputs.

Looking at Figure 3.42, artificial agent completed the turn with lower deviation values from the expert’s turn. Distance from obstructions was lower than with the expert’s turn; however, in the real world, expert’s turn could be considered safer. RMSE is 87.2 m for the artificial agent’s performance.

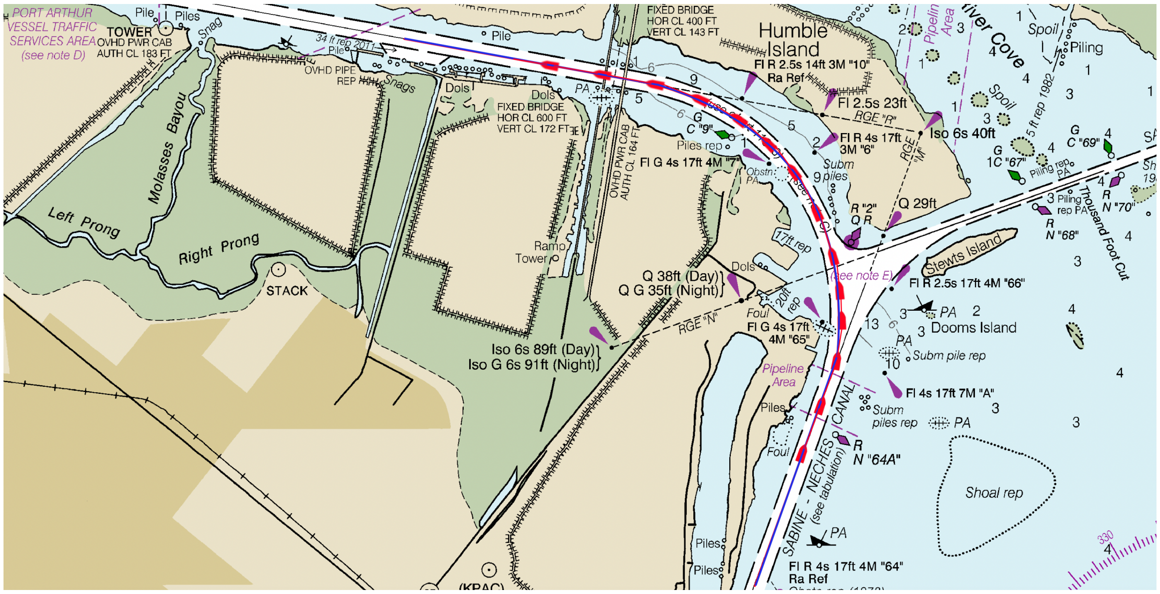


Figure 3.41 – Artificial Agent's turn maneuver

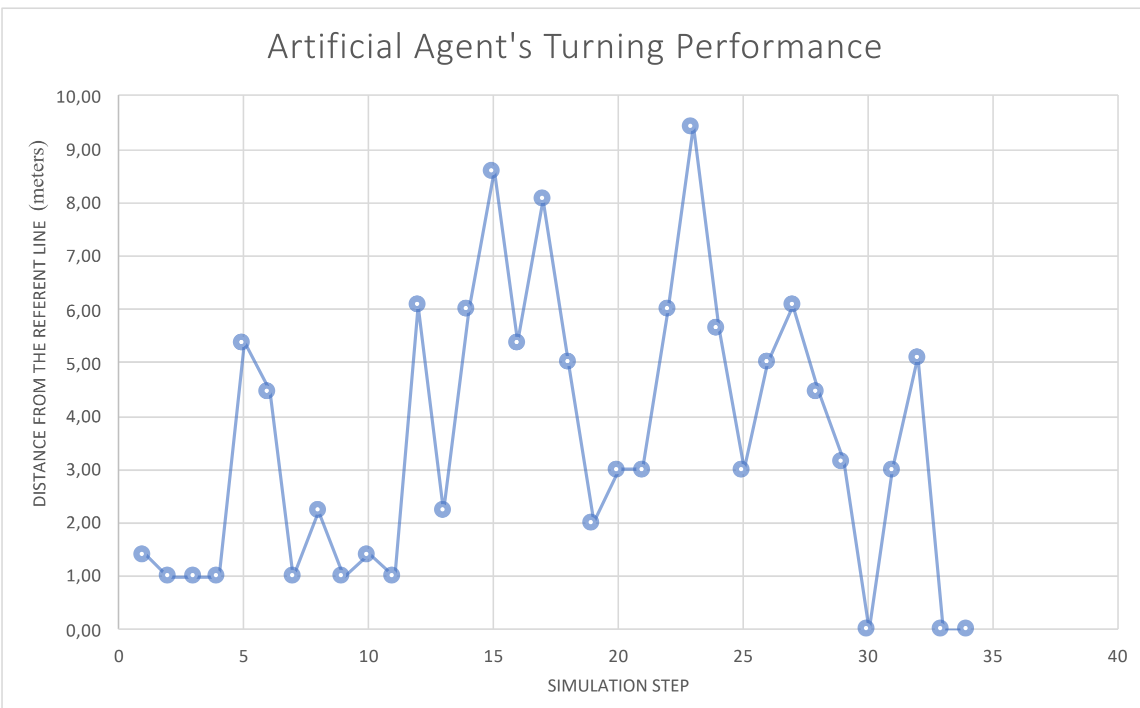


Figure 3.42 – Artificial Agent's turning performance with deviation from the referent line

Further on, if position deviations are considered, similar results which are presented in figures 3.43 and 3.44 are evident. It is noticeable that there is larger deviation in the case of the expert's handling and that artificial agent managed to steer the turn with lower deviation, which shows that artificial agent is able to perform steering of the simulated sea surface vehicle without incurring risks. By integrating a proper risk assessment in the rewards space design, artificial agent would be able to detect obstructions and modify behavior to find the optimal distance from obstructions and outer line of the narrow channel, which is shown in the following chapters.

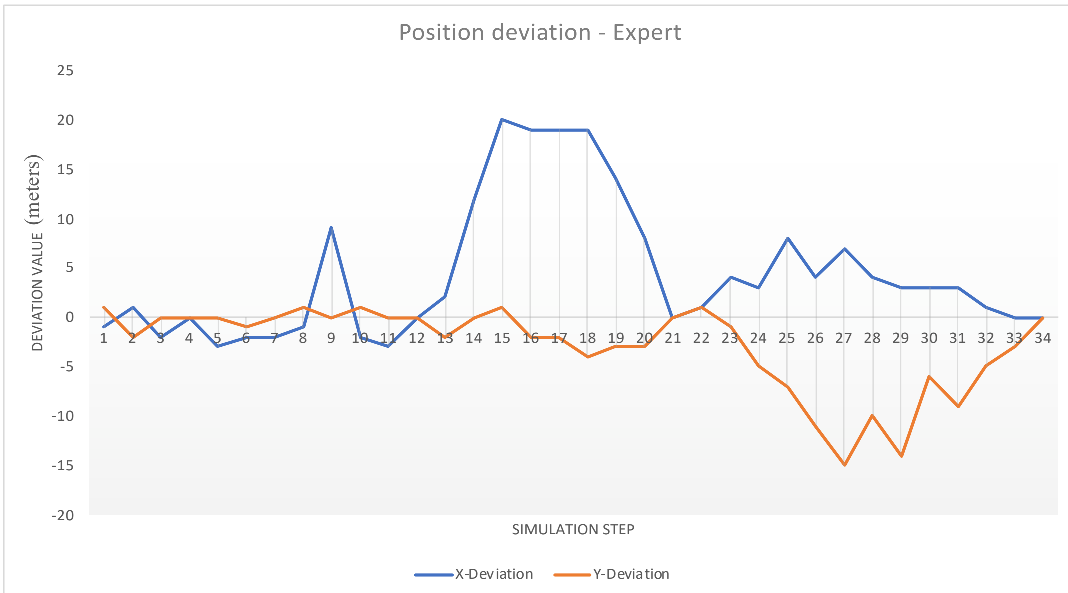


Figure 3.43 – Expert’s turning performance with position deviation

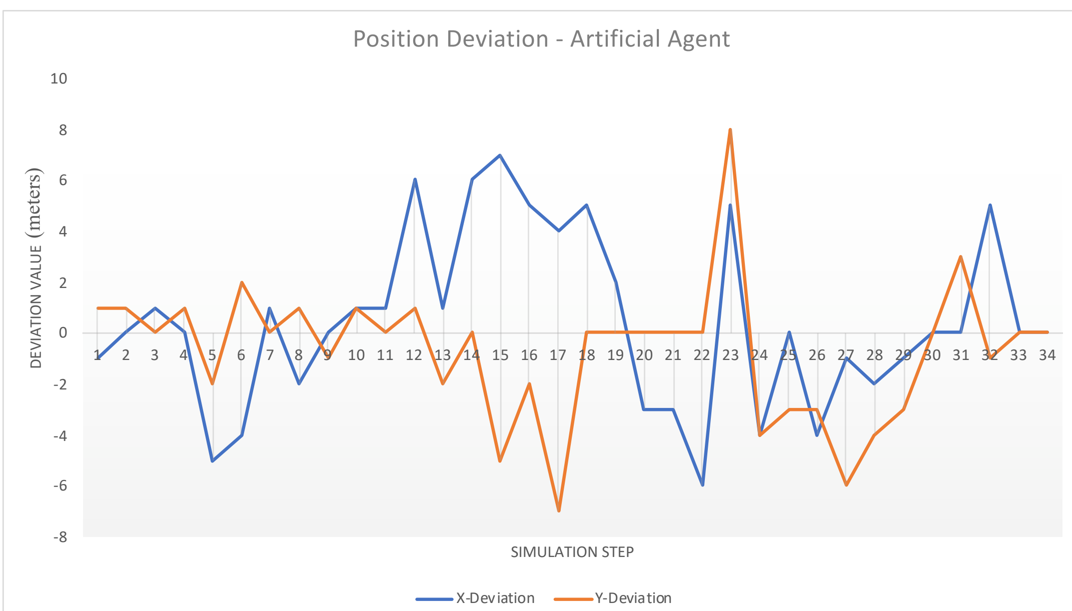


Figure 3.44 – Artificial agent’s turning performance with position deviation

3.3.9 Auto-Telegraph

Auto-telegraph is a feature allowing for automatic control of engine telegraphs. Considering that there are various propulsion systems available in commercial shipping, it is not possible to design one controller that will fit all approaches. Own sea surface vehicle is a steam ship where telegraph controls the opening of maneuvering valves and the quantity of steam to a turbine. A classic slow-turning diesel engine will have a more direct fuel control, while diesel electric will control electric propulsion motors that are connected to generators via switchboards and converters. All of these propulsion solutions have specific ways of developing power and often require manual starts of machinery or supervision by duty engineers, so automatization of propulsion control is still a challenge. For example, on LNG powered vehicles, various machinery on deck has to be working in order to initiate a sensitive sequence of burning NG in engines. Engines are usually sensitive to the quality of NG carried onboard, so calorific value will play a significant role. When a sea surface vehicle is rolling and pitching, resistance on propeller is changing and so is power demanded, so the process sometimes requires a manual control. All of these factors require that automatic control is finely tuned for that particular plant.

Only the LNG carrier model is considered in this chapter, so the interest of this section is on how the process of automation could be implemented for that particular steam plant. External disturbances are hard to predict and have strong influence on speed. Reinforcement learning model is used to predict changes in speed and use that information to adjust Revolutions Per Minute (RPM) to get the speed navigators set for the segment of a voyage. Only limited data of performance is extracted from sea trials, but yet again simulator needs to be utilized to get more data and then update it with a real data from exploitation to get realistic predictions and knowledge. Similarly, fuel consumption could be traced with RPMs and then get realistic predictions for a voyage. For example, as currents are mostly known in advance, good prediction of speed influence for the entire voyage could be computed. Auto-throttle can be engaged to maintain a certain speed if this is possible on a vehicle navigator is trying to control. Auto-throttle can be used in collision avoidance situation, but as the information from sea trials for selected vehicle will show, engine response on large underactuated vehicles is very slow and suboptimal way to avoid collision due to reaction times. On fully actuated vehicles, such are DP vessels, actuators and power delivery are fast enough to power-control vehicles, and in those cases, throttle control tuning approach would be different.

In regards of the RPM control, focus is on SOG that is affected by external disturbances in relation to the demanded speed to maintain, also actual RPMs after the request for certain RPMs or speed needs to be tracked, so state space is defined as follows:

$$S = (d_{LOG}, SOG_R, SOG_A, RPM_R, RPM_A, B_{BM}) \tag{3.125}$$

The depth is part of the state space because in shallow waters it is difficult to gain full RPMs as a sea surface vehicle is affected by the shallow water effect. This is investigated further on in simulation research and a separate table for shallow water performance is made, as speed and RPMs are different in the shallow waters than in deep waters. Speed over ground requested and actual are other members of the state space. This is because the idea of an auto-throttle control is to allow for navigator to select a certain speed and that auto-throttle manages to sustain that speed within the possible performance of a propulsion. Other mode is to simply select RPMs and then propulsion system tries to deliver the required RPMs. The RPM is also part of state space as it delivers information on how much RPMs agent is getting out of the propulsion system. Finally, the B_{BM}, which stands for boiler burner mode, is specific for depicted model of an LNG vessel, which has a steam propulsion. Steam is produced in boilers and with higher demand of RPMs, more burners have to be used on the boiler to produce more steam. This is a rough generalization of the propulsion system, but it is sufficient for this study. Assumption is made that each boiler has three burners, so the $B_{BM} = \{1B, 2B, 3B\}$. Therefore, depending on the required RPMs, boilers could be working on a single burner, two burners or three burners when full power is required. B_{BM} is included as a showcase only, so burners mode will not be considered in the algorithm, as simulator does not have capability of testing power output and burners request, but rather assumes consistent availability. However, once the performance limits are known, burner modes could be easily added.

The action space remains discrete, as navigator commonly chose an RPM value that will remain for a certain period of time. Own agent has only an integer number option as selected RPMs, so fractions are irrelevant. As it is visible from the wheelhouse poster of the simulated model Wärtsila LNG 2 (Dis.89634t), sea trials allow for connecting RPMs with speed:

Table 3.7 – Excerpt from the Simulated vessel's Pilot Card

PROPULSION PARTICULARS			
Type of Main Engine	Steam turbine	Number of propellers	1
No. of Main Engines	1	Propeller rotation	Right
Max. power per shaft	1 x 26800 kW	Propeller type	FPP
Astern power	41 % ahead	Min. RPM	20
Time limit astern	N/A	Emergency FAH to FAS	38.2 seconds

Engine Telegraph Table				
Engine Order	Speed, knots	Engine power, kW	RPM	Pitch ratio
"FSAH"	21.3	26800	89	0.84
"FAH"	11	3700	46	0.84
"HAH"	9	2086	38	0.84
"SAH"	6.8	927	29	0.84
"DSAH"	5.2	462	23	0.84
"DSAS"	-3.1	-111	-23	0.84
"SAS"	-3.9	-222	-29	0.84
"HAS"	-5.2	-499	-38	0.84
"FAS"	-6.3	6410	-46	0.84

Source: Wartsila, 2021.

In line with the above Table, the following action space is presented:

$$A_{RPM} = \{-46, -38, -29, -23, 0, 23, 29, 38, 46, 47, 48, \dots, 87, 88, 89\} \qquad (3.126)$$

As it is visible from (3.126), all RPM orders are fixed with a certain number, except between 46 and 89 RPMs, where any integer can be selected. This is a design of this particular propulsion. Therefore, own agent can only select as described above and a study of constraints in building up RPMs or slowing down is delivered. The change of RPMs also depends on the fuel used, so for example it takes longer to increase or decrease RPMs when boilers burn natural gas, then when burning Low Sulphur Heavy Oil (LSHFO) or Marine Gas Oil (MGO). In this study, different types of fuel are not taken into account, but this could be easily implemented by setting a constraint in performance. Developing certain RPMs does not necessarily mean that the corresponding speed will be immediately reached. This also takes time and has to be taken into account.

It would be rare to have same output speed with selected RPMs. This is because of external influences, current maneuvering, or even output power limited by the quality of fuel. The auto-telegraph system design allows operator to select desired SOG and simply turn on the system. In this approach speed through water is omitted, but this can be easily added to the model. The main benefit of auto-throttle based on reinforcement learning is that the system predicts changes in speed because of planned turns or external disturbances and endeavors to adjust RPMs required to maintain required speed well in advance and without waiting for the effect to take place. This will also allow for RPMs to be initially developed to a higher number than required for selected speed, but as the speed is approaching the desired level, RPMs would be reduced to the equilibrium level. However, sometimes at maneuvering speeds, navigator wants to use the benefit of a faster turn, which happens when the rotational speed of propeller is higher than the speed of a sea surface vehicle, so in these cases navigator should use the system in manual mode and adjust for RPMs, rather than adjusting the speed. Simulator allows for ballast and laden condition, so this has to be taken into account as well, as sea surface vehicle will require different amount of power produced by the propulsion system for different weight of a vehicle.

Keeping above setup in mind, the following reward space is modeled as:

$$R(S_i, A_j) = \begin{cases} 100, & \text{if} \quad SOG_A = SOG_R \\ 50, & \text{if}\, SOG_A = SOG_R - 0.2\text{ kt} < SOG_R < SOG_R + 0.2\text{ kt} \\ 10, & \text{if}\, SOG_A = SOG_R - 0.5\text{ kt} < SOG_R < SOG_R + 0.5\text{ kt} \\ -1, & \text{otherwise} \end{cases} \tag{3.127}$$

As agent’s goal is to maintain requested speed, only one reward function is required and it describes a reward for keeping the actual speed over ground at requested speed with allowed deviation of up to 0.5 kt. Otherwise, the agent gets punished. It is necessary to keep in mind that at sea trials vehicles are tested for top speed at favorable conditions and if weather conditions do not allow it, vehicle will not be able to reach the desired speed even at maximum RPMs. In order to preserve seaworthiness of a sea surface vehicle, at certain environmental conditions, especially with the impact of waves, navigators have to reduce the speed to maintain seaworthiness. In these cases, navigators will have to either select lower speed or turn to RPM mode and control RPMs manually. Torque limitation can be also a parameter to limit the performance of a propulsion, so in case that agent is trying to reach the preset speed, a message of torque limitation will appear notifying navigator that only RPM control is possible and maintaining speed higher than achievable with that power is not possible. Following the above mindset, the following algorithm is proposed:

Algorithm 6 – AUTO-TELEGRAPH Control Algorithm

Input: States S, Actions A_{RPM}

Reward function $R : S \text{ x } A \rightarrow \mathbb{R}$, Discounting $\gamma = 0.2$, ε-greedy factor 0.10,

Learning rate $\alpha = 0.2$.

Output: Q

Initialization: $SOG_R = SOG_A,\ Q(s, a) = 0$.

While (*SPEED* mode is engaged) do:

  for each episode (time horizon of 30 minutes) do:

    Initialize s

    Chose $a = RPM_R$ as first action.

    for each step of episode do:

      Read *SOG*

      Update *SOG* according to motion vectors and external disturbances

      Read current *RPM*

      Read current d_{LOG} and determine *shallow* or *deep* water

      Update s' according to data available in the $B_{SIMULATION}$

      Take action a, observe R, s'

      Choose a' from s' using policy derived from Q (ε-greedy)

      $Q(s, a) \leftarrow Q(s, a) + \alpha\left[R\left(S_i, A_j\right) + \gamma Q(s', a') - Q(s, a)\right]$

      $s \leftarrow s'; a \leftarrow a'$

end for when s reaches time horizon

end for and **return** Q

end While

end

The most distinctive part of the auto-telegraph algorithm is the infinite loop. The algorithm has to go on until switched off by a user. In this case the discount factor is $\gamma = 0.2$, as earlier rewards are valued more than the distant ones. Fixed learning rate is maintained at $\alpha = 0.2$, as exploration is discouraged, considering that we have sufficient data available for the RPM control. However, the learning rate will allows for intermittent exploration and allows for verification if there is a better solution outside of the bounded search space. The ε-greedy policy remains as in (3.74). The algorithm initiates with $Q(s, a) = 0$ because when a navigator engages the SPEED mode, algorithm commences and remains until the mode is switched off. In the moment of initiation, current state of a vehicle is irrelevant, as already in the second step all parameters will be updated. Null initiation will not be expensive due to slow dynamics of propulsion control in this case of an underactuated vehicle. In the case of DP vessels, propulsion approach would be different, and the relevant input would be acceleration in various degrees of freedom for which reward space would be more complex. The central focus of the algorithm is to update states with a sensory data and to do predictions based on the history data available in the simulation buffer $B_{SIMULATION}$. In this study only a simulation buffer is envisioned, but once a sea surface vehicle is on the open seas, similarity function can be used to compare simulated and real-world data to generate more accurate trajectories.

As presented in the Appendix B, Table B.7 depicts 16 scenarios with external disturbances. Scenarios are same as used in the autopilot trainings and simulations. However, a column is added for set speed of simulated sea surface vehicle. It is in the interest of this research to find if the proposed algorithm is able to determine correct RPMs for the desired speed related to the external disturbances. Input data is derived from sea trials of vehicle used in training and simulation, which is then used to design lookup tables suitable for interpolation. Desired speed is chosen randomly within the range of speeds simulated vehicle can accomplish.

Trainings were conducted within the same environment as autopilot trainings and simulations. Aim was to see if the auto-telegraph algorithm can learn and handle dynamic environmental loads. Training was conducted until the average rewards value for the last 20 training episodes reaches 15,300. This number was selected by multiplying highest reward value that can be reached per step by the maximum number of steps per episode, which for this training is selected to be 300 in accordance with the tests done earlier in this chapter. As per the reward space (3.127) 100 points is reserved for the situation where desired speed exactly matches achieved speed. It is difficult to get the exact match, so 100 points for calculations was not considered, but rather taken 50 points for the second reward option and 1 point for the third reward option, which totals 51 points per step, summing to 15,300 for 300 steps.

In Table B.8 of the Appendix B, columns for desired speed agent would like auto-telegraph maintains is presented, as well as RPMs that are required to maintain desired speed according to the sea trials data for the calm weather and sea, longitudinal external influence on speed data that is extracted from each of the simulated scenarios (taken as an average value for external disturbances), achieved speed for each of the scenarios, and RPMs maintained for that speed. Depending on the longitudinal influence, higher or lower number of RPMs than derived from the sea trials data is required to achieve desired speed. As proposed simulated vehicle is equipped with steam propulsion and is underactuated, the action space (available RPMs) is discrete and selected integers are strictly defined by the action space. If vehicle with electric propulsion was considered, action space would have been continuous. Therefore, it is evident from the results that some of the desired speeds could not be achieved without deviations, and the closest possible RPMs are selected as solution. This resembles the real world where rarely exact speed can be matched, especially as environmental loads are highly dynamic, and consistent change of RPMs to maintain certain speed would be required. This is unfavorable as it would lead to unwanted difficulties with the complex propulsion plant.

With the environmental loads of the first scenario, it is evident that the vehicle gains

1.38 knots of speed. It is, therefore, valid to expect that agent would require less power, and subsequently less RPMs, to achieve desired speed. This is exactly evident, as own agent required 65 RPMs to achieve desired speed, while it would require 69 RPMs to maintain desired speed without any external disturbances. Similar approach and analysis were made for other scenarios. However, once reaching scenario 3,it was noticed that some adjustment to the rewards space is required in order for the model to be viable. This is mainly because the desired speed is lower than the minimum speed vessel can make with minimum RPMs. Proposed algorithm was not converging as the rewards space was constricted and allowed for only 0.5 kt difference from the desired speed to gain any rewards. Therefore, in order for the algorithm to converge, the rewards shaping was designed in a way that allows for wider rewards deviation within the rewards space when the desired speed is below Dead Slow Ahead / Astern RPMs range. Considering the environmental loads of the scenario 3, algorithm correctly selected 0 RPMs. As an opposite case from the first scenario, a closer look is taken at the scenario 7 where environmental loads are slowing down own vehicle, and it was noticed that the algorithm correctly selected 56 RPMs, which is higher than sea trials RPMs in order to maintain desired speed. Identical situation is evident in the scenario 8. Particular scenario is scenario 12, where higher deviation of the achieved speed from the desired speed is noticed. The reason for this is the limitation of the propulsion plant. Algorithm correctly selected Dead Slow Ahead (23 RPMs) as if it selected Slow Ahead (38 RPMs), the deviation would be higher, so 23 RPMs remain optimal.

The benefit of auto-telegraph option for the decision support system is not only that it can autonomously control the power generation in cooperation with other auxiliary systems, but also that it can provide navigators with instant information about which RPMs would vehicle require to maintain desired speed taking into consideration present environmental loads. Another benefit is also that the system is able to provide a navigator with the information how long it would take to increase or decrease certain RPMs or how much time it would take to increase or decrease speed. This is especially beneficial in collision avoidance, so this information is used as an input signal to the collision avoidance algorithm and it is used in cases where course change does not provide optimal solution and speed increase, or reduction, is deemed beneficial to clear obstructions.

In order to aid algorithm convergence to optimal solutions, enough data is simulated to generate a standalone shaping function that could be called whenever a collision algorithm fails to converge. Comparably to auto-pilot sections, exploitation data could be collected, and simulated data updated using similarity function. Having external disturbance data, it is possible to predict influence on speed and RPMs, but there is also influence of turning, which is taken into account. Speed loss data caused by turning is extracted from zig-zag maneuvers done at sea trials. This is paired with the external disturbance data and uploaded to the knowledge buffer that is updated during exploitation. This data is beneficial for planned maneuvers and collision avoidance algorithm. If there is enough power reserve, auto-throttle algorithm can increase RPMs just before the turn to maintain consistent speed throughout the turn, if the physics of the turn allows for it. As evident from sea trials data, larger turns would cause significant drop in speed regardless of the compensation by the algorithm.

3.4 Discussion

This chapter investigated approaches to improve reliability of input data, which is critical component of collision avoidance system. Uncertainty in sensing data significantly influences stability of the collision avoidance system, therefore it is necessary to reduce uncertainty by increasing reliability of data through data fusion and filtering. Once the quality of the input data is improved, it is possible to increase observability of inherent state space and ensure appropriate motion control of sea surface vehicles.

An improvement of the sensing data filtering process is proposed by introducing nonlinear state estimator that is named Foraging Particle Filter (FPF). FPF is a variation of swarm algorithmic approaches, where foraging process is utilized in algorithmic design. Main focus of the FPF design was solving impoverishment and particle degeneracy issues. A scalar growth model has been used to verify effectiveness and usability of the proposed particle filter and comparing to other particle filter models used as bassline cases, it was noticeable that the FPF has performed well with lowest RMSE and STD results among compared particle filters. The main challenge remains to select appropriate number of particles and iterations, as it was visible from experiment results that increasing number of iterations above a certain level does not improve results, but only increase computational loads. Assumption is made that for the rest of the research FPF is used to increase observability of the state space estimations for sensing data. Together with data fusion, more reliable inputs of position, heading and speed would be achieved.

In order to ensure feasibility of trajectories that are generated by collision avoidance algorithms, appropriate motion control solution was sourced. Even though approach in this thesis is model-free motion control, framework of Model-Predictive Control (MPC) is utilized to propagate signals to actuators. If the sea surface vehicle is just built and there is no data to ensure safe exploitation in commercial waters, nonlinear observer is proposed that will capture initial data at sea trials. Experimental results verified accuracy of the proposed model for underactuated sea surface vehicles. Together with the captured data at sea trials, it was also proposed that initial database is developed by developing simulation models of sea surface vehicles and that simulated data is used to reduce exploration on initial voyages. This is crucial step to avoid any actions that would be deemed unsafe once a sea surface vehicle is in the real world interacting with other equipped and non-equipped vehicles.

Intelligent autopilot has been developed to manage all motion-control requirements from the collision avoidance algorithm. Autopilot is envisioned as a system of various modes and controls. Even though it can be used in autonomous control, the focus of this thesis is decision-making support, so the system is designed having in mind usual steering and propulsion setups found on commercial sea surface vehicles. Intelligent autopilot is adaptive with regular updates of the knowledge buffer. Learning commences with learning from demonstration. This is crucial step to reduce exploration and to confine state space and action space search. A novel reward shaping approach has been proposed and used in all experiments to increase efficiency of reinforcement learning techniques. From own experiments it is evident that proposed rewards shaping provided more stable learning with decreased amounts of exploration.

The interest of this study is in motion control that can cope well under dynamic environmental loads. This is the reason why different modes of controls were proposed with different tuning and algorithms. Heading, course, turning and propulsion controls that are able to manage motion control of sea surface vehicles in deep and shallow waters, with precision and economy modes, and under various environmental disturbances is proposed. Simulation and experiment results showed that proposed algorithms were successful in controlling the motion of simulated sea surface vehicle, and that the learning was efficacious. Auto-Telegraph option was added to the motion control challenge to utilize option of speed control during collision avoidance maneuvers. Auto-Telegraph model thrived in all environmental conditions and was able to maintain desired speeds and RPMs, as well as to advise navigators how long it would take to slow down or speed up, which is information valuable for taking a proper action when avoiding collision.

[THIS PAGE INTENTIONALLY LEFT BLANK]

Chapter 4

Dynamic collision avoidance

As the central chapter of this thesis, Chapter 4 delivers dynamic collision avoidance modeling utilizing HMMs. As proposed model should utilize Collision Avoidance Regulations (COLREGs), this chapter begins by providing an overview and challenges of COLREGs implementation. A solution to a common problem of categorization of which COLREG rule should be used in a certain situation is provided. Uncertainty related to COLREGs is then quantified, and study of its implementation in the target trajectory generation discussed. Based on own vehicle and target vehicle trajectories, an algorithm that would successfully resolve collision situations, leaving required minimal distance from dynamic and static objects is delivered.

4.1 COLREGs implementation and compliance

The International Regulations for Preventing Collisions at Sea 1972 or COLREGs came into force in July 1977. Since 1972 there were no major revisions or rewrites to the COLREGs, while the shipping industry changed drastically in size and number. Today, technology allows for automated and even autonomous vehicles; however, the COLREGs have not evolved to follow the advance of technology. In parallel, if we take a criminal law as an example, we can see that today we have criminal acts and intents that did not exist in 1970-s, simply because of the lack of technology. If we did not change criminal jurisprudence around the Globe, we would face significant issues finding justice for the affected parties.

When reading the COLREGs it is evident that the language used serves mostly legal profession and that professional mariners struggle with interpretation of some of the Rules. COLREGs' main goal is to prevent collisions at sea, and this is the reason why the language should be adjusted for mariners. Ambiguity of the COLREGs is spread consistently among the Rules that even today studies measuring interpretation and understanding of some specific wordings are present [Mohovic et al., 2016]. In addition, mariners, educational professionals, and even admiralty law experts still disagree when interpreting rules, so it is sometimes challenging to find a correct interpretation. As with other rules and regulations, when larger number of persons do not understand some rules, it says more about the quality of wording then the intellect of persons. At sea, younger seafarers are often told that they can learn a lot in school, but only experience will make a seafarer out of them. In order to fully comprehend collision situations at sea, an able-bodied mariner needs to develop seamanship that is intersection between experience and common sense. The COLREGs appear to be vague enough that mariners feel as the blame is unavoidable whatever the action taken [Kemp, 1973]. Even before the introduction of the latest revision of COLREGs authors criticized the use of unnecessary and vague language [Azad, 1959]. Stitt [2002] delivered a thorough overview of complications and challenges that COLREGs bring to an everyday mariner.

Maritime industry would like to see evolution, especially to track technological changes. When thinking about decision support systems, or autonomous marine control, a big gap in regulation is noticeable. Even though technology allows for intent aware solutions, where risk of collision could be brought down to minimums, navigators would have difficulties combining that kind of technology with the COLREGs that are known today. As mentioned in Chapter 1, collisions still occur alarmingly frequent. This becomes even worse if near misses that happen daily are included, but these are not tracked as well as incidents, so the real number of near misses could only be estimated.

It is still possible to experience vessels signaling Not Under Command when they get information to drift and wait further notice. It is confusing to understand what is “making way” and “underway”. It is evident that the Rules were written having in mind smaller outreach, while today they are used and misunderstood globally. Mohovic et al. [2016] showed that misunderstanding of COLREGs is more severe than thought. A number of commercial ships that sail with navigators not knowing or not understanding the COLREGs is still a challenge that requires global solution.

As efforts of Convention members to advocate for a change is not prominent, the only approach in resolving collision and close-quarter situations is improvement in technology and intent communication. In line with the criticism, this study mentions only the rules that affect proposed algorithm. The scope of this research is not to find solution to full implementation of all COLREGs, but rather to showcase that COLREGs could be taken into account to reduce collision risk and increase maritime safety.

Designing collision risk averse algorithms without COLREGs compliance does not add value for the real-life use. COLREGs are equally binding for human and non-human decision makers, especially as both human and autonomous systems thrive in the same environment. Porathe [2019] has delivered a comprehensive overview of challenges in mixing manned and unmanned vehicles, while Stentz [1994] showed that maintaining protocols designed and developed for humans allow for safer and predictable maneuvers. Human involvement can't be disregarded in the process, and there is an obligation in designing machine protocols that would be compliant with human operational styles. If autonomous vehicles would be allowed to disregard human protocols, there is a possibility of getting a counter effect and having even more near misses and incidents at sea due to increased level of confusion and misunderstandings. Development of algorithms in vacuum is never a good idea, as humans will still be responsible in following protocols and rules, while algorithms would have freedom to find the optimal solution without same level of constraints. With enough computing power, machines are able to look further in the future and make a sequential decision chain that would be hard to follow by counteracting human navigator.

Some studies, like Lee and Kim [2004] claim to be COLREGs compliant but allow for some violations in certain scenarios. It is paramount to keep the full compliance to the COLREGs and embrace the ambiguity by discussing and developing solutions that would remove those ambiguities. This is not an easy task, but technological solution to any kind of protocol is comparatively easier part than agreeing on the protocol itself. This research has to stay within the challenges of current COLREGs, so it is necessary to keep in mind that some of the collision scenarios would be resolved in a different way by human navigators than the solution offered by any kind of COLREGs compliant algorithm. Turning to port when there is a risk of collision goes clearly against COLREGs, so this is something that has to be transferred to optimization algorithm as constraint, regardless of the lucrative maxima for a reward or minima for penalty. As it will be seen in the next chapter, some humans are going to decide to go against COLREGs, but at that moment they assume all the risk behaving this way.

Benjamin et al. [2006] were first to introduce algorithms following COLREGs between autonomous vehicles and there were many other approaches after that. However, all of those approaches mostly focused on Rules 13-18 omitting that some of the rules are interacting and that there are various types of vehicles covered by COLREGs. Some of the authors used "COLREGs-compliant" in the title of their work, but still focused only on few Rules [Hoekstra et al., 2002, Lee and Kim, 2004, Stentz, 1994]. Many of the mentioned studies neglected COLREGs Rule 8 requirement for alteration of course to be large enough to be readily apparent to another vessel observing either visually or by radar. Algorithms were successful in avoiding targets, but the fact remains that compliance with COLREGs requires adherence to all Rules. COLREGs are vague in defining what is a large enough course alteration, so it is necessary to turn to Case law, insurance recommendations and experience in some of the aspects of COLREGs, which in this Rule state that large enough course alteration is not less than 35° according to Case law, or minimum of 30° according to an experience [Benjamin, 2002, Benjamin et al., 2006. Porathe and Shaw, 2012]. It is necessary to keep in mind that visual reference is a major focus of COLREGs and just because advanced sensors are installed onboard, water-stabilized RADARs still have to be used and information collected confirmed by sight and the correct attitude of target vessel determined before concluding which Rules should be applied to resolve collision risks. When modeling collision avoidance algorithms, it is necessary to penalize for kinematic and dynamic constraints violations, as well as for large deviations from the intended course towards destination. However, care has to be taken when assigning weights to allow for large enough alterations and to actually code that course alteration is initially at least 30° and that vessel returns to the parallel course until the risk of collision diminishes, after which a vehicle can go back to its original course.

As visible from Figure 4.1, proposed collision avoidance algorithm utilizes predictor that takes into account motion control algorithm and external disturbances in order to predict alterations of headings and return to the route after the collision conflict has been resolved with the target sea surface vehicle.

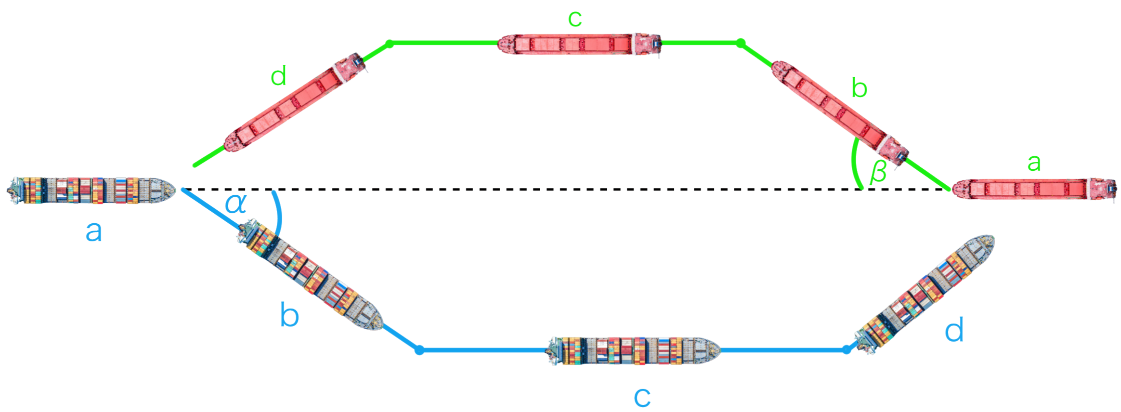


Figure 4.1 – Collision avoidance geometries for head-on situation

Different approaches to trajectory optimization have led to a rich academic pool of ideas but focus on human control has to remain. Basically, proposed algorithms have to offer solutions that would be similar as if decisions were made by a human navigator. In the commercial shipping, passage planning is supported by a great deal of planning, risk assessments, and approvals from managing teams ashore. Planned tracks are evaluated by various software and provide navigators with assurance that a vehicle will not come in any risk with stationary objects or shallow waters. Any significant deviation from that plan has to be assessed and approved. Therefore, solutions of generating paths arbitrarily after collision risk has passed is not a viable solution. Vehicles should return back to the safe pre-planned track.

Most of the mentioned research considered only a pair of vehicles and their interaction, while completely disregarding the human factor. External disturbances and speed variations were not taken into account, while COLREGs were only partially utilized for compliance. What makes COLREGs compliance even more difficult is that experienced mariners already expect a certain behavior from targets they meet. So, even though proposed algorithm would be solving collision situations successfully and according to COLREGs, some behavior would be considered strange and unnatural, so seamanship has to be kept in mind while tuning algorithms. For example, maneuvering of sea surface vehicles will affect the attack angles of external disturbances and larger or faster turns will affect the speed of a vehicle. Research considering speed changes in collision avoidance maneuvers is scarce. When maneuvering, commercial vehicles rarely slow down significantly and is used as a last resort simply because of the inertia and sluggish response from the propulsion system.

4.1.1 Evaluating COLREGs and safety parameters

Considering the ambiguity found in many rules of the COLREGs, certain rules have to be evaluated in order to remove ambiguity as much as possible for collision avoidance algorithms to work. It is once again emphasized that scope of this study are underactuated sea surface vehicles and that this thesis provides solution for decision support system, rather than autonomous system. Autonomy would be possible with availability of sensors that would allow for COLREGs compliance. However, most of the commercial vessels do not have required audio-visual sensors that would ensure full compliance. This is the reason why some of the rules are omitted as well, as still human navigators have to watch and listen for light and sound signals.

The intentional vagueness of COLREGs is there mainly to allow for experienced navigators to select an optimal action with liberty to interpret complex collision scenarios without a significant restriction, but also COLREGs are vague to make blame assignment easier. Due to vagueness, some of the rules of COLREGs are open to interpretation to Admiralty experts. COLREGs were written in early 1970's and were appropriate for that time. However, in the present time we can only turn to professional experience, interviews and Case law to get some interpretation of the ambiguous phrases and weasel words. This proves a point that experience is a key factor in appropriately utilizing COLREGs in situations where collision risk exists. That is why it is beneficial that researchers have exposure to COLREGs in real-life situations at deep sea, but many researchers work only in academia. This promotes a disconnect between experienced mariners and algorithm designers. The situation is similar to situations when a professional supervisor of a complex process requires assistance from an IT department, but a technician only understands IT, while he has no understanding of process. IT-process frustration is an everyday occurrence in technological organizations.

In this chapter most of the Rules are presented, while quantifiable way for COLREGs algorithm to determine which Rule is applicative to a certain situation is delivered. In order to determine a relevant Rule, it is necessary to know if the risk of collision exists and for this relative geometry of interacting vehicles and stationary object is required. Many authors used various types of metrics to measure risk; however, in this research focus is on real-world situation and all available sensory information. Sensors commonly found onboard are regulated and information proposed algorithm has to deal with is the same information a human navigator finds on commercial ships. Lean approach is preferred. If the information is available and is already processed, there is no benefit of adding computing complexity to the system, but rather utilize sensor uncertainty provided by a manufacturer or governing body for state estimation. As mentioned in Chapter 3, benefits of sensor data fusion is exploited to reduce uncertainty. The most important information to determine collision risk is dCPA (distance at Closest Point of Approach), TCPA (Time to CPA) and actual attitude of a target (the relative bearing to own vehicle).

The CPA is the point on an own vehicle's trajectory where the distance to the target is at its minimum value. Benjamin et al., 2006 described thoroughly how the CPA is calculated, but this information is taken from ARPA installed onboard. Only small coastal vehicles do not have requirement for ARPA installation onboard and for them additional calculations could be used by utilizing data from RADAR. CPA and TCPA sensor error and uncertainty has already been discussed and this uncertainty is considered by expanding the safe radius around each acquired target. The process of uncertainty influence will be discussed later in text. Another important value to determine collision risk is the attitude of a target. It is necessary to know own vehicle's course through water and over ground, but also to know target's course over water and over ground. The attitude is affected by the course through water and by the relative bearing of a target. With these values it is possible to determine how ownship would see a target with own eyes and to determine which Rule of COLREGs would be used in that situation.

The importance of having water stabilized RADAR for collision avoidance is emphasized, simply because COLREGs rely on visual reference more than on ARPA calculations. So, allowance for external disturbances have to be made, and only then Rules would be applied as if the target is apparent to us with a naked eye. For actual collision avoidance, true over ground movement is required in order to have safe distance from a target until the collision danger is gone.

Some authors [Dinh and Im, 2016; Goodwin, 1975; Liu et al., 2016; Pietrzykowski, 2008; Rawson et al., 2014; Rudan et al., 2019; Szlapczynski, 2006; Szlapczynski and Szlapczynska, 2016; Szlapczynski and Szlapczynska, 2017; Wang et al., 2010] developed domain based collision risk determination approaches, but as their research shows, domain based approach is more complex and computationally expensive than a more traditional dCPA and TCPA approaches. Remaining with dCPA and TCPA brings a benefit of lower computational complexity. The main criticism of dCPA and TCPA approach from the perspective of the domain approach authors is that domain-based methods favor maneuvers recommended by COLREGs, but yet again COLREGs were not utilized in their entirety and still some situations lead to maneuvers that avoided collision but were not according to COLREGs. This research shows that dCPA and TCPA approach is easier to envision, professional mariners do not think about domain space, but rather distance from a danger, and that maneuvers are still COLREGs compliant without unnecessary computational complexity. In their latest state of the art review, Szlapczynski and Szlapczynska [2017] concluded that for collision avoidance and after almost 50 years since the idea of ship domain was introduced, only a partial success is visible, as dCPA and TCPA approach is simpler to implement and interpret. Nevertheless, domain methods are still researched and actually growing popular for applications other than collision avoidance, where accuracy of modeling is more important than computational time or sequential decision making.

Before assessing the Rules, question is made on how do human navigators asses the success of collision avoidance maneuver? Intuitively, sailing is successful when there was no contact with another dynamic or static object. However, near misses are something that causes a lot of stress and sometimes even panic to navigators, so it is necessary to ensure that near misses do not occur. Hence, it is possible to sum the human optimization factors to COLREGs compliance, smallest deviation from the original course, minimum time added due to maneuvering and safe distance when passing objects. These efficiencies are integrated in reward functions and agent weighs all factors during the optimization process. Similarly as masters' of ocean going vessels write their safety preferences in line with their standing or night orders, it is possible to define preferred CPA, minimum CPA, CPA below which a near-miss is considered and a collision distance. These distances could be depicted as circles around the own vehicle with various radii. So, for example, a common preferred CPA would be 2 NM, CPA of 1 NM would be a minimum, while CPA below vehicle's transfer when turning with rudder hard-over (for the LNG 2 vehicle used in experiments, this is 5 cables when vessel is at full speed and with a wheel hard over) would be near miss. Finally, a collision is anything below 2 cables (value that depends on the type and size of a vehicle). In line with this, a set of circles could be envisioned around a sea surface vehicle describing safety zones:

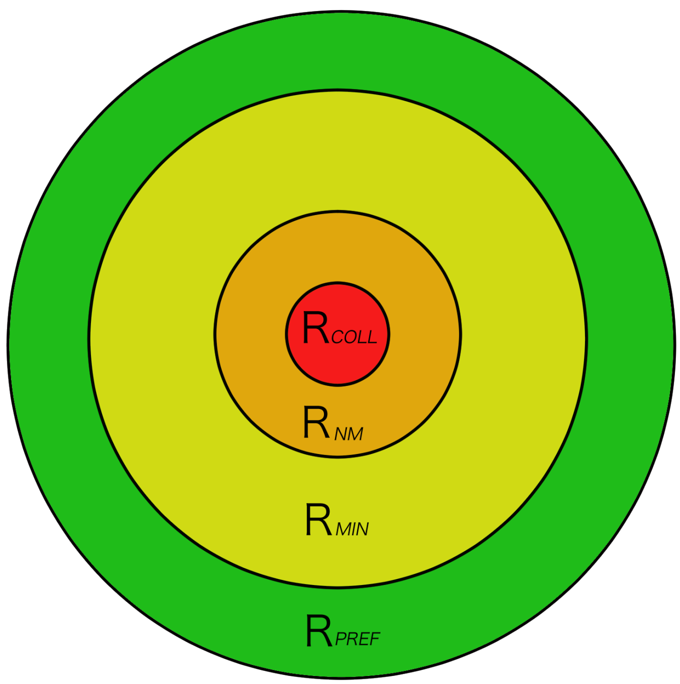


Figure 4.2 – Safety zones around sea surface vehicle

This is a helpful way to assign safety zones. Safety zones are envisioned as dynamic, so that when a vehicle is in busy waterways and have a lot of ships around, these boundaries could be moved to ensure convergence of the algorithm. Dynamic CPA settings is a common practice with experienced navigators, as it is not practical to expect the same level of separation in Singapore strait and at open seas. Also, these radii are dynamic, as vehicles have different turning circles at different speeds, so the speed has to be taken into account. As we are looking at collision avoidance, we will always use turning circle of rudder hard over to avoid contact.

4.1.2 COLREGs - individual Rules' assessment

As some of the Rules are applied consecutively, it is necessary find a way how to determine which rule is applicable at certain collision situation. The key influencing factor is the attitude of a target in relation to own vehicle. It is important to keep in mind that there are performance limits with sensory equipment on commercial vessels. For the sake of collision avoidance, it is not possible to ensure full autonomy without having audio and visual sensory equipment that could hear and see for human navigator. The scope of this thesis are commercial vehicles with regulations as they stand today, so it is necessary to involve human navigators in the decision support process. In this study solutions for full autonomy is mentioned as well. The following text uses phrases and words from the International Regulations for Preventing Collisions at Sea [2020].

GENERAL RULES (1-3)

The general rules deliver information about application, responsibility and general definitions. If looking at these rules through the eyes of an autonomous or decision support agent, it is required to find parts that would require inclusion in COLREGs Entry Criteria algorithm.

In the **Rule 1** Rules apply to all vessels on all waters navigable by seagoing vessels. The Rule 1 does not make any distinction in regards of controlling inputs; be it human, automated, autonomous or combination of it. The rest of the Rule 1 is related to how Governments interact with COLREGs and does not have any direct influence on proposed collision avoidance model.

The **Rule 2** delivers regulation about responsibility. The (a) paragraph states that vessel, owner, master or crew have to comply with COLREGs and could not be exonerated by nothing written in the Rules from the consequences of neglect and any precaution that could be required by the ordinary practice of seamen. Vagueness of the Rules is immediately noticeable, as it is hard to determine what is the ordinary practice of seamen. In this approach, endeavor is made that that Rules are followed similarly as professional navigators would do. The (b) paragraph delivers an interesting notation that each collision situation has to be observed holistically and that navigators could face special circumstances in which departure from COLREGs could be necessary to avoid immediate danger. This brings us to the first applicative Rule that should be used consecutively with all other Rules in the proposed algorithm. In order to comply with the Rule 2 (b), it is necessary to define what an immediate danger is. In this thesis anything occupying a near miss radius is an imminent danger and risk averse approach is required to avoid contact on any way possible. In the case of selected vehicle LNG 2, decision has been made that there should not be any dynamic object occupying area of 5 cables when the vessel is at full speed and with a wheel hard over. The safety radii change according to the speed and turning radius of a vehicle. Vehicle's over ground movement has to be taken into account in order to compensate for the drift when determining an immediate danger. Hence, if any dynamic object occupies a near miss zone, collision avoidance algorithm would be allowed to find an optimal trajectory outside of the COLREGs zone to either increase the distance to a target or to avoid contact by either turning to port, starboard, make a full circle or reduce speed. Therefore, COLREGs Entry Criteria algorithm has to show notification to a navigator or enter a code to autonomous agent: "IMMEDIATE DANGER MODE ENGAGED", which will allow for behavior that can be outside of COLREGs.

In the **Rule 3** general definitions are delivered. These definitions are especially important when other Rules require that navigator has to identify type of sea surface vehicle acting as target vehicle in order to correctly apply those Rules. The paragraph (a) reinstates that any water craft, including seaplanes, that have ability to be used as a means of transportation on water is considered a vessel. So, whenever term sea surface vehicle is used, it includes the term vessel. Paragraph (b) depicts that power-driven vessel is any vessel propelled by machinery. Intuitively, paragraph (c) defines sailing vessel as any vessel under sail provided that even if fitted with propelling machinery, this machinery is not used. Paragraph (d) delivers details about vessels engaged in fishing and it describes that vessels engaged in fishing are only vessels that use fishing apparatus that restrict their maneuverability. Paragraph (e) defines seaplane as any aircraft designed to maneuver on water. In the paragraph (f) vessel not under command is defined as a vessel which is unable to maneuver as required by COLREGs due to some exceptional circumstances and can't keep out of the way of another vessel. Paragraph (g) delivers definition of vessel restricted in her ability to maneuver as vessel that is engaged in work that restricts her ability to maneuver as per COLREGs. Paragraph (g) delivers non exhaustive list of examples. Interesting vessel is vessel constrained by her draught defined by paragraph (h) as a power-driven vessel which is severely restricted in her ability to deviate from the course she is following due to her draught in relation to the available depth and width of navigable water. This means that a vessel in question is passing a certain body of water and can't deviate from its course otherwise it will run aground. For example, in some channels or straits vessels with deep draughts will have a limited space where they can navigate, while other vessels can use much wider space. Another paragraph of our interest is (k) which defines that vessels are in sight of one another only when they can visually observe each other. In line with this definition, paragraph (l) states that restricted visibility is any condition in which visibility is restricted by some external disturbance, such are rain, fog, mist, snow, and similar. Finally, paragraph (m) defines Wing-In-Ground (WIG) vessels as any multimodal craft that can fly in proximity to the surface when in operational mode.

It is important to identify types of vessels in order to ensure state estimation is correct. AIS can be used to identify types of vessels, or it is done visually. Both methods of identification are prone to human mistake, so there is always an uncertainty if the vessel is actually displaying correct visual signs or turn AIS to appropriate sailing mode. In any case, if a vehicle is displaying visual or AIS notification, own vehicle has to ensure to act as required by COLREGs. Identification of vehicles can be done automatically, but this requires visual sensors capable to see as human navigators and distinguishing the navigational lights. Visual reference can be fused with AIS data to confirm navigational signals. In case that there are no visual sensors installed onboard, a human navigator has to input the visual reference to the decision support system in order to have accurate trajectory generation.

CONDUCT OF VESSEL IN ANY CONDITION OF VISIBILITY (4-10)

This part of COLREGs is concerned about Rules that should be applied to all types of vessels and in all conditions of visibility.

Rule 4 only mentions application and that Rules of this section apply in any condition of visibility.

Rule 5 requires that all vessels at all times maintain a proper look-out by sight, hearing and all other available means in order to make a full appraisal of the situation and to measure risk of collision. This is where a distinction of a decision support system with human navigators and fully autonomous systems appears. It is evident that sensors are required so that own agent can hear and see other vehicles at sea. For this purpose, cameras, infra-red sensor and similar can be used. Due to security and firefighting control some of the vessels have Closed Circuit Television systems onboard with capacity to store data. However, these systems are not fit to see other objects as their purpose is to record low resolution information, so a similar system with high resolution cameras and visual recognition software would be required. Audio signals have distinctive frequencies, so a system of omnidirectional microphones could be installed in order to hear signals from other vessels. Some of the vessels with fully enclosed bridges already have an audio system installed to ensure navigators could hear signals and ensure compliance with Rule 5; however, these systems sometimes do not have sophisticated tuning to hear signals early enough and for an autonomous agent to distinguish signals properly, so better recording equipment is required to satisfy autonomy. In general, proposed collision avoidance algorithm should use as much as possible data available and fuse all information to identify risk and deliver optimal trajectory. In the decision support system, a simple prompt of confirmation from a navigator would be required to identify audio and visual signals. Only with that kind of cooperation sustainable results could be achieved.

Rule 6 delivers regulation about safe speed. The Rule 6 states that all vessels should at all time proceed at a safe speed so that effective and proper actions could be taken to avoid collision and that vessels can stop appropriately to the prevailing circumstances and conditions. Another vague but very important Rule of COLREGs. Safe speed rule is intentionally vague because it would be extremely difficult to regulate a speed knots that would satisfy safe navigation for all vessels. Each vehicle will have different safe speed in the similar situation. Safe speed largely depends on maneuvering characteristics. Therefore, navigators have to realize what is the safe speed for their vessel. One of the most common contributary reasons for collisions in the Case Law [Benjamin, 2002, Benjamin et al., 2006] is exceeding safe speed. The Rule 6 does not give a speed that vessels should follow, but it does provide a non-exhausting list of factors that should be considered when thinking of safe speed. Among the intuitive factors such are visibility, traffic density, maneuvering limitations and external disturbances, the Rule 6 also talks about limitation of radars. Many of electronic appliances were not available at the time of writing this Rule, but the same approach can be taken in using other electronic solutions, including ARPA. Logically, reducing the RPMs to the maneuvering speed is one of the first things to do when sailing in areas with increased traffic, shallow waters or other prevailing dangers. This is because an engine will react faster than on navigation full speed due to various technical reasons. The challenge is, of course, for autonomous or decision support agents to determine the safe speed. Metrics for safe speed determination is proposed below. However, lacking further studies, some metrics are only taken arbitrarily. In case that proposed algorithm can't find convergence and optimal trajectory by turning, slowing down will be considered and advised to either autonomous or DSS agent. The only difference between an autonomous and decision support systems is that autonomous agent has a direct control, while DSS agent prints advisory to navigator which navigator chooses to obey or not. So, in order to determine safe speed, following is taken into consideration as per the Rule 6:

- Check the visibility and if the visibility is less than 3 NM, then consider slowing the speed to maneuvering range until clearing the situation;

- Check the visibility and determine if the visibility is less than 3 NM and there are any radar targets within the 3 NM radius, reduce the speed to maneuvering range until clearing the situation;
- If the sea room is restricted by shallow water, TSS, narrow channel, safety fairway, or objects detected by the look-ahead (human navigators have to set up look-ahead parameters carefully), and there is a concentration of more than 4 following, crossing or reciprocal targets within the 3 NM radius, then reduce speed to maneuvering range until clearing the situation. Please note that number of targets is arbitrarily selected, and further study is necessary to find correlation of number of targets and increased risk of collision, as for now research is limited and not decisive [Rutkowski, 2016]. For both DSS and autonomous agents this can be selected by designer or navigator;
- If unable to maintain minimal dCPA with any of interacting targets by turning a vehicle due to static and dynamic restrictions, then reduce speed to maneuvering until clearing the situation. This measure is here to ensure stopping distance is reduced in case collision avoidance algorithm determines that stopping is required to avoid collision or reduce impact. Speed reduction is initiated by the collision avoidance algorithm rather than COLREGs classification algorithm;
- If the depth of available water is less than 3 times the draught, consider reducing the speed to allow for maneuvering performance assurance. If the depth of available water is below 1.5 times the draught, reduce speed to maneuvering. If the depth of available water is below 1.2 times the draught, adjust speed according to the calculated squat for the particular vessel.

In the previous chapter it was determined that the depth below 4 times the draught was used in the algorithm for turns and this is still applicative for turns. However, many factors will influence how a particular vessel will behave in shallow waters. The shape of hull, type of seabed, the distribution of available water (is it a dredged channel, equal plain, etc.), external influences, including currents, density of available water, speed of a vehicle, and similar. This is why navigators always plan the voyage and include UKC and squat calculations, which will allow them to know exactly how fast the vehicle they are on can pass restricted waters at a planned time. PIANC [1992] made a general distinction between deep $\left({h}/{D} > 3.0\right)$, medium deep $\left(1.5 < {h}/{D} \leq 3.0\right)$, shallow $\left(1.2 < {h}/{D} \leq 1.5\right)$ and very shallow water $\left({h}/{D} \leq 1.2\right)$, where D is the depth of water, while h is draught of a sea surface vehicle.

Other factors from the Rule 6 are not considered, as the background lighting is related to human only operations, while external disturbances and radar limitations were already integrated in control algorithms.

Rule 7 delivers notation about the risk of collision. The main premise of this rule is that all vessels should use all available means to determine risk of collision and if ever in doubt, assume that the risk of collision exists. This is a straightforward notation, so all sensory equipment is utilized to determine risk of collision. Considering the consequence of collision or running aground, risk of false positive is accepted and whenever in doubt maneuver to avoid target sensed by the equipment onboard. The interesting notation is further delivered in paragraph (d) and it states that when determining if risk of collision exist, assumption is made that risk exists if the compass bearing of an approaching vessel does not appreciably change, but to keep in mind that risk of collision can still exist even when the bearing changes if approaching a very large vessel or a tow or approaching vessel at close range. This requirement is added to already extensive list of sensory information that is used to determine a risk of collision. Professional mariners assess risk in a different way. Some will allow closer approaches to some types of targets and be comfortable, while other will give wide berth to all targets. In general, navigators of fishing boats are more comfortable to come closer to the larger vessels, than it would be the case of navigators of a Very Large Crude Carrier (VLCC) approaching any other targets considering it takes a long time to provoke a yaw change in VLCCs. As own safety radii are defined, imperative is to keep all targets away from the danger zone.

Rule 8 delivers rules on action to avoid collision. From the perspective of collision avoidance algorithm, this Rule is very important. The Rule states that any action to avoid collision shall be positive and made early if circumstances allow. This gives us a general instruction in when to act to avoid collision, which is as early as possible. In the case of algorithmic maneuvers, it is necessary to act as early as possible after the sensory information has been confirmed valid. In this case, RADAR acquire targets in 12 NM radius and then ARPA give us estimates. Taking in consideration limitations and error uncertainties of RADAR and ARPA, data fusion of sensory equipment is conducted for at least one minute before generating trajectories. Trajectory quality should improve after an additional two minutes. The Rule 8 also states that any alteration of course and/or speed to avoid collision should be large enough to be readily apparent to another vessel observing visually or by rudder. This requirement is a key to collision avoidance maneuvering approach. As stated earlier, course should always be altered by more than 35° in order to make it apparent to another vehicles that own turn is related to collision avoidance. This does not mean that after alteration of 35° navigator should continue along that path until fully clear of the target. This will, of course, depend on the situation, but in the case of head-on situation, navigator should alter own course by 35° and continue that path until target trajectory is in the preferred or further than preferred CPA radius. After that, course can be altered back to the same general direction ownship was keeping before the collision avoidance maneuver. Finally, after the target has passed behind own vehicle, course would be altered back to the track and then continue originally planned voyage. In case of crossing traffic, course would be also altered by at least 35° and that path maintained until the target is well passed so that own vehicle can return back to the planned route. We depict these situations below:

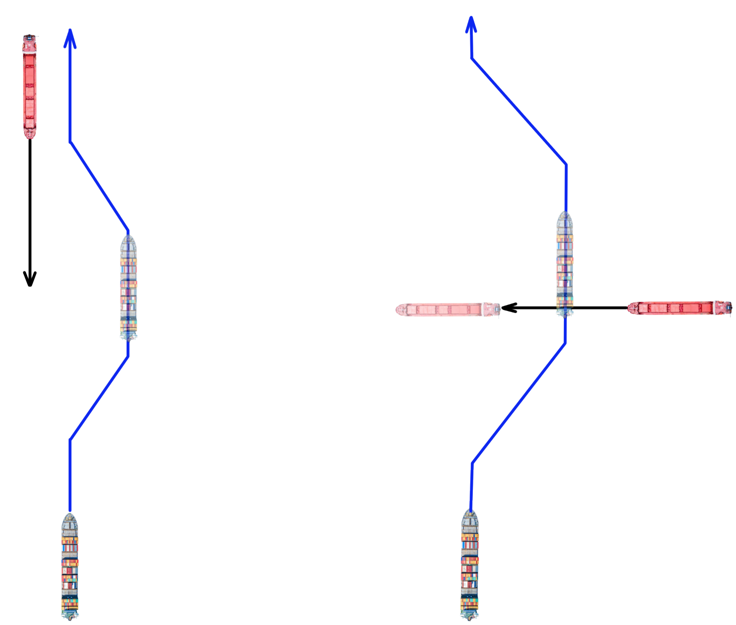


Figure 4.3 – Actions to avoid collision

The Rule 8 also advises an intuitive notion for underactuated large vessels that if there is a sufficient sea-room, alteration of course alone might be the best and most effective action, but it has to be done on time and that it does not result in another close-quarters situation. Proposed collision avoidance algorithm takes this part into account as well. Even though it prefers to do a 35° turn and make enough room between vehicles, it will firstly check if this maneuver will lead to another issue. If there is no enough sea-room for the vehicle, it will try to find optimization allowing for closer approach of targets and smaller turns, but it will not allow for targets to occupy a near-miss radius. If turning alone is not giving us a desired effect, algorithm will consider reducing speed and check for optimization again until finding a viable solution. In a case of DSS, it prompts a user to reduce the speed, while in case of full autonomy, it issues an order to the engine telegraph. After the logical paragraph (d) requiring that vessels are passing at a safe distance, paragraph (e) delivers regulation that when necessary for collision avoidance or more time to assess the situation, vessels should reduce speed, or even reverse the propulsion. There is a benefit of algorithms assessing collision almost continuously, so a slight change will immediately be brought up to the autonomous or human operator. Final three subparagraphs talk about vessel required by COLREGs not to impede the passage of another vessel and again reinstates that all actions should be done in ample time, but also mentions that a vessel the passage of which is not to be impeded remains obliged to comply with COLREGs when there is a risk of collision, which basically means that even if its passage is impeded it is required to avoid collision by any means. This brings us back to the earlier requirement of Rule 2 that vessels can deviate from the Rules in order to avoid collision if the danger is imminent. In this research it is defined that this is happening when impeding vessel is occupying a near miss radius.

The next rule is the **Rule 9**, Narrow channels. In the first paragraph, Rule 9 requires that vessel keeps as near to the outer limit of the channel or fairway which lies on her starboard side. This is to ensure safe distance of meeting vessels. Passing fairways and narrow channels is usually planned well in advance and very often these areas require pilots to guide the vessels. Meeting any vessel on reciprocal course in narrow channel is in practice always discussed between a master of each vessel and pilot, while pilots agree approach by communicating with each other via VHF. A passage plan will keep the vessel close to the outer limit of a channel or a fairway and collision avoidance algorithm will stick to the plan. When very large ships are passing these areas, they are usually accompanied with a pilot or tug in order to clear other potential vessels. When visualizing a vessel passing a narrow channel, the paragraph (b) comes as a logical requirement where vessels of less than 20 meters or sailing vessels should not impede the passage of a vessel which can safely navigate only within the narrow channel or a fairway. We can ask ourselves, how would vessel less than 20 meters long and sailing vessels know that a vessel in fairway can only navigate within the fairway? There is no definite answer to this question and, again, the vessel in the channel or fairway will need to use all available means to avoid collision and inform vessels impeding his way. The only way how can a vessel in the fairway be ready for the uncertainty of the behavior of the vessels that should not impede its way is to maintain safe speed and be ready to stop the vessel if necessary to avoid collision. In case of a human navigator, radio communication can be used to warn other traffic. Areas that have narrow channels and fairways in which vessels large enough to safely sail only within those channels or fairways usually have VTS surveillance, pilotage requirement or even escort boats to ensure no impeding will happen during the transit. This is crucial as many participants in maritime world do not follow COLREGs and some areas are notoriously difficult to transit without escorts (entrance to Bonny port in Nigeria, for example). Considering the inability to maneuver outside the narrow channel or fairway, proposed approach is to compensate for increased uncertainty by not allowing any target entering the minimum CPA. The collision avoidance algorithm should tract TCPA and time it takes for the own vehicle to stop. If the steady trajectory of a target is not changing before the TCPA equals the time to stop, algorithm will send a DSS agent warning to reduce speed and/or stop engine. In case of an autonomous agent, signal from the collision avoidance algorithm will be an input information to the telegraph. The signal has to be issued on time, allowing own vehicle to stop on time.

The paragraph (c) of the Rule 9 is more deterministic than paragraph (b). It states that a vessel engaged in fishing shall not impede the passage of any other vessel navigating within a narrow channel or a fairway. This is clear instruction to fishing vessels and, in this paragraph, conditional sentence that applies this paragraph to vessels that can only navigate safely within a narrow channel or fairway does not exist. In this paragraph it is obvious that fishing vessels should stay clear of other vessels using narrow channels and fairways. From the perspective of collision avoidance algorithm, it is still imperative to maintain conservative approach and include the uncertainty of fishing vessel behavior and maintain stringent collision avoidance trajectory tracking. The paragraph (d) states that any type of vessel crossing a narrow channel or fairways should not cross if such crossing impedes the passage of vessel which can safely navigate only within such channel or fairway. Vessel using a channel or fairway is also instructed to use a sound signal prescribed in Rule 34(d) if in doubt of intentions of the crossing vessel. Risk averse approach is maintained and crossing vessel is monitored well in advance, so that ownship can adjust speed to ensure no contact with a crossing vessel. The control of sound signals is easy and if the crossing vessel's trajectory is entering the CPA preferred radius, signal as per Rule 34 (d) will be used. If in the next minute there is no change in trajectory of a crossing target, signal to DSS or autonomous agent will be issued to slow down or stop the vessel, depending on the resulting distance clearance from the crossing target. Subparagraphs (i) and (ii) of the paragraph (d) deliver requirements for overtaking situation in narrow channels or fairways. When overtaking can take place only if vessel to be overtaken has to take some action to permit safe passage, then vessel that intends to overtake should sound the appropriate sound signal as prescribed in Rule 34(c)(i), after which the vessel to be overtaken should sound the appropriate signal as per the Rule 34(c)(ii) and take steps to permit safe passage. If in doubt, vessels should sound signals prescribed in Rule 34(d). This Rule does not relieve overtaking vessel of her obligation as per Rule 13. Therefore, in the case of overtaking, it is necessary to ensure that both Rule 13 and Rule 9 are considered. Autonomy is possible if appropriate sensory equipment is installed onboard, otherwise a human navigator has to listen to signals and input info to DSS or respond manually. The future could bring a messaging system that allows for communication between vessels automatically or manually so that the uncertainty is lowered to even lower levels.

Paragraph (f) instructs vessels to proceed with care and use appropriate signals from the Rule 34(e) when getting close to a bend of a narrow channel or fairway, or if there is any obstruction, while paragraph (g) instructs vessels not to anchor in a narrow channel unless there is an emergency. These paragraphs are clear and easy to implement.

Rule 10 delivers regulation about Traffic separation schemes (TSS). At the time of writing the COLREGs, TSS was a revolutionary and authors show that introducing TSS contributed largely to collision prevention [Benjamin, 2002, Benjamin et al., 2006]. The paragraph (a) is probably the most important paragraph in the Rule 10, as it states that this Rule applies to TSS adopted by the Organization and that it does not relieve any vessel to comply with other rules. One of the common misconceptions of the Rule 10 among navigators is that once in TSS, it is not required to follow other rules, but just follow own lane. However, obligation of avoiding crossing traffic from own starboard side still exists, which is explained further in Rules overview. The paragraph (b) states that vessels using TSS should proceed in the appropriate traffic lane and navigate in the general direction of traffic; should also stay as practically as possible away from the separation line or zone; and should join or leave a traffic lane near the termination of the lane and at a smallest angle to the traffic flow, which is a similar of the road traffic where vehicles are joining one direction traffic flow in the same direction. This part is ensured by a proper passage planning. The paragraph (c) states that vessels should as far as practicable avoid crossing traffic lanes, but if required to do so, should cross traffic lanes as close to right angles of a traffic flow. This paragraph is a good example why water stabilized RADARs are still required for COLREGs. As mentioned earlier, COLREGs were written mainly for visual reference, so proper lookout by sight is a crucial element of compliance. Many navigators would correct for drift when crossing lanes, while this could confuse other vessels navigating traffic lanes. Basically, ownship has to appear as 90° crossing to the other traffic if the plan is to cross traffic lanes. Again, this has to be planned in advance and if done unplanned, a human navigator has to ensure this paragraph is followed. Paragraph (d) instructs vessels not to use an inshore traffic zone when able to use the appropriate traffic lane with an exception of vessels with length less than 20 meters, sailing vessels and vessels engaged in fishing. The other exception for vessels is that inshore traffic zone can be used by vessels en route to or from a port, anchorage, pilot station or any other place situated within the inshore traffic zone, or to avoid immediate danger. This is very interesting regulation for the proposed collision avoidance algorithm. Basically, entering inshore traffic zone is discouraged, however not prohibited if navigator is avoiding immediate danger. Immediate danger definition still stands as when target occupies own minimum

CPA radius and trajectory keeps target vehicle towards the near miss or collision radius, so at that moment, no constraint is enforced on the collision avoidance algorithm. Avoidance of inshore traffic zone is usually planned well in advance; however, situation can arise when own vehicle has to avoid another and needs to enter in the inshore zone. This especially happens when avoiding crossing traffic from own starboard side. As TSS is marked on ECDIS, collision avoidance algorithm is permitted to scan for distance from both inshore zone and TSS separation zone and try to keep vessel in the middle and if avoiding traffic closer to one side, to prefer to turn to the opposite side. However, this can be burdensome and computationally expensive, so approach is to avoid targets as early as possible and if the optimization forces ownship to go to the inshore zone, ownship will wait until target enters minimum CPA zone and then turn to avoid as necessary. It is also possible to track the trajectory and slow down when necessary for better clearance with a target.

Another challenge for all situation where vessels have to navigate in parallel to each other for prolonged times is introduced. This is applicable to narrow channels, fairways or TSS, or to any situation where ownship does not have enough sea space to maneuver away from targets with satisfactory clearance. For example, Singapore strait or Dover strait are examples where high number of vehicles have to follow the same traffic flow for prolonged times. The challenge becomes when there are scheduled turns and several vessels meet the turn at the approximately same time, so that uncertainty about intent of other vehicles becomes worrisome. With the intent aware and communication module, the issue of further prospects could be avoided. Figure 4.4 depicts the challenge.

If we analyze the situation below, vessels 2 and particularly 5 will have uncertainty challenges. If the speed of vessel 5 is same or higher than 4, navigators of vessel 5 will be concerned about intentions of vessel 4. Of course, they will expect that vessel 4 will turn when reaching the turning point, but as there are ports on the starboard side inshore zone, there is a possibility that 4 wants to exit the TSS and they could slack their speed, they could prolong their straight movement or turn on time with others. If we then also add vessel 7 crossing and making close quarter situation with 5, then we have a situation that can potentially be hard to resolve. Therefore, approach is risk averse and does not take into account any commercial pressure that navigators sometime feel. Safety of vehicles is prioritized, and if slacking of speed is optimal, it is assumed that this is something that has to be done regardless of any other non-collision avoidance related factors. This is why collision avoidance algorithm is modeled to maintain separation of minimum CPA radius for that part of the passage and that checks possibility to either overtake vessel 4 and turns before vessels 4, 6 and 7 reaches this area, or slackens speed so that all pass clear before vessel 5 reaches the turning point. The similar situation would be with vessel 2 or vessel 6 overtaking. They are faster than the other vehicles, but the uncertainty of turning simultaneously on a turning waypoint still exists.

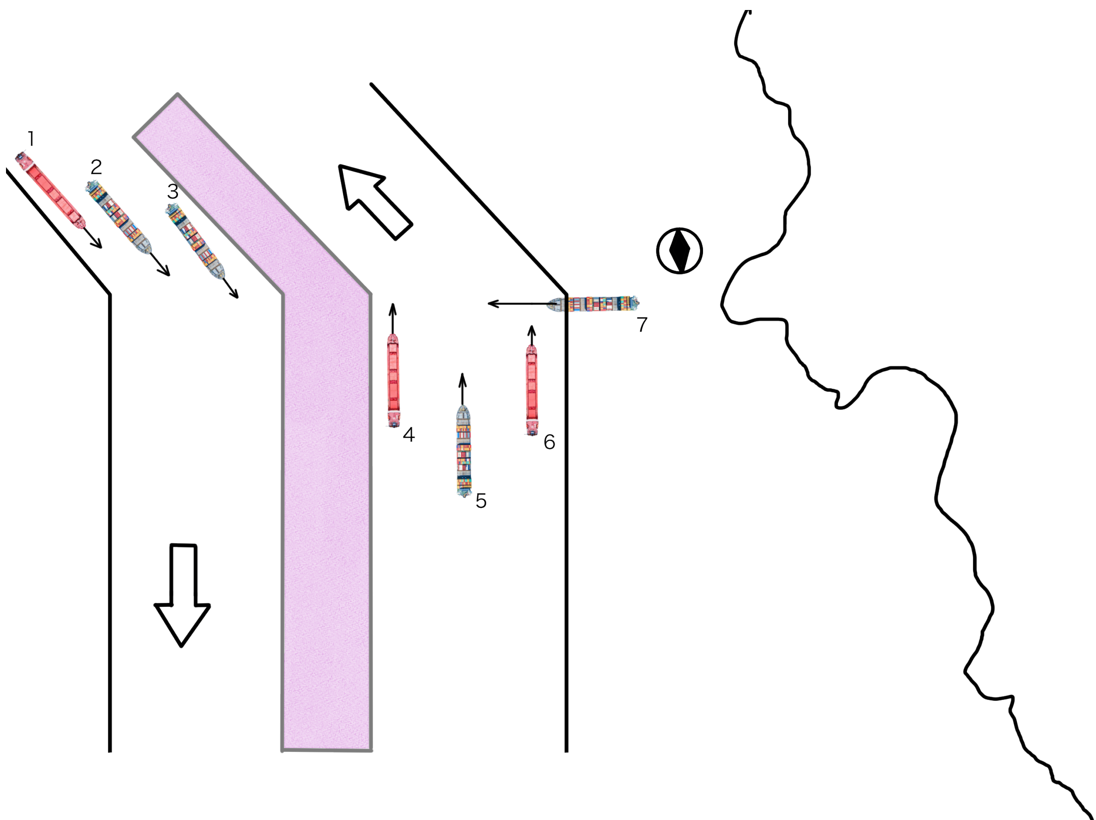


Figure 4.4 – TSS uncertainty challenge

Similarly as for inshore zone, the paragraph (e) delivers restrictions on using the separation zone. Unless crossing a TSS or joining or leaving a traffic lane, vessel should not enter a separation zone unless avoiding immediate danger or to engage in fishing. Paragraph (f) conveys that vessels should navigate carefully near the entrance and terminations of TSS. The entrance can be busy with many vessels joining with different speeds, while termination can be complex due to different destinations and courses vessels could take upon exiting the TSS. The paragraph (g) states that vessels should avoid anchoring. It does not explicitly forbid it, because there is a possibility that vessels would have to do it in emergency situations. Vessels not using TSS should avoid the separation zone with as wide margin as possible according to the paragraph (h), while paragraph (j) obliges vessels engaged in fishing in the separation zone should not impede passage of any other vessel using a TSS. Paragraph (j) delivers similar notation as in Rule 9 where vessels less than 20 meters in length, vessels engaged in fishing, or sailing vessels should not impede a power-driven vessel following a traffic lane. The main difference is lack of a phrase ”safely navigate only within”. In the case of Rule 10, distinction is clearer. The paragraph (k) brings an exception for complying with the Rule 10 for any vessel restricted in her ability to maneuver when engaged in operation for the maintenance of safety of navigation in TSS. However, there is still a challenge of identifying vessels that are engaged in safety of navigation maintenance. It is possible to see from AIS, or visually, that vessel is restricted in her ability to maneuver, but how can agent know if the target vehicle is engaged in maintenance operation? The clarification can only be provided by usage of communication equipment and input from human navigators. In any case, there is a requirement within modelled collision avoidance algorithm to stay clear of vessels restricted in their ability to maneuver regardless of the reason and it is imperative to stay clear from them allowing them enough space. Similarly, paragraph (l) conveys that a vessel restricted in her ability to maneuver when engaged in submarine cable laying, servicing or picking up is exempted from complying with Rule 10.

CONDUCT OF VESSEL IN SIGHT OF ONE ANOTHER (11-18)

This section delivers regulations about conduct of vehicles that meet each other, and a visual reference can be established. Hence, a large emphasis is put on actually seeing targets. For the purpose of COLREGs, human sight is referred to in the Rules. Even if there are sensors onboard that could “see” better than human eyes, COLREGs are written for humans, so vessels have to be in human sight of each other. This requirement is not particularly difficult with DSS systems, but significantly challenging for autonomous systems from the perspective of compliance. Technology installed onboard sea surface vehicles is not ignored, but in regards of compliant actions to be taken, visual confirmation is required before proceeding. In this model assumption is made that visibility lower than 3 NM is low enough to trigger restricted visibility part of the Rules, however, this can be easily adjusted for any value once the agreement among Administrative states is achieved. Algorithm will deliver collision avoidance trajectory as soon as able, even before the visual reference is made. Navigator will have a choice to act upon the proposal from the algorithm, or to wait for visual confirmation and then follow the collision conflict resolution. In the fully autonomous case, the agent would wait for sensory confirmation from cameras acting as human observer.

Rule 11 explains application stating that rules 11-18 apply to vessels in sight of one another.

Rule 12 is disregarded as it is relevant for sailing vessels acting with each other, and this is out of scope of this research. Interaction with sailing vessels in order to avoid collision is included, but we do not consider interaction of two sailing vessels.

Rule 13 regulates overtaking. The paragraph (a) uses word "notwithstanding" anything contained in the Rules of part B, sections I and II, which basically means in spite or regardless of anything stated in the Rules of part B, sections I and II, any vessel overtaking any other vessel must keep out of the way of the vessel being overtaken. In simple terms, overtaking vessel has to complete overtaking with sufficient distance clearance from the vessel she is overtaking. The vessel that is overtaking will have higher speed and can be considered as a give-way vessel as per the Rule 16, while overtaken vessel would be a stand-on vessel [Benjamin, 2002, Benjamin et al., 2006]. The paragraph (b) describes the geometry of overtaking vessels and states that a vessel shall be deemed to be overtaking when approaching another vessel from a direction more than 22.5° abaft her beam, so that she can only see the stern light of the vessel being overtaken, but neither of her sidelights. This is clear reference for human navigators and because sectors of navigational lights are known, this is easily transferable to machine reading. The paragraph (c) states that if a vessel is in doubt if she is overtaking, she shall assume that this is the case and act as overtaking. A very significant paragraph (d) states that overtaking vessel should not overtake and adjust course to become a crossing vessel and after course alteration claim that she was a crossing vessel, so her duty of keeping clear of the overtaken vessel stands until finally past and clear. This is particularly apparent in Figure 4.5.

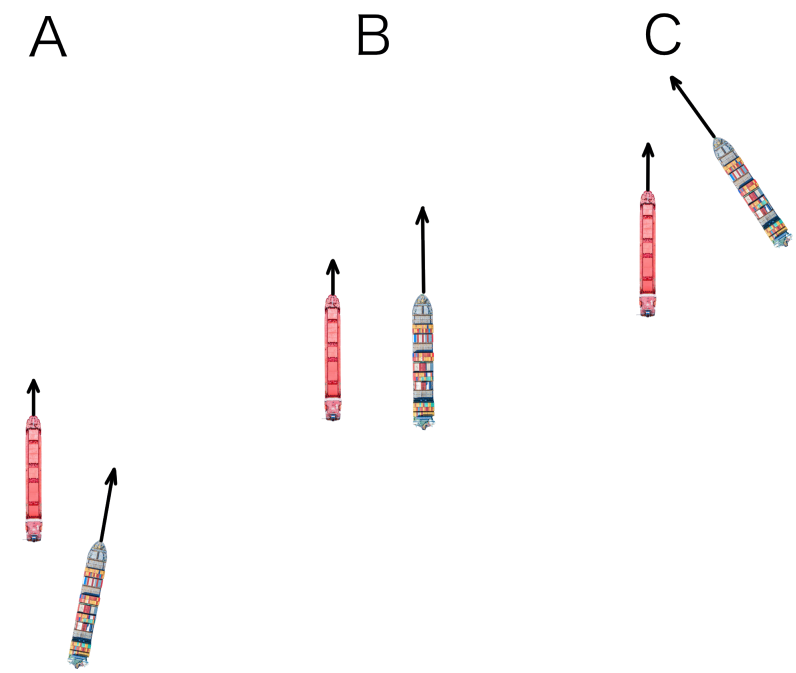


Figure 4.5 – Overtaking challenges

As depicted in Figure 4.5, overtaking vessel decided to cross the path of the vessel being overtaken raising a doubt if the overtaking has been abandoned and crossing initiated. The paragraph (d) is clear in stating that the overtaking maneuver has to be completed and overtaken vessel well clear before any other maneuver is possible. Similar situation is seen as described in Rule 10 when overtaking is happening in TSS or in narrow channel or fairway. If ownship is the overtaking vehicle, proposed algorithm would check if the overtaking maneuver can be completed before the turn and if the vessel should increase speed to complete the maneuver or slacken the speed to abolish intention of overtaking. Sometimes ownship will have an intention of crossing the TSS to port in order to reach anchorage, pilot station or similar, for which it is necessary to position own vehicle more towards the separation line and cross the opposite traffic lane of the TSS. This is a common occurrence in Singapore straits when sailing East and crossing the west travelling traffic lane to reach anchorage or pilot station. In this case, proposed collision avoidance algorithm will have to complete all overtaking or slacken the speed before turning perpendicular to the traffic flow. It is also necessary to ensure that if own vehicle is being overtaken, ownship has to maintain heading and speed, but if there is clear indication that the point of turn would be reached before the overtaking vessel clears own vehicle, ownship is required to slacken speed and prevent close quarter situations.

Overtaking in parallel is somewhat easier situation as both vessels will be sailing similar courses, but if overtaking begins under any relative angle, own vehicle has to ensure that overtaking does not finish with crossing of the bow of vessel being overtaken. Hence, ownship has to ensure to overtake passing the stern of the vessel being overtaken. To ensure this is followed, reward function is modeled in a way to penalize crossing of the bow until vessel is clear. In this study, this happens when the target vehicle is in the CPA preferred radius abaft the beam of the own vehicle.

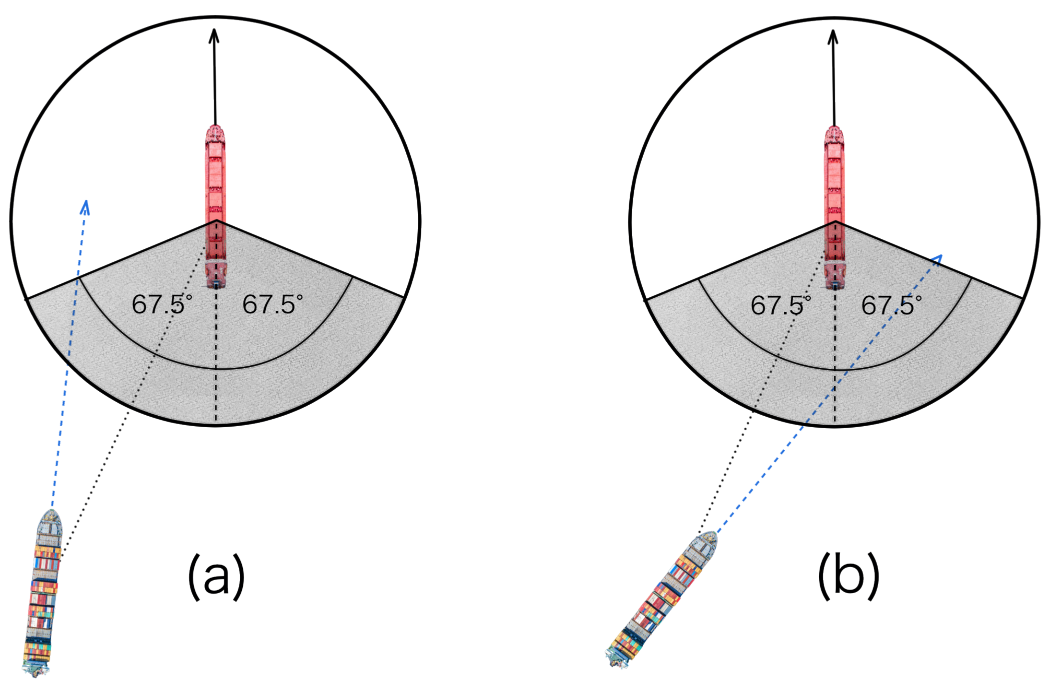


Figure 4.6 – Overtaking geometries

Figure 4.6 depicts situation where a container vessel has a choice of overtaking the bulk carrier over her port side (a) or over her starboard side (b). In the case (a) it is assumed that the container vessel can't alter any more to port. By altering her course to starboard, container vessel ensures that there are no close quarter situations during the overtaking maneuver and that stand-on vessel will not be forced to alter course or speed in order to prevent contact.

As mentioned earlier, a vessel being overtaken is a stand-on vessel, which requires a vessel being overtaken to maintain its course and speed. In case of a straight course and unlimited seas without perils for the own vehicle, it is possible to maintain the speed and course. However, if in TSS, safety fairway or narrow channel and ownship has to take a turn to remain in safe waters, reducing speed to allow for overtaking vessel to pass faster should be considered. The collision avoidance algorithm will take a sequential decision look ahead and determine if the overtaking can be completed before the turn so that target vessel occupies preferred CPA radius. If this is not achievable, request for slacking the speed will be sent to DSS navigator or in the case of full autonomy to the propulsion actuator.

Rule 14 Head-on situation. Rule 14 is one of the frequently disputed rules, both in practice and Case Law mostly because of the difficulty in figuring out is a vessel in head-on or in crossing situation. Paragraph (a) states that when two power-driven vessels meet on reciprocal or nearly reciprocal courses and the collision risk exist, each vessel should alter their course to starboard in order to pass on the port side of other. Paragraph (b) states that such situation exists when a vessel sees the other ahead or nearly ahead and by night can see masthead lights in a line or nearly in a line and both sidelights, while by day she observes corresponding aspect of the other vessel. It is noticeable that this rule is vague and is left to interpretation. Considering many types of vessels, as well as external disturbances when vehicles are rolling, yawing, and pitching significantly, determination of visual attitude and reference is not easy. Authors Bukaty and Morzova [2010] showed that visual and radar confirmation of reciprocal or nearly reciprocal targets is rarely conclusive and can't be done with confidence. This is the reason why COLREGs introduced paragraph (c), which states that in case of uncertainty, assumption is made that there is a head-on situation and to make a starboard turn on time, so that the course change is apparent to the target vehicle, and that the intent of collision avoidance is clearly shown. If human navigator is optimizing collision avoidance, then the paragraph (c) can be left to stand as it is and rely on individual perception and decision-making skills. However, for machine optimization purpose, it is necessary to define reciprocal in order to use the COLREGs determination algorithm correctly. It is necessary to reemphasize that determination of the own vehicle's and target vehicle's attitude is based on visual reference; therefore, water stabilized radar has to be in use.

It is not easy to find consensus on reciprocal and near-reciprocal course reference, so Case law and P&I insurance companies are considered, as they stated that near reciprocal course is roughly within half a point (6°) on either side of own vehicle bow [Benjamin, 2002, Benjamin et al., 2006]. In practice, whenever in doubt, ownship turns to starboard. Turning to port in head-on situation could be catastrophic, especially if the target vessel turns to starboard as mandated by the COLREGs. Therefore, if the own vehicle is within the 6° port and starboard threshold from the current course through water that target is doing, head-on mode is on; otherwise, it is not a head-on situation. Considering possibility of significant uncertainty, ownship should always act by turning to starboard if the target vessel occupies minimum CPA radius and the target vehicle's trajectory is consistent with no apparent change.

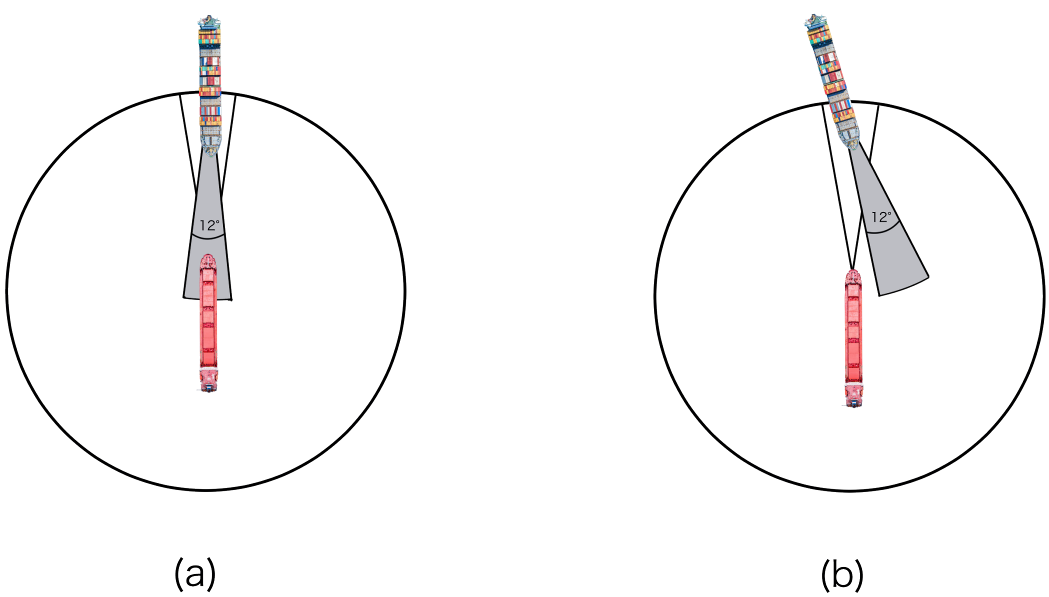


Figure 4.7 – Head-on geometries with 12° sectors

Figure 4.7 depicts two situations of the head-on encounters, where (a) shows how head-on situation is determined with designed angle and range geometries, while (b) shows a case where range is satisfying the heads-on situation, but the relative angle does not, so this would not be a case of heads-on situation, but rather a crossing situation.

Rule 15 has only one paragraph and it delivers regulation in regard to crossing situation. The crossing Rule states that when two power-driven vessels are crossing and there is a risk of collision, the vessel which has the other vessel on her own starboard side shall keep out of the way and, if possible, should avoid crossing the bow of the other vessel. Regulators did not restrict maneuvering to starboard only, but if thinking about crossing from the starboard side, vessel should alter to starboard and pass astern of the crossing vessel, if the circumstances of the case permit.

The Rule 15 assigns give-way and stand-on responsibilities among two crossing power-driven vehicles, but only when there is a risk of collision. Even though own threshold for collision risk might be different than the target vessel, own vehicle can act conservatively and risk averse, which is always a good approach to avoid close-quarter situations. If thinking about other COLREGs geometries, it is noticeable that crossing situation is anything that is not classified as head-on or overtaking situation. Therefore, when the relative bearing of the target vessel is in the spectrum of [6°, 112.5°] and [247.5°, 354°], and if the trajectory is bringing the target vessel to the minimum CPA radius, then navigator confirms that there is a risk of collision and then decides on the crossing action. In several Admiralty cases, a notion that crossing give-way vessel should not cross ahead of the stand-on vessel has been confirmed [Benjamin et al., 2006], therefore it is necessary to ensure that algorithm verifies crossing the stern is optimal behavior when verifying generated trajectories.

Figure 4.8 depicts container vehicle crossing the determined range of the target bulk carrier with relative bearings taken from the container vessel (a) and taken from the bulk carrier (b). Similarly, Figure 4.9 showcases geometries of a stand-on vehicle, where container vessel is stand-on vehicle with relative bearings taken from the container vehicle (a) and taken from the bulk carrier (b).

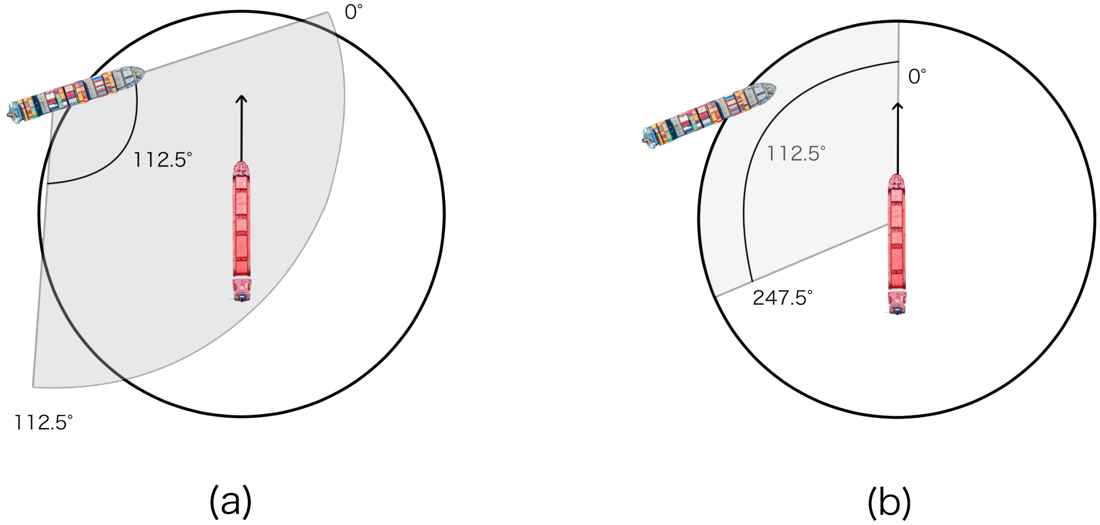


Figure 4.8 – Give-way geometries

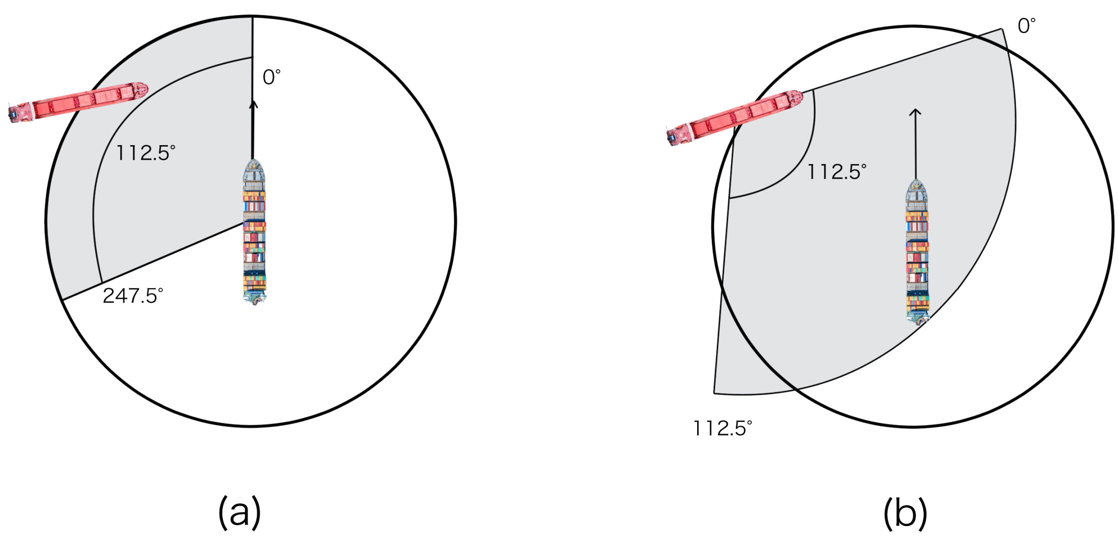


Figure 4.9 – Stand-on geometries

Rule 16 is related to actions that give-way vessels have to take. In a single paragraph the Rule 16 states that every vessel which is directed to keep out of the way of another vessel has to take early and substantial action to keep well clear, if circumstances allow so. This Rule is straightforward and is actually deeply interconnected with other rules. Vessel should take actions early in order to minimize the risk of collision. The maneuver has to be apparent to other targets and that is the reason why the Rule 16 mentions substantial action. It is necessary to emphasize that the Rule is simple and is not constrained by the type of a vehicle or collision situation. Rule 16 interacts with Rules 12, 14, 15, and 18. In order for machines to "understand" the Rule 16, metrics for when to initiate maneuvers are required and to determine what is the substantial action in certain collision situation. These metrics are already provided in relevant sections of the Rules.

Rule 17 is somewhat more complex, as it requires a sort of "patience" and decision making that is not straight forward. Unlike Rule 16 where a give-way vessel is required to act early and substantially, Rule 17 requires that a stand-on vessel is keeping its course and speed until it is evident that target vessel is not acting as per the Rule 16 and then it is required from a stand-on vessel to act in any way that will avoid contact. Continuance of vagueness of the Rules is noticeable, as it would be impractical to impose thresholds for all vessel types. Therefore, it is necessary to rely on individual navigators' ability and sense of risk. Machines do not have human intuition, so CPA metrics should be used similarly as in other Rules. The Rule 17 starts with paragraph (a) that states where one of two vessels is to keep out of the way, the other vessel has to keep her course and speed. However, as soon as it becomes apparent to the stand-on vessel that the give-way vessel is not taking appropriate action, the stand-on vessel has to take action to avoid collision. The paragraph (b) states that if the stand-on vessel realizes collision can't be avoided by the give-way vessel alone, the stand-on vessel has to take action that will best aid to avoid collision. The paragraph (c) states that if two power-driven vessels are in crossing situation, the stand-on vessel should avoid altering her course to port if the circumstances of the case admit. The paragraph (d) delivers important notation that this Rule does not relieve the give-way vessel of her obligation to keep out of the way. As stated earlier, Admiralty courts commonly assign blame in percentages between involved parties. In regard to Rule 17, a usual fault of 25 % is assigned to any stand-on vessel not maneuvering when there is a risk of collision and collision occurs [Benjamin, 2002]. As with other Rules of the COLREGs, it is necessary to determine the attitude and assign relevant Rules before decision about appropriate maneuver is taken. In the busy waterways, vehicles would usually interact with more than one target. Even though there is an obligation to maintain course and speed, sometimes own vehicle will interact with other targets and the holistic approach is necessary to maintain separation. Often maintaining course and speed will be impractical, but it is necessary to ensure risk of collision is as low as possible. That is why dCPA separation radii is still enforced, and ownship acts according to the collision risk determined at TCPA. An imminent risk criterion is enforced, as whenever a target occupies safe CPA radius sector and targets' trajectory is entering minimum, near-miss, or collision CPA radii sector, where immediate action is required to increase separation. When there are no vehicles in the safe CPA radius and there are no imminent threats, Rule 17 requires that stand-on vessel maintains course and speed. In order to achieve this, reward function will be designed to penalize changes of speed and course own vehicle is keeping at that time. Reward function is designed in a way that there is some tolerance for the speed change, after which stand-on vessel is penalized, as well as with course changes, where gradual change of course more than 2° would incur increasing penalties. Of course, if there is a turn on the way, or any other imminent danger within the safe CPA zone, this requirement would be disregarded. The goal is, therefore, to correctly identify collision risk and act accordingly. When decision is made not to keep course and speed, own action has to be substantial and intention visible to all targets within the range.

Rule 18 This Rule requires differentiation of various types of vessels, and as already stated before, to assure full autonomy, specific sensory equipment that can read navigational lights and daylight shapes is required. It is possible to get the information about the type and condition of commercial vessels through AIS, but there is an uncertainty of incorrect data. Therefore, fusion of data would be the best approach. For the current sensory equipment found on commercial sea surface vehicles, involvement of human navigators is necessary to input type of a ship. Related to the Rule 18, but also COLREGs in general, intent aware and cooperative model is proposed that would aid in a sea surface vehicle type classification. Once the ship type and restrictions are determined, Rule 18 is straight-forward to follow.

CONDUCT OF VESSEL IN RESTRICTED VISIBILITY (19)

Rule 19 is the single Rule of the restricted visibility section. The Rule 19 applies to all vessels not in sight of one another when navigating in or near areas of restricted visibility. In order to apply Rule 19, a threshold for restricted visibility has to be defined. COLREGs do not offer any number, but common threshold of three nautical miles below which the visibility is considered restricted [Benjamin, 2002] is used in this study. The paragraph (b) delivers a significant restriction stating that every vessel shall proceed at a safe speed adapted to the prevailing circumstances, while a power-driven vessel must have her engines ready for immediate maneuver. In this case, focus is on underactuated commercial power-driven sea surface vehicle, so whenever the visibility is measured below 3 NM and there is traffic present in the vicinity, own RPMs are reduced to Maneuvering Full Ahead in order to be ready for immediate maneuvering as per the Rule 19. Safe speed selection is described in more detail in the Rule 6. For example, on LNG ships that use LNG as fuel, slowing down from Navigational Full to Maneuvering Full takes up to 30 minutes, so it is important to be ready for an immediate maneuvering action. The paragraph (c) reinstates that every vessel has to consider prevailing circumstances and conditions of restricted visibility when complying with section I of this Rules. The paragraph (d) delivers detailed instructions on how to act when detecting targets by radars alone. In such circumstances, vessels have to determine if there is a risk of collision and then act early; however, an alteration of course to port for a vessel forward of the beam other than for a vessel being overtaken should be avoided, as well as an alteration of course towards a vessel abeam or abaft the beam. This requirement is emphasized in order to allow for additional uncertainty of sailing in restricted visibility. Final paragraph (e) requires that each vessel which hears apparently forward of her beam the fog signal of another vessel, or vessel that has determined a collision risk with a target forward of her beam, has to reduce her speed to the minimum at which she can maintain her course. The paragraph (e) continues by instructing vessels to further reduce her ahead movement if necessary, for maintaining safe separation.

PART C – LIGHTS AND SHAPES (20-31)

COLREGs are explicit in requirements for lights and shapes, so there is no need for further explanation. It is difficult to envision autonomous light and signal control at this stage. Vessel should recognize certain conditions, such are constrained by draft, restricted maneuvering, restricted visibility, underway, anchored, etc. Some of these states are easier to determine, but some still require human decision making. Once the state is determined, certain visual and sound signals have to be prominently displayed and used according to the COLREGs. Another difficulty is recognition of targets' lights and shapes, for which accurate and certified sensory equipment is required. Therefore, human interaction can't be completely avoided, but can be enhanced by automation. There is a potential benefit of amending the COLREGs with special signals that autonomous sea surface vehicles have to display; however, a consensus has to be achieved on the IMO level.

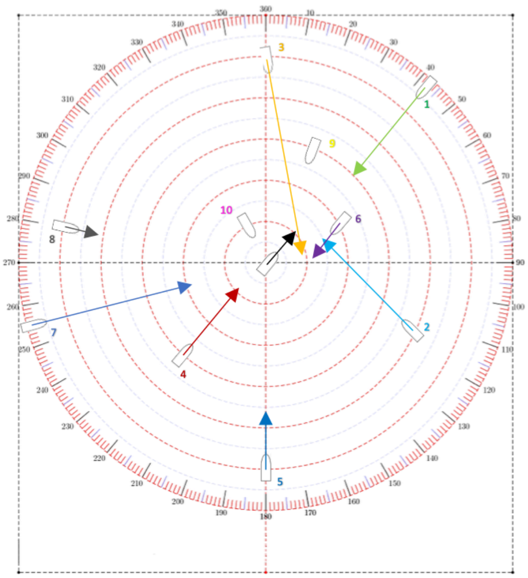
PART D – SOUND AND LIGHT SIGNALS (32-37)

Similarly as with lights and shapes, sound and light signals require setup self-awareness and accurate determination of targets' signals.

In the Appendix C, COLREGs classification algorithm is delivered. Classification algorithm is utilized only to determine appropriate Rule in certain circumstance and not to generate evasive maneuvers. Therefore, reward functions with all constraints relevant to COLREGs will be presented in the next chapter, where 8 cases that would be used for testing collision classification algorithm are delivered, but collision avoidance algorithm as well. Some scenarios are similar, but have obstacles, shallow water, TSS, narrow channels, or safety fairways introduced to make maneuvering challenging.

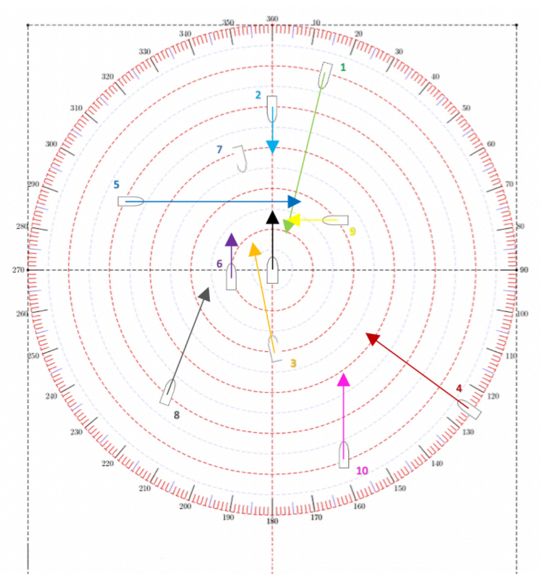
4.1.3 COLREGs classification testing and evaluation

In order to verify compliance with COLREGs, 8 scenarios where COLREGs classification algorithm has to correctly identify relevant Rules for that scenario were developed. These scenarios will also be used for the collision avoidance algorithm. All vectors in the following scenarios are of 12 minutes length.



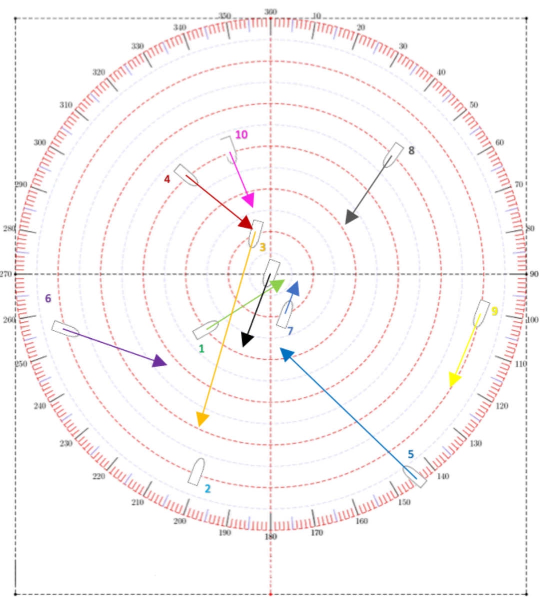
Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	040	5.2	1.04	LNG
Vehicle 1	220	11	2.2	Feeder
Vehicle 2	315	15.7	3.14	VLCC
Vehicle 3	170	24	4.8	Pass
Vehicle 4	040	10.7	2.14	Supplier
Vehicle 5	000	6.8	1.36	LNG
Vehicle 6	220	5.5	1.1	VLCC
Vehicle 7	075	20.4	4.08	Feeder
Vehicle 8	100	3.7	0.74	Pass
Vehicle 9	200	0	0	LNG
Vehicle 10	150	0	0	Supplier

Figure 4.10 – Scenario 1



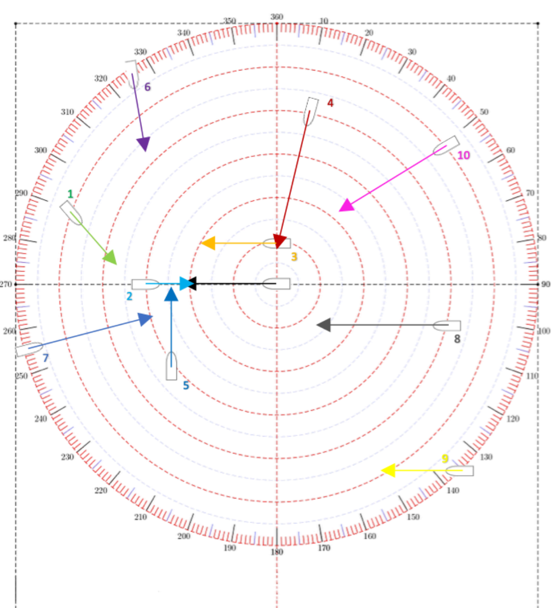
Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	000	6.8	1.36	LNG
Vehicle 1	195	20.4	4.08	Feeder
Vehicle 2	180	5.5	1.1	VLCC
Vehicle 3	350	14.9	2.98	Pass
Vehicle 4	305	16.2	3.24	Supplier
Vehicle 5	090	21.3	4.26	LNG
Vehicle 6	000	5.5	1.1	VLCC
Vehicle 7	165	0	0	Feeder
Vehicle 8	020	14.9	2.98	Pass
Vehicle 9	270	6.8	1.36	LNG
Vehicle 10	000	10.7	2.14	Supplier

Figure 4.11 – Scenario 2



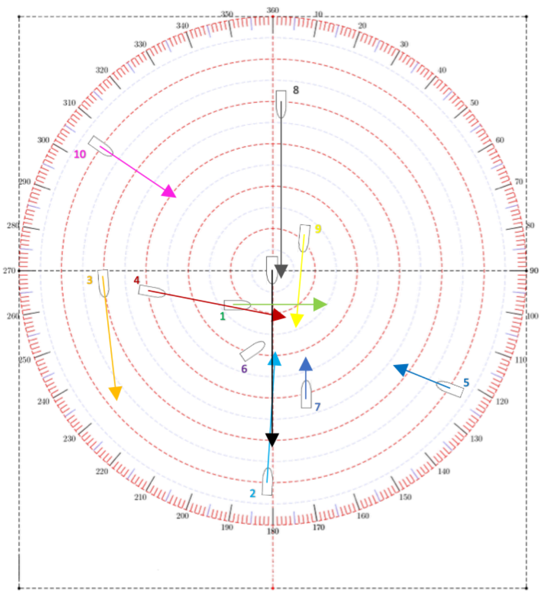
Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	200	9	1.8	LNG
Vehicle 1	060	11	2.2	Feeder
Vehicle 2	020	0	0	VLCC
Vehicle 3	195	24	4.8	Pass
Vehicle 4	130	9.5	1.9	Supplier
Vehicle 5	310	21.3	4.26	LNG
Vehicle 6	110	12.8	2.56	VLCC
Vehicle 7	020	4	0.8	Feeder
Vehicle 8	215	9.7	1.94	Pass
Vehicle 9	210	9	1.8	LNG
Vehicle 10	160	7.2	1.44	Supplier

Figure 4.12 – Scenario 3



Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	270	11	1.8	LNG
Vehicle 1	140	8	2.2	Feeder
Vehicle 2	090	5.5	0	VLCC
Vehicle 3	270	9.7	4.8	Pass
Vehicle 4	195	16.2	1.9	Supplier
Vehicle 5	000	9	4.26	LNG
Vehicle 6	170	9	2.56	VLCC
Vehicle 7	075	15.5	0.8	Feeder
Vehicle 8	270	14.9	1.94	Pass
Vehicle 9	270	9	1.8	LNG
Vehicle 10	240	14.1	1.44	Supplier

Figure 4.13 – Scenario 4



Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	180	21.3	4.26	LNG
Vehicle 1	090	11	2.2	Feeder
Vehicle 2	003	15.7	3.14	VLCC
Vehicle 3	175	14.9	2.98	Pass
Vehicle 4	100	16.2	3.24	Supplier
Vehicle 5	290	6.8	1.36	LNG
Vehicle 6	055	0	0	VLCC
Vehicle 7	000	4	0.8	Feeder
Vehicle 8	180	20.7	4.14	Pass
Vehicle 9	185	11	2.2	LNG
Vehicle 10	125	10.7	2.88	Supplier

Figure 4.14 – Scenario 5

Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	149	21.3	4.26	LNG
Vehicle 1	149	5.5	1.1	VLCC
Vehicle 2	060	9.5	1.9	Supplier
Vehicle 3	325	20.4	4.08	Feeder
Vehicle 4	325	9.7	1.94	Pass
Vehicle 5	240	5.5	1.1	VLCC

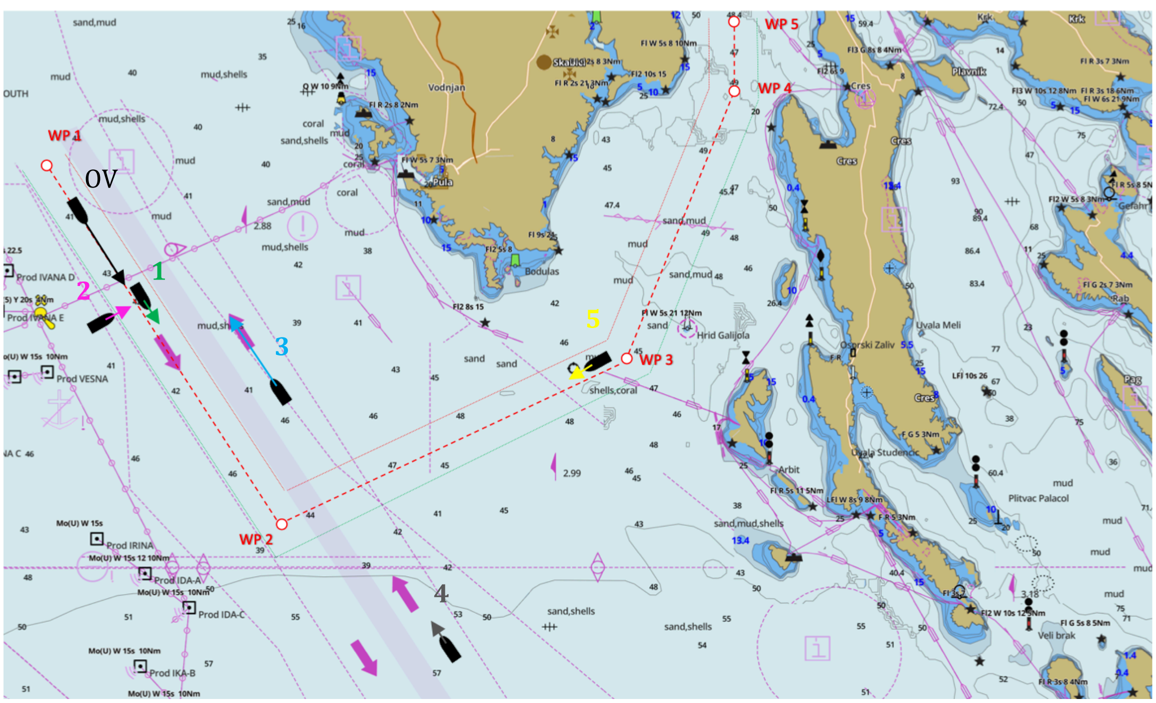


Figure 4.15 – Scenario 6

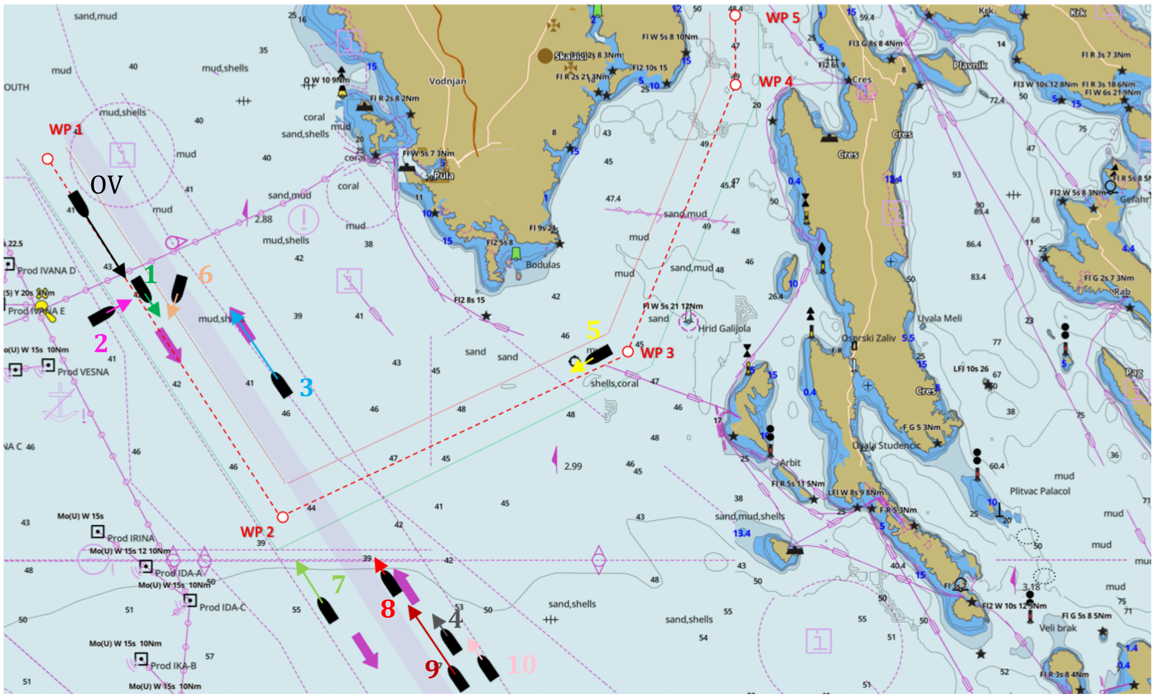


Figure 4.16 – Scenario 7

Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	149	21.3	4.26	LNG
Vehicle 1	149	5.5	1.1	VLCC
Vehicle 2	060	9.5	1.9	Supplier
Vehicle 3	325	20.4	4.08	Feeder
Vehicle 4	325	9.7	1.94	Pass
Vehicle 5	240	5.5	1.1	VLCC
Vehicle 6	196	5.5	1.1	VLCC
Vehicle 7	327	16.2	3.24	Supplier
Vehicle 8	326	5.2	1.04	LNG
Vehicle 9	325	20.4	4.08	Feeder
Vehicle 10	325	9.7	1.94	Pass

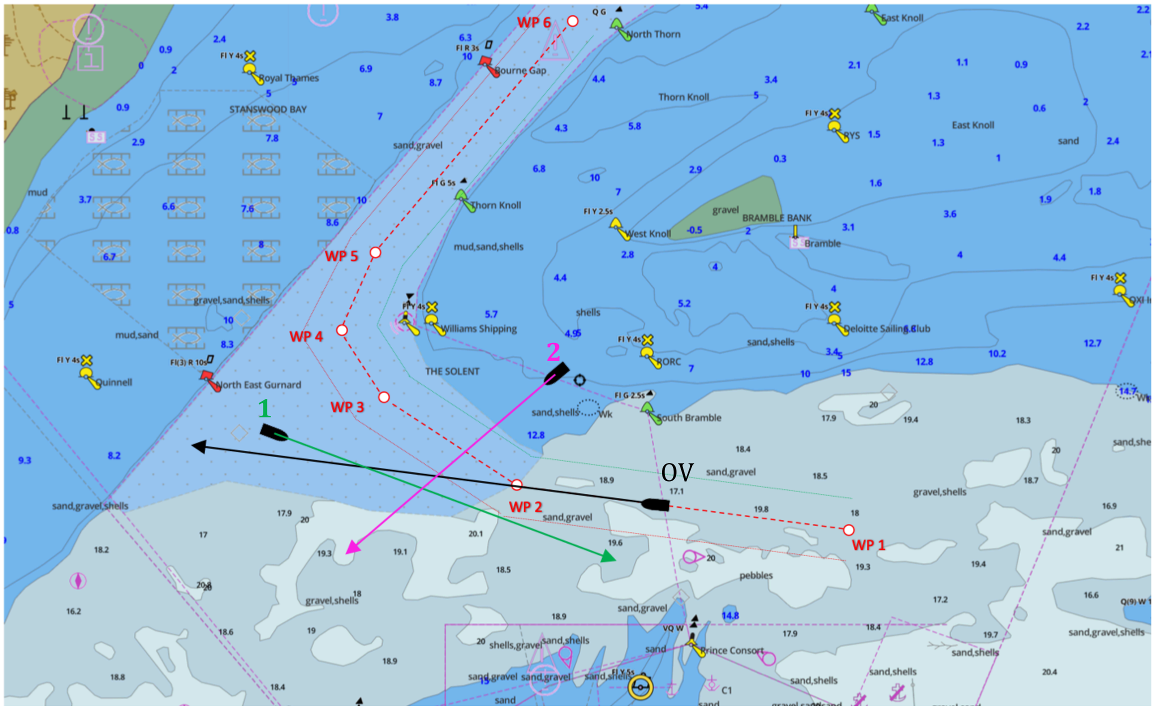


Figure 4.17 – Scenario 8

Vehicle	COG (°)	SOG (kt)	Vector (NM)	Type
Own	275	11	1.1	LNG
Vehicle 1	110	8	0.8	Feeder
Vehicle 2	230	6	0.6	Supplier

In this study intention is not to cover all possibilities, but rather to investigate how would proposed algorithms behave in some of the critical scenarios. How COLREGs Classification Algorithm copes with first five scenarios is the focus of this section; therefore, it is assumed that all speeds and courses are water stabilized as required by the COLREGs. Resolution of conflicts is not considered at this time. Matlab_R2021a on a personal computer with 2.8 GHz Quad-Core Intel Core i7 processor and 16 GB RAM is used to simulate above cases and determine if the COLREGs determination and classification is done accurately by the algorithm.

In the first run, there are no external influences and that all vectors are as mentioned in the case descriptions above, which means that heading, speed, course over ground and through water are all same without any set or drift. Afterwards, random external disturbances are utilized to verify results with different geometries. For the COLREGs classification algorithm, external disturbances do not add to complexity, as only change in heading due to environmental loads is tracked and then pairwise geometries determined to establish classification. Tables below confirm that proposed classification algorithm is capable of determining appropriate Rules. Classification can be done offline with certain segments online (data transmission), but also it is possible to reduce computation time by taking some information directly from viable sensors.

Table 4.1 – COLREGs Classification Scenario 1

Rule	Description
7	COLLISION RISK EXISTS WITH T1
7	COLLISION RISK EXISTS WITH T2
7	COLLISION RISK EXISTS WITH T3
7	COLLISION RISK EXISTS WITH T4
7	COLLISION RISK EXISTS WITH T6
7	COLLISION RISK EXISTS WITH T7
7	COLLISION RISK EXISTS WITH T9
13	OVERTAKEN BY T4 - KEEP COURSE AND SPEED
14	T1 HEAD-ON
15	T2 CROSSING BOW FROM STARBOARD – GIVE-WAY
15	T3 CROSSING BOW FROM PORT – STAND-ON
15	T7 PASSING STERN FROM PORT – STAND-ON
16	GIVE-WAY TO T1
16	GIVE-WAY TO T2
16	GIVE-WAY TO T9 (T9-NUC)
17	(T3) STAND-ON
17	(T4) STAND-ON
17	(T5) STAND-ON
17	(T7) STAND-ON
17	(T8) STAND-ON
17	(T10) STAND-ON
////	Mean Execution Time (100 iterations) = 0.27308 sec

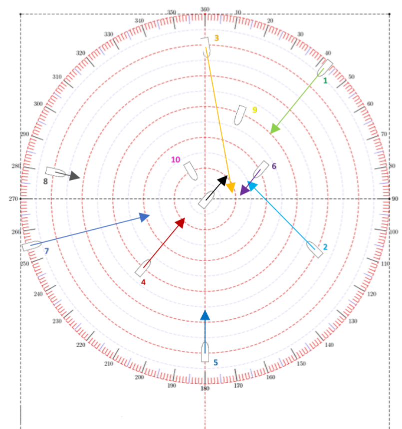


Table 4.2 – COLREGs Classification Scenario 2

RULE	DESCRIPTION
7	COLLISION RISK EXISTS WITH T1
7	COLLISION RISK EXISTS WITH T2
7	COLLISION RISK EXISTS WITH T3
7	COLLISION RISK EXISTS WITH T5
7	COLLISION RISK EXISTS WITH T6
7	COLLISION RISK EXISTS WITH T7
7	COLLISION RISK EXISTS WITH T8
7	COLLISION RISK EXISTS WITH T9
13	OVERTAKEN BY T3 - KEEP COURSE AND SPEED
13	OVERTAKING T6 – GIVE-WAY
14	T2 HEAD-ON
15	T1 PASSING STERN FROM STARBOARD – GIVE-WAY
15	T5 PASSING STERN FROM PORT – STAND-ON
15	T9 PASSING STERN FROM STARBOARD – GIVE-WAY
16	GIVE-WAY TO T1
16	GIVE-WAY TO T2
16	GIVE-WAY TO T7
16	GIVE-WAY TO T9
17	(T3) STAND-ON
17	(T4) STAND-ON
17	(T5) STAND-ON
17	(T6) STAND-ON
17	(T8) STAND-ON
17	(T10) STAND-ON
////	Mean Execution Time (100 iterations) = 0.2817 sec

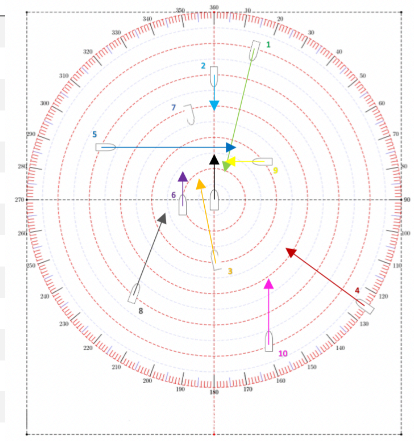


Table 4.3 – COLREGs Classification Scenario 3

RULE	DESCRIPTION
7	COLLISION RISK EXISTS WITH T1
14	T2 HEAD-ON
15	T1 CROSSING BOW FROM STARBOARD – GIVE-WAY
15	T5 PASSING STERN FROM PORT – STAND-ON
15	T6 CROSSING BOW FROM STARBOARD – GIVE-WAY
16	GIVE-WAY TO T1
16	GIVE-WAY TO T6
17	(T3) STAND-ON
17	(T4) STAND-ON
17	(T5) STAND-ON
17	(T7) STAND-ON
17	(T8) STAND-ON
17	(T9) STAND-ON
17	(T10) STAND-ON
////	Mean Execution Time (100 iterations) = 0.2841 sec

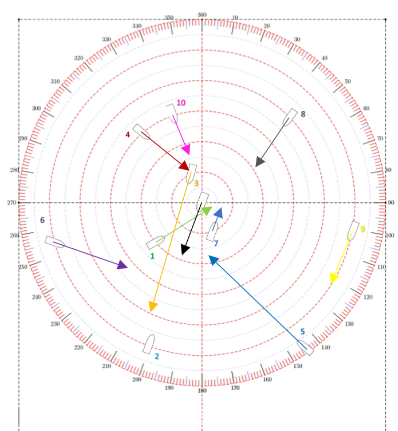


Table 4.4 – COLREGs Classification Scenario 4

RULE	DESCRIPTION
7	COLLISION RISK EXISTS WITH T1
7	COLLISION RISK EXISTS WITH T2
7	COLLISION RISK EXISTS WITH T3
7	COLLISION RISK EXISTS WITH T5
7	COLLISION RISK EXISTS WITH T7
14	T2 HEAD-ON
15	T1 CROSSING BOW FROM STARBOARD – GIVE-WAY
15	T5 CROSSING BOW FROM PORT – STAND-ON
15	T7 PASSING STERN FROM PORT – STAND-ON
16	GIVE-WAY TO T1
16	GIVE-WAY TO T3
16	GIVE-WAY TO T6
17	(T5) STAND-ON
17	(T7) STAND-ON
17	(T8) STAND-ON
17	(T9) STAND-ON
17	(T10) STAND-ON
////	Mean Execution Time (100 iterations) = 0.2939 sec

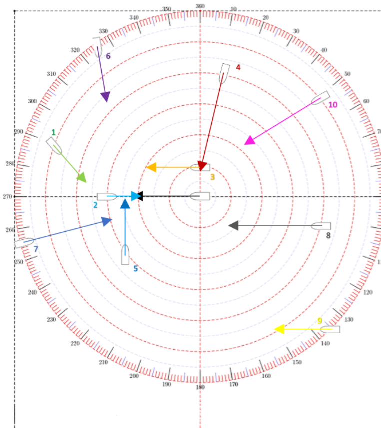
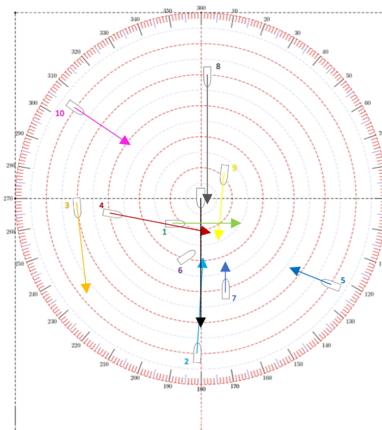


Table 4.5 – COLREGs Classification Scenario 5

RULE	DESCRIPTION
6	CONSIDER SLOWING DOWN TO MANEUVERING FULL
7	COLLISION RISK EXISTS WITH T1
7	COLLISION RISK EXISTS WITH T2
7	COLLISION RISK EXISTS WITH T6
7	COLLISION RISK EXISTS WITH T7
14	T2 HEAD-ON
15	T1 PASSING STERN FROM STARBOARD – GIVE-WAY
15	T6 PASSING STERN FROM STARBOARD – GIVE-WAY
16	GIVE-WAY TO T1
16	GIVE-WAY TO T3
16	GIVE-WAY TO T4
16	GIVE-WAY TO T6
16	GIVE-WAY TO T7
17	(T5) STAND-ON
17	(T8) STAND-ON
17	(T9) STAND-ON
17	(T10) STAND-ON
////	Mean Execution Time (100 iterations) = 0.2879 sec



4.2 Target uncertainties

Even if perfect sensing and controlling of the own sea surface vehicle is possible, the issue of the target behavior still exists. If all navigators had an equal understanding of the COLREGs and if they always obliged the rules, there would be far less incidents at sea. It was already mentioned that COLREGs deserve a rewrite in the light of the newer age, but even if the COLREGs remained the same, a proper education and practice of COLREGs would lead to lower number of collisions and allisions [Mohovic et al., 2016]. Uncertainties related to navigational behavior of the targets met is hard to quantify and implement in models. However, in this chapter utilization of behavioral uncertainties by expanding safety zones of the ownship is proposed. The safest approach is cooperative collision avoidance, so proposal of intent-aware model is delivered in the Section 5.1, but in the case where intent is not known, a risk of not following the COLREGs is assumed.

Throughout the year 2014 a large-scale research was conducted to investigate understanding of the COLREGs [Mohovic et al., 2016]. As depicted in the Chapter 2, failure to comply with the COLREGs is the most influential reason for collisions. The Avoiding Collisions at Sea (ACTS) research [Mohovic et al., 2016] examined understanding and knowledge of COLREGs by both students and experienced navigators in order to verify which Rules are difficult to understand and what is the influence of navigational experience on COLREGs understanding. Even though the major goal of the ACTS research was to identify gaps in knowledge and design better tools to aid in educating students and seafarers to bridge that gap, the data can be utilized to quantify misunderstanding of individual Rules and apply results in proposed algorithms.

Mohovic et al. [2016] created questionnaire that was completed by 1538 participants from 68 different countries. 46 % of the participants were professional seafarers, 36 % were seafaring students, while 18 % represented STCW license holders for various types of smaller crafts. A very prominent result of this research was that even though some administrations still believe that citing COLREGs by heart will improve collision avoidance behavior, the ACTS research actually showed that understanding the Rules and ability to apply the relevant Rules to specific situations does not correlate with the ability to cite Rules. In Figure 4.18 it can be noticed that some Rules are harder to understand, and more specifically harder to apply in real world scenarios. Rules that are hard to understand are Rule 6 – Safe Speed, Rule 10 – TSS, Rule 13 – Overtaking, Rule 14 – Head-on situation, Rule 17 – Action by stand-on vessel, Rule 18 – Responsibilities between vessels and Rule 19 – Conduct of vessels in restricted visibility [Mohovic et al., 2016].

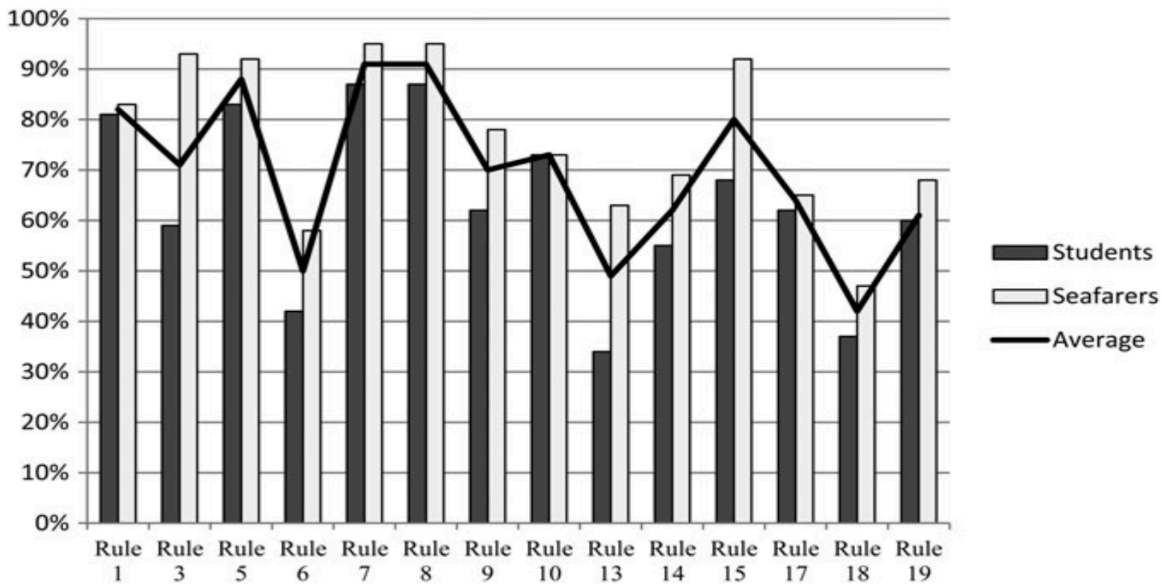


Figure 4.18 – Percentage of correct answers by each Rule (Source: Mohovic et al., 2016)

To understand what the consequences of poor understanding of the Rules are, further study of the Marine Accidents Investigation Branch (MAIB) collision statistics where violation of Rules was determined as contributory cause of marine accidents was done. Figure 4.19 offers an interesting insight in which it is possible to see that understanding Rules still does not safeguard safety of navigation. Even the Rules that are understood well can lead to accidents when navigators are complacent and do not follow the Rules for whatever the root cause is identified in that particular incident. It is not uncommon that professionals get distracted from their daily tasks; therefore, a decision support system is a good fit to reduce a number of incidents and safeguard waterways.

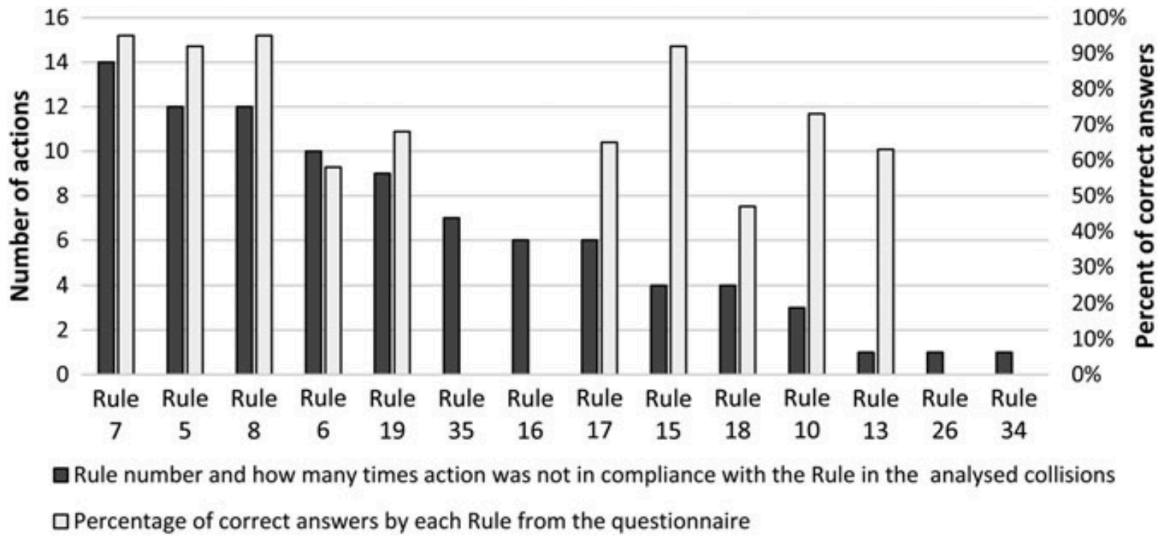


Figure 4.19 – Compliance and correct answers percentage comparison (Source: Mohovic et al., 2016)

After analyzing MAIB collision investigation database, it is evident that the highest number of cases had violated COLREGs' Rule 5 (Look-out), Rule 7 (Risk of Collision) and Rule 8 (Action to avoid collision), even though these Rules are well understood by navigators and students. The reasoning for such results lies deep in the investigation reports and it is often the case that complacency plays a major role in marine accidents. Navigators understand well the meaning of the proper look-out, but, as it is evident from MAIB's incident reports, decide to bypass well-established internal standards to ensure proper look-out at all times. Failure to understand maneuvering characteristics of their own vessels, or underestimating effects of sailing close to another vessel, complacency assured by "we did it so many times before", or reluctancy to use propulsion in collision situations are some of the examples seen in MAIB's incident reports that lead to violation of Rules 7 and 8.

In 40 % of the collisions Rule 17 (Action by stand-on vessel) and Rule 18 (Responsibilities between vessels) are violated, but also from Mohovic et al. [2016] ACTS study it is evident that Rule 17 has 62 % of correct answers, while Rule 18 has 48 % correct answers, which makes them hard to understand and apply in real world scenarios. Similarly, Rules 5, 10, 13 and 19 are considered difficult to understand and lead to collision situations. This is where proposed COLREGs classification algorithm, together with risk quantification can aid the overall collision avoidance algorithm to make encounters safer by extending safety zones of the own vessel.

It is necessary to keep in mind that we can't remove humans from the process, but rather influence root causes of incidents where human factor played a major role. In the content of this section intention is to safeguard ownship from behavioral uncertainties of any targets met during navigation. Trajectory uncertainties are already taken into consideration by developing safety zones, however it is imperative to note that human navigator still has an obligation to choose safety zone, similarly as the parameters of the safety cone have to be selected at each leg of the voyage on the ECDIS. Therefore, unless fully automated and autonomous system is created, human factor will never be removed completely, but risks associated with human operators could be mitigated. As autonomy is not possible in the present legal environment, the focus of this study remains on reducing human errors, rather than eliminating human interaction completely.

For the reasons depicted in Section 4.1, selected number of the COLREGs are considered; however, this model can be extended if found necessary. In this research idea is to expand safety zones of the own vehicle depending on the active rule that is detected by the COLREGs classification algorithm. The logical question is presented: by how much is it necessary to increase the radius of each safety zone for a particular COLREGs rule? Approach in this thesis is based on a traditional qualitative risk assessment where consequence is multiplied by a likelihood. Safety zone expansion factors are, therefore, calculated as follows:

$$e_{XP} = coll_{freq} \times f_{ACTS} \tag{4.1}$$

$$r_{EXP} = r_{SEL} + (r_{SEL} \times e_{XP}) \tag{4.2}$$

where, e_{XP} is the expansion factor, which is used to increase radius of all safety zones for the own ship. At each stage of a passage, navigator choses safety zones, which are then increased by the expansion factor using the formula (4.2). Findings from analyzing MAIB collision reports are utilized to see in how many cases was a violation of a certain

COLREGs rule a contributory factor and use this information as the collision frequency denoted as $coll_{freq}$. If Rule 5 is taken as an example, in 12 cases out of 14, Rule 5 was a contributory factor, which means that $coll_{freq} = \frac{12}{14} = 0.86$. As a result of the ACTS survey [Mohovic, et al., 2016] an overview of which COLREGs rules are easier to understand and which require more comprehensive knowledge and experience is gained. By utilizing the number of incorrect answers in the ACTS questionnaire from experienced seafarers f_{ACTS}, it is possible to compute the expansion factor. r_{EXP} represents expanded radius of the safety zone, while r_{SEL} stands for the selected radius prior expansion, which is chosen by a navigator. Finally, Table 4.6 depicts the expansion factor for each individual COLREGs rule depending on the ACTS survey and MAIB collision database:

Table 4.6 – Expansion factors for safety zones

COLREGs Rule	$coll_{freq}$	f_{ACTS}	e_{XP}
Rule 5 – Look out	0.86	0.08	0.07
Rule 6 – Safe speed	0.71	0.42	0.3
Rule 7 – Risk of collision	1	0.06	0.06
Rule 8 – Action to avoid collision	0.86	0.06	0.05
Rule 9 – Narrow channels	0	0.22	0
Rule 10 – TSS	0.21	0.28	0.06
Rule 13 – Overtaking	0.07	0.37	0.03
Rule 14 – Head-on situation	0	0.31	0
Rule 15 – Crossing situation	0.29	0.08	0.02
Rule 16 – Action by give-way vessel	0.43	0.37	0.16
Rule 17 – Action by stand-on vessel	0.43	0.37	0.16
Rule 18 – Responsibilities between vessels	0.43	0.52	0.22
Rule 19 – Restricted visibility	0.64	0.31	0.2

Once the COLREGs classification algorithm determines applicable rules, the largest e_{XP} of the relevant rules selected by the algorithm is used. For example, in scenario No. 1 COLREGs classification algorithm determined that following rules are relevant: Rule 7, Rule 13, Rule 14, Rule 15, Rule 16, and Rule 17. Rules 16 and 17 have the highest e_{XP} value of 0.16, so in this case safety zone radii will be increased by 16 %. If minimum CPA is selected as 1 NM by the navigator, until the COLREGs algorithm reads that there is a collision danger with that target, minimum CPA radius will be increased by 16 % to 1.16 NM.

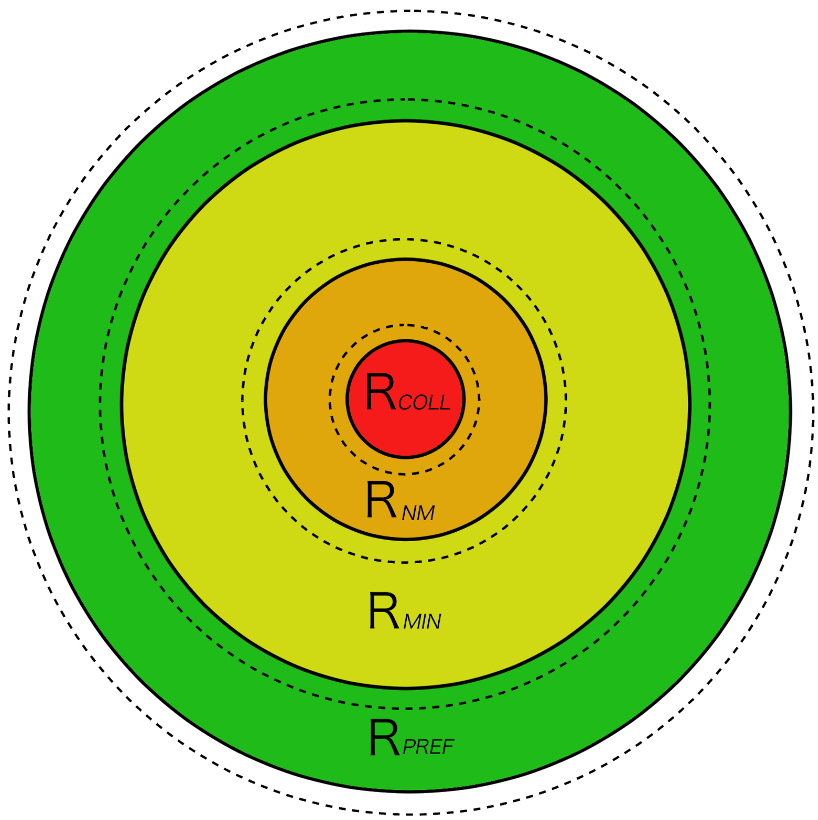


Figure 4.20 – Dynamically expanding safety zones due to inherent uncertainties

4.3 Trajectory generation - predictor

Considering that the proposed solution for collision avoidance relies on algorithmic integration of multiple systems presented in this thesis, and most of the constraints for effective maneuvering are defined by various reward functions, a predictor is required that will efficiently generate trajectories of own vehicle and targets. In majority of the real-world cases, there are readily available target information from RADAR and ECDIS stations, so it is possible to reduce computational complexity. However, predictor will always be required for the own vehicle to find optimal trajectories for hazard avoidance.

Having a predictor that can utilize dynamic trajectory replanning in space-time domain allows us to extend online model-free HMM framework to collision avoidance. Finding course alterations that would resolve conflicts is of interest, but the benefit of this methodology is that it is possible to increase dimensions of state, action, and observation spaces. Therefore, speed alterations could be utilized as well. However, it is necessary to keep space dimensions low in order to ensure feasibility of the collision avoidance system in the real world. Various techniques are utilized to increase observability and decrease number of parameters required to describe a state to ensure computational efficiency. Markovian solutions are applicable as many parts of the collision avoidance problem can be solved geometrically utilizing readily available solutions onboard commercial sea surface vehicles.

Figure 4.21 depicts process of online policy modification through trajectory replanning. Instead of offline learning and developing policies that cover all possibilities, sensing data and dynamic trajectory replanning in 0.1 Hz timestamps is utilized, which is consistent with previous algorithmic solutions. For each 10 seconds one waypoint is generated and connected linearly to form an evasive sailing trajectory.

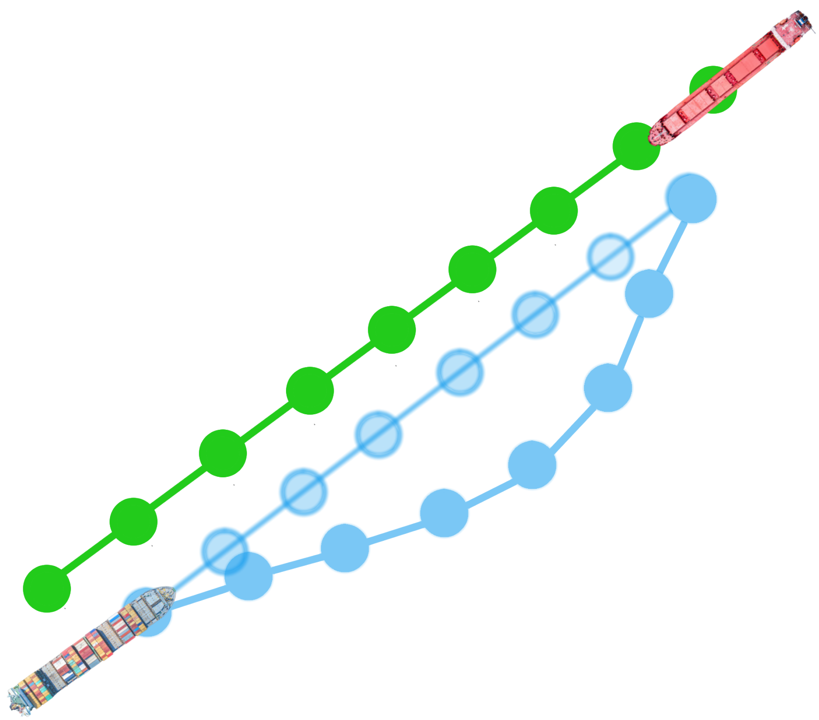


Figure 4.21 – Online policy modification and trajectory generation

In the case of underactuated vehicles, state and observation spaces are continuous, while action space is usually discrete. Even though the action space is usually discrete, it is still within high dimensional spaces, so it is important to carefully select representatives of state and observation spaces. This was done by developing sets of motion control algorithms in the previous chapters of this thesis. Similarly, issue of sensor imperfections and increased observability is addressed through sensor fusion and sensing data filtering. Another challenge addressed was target detection and intention uncertainties in order to have a sufficiently large safety buffer for predictions, especially for the predictions made further in the future. Built on the premises of kinodynamic planning, known motion constraints are utilized to ensure feasibility of the nonholonomic motion control for trajectory generation. This approach mandates that all known constraints are applied when generating trajectories so that only achievable space-time waypoints are generated. For example, if we know that there is a constraint of turning to port for head-on situations, then we can impose that constraint to a trajectory generator and allow only to search for solutions to the starboard side of the own vehicle. If ownship had knowledge of a target vehicle, performance constraints to targets' trajectories could be implemented, which is exactly what will be covered in the next chapter.

When generating future trajectories, implementation of incremental waypoint uncertainties is avoided by utilizing sensor fusion and filtering. This allows for improved observability and confidence of future predictions. If waypoint uncertainties were incrementally increased, it would be difficult to determine own vehicle and target attitudes at predicted position and time of conflict. With proposed approach it is possible to also estimate in which zone of the safety radius would a target be in the time of conflict. During exploitation, algorithm could track performance with various targets and utilizing information that is available through AIS it would enforce classification of targets and store risk data. Depending on the risk classification, targets would get different safety radii, smaller for the lower risk and larger for the higher risk vehicles. In that way, larger separation at the predicted point of conflict would be ensured.

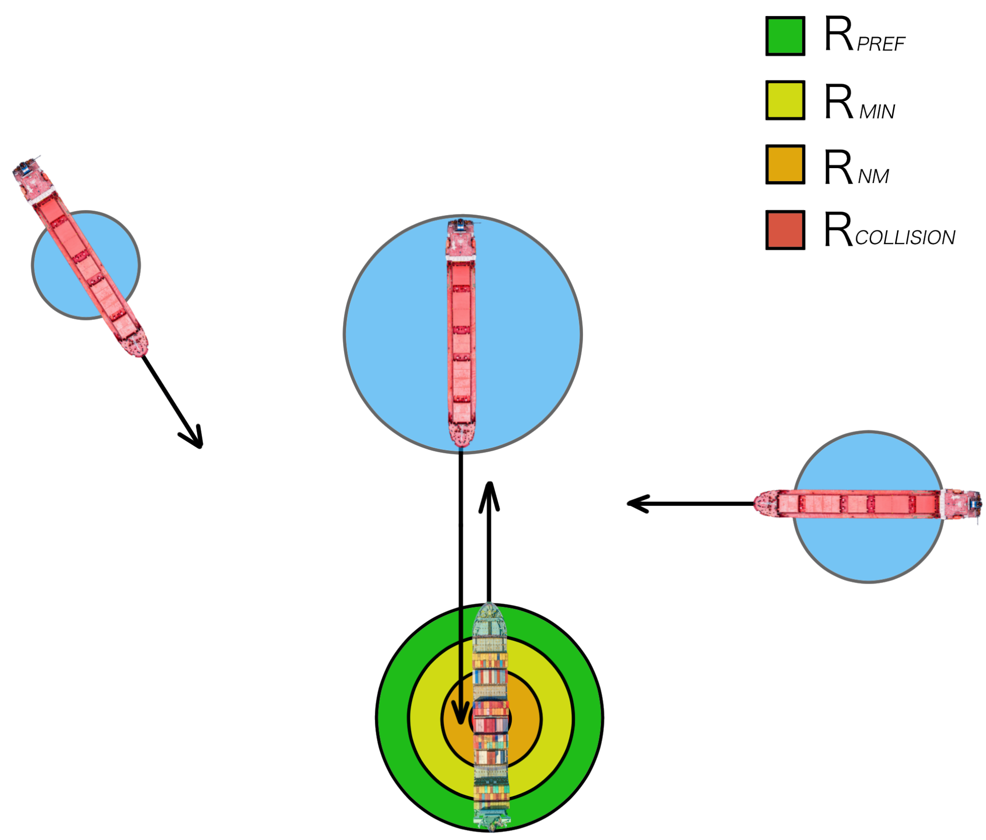


Figure 4.22 – Variable safety radii depending on risk classification

In order to ensure collision avoidance algorithm accurately generates trajectories, input information has to be extensive. Instead of raw sensor readings, filtered and fused information are utilized as observations. FPF filters are also used to aid in state estimation. Focus is maintained on limiting elements of a state space to reduce computational complexity; therefore, position, velocity, and attitude (important for the COLREGs classification) are important. State estimation can be done for own vehicle and targets, but as a minimum, estimation for the own vehicle is necessity. Target estimation can be taken as an input from already existing equipment on commercial sea surface vehicles. However, various manufacturers have different standards and accuracy levels, which needs to be kept in mind when determining risk levels of state estimation. State estimation is represented by a waypoint that contains information about position, speed and attitude of all targets, but for the own vehicle more than regular information of the sea surface vehicle state vector is shared. Mostly, this will be control command vector that describes action taken at that time (steering input and engine power for underactuated vehicles). Finally, as a sequence of predicted waypoints, the predictor generates a trajectory with a predefined number of waypoints. In this case, the horizon is limited to the next 30 minutes. In addition to the own planned route represented as trajectory $\mathbb{t}^r$, the predictor generates optimal trajectory $\mathbb{t}^o$. The set of all trajectories is denoted as $\mathbb{T}$, while set of all viable trajectories is written $\mathbb{T}^v$, where $\mathbb{T}^v \subseteq \mathbb{T}$. Viable trajectories represent all solutions that a navigator could use to ensure safe separation, regardless if they are considered suboptimal.

Collective reward functions are used to optimize trajectories. Reward functions for maneuvering, COLREGs classification, collision avoidance and auto-telegraph are used collectively to find trajectories without conflicts. Constraints are also embedded in observations; such are shallow water positions or position of other hazards directly extracted from ECDIS. Collective reward functions contain maneuvering, deviation and collision constraints that shape optimal trajectory selection.

State estimation for target vehicles is based on utilizing fused sensor information and filtering techniques in order to increase observability and reduce uncertainties. The process of reading RADAR, AIS and ECDIS information is followed by FPF filtering after which a risk factor is assigned to a target state estimation. In case that same target, or similar type of target has been used in the past, assigning risk will be easier. If there are no past information about a target, then risk classification technique based on quality of sensor information and inherent intent uncertainties will determine if a target is considered low risk, medium risk or high risk. Higher risk targets will have safety zones with larger radii. For the target uncertainties, process as described in Table 4.6 is used. In case of a theoretic perfect sensor, safety radius remains the same. If the sensor information is fused as a combination of various sources and filtered by an FPF filter, low risk is assigned and increase of the safety radius is 5 %. In case that fused sensor information is received with inherent errors and there is no filtering, or if there is filtering but only from limited sensor sources, the risk is deemed to be medium and safety radius increased by 20 %. If sensor information is received from one source only and without any filtering, this information is considered unreliable and high risk is assigned to the state estimation, so that the safety radius is increased by 50 %. Therefore, a function $E_T : O \times \mathrm{R} \rightarrow \mathrm{S}$ estimates the state of a target vehicle (T) and determines risk factor. O is a set of available sensor information, R represents set of risk factors, while S denotes set of estimated target vehicle states.

Trajectory generation for target vehicles employs state estimation for a target vehicle to generate predicted trajectory $\mathfrak{t}$ for that target. The function $P_{\mathfrak{t}} : E_T \rightarrow \mathbb{T}_T$ is bounded by the time horizon and number of waypoints. In most cases time horizon of 30 minutes is used, and considering that sampling frequency is 0.033 Hz, maximum number of waypoints is limited to 60.

Trajectory generation for own vehicle is the critical part of the predictor, as the performance and accuracy of the collision avoidance algorithm depends on valid state estimations and waypoint generation. Trajectories of target vehicles are generated first and then function $W : \mathbb{T}_T \times O_H \rightarrow \mathbb{T}_{OWN}$ is used to generate trajectories for the own vehicle. This is because it is necessary to utilize target trajectories $\mathbb{T}_T$ and hazard information O_H to determine safe waters and optimal trajectory for that situation. The process initializes with function W selecting either a random initial own vehicle trajectory $\mathfrak{t}_R$, or planned route $\mathfrak{t}_P$ for the own vehicle and retrieving trajectories of target vehicles to compute optimal trajectory for the own vehicle $\mathfrak{t}_{OWN}$. Optimal trajectory is computed by maximizing collective reward functions. Function W is also used to determine all trajectories that satisfy specific reward functions and are deemed safe for navigation in order to get the safe sector for a navigator to navigate if for any reason he/she decides to sail on a different trajectory then the optimal trajectory.

In order for function W to optimize own vehicle trajectory, it is necessary to have planned route in the form of nominal trajectory and nominal waypoints that correspond to time stamps of waypoints generated for target vehicles. Waypoints are generated with time difference of 30 seconds; therefore, maximal number of waypoints is 60 $(1 \leq n < 60)$:

- If risk of collision is determined by COLREGs classification algorithm then continue with other steps, otherwise no action is required, and own vehicle can proceed along the planned route. This reduces computational burden of checking every target in the surveyed area.
- Compute target trajectory by adding waypoints in the direction a target vehicle is heading (COG) with 30 seconds increments. SOG determines the distance between waypoints. In case target vehicle is turning, predictor records the ROT at the time of prediction and changes direction of travel by the ROT amount. In this way it is possible to get a trajectory of a target vehicle and target's attitude is transferred to the last waypoint. In case of coordinated collision avoidance where target vehicle is equipped as well, then more accurate trajectory can be extracted from a target vehicle by sharing the planned route with own vehicle.
- Extract from COLREGs Classification Algorithm applicable rules for the collision situation with a target vehicle and determine initial course alteration ($\pm$ 35°). Once initial course alteration is selected, generate own vehicle trajectory $\mathbb{t}_{OWN}$ in the direction of the altered course using same principle as for target vehicles.
- Iterate over $\mathbb{t}_{OWN}$ for the own vehicle by performing following steps. Iteration is to be performed on each waypoint of the predicted trajectory $w_n \in \mathbb{t}_{OWN}$.
- Calculate CPA_T for each target vehicle at each waypoint using process explained further in the text. Subtract risk radius for target vehicle and minimum radius of own vehicle (expressed in Nautical Miles) to the

calculated closest point of approach to get $CPA_{T_R} = CPA_T - r_{R(T)} - R_{MIN}$. In this way calculated CPA is reduced for the risk factored in safety radius of own vehicle and each target vehicles.

- Compare calculated CPA with the desired CPA (set with the safety zone for the own vehicle). If $CPA_{T_R} \leq R_{MIN}$, terminate iteration, add or subtract 1° to/from the course of the previous iteration, depending if the initial turn was to starboard or to port respectively. If $CPA_{T_R} > R_{MIN}$, set w_{n+1} as the current waypoint and proceed with the iteration.
- After passing the waypoint with lowest CPA, alter course to go back to the planned route (the same amount the course was altered at the beginning of the trajectory) by ensuring that target remains at the R_{SAFE} zone all the time.
- Numerically compute collective rewards. If no CPA violations and reward functions score is above selected threshold, nominate generated trajectory as optimal and terminate the optimization. Else, store results to compare results of the generated trajectory with other suboptimal trajectories.
- Compare all suboptimal trajectories and select the trajectory with the highest collective rewards score. If none of generated trajectories manages to find acceptable solution, allow for trajectory with CPA in the near miss zone. If this is not achievable, request change in speed from the operator (increase or decrease depending on the trajectory with the highest rewards score).

Considering that presented collision avoidance is done at low ranges, field of view is maintained up to 24 NM, which allows us to use simpler transformations of 2D geodetic coordinates (φ, λ) into corresponding Cartesian coordinates (x, y) by utilizing WGS 84 constants when approximating sphere to an ellipsoid. As depicted by Borčić [1955], following equations are used to transfer geodetic to Cartesian coordinates:

$$x = \frac{a_{elip} \cos\varphi \cos\lambda}{(1 - e^2 \sin^2 \varphi)^{1/2}}$$

$$y = \frac{a_{elip} \cos\varphi \sin\lambda}{(1 - e^2 \sin^2 \varphi)^{1/2}}$$

$$z = \frac{a_{elip}(1-e^2)\sin\varphi}{(1-e^2\sin^2\varphi)^{1/2}} \quad , \tag{4.3}$$

where x, y, and z are Cartesian coordinate axes abscissa, ordinate, and applicate respectively. a_{elip} represents semi-major axis of the Earth reference ellipsoid, while e^2 is the first eccentricity squared. For the WGS-84 Geodetic System ellipsoid constant values are $a_{elip} = 6\ 378\ 137$ m, while $e^2 = 6.694\ 379\ 990\ 14 \times 10^{-3}$ [Borčić, 1955].

It is then possible to propagate waypoints with the same timestamp for own vehicle and targets in the vicinity. In that way it is possible to get a straightforward time of closest point of approach, while the CPA is calculated with Euclidian distance formula based on Pythagoras's theorem. Therefore, if we had two waypoints $w_1(x_1, y_1)$ and $w_2(x_2, y_2)$, CPA formula would be:

$$CPA = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{4.4}$$

This simpler approach can be used because interest lies in closest distance in every waypoint that is generated at the trajectory and because waypoints between own vehicle and target vehicles are at the same timestamp. To avoid differentiation between previous and current position to get the speed value, SOG and COG values could be utilized with ROT direction to propagate future waypoints in position space.

Trajectories are generated by separating waypoints, which entails a problem of geometrically connecting those waypoints. However, as the separation is only 30 seconds, generalization that waypoints are connected with a straight line can be accepted without loss of performance or accuracy. In this way trajectories are optimized for collision avoidance by avoiding safety zones around own and target vessels. In order to avoid local maxima, allowance is made for optimization algorithm to have a certain amount of exploration; however, as reward functions contain rewards and penalties to stay close to the planned route, optimization of trajectories that would take own vehicle to directions not feasible for the intended voyage is prevented.

In order to avoid high computational costs, a solution where minimal number of members are used to describe a state space of each participating vehicle is proposed. Decision was made to utilize sensor readings rather than to make multiple calculations to generate trajectories. It is necessary also to stay away from discretization in the position space within the HMM framework, but rather do all modifications in the control space of the own vehicle with the addition of the safety zone around target vehicles. Another reason why trajectories are generated in control space is that it would be difficult to guarantee feasibility of generated trajectories if waypoints were constructed in position space, as it would require additional steps to ensure nonholonomic vehicle motion control.

As the proposed MCAS system is designed to be dynamic, proposed collision avoidance algorithm would be receiving observations consistently with the rate of 1 Hz, which entails dynamic replanning of generated trajectories. Considering complexity of the collision avoidance system, especially in situations with larger number of targets, proposed algorithms were designed to perform within the boundary of the sample rate, even though solution is required every 30 seconds (before replanning of generated trajectories occur). The collision avoidance algorithm has been optimizing within 0.35 seconds timeframe for a single target; however, when the full MCAS system is working in parallel, optimization time is increased to 0.84 seconds. In the case when there are multiple targets, this time is further increased with an average of 14.8 seconds, but maximum of 23.8 seconds. These results are for a single computer doing all optimizations in a series and are still within the real-time computation boundaries; however, if parallel computing available on modern navigation bridges was utilized, and as long as one pairwise trajectory generation is possible in real-time, there can be any number of targets on the horizon. This is the reason why proposed system leads to decentralization of algorithms and processing power. In that sense, look-up tables allow us to maintain some of the logic computation offline, while keeping the collision avoidance online. There is no reason that one computer runs both FPF filtration of the sensor data, vehicle motion control algorithms, auto-telegraph, COLREGs classification algorithm, collision avoidance algorithm and cooperation algorithm at the same time, when it can be assigned to other available computers. The system could be designed in a different way as a central processing unit, but this would lead to high computational cost that could result in solution generation that is lower than 1Hz. This, again, would not be a large issue for the motion control and collision avoidance of underactuated ocean-going vehicles, but would require different processing solutions for situations where more precise position control has to be conducted in narrow and busy waterways.

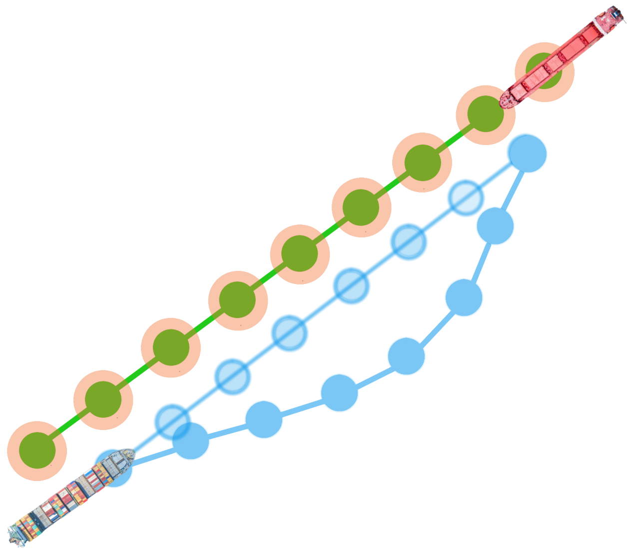


Figure 4.23 – Assigning safety zone circles on trajectory waypoints of target vehicle due to inherent uncertainties

In Figure 4.24, it is possible to see an example how trajectory generation populates future waypoints. The process starts with an initial waypoint that is described as a belief state (taking all sensor uncertainties into account). The process continues by selecting one of the actions available in the action space (action space is bounded by maneuvering limitations and reward function) after which observation is utilized to generate the next waypoint. This process is continued until the last waypoint of the trajectory is reached. Sampling time of 30 seconds and time period of 15 minutes is selected, so the final waypoint will be waypoint 30.

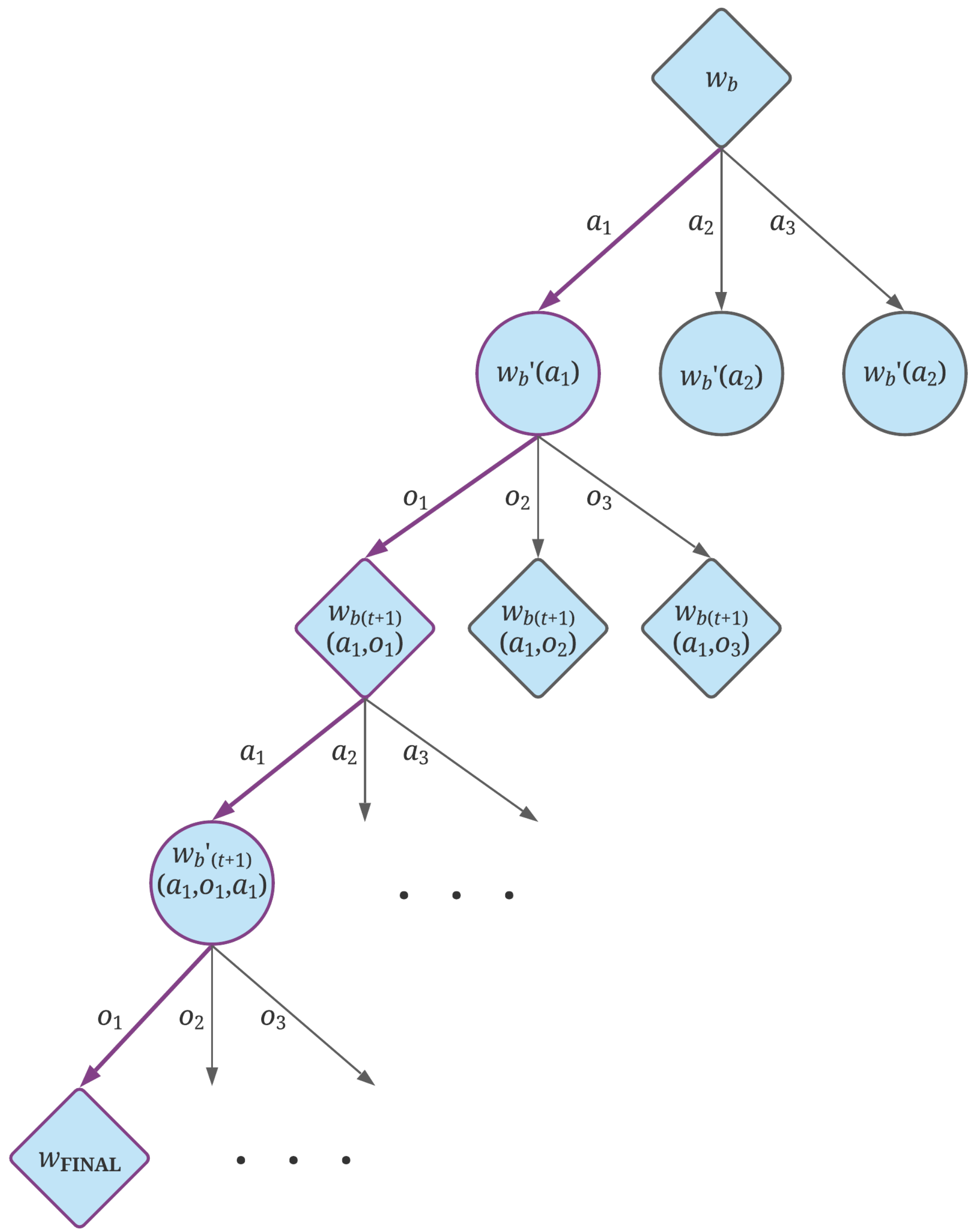


Figure 4.24 – Trajectory generation utilizing belief state expansion

4.4 Collision avoidance algorithm

The central focus of this section is to define parameters and determine reward function of the overall collision avoidance process, as well as to present results of numerical simulations of scenarios exhibited in the Section 4.1 of this chapter.

The proposed collision avoidance algorithm is highly interconnected with other algorithms developed in this thesis. Some of outputs from other algorithms are used as input data for this algorithm. For example, the COLREGs classification algorithm will verify if the collision risk exists and which rules are applicable for that situation. This information is used as an input guidance for the collision avoidance algorithm to select appropriate reward functions and, in cooperation with the predictor, finds the best action to take and optimal trajectory.

Other algorithms in this thesis cover many aspects of the collision avoidance, so the main goal with the collision avoidance algorithm is to utilize maximized expected and discounted return on rewards for the potential trajectories. The algebraic vector predictor is verifying potential trajectories and summing rewards for each, finding the optimal trajectory. It is done holistically taking into consideration the whole traffic situation; however, multiple targets are covered in the next chapter of the thesis. This is necessary, as there are different approaches to multiple target situations depending on the computational burden and intent sharing possibilities. Therefore, collision avoidance algorithm is presented as egocentric resolution that solves resolution for one target only at a time. The algorithm can still handle multiple targets, but it will prioritize targets with higher risk and resolve firstly those targets and then continue with other targets on a later stage. This can lead to successful resolutions but not selecting optimal trajectories as if all targets in the observable area would be taken into account.

Collision avoidance algorithm has to be tuned for a specific sea surface vehicle. As COLREGs distinguish between type of vehicles and there are differences among maneuvering characteristics of vehicles, specific tuning of parameters is necessary for the sustainable and robust application of the algorithm. Tuning is required only for the ownship and there are no limitations for the target vehicles it would meet. In this thesis ownship

LNG vehicle is utilized, but tuning parameters for a different vehicle type is a fairly simple process.

Rewards design and shaping remain the key element for optimizing trajectories. Therefore, it is required to define rewards in a similar way as with previous algorithms. Rewards are defined depending on the situation and COLREGs Rule that is applicable to that situation. As motion control algorithm is in charge of setting up an autopilot control, we are only interested in COG and SOG information within this predictor. Trajectories are computed keeping in mind only the over-ground values and then this information is fed to the motion control algorithm in order to compute what heading is required for the computed COG.

Collision avoidance algorithm has a preferential approach when searching for optimal solution. It will firstly try to search the most common space for an optimal trajectory. For example, in a head-on situation, it will search for an optimal trajectory by turning 35º to starboard and if this COG would make sufficient alteration to guarantee required CPA, then this would become an optimal trajectory to take. If this COG would not provide optimal solution, the algorithm would then search further to starboard up to 90º from the original course. If there is no optimal solution within this space, then algorithm reduces safety radii and tries to find satisficing solution from 35º all the way up to 180º. To increase efficiency, algorithm first searches for the COG in which the largest CPA is possible during the previous step (searching for optimal solution up to 90º). If there is no available solution even in this space, the algorithm searches for solution with reduced speed taking into account how long it would take to reduce speed. This is done by determining how much reduction of speed is possible within the TCPA time and then algorithm searches for the reduced speed optimal or satisficing solution. If no solution exists in this space, turning to port is checked. If no solution is available at all and collision is inevitable, crash maneuver is selected by the system.

The trajectory is generated by keeping track of a target and by keeping it within the desired safety radius zone. When initial course change is found (for example it is +35º from the original course), then ownship alters back to the parallel course as on the route when earliest possible after the course change (determined by predictor when a target can be safely within the selected safety radius). Then after a target has passed abeam of ownship, trajectory towards back to the route (smoothness can be applied so that it is not done aggressively), or towards the next waypoint and slowly going back to the track. As route verification is important part of passage planning, it would be preferred that the ownship goes back to the track limits, as that is the part that was verified by ECDIS prior the passage.

Figure 4.25 shows an example head-on encounter in which the optimal (green) and satisficing (yellow) areas are considered. The definition of optimal and satisficing is related to the safety zone ownship selected and if it is possible to stay out of the minimal CPA safety zone. It is noticeable that the green area extends only up to perpendicular to the route of the ownship. This is on purpose, as we do not envision requirement to sail out of the green zone in this situation. However, if optimal solution could not be found, the search for optimality continues further and can even be reciprocal to the current heading. In practice, navigators would like to follow the first green heading available, as this means that deviation from the original course would be as small as possible.

Rule 6. The Rule 6 is governed by the COLREGs Classification Algorithm. As decision support system is designed, allowance for violation of safe speed requirement has to be agreed, so if a human navigator ignores Safe Speed warnings, proposed collision avoidance algorithm still has to do predictions in accordance with the current speed a vehicle is doing. The predictor can read information about intent, so if a human navigator decides to follow the advisory from the COLREGs Classification Algorithm, then intent can be fed to the vector predictor and taken into account for more accurate predictions. Therefore, collision avoidance algorithm will not be specifically tracking the safe speed requirements and there are no reward functions outside the ones defined in the COLREGs Classification Algorithm. Similarly, Rule 7 is governed by the COLREGs Classification Algorithm and then the information is fed as a binary value to the collision avoidance algorithm. The risk of collision either exist or it doesn't, and in accordance with that information, collision avoidance algorithm decides if action is needed or not.

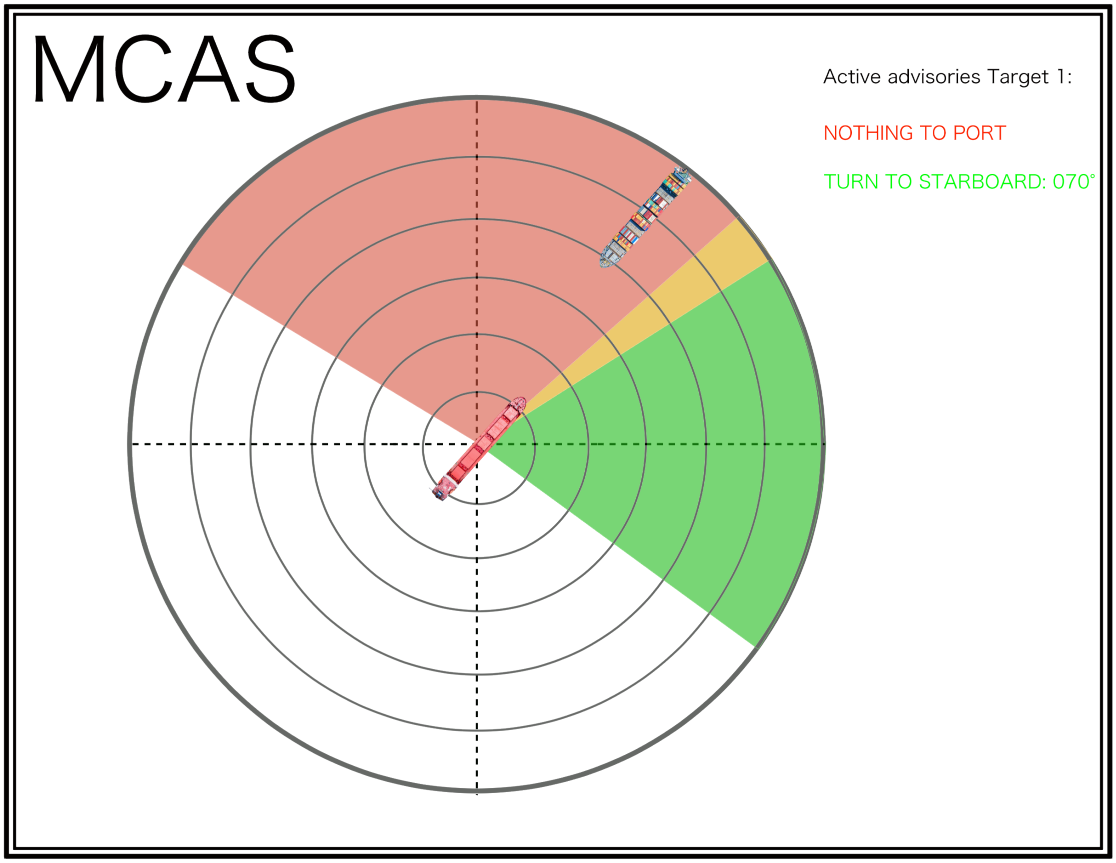


Figure 4.25 – MCAS individual screen with advisories

Rule 8. As described in the COLREGs classification section of this thesis, Rule 8 describes requirement of an early and significant action so that all participants of that traffic situation are clear of an intent. It is not easy to quantify this requirement, but the case law showed that from the point of Admiralty courts, minimum of 35º course change is expected in order to show clear intent, given that there is enough space to do such a maneuver. That does not mean that ownship has to keep that course until the collision risk is clear, but only to show intent. That is why proposed algorithm is designed to firstly check possibility of altering 35º from the original course and then track the CPA from a target vehicle when deciding on a trajectory. Except in a case of overtaking, preferred action would be turning to starboard, so the algorithm will determine if there is a possibility of turning to port. There will always exist a small possibility of turning to port as COLREGs require to avoid collision even if it takes to break some of the Rules. As described above, preference system is engaged and search for optimal or satisficing solution would be in progress.

$$R_{8A}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad COG_P = COG_A \pm 35° \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

$$R_{8B}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad CPA < \mathrm{R}_{PREF/SAFE/\,MIN} \\ -10, & \text{otherwise} \end{cases} \tag{4.6}$$

$$R_{8C}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad CPA \geq \mathrm{R}_{SAFE/MIN/NM} \\ -10, & \text{otherwise} \end{cases} \tag{4.7}$$

$$R_{8D}(s_i, a_j) = \begin{cases} 50, & \text{if} \quad d_{TRACK_{PLANNED}} = d_{TRACK_{ACTUAL}} \\ -1, & \text{otherwise} \end{cases} \tag{4.8}$$

$$R_{8E}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad CPA_{SHALLOW} \geq \mathrm{R}_{SAFE/MIN/NM} \\ -1000, & \text{if} \quad CPA_{SHALLOW} < \mathrm{R}_{NM} \\ -50, & \text{otherwise} \end{cases} \tag{4.9}$$

Rule 8 is applicable to all situations where risk of collision exists, but it will be applied in a different way depending on the situation with a particular target. With the expression (4.5) the Admiralty courts practice to consider 35º alteration of courses as sufficient is rewarded. Depending on a traffic situation, this expression will allow alteration to port, but in the most cases it will be to starboard. There are no penalties for altering smaller than 35º just to make sure that the agent searches for satisficing solutions as well. Expression (4.6) is used to reward agent when finding trajectories that will not take an ownship too far away from a target, while expression (4.7) rewards agent to select trajectories that are not too close to a target. Both expressions (4.6) and (4.7) are tunable to a desired safety radius. With the expression (4.8) vehicles that stay on the track are rewarded, so even when trajectories are selected and path developed, agent wishes to go back to the track as soon as it is practically possible. Special tuning is required depending on the maneuvering characteristics of a vehicle. The expression (4.9) is related to shallow water and it rewards agent that stay away from a shallow water by the preselected distance and penalize heavily for getting closer than minimum safe radius (entering in a near miss radius). This is of great importance to prevent incidents like Wakashio.

Rule 9. From the perspective of collision avoidance, requirement to keep as near to the outer limit of a narrow channel is in focus. As proposed model is decision support model, it is important that a human operator makes appropriate passage plan and ensures that the planned passage is on the outer limit of a narrow channel. For the autonomous navigation, reward function would be shaped to prefer sailing on the outer limit of a narrow channel. Ownship is a large LNG vehicle, so it is not necessary to encode the requirement for vehicles less of 20 meters in length, sailing vehicles or fishing vehicles. Another important part is overtaking, and it is necessary to ensure that overtaking in narrow channels is done only if it is safe to do so (vehicles would be keeping sufficient distance) and that overtaking can be taken before the significant alteration of course when there are multiple branches of channels and there is possibility that overtaking vessel would turn to another direction. If the intent is shared among vehicles, then this information can be used as an input to the collision avoidance algorithm in order to improve state estimation. Sound and light signals are deterministic and easily implementable, so they are not a part of the collision avoidance algorithm.

$$R_{9A}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad d_{COURSE} = \pm\, 0.3 \text{ NM} \\ -10, & \text{otherwise} \end{cases} \tag{4.10}$$

$$R_{9B}(s_i, a_j) = \begin{cases} 1, & \text{if} \quad RPM \leq NAV\ FULL \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

$$R_{9C}(s_i, a_j) = \begin{cases} 20, & \text{if} \quad CPA_T \geq \mathrm{R}_{MIN/NM} \\ -10, & \text{otherwise} \end{cases} \tag{4.12}$$

$$R_{9D}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad TCP_{WP} > TCP_{TARGET} + 12 \text{ min and } 112.5° \leq \theta_T \leq 247.5° \\ -10, & \text{otherwise} \end{cases} \tag{4.13}$$

The expression (4.10) rewards vehicles that stay within 3 cables of the planned course line. This is important in narrow channels and it is one of the ways to maintain vehicle closer to the outer limits of a channel. Even though proposed COLREGs Classification Algorithm is rewarding safe speed in narrow channels, additional layer with expression (4.11) is introduced. The reward is small and there is no penalty, so it has a small but distinctive influence when searching for the optimal trajectory. The expression (4.11) is only applicable when overtaking targets, so the Rule 13 will be active and allow for (4.11) to be included in rewards summation. This expression rewards vehicles that would overtake other vehicles only if they can keep them within desired safety radius. In this case minimum and near-miss radii are stated, as the narrow channels usually don't have much space for overtaking. Also, there is a penalty for overtaking a vehicle with smaller CPAs. If overtaking can't be done with appropriate space, human navigator would get a warning that the overtaking is unsafe. Similarly, the expression (4.13) is related to overtaking and will be active only when COLREGs Classification Algorithm activates Rule 13. However, the expression (4.13) is only applicable to unequipped vehicles, so if the intent sharing is possible, then (4.13) can be disregarded. Otherwise, (4.13) ensures that overtaking can be completed before the next waypoint where change of course will be done. This is an additional safety layer. Only if the predictor can verify that own vehicle will reach the next waypoint with 12 miles TCPA from the overtaken vessel and that the overtaken vessel is abaft the beam of own vehicle will the overtaking be supported. Otherwise, human navigator will get a warning that the overtaking is unsafe and can't be completed before the next turn. This requirement can also be tuned by defining how big of a turn it has to be to activate this requirement, or to verify if there are any branches of the narrow channel that other vehicle could be taking while ownship is overtaking it.

Rule 10. Rule 10 is somewhat similar to Rule 9 with reward design. However, additional requirements are laid out in the Rule 10 for which some additional reward functions are required. It is necessary to ensure that a vehicle is following a traffic flow of a traffic lane, that it keeps clear form the separation zone, that it joins the traffic lane with small angle to the traffic flow and when crossing that it is done on a heading that is as close as 90° to the traffic flow. Without loss of generality, reward space is simplified by not introducing any limitation for the inshore zone and this is because other rewards will penalize leaving traffic lanes and ownship would then use inshore zone only for the collision avoidance.

$$R_{10A}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad d_{COURSE} = \pm\, 0.3 \text{ NM} \\ 0, & \text{otherwise} \end{cases} \tag{4.14}$$

$$R_{10B}(s_i, a_j) = \begin{cases} 5, & \text{if} \quad d_T < 3\text{NM and } CPA_T \leq \mathrm{R}_{SAFE} \text{ and } RPM \leq NAV\ FULL \\ 0, & \text{otherwise} \end{cases} \tag{4.15}$$

$$R_{10C}(s_i, a_j) = \begin{cases} -10, & \text{if} \quad n_F, e_F = n_{SEP\ ZONE}, e_{SEP\ ZONE} \\ 0, & \text{otherwise} \end{cases} \tag{4.16}$$

$$R_{10D}(s_i, a_j) = \begin{cases} 20, & \text{if} \quad CPA_T \geq \mathrm{R}_{MIN/NM} \\ -10, & \text{otherwise} \end{cases} \tag{4.17}$$

$$R_{10E}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad TCP_{WP} > TCP_{TARGET} + 12 \text{ min and } 112.5° \leq \theta_T \leq 247.5° \\ -10, & \text{otherwise} \end{cases} \tag{4.18}$$

Even though Rule 10 has very similar expressions as Rule 9, there are some very distinctive requirements for the navigation within TSS, therefore reward function is slightly different. The expression (4.14) rewards a vehicle to stay within 3 cables of the planned course, but because Traffic Separation Schemes can sometimes be substantially wide, no penalty will occur if a vehicle is further away, but it is rewarded to be closer. As the TSS can be wide, deep and long enough to sustain full speed navigation, insisting on safe speed is not always optimal. Therefore, safe speed is rewarded only when there is a target within 3 NM of own vehicle, and when the CPA for that target is closer than safe radius. The expression (4.16) is penalizing if own vehicle is in separation zone. However, large penalty was not applied, as there are cases when vehicle would have to enter separation zone to avoid collision. Expressions (4.17) and (4.18) are identical to expressions used in Rule 9, so they are only used for overtaking within TSS zone. If Rule 13 is not active, these rewards would not be considered.

Rule 13. The main input information for designing a reward function for overtaking is active Rule 13 signal from the COLREGs Classification Algorithm. Therefore, it is not necessary to track the pose of ownship and target vehicles; this is always checked by the COLREGs Classification Algorithm.

$$R_{13A}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad d_{COURSE} = \pm\, 0.3\; NM \\ -1, & \text{otherwise} \end{cases} \tag{4.19}$$

$$R_{13B}(s_i, a_j) = \begin{cases} 50, & \text{if} \quad CPA_T \geq \mathrm{R}_{MIN/NM} \\ -10, & \text{otherwise} \end{cases} \tag{4.20}$$

$$R_{13C}(s_i, a_j) = \begin{cases} 10, & if \quad COG = COG_{INITIAL} \text{ and } SOG = SOG_{INITIAL} \\ -1, & \text{otherwise} \end{cases} \tag{4.21}$$

Expression (4.19) is similar to reward functions used in other Rules and it rewards own vehicle to stay within 3 cables of planned course. This reward will promote return to a planned track after overtaking of a target vehicle. For an overtaking vehicle, the expression (4.20) is relevant and it rewards overtaking vehicle to maintain sufficient distance for a vehicle being overtaken. Rewards are designed to be sufficiently large to maintain distances appropriately. The final reward for the Rule 13 is defined by the expression (4.21) and it is relevant when an ownship is being overtaken. In this case ownship is rewarded to maintain its course and speed. However, as incidents still could occur during an overtaking action, lower rewards are maintained, and small penalty introduced to deviate from the course and speed in order to avoid any near misses. That is the reason not to exclude expression (4.20) even when ownship is vehicle that is being overtaken, as we want to move from a target that is trying to intrude desired safety radius.

Rule 14. As COLREGs Classification Algorithm determines if the Rule 14 stands in certain situations, application of the Head-On Rule is quite straight forward, so reward space design is simple.

$$R_{14A}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad COG_P = COG_A + 35° \\ 0, & \text{otherwise} \end{cases} \tag{4.22}$$

$$R_{14B}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad ROT \geq 0° \\ -25, & \text{otherwise} \end{cases} \tag{4.23}$$

$$R_{14C}(s_i, a_j) = \begin{cases} 50, & \text{if} \quad CPA_T \geq \mathrm{R}_{MIN} \\ 20, & \text{if} \quad \mathrm{R}_{MIN} > CPA_T \geq \mathrm{R}_{NM} \\ -50, & \text{otherwise} \end{cases} \tag{4.24}$$

With the expression (4.22) turns to starboard are rewarded. Turns to port are not penalized, but rather discouraged. This is because of the COLREGs requirements that vehicle should do whatever necessary to avoid collision. In order to support turns to the starboard side, expression (4.23) is utilized where again turns to starboard are rewarded, while turns to port are penalized in order to emphasize the importance of the Rule 14. Finally, expression (4.24) is utilized to ensure that proper CPA is maintained from target vehicles by enforcing rewards for staying away from the minimum and near miss zones (in case it is not possible to maintain distance from the minimum zone, staying away from the near miss zone still delivers rewards), while it penalizes target vehicles entering near miss zone. Reward functions of other rules ensure that own vehicle returns back to the original route after the head-on situation has been resolved.

Rule 15. When there is a risk of collision and two vehicles are determined to be in a crossing situation, reward function has to ensure that vehicles stay well clear of each other.

$$R_{15A}(s_i, a_j) = \begin{cases} 20, & \text{when } 1 < w_{T_n} < 179°, \quad COG_P = COG_A \pm 35° \\ 10, & \text{when } 181° < w_{T_n} < 359°, \quad COG = COG_{INITIAL} \text{ and } SOG = SOG_{INITIAL} \\ -5, & \text{otherwise} \end{cases} \tag{4.25}$$

$$R_{15B}(s_i, a_j) = \begin{cases} 10, & \text{if} \quad ROT \neq 0°, \ BCR > 0\, NM \\ -5, & \text{otherwise} \end{cases} \tag{4.26}$$

In order to ensure that own vehicle acts according to the Rule 15, expression (4.25) is introduced where in case that the target vehicle is positioned on the starboard side from the own vehicle, collision avoidance maneuver is initialized. Otherwise, own vehicle is a stand-on vehicle, and it maintains its original course and speed. Turning to starboard is always preferred, but it is not strictly enforced, so there is an option of turning to port or to starboard. However, it is known that vehicles should avoid crossing ahead of the target vehicle, so expression (4.26) is used to ensure that Bow Crossing Range (BCR) is above 0 when own vehicle is a give-way vehicle. This will give preference to starboard turns unless collective risk of collision is greater when turning to starboard and remaining options are turning to port or change of speed.

<u>Rule 16.</u> Actions of a give-way vehicle do not require separate reward function, as it is included in other rules and promote ample time and substantial alternation of course.

<u>Rule 17.</u> The main challenge of the Rule 17 is how to ensure that own vehicle takes action if give-way vehicle is not following the Rule 16. Regular requirements of a stand-on vessel have been covered by reward functions of other Rules, but for taking actions in case of an unresponsive target, following reward function is proposed:

$$R_{17A}(s_i, a_j) = \begin{cases} 20, & \text{when } CPA_{T_n} \leq 3\text{ NM, and/or } TCPA_{T_n} \leq 12\text{ min, and } CPA_{T_n} < \mathrm{R}_{MIN,} \\ & \text{then } COG_A - 35° \geq COG_P \geq COG_A + 35° \\ -50, & \text{otherwise} \end{cases} \tag{4.27}$$

$$R_{17B}(s_i, a_j) = \begin{cases} 10, & \text{if } R_{17A}(s_i, a_j) \text{ is active, } CPA_{T_n} \geq \mathrm{R}_{MIN} \\ -5, & \text{otherwise} \end{cases} \tag{4.28}$$

As discussed earlier in the previous subchapter, some of the Rules are vague and approach is made where at any time CPA from a target vehicle is less than 3 NM, and/or TCPA is less than 12 minutes, and target vehicle is not taking required action while occupying collision or near miss zone, own vehicle will commence collision avoidance maneuver. Described requirement is included in the expression (4.27), while (4.28) promotes larger change of course and increased ROT in order to achieve CPA that is safe and close quarter situation avoided.

Other rules are already included in other reward functions, including the restricted visibility Rule 19.

4.5 Are we better off? – Own vehicle equipped

In this subchapter combined system of sensor filtration is deployed, motion control, auto-telegraph, COLREGs classification, trajectory generation and collision avoidance algorithms. Scenarios two, five, and eight were randomly selected to verify feasibility and accuracy of proposed algorithms.

All experiments are conducted on one computer with all algorithms running concurrently. Once one algorithm can provide input information, that information is fed to the next one until the optimal evading trajectory is selected and advisories displayed. As mentioned earlier, sampling period for the collision avoidance and trajectory generation is 30 seconds. In all examples proposed algorithm was able to generate trajectories within the sampling time. However, as different sampling periods are used with other algorithms, such are FPF and motion control, there is a significant benefit of conducting these processes on separate machines.

Figure 4.26 depicts the still frame of the simulation for the scenario 2. At the moment of capturing this geometries, collaborative algorithms were already in action and all necessary information from the COLREGs classification algorithm has been received. From the CPA, TCPA and dCPA information taken from sensors, four targets resulted with collision alerts. Those are target 1, as the TCPA is less than 15 minutes before the target vehicle 1 would occupy COLLISION safety zone. Target vehicle 3 is also active as motion vector (12 min) already occupies NM safety zone and overtaking own vehicle. Vehicle 3 would very soon be considered as NC vehicle, as the distance to the own vehicle is dropping and ownship would then get the advisory to act in order to avoid collision. Vehicle 5 is also active with non-compliance alert, so expedited maneuver is requested from own vehicle. Finally, vehicle 9 is the last vehicle with active collision alert, as it would within 15 minutes occupy NM safety zone. Other vehicles are currently considered as a non-thread, but the relative geometries are compared in each iteration of the collision algorithm check. In case own vehicle would not follow advisories, some other vehicles would become active as well. For example, vehicle 1 would be considered risky in case it would not follow COLREGs and would maintain its trajectory.

Scenario 2

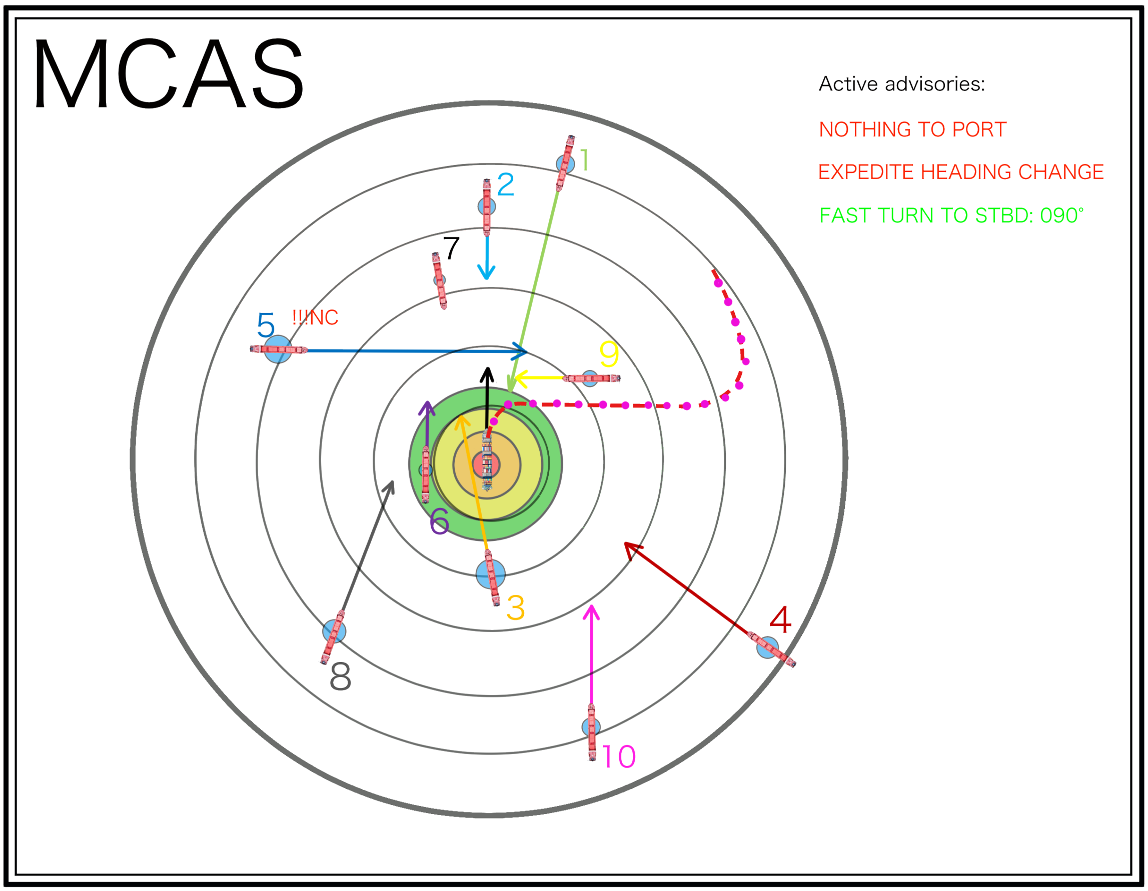


Figure 4.26 – MCAS overview of the optimal trajectory for the Scenario 2

The predictor has selected the significant alteration of course of 90 degrees to starboard, as this approach was found as an optimal one, namely because the target 5 is non-cooperative and did not turn to starboard according to COLREGs. Turning to 090°

allowed for safe interaction with targets 1, 3, 5, and 9 by maximizing the reward function. 100 of simulations were carried out against the same scenario and the results were shown to be stable, without convergence to local maxima. Trajectory successfully cleared all targets and returned back to the planned route. Motion control algorithm was utilized to verify feasibility of maneuvers. 100 simulations were conducted with various external disturbances according to the Table 4.7. Figure 4.26 depicts situation without external disturbances. All simulations performed well, with some vehicles having difficulty to navigate in some iterations; however, the result was consistent with variability of speed and headings, while over-ground trajectories remained within the smallest margin. Feasibility was acceptable, ranging from 9.8 seconds to 24.8 seconds to converge.

Table 4.7 – External disturbances Scenario 2

Simulation	Relative Wind Direction (°)	Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Relative Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1-25	0	0	0	0	0	0	0	0
26-50	112.5	63	090	14	100	1	180	6
51-75	67.5	21	45	6	74	2	67.5	4
76-100	270	33	292.5	2.5	286	2	315	2

With Figure 4.27 relative geometries of own vehicle and targets for the scenario 5 are visible. In this scenario active targets are 1 where, due to own vehicle speed TCPA shows that the contact with the target 1 would be probable. In this instance the fact that navigators should not allow to be in this position with own vehicle is disregarded, but rather use the fact that navigators did not act on time and now decision support system aids the recovery. Target 2 is also considered as active and it is classified as non-compliant as the motion vectors show that without acting immediately, there is a significant risk of collision (especially as predictor takes into account that target 6 is not moving, so the system utilizes NC classification to force own vehicle to act immediately and avoid collision. Target 4 is active as well, which is evident from the relative geometries and speed vectors of own vehicle and target vehicle 4. Not only that there is a collision risk evident from the relative geometry, but also due to the fact that according to COLREGs target 4 should be avoided. The next active target is target vehicle 6. If the speed of own vehicle was lower, there is a possibility that the system would not include this target, but it is necessary to act early in order to avoid complications that would happen soon after this iteration. Finally, target 9 is also active target, even though ownship is overtaking, but within the simulation target 9 has a positive rate of turn, which indicates that the vehicle is turning to starboard and there are safety zone violations in the future waypoints of the generated trajectory.

Scenario 5

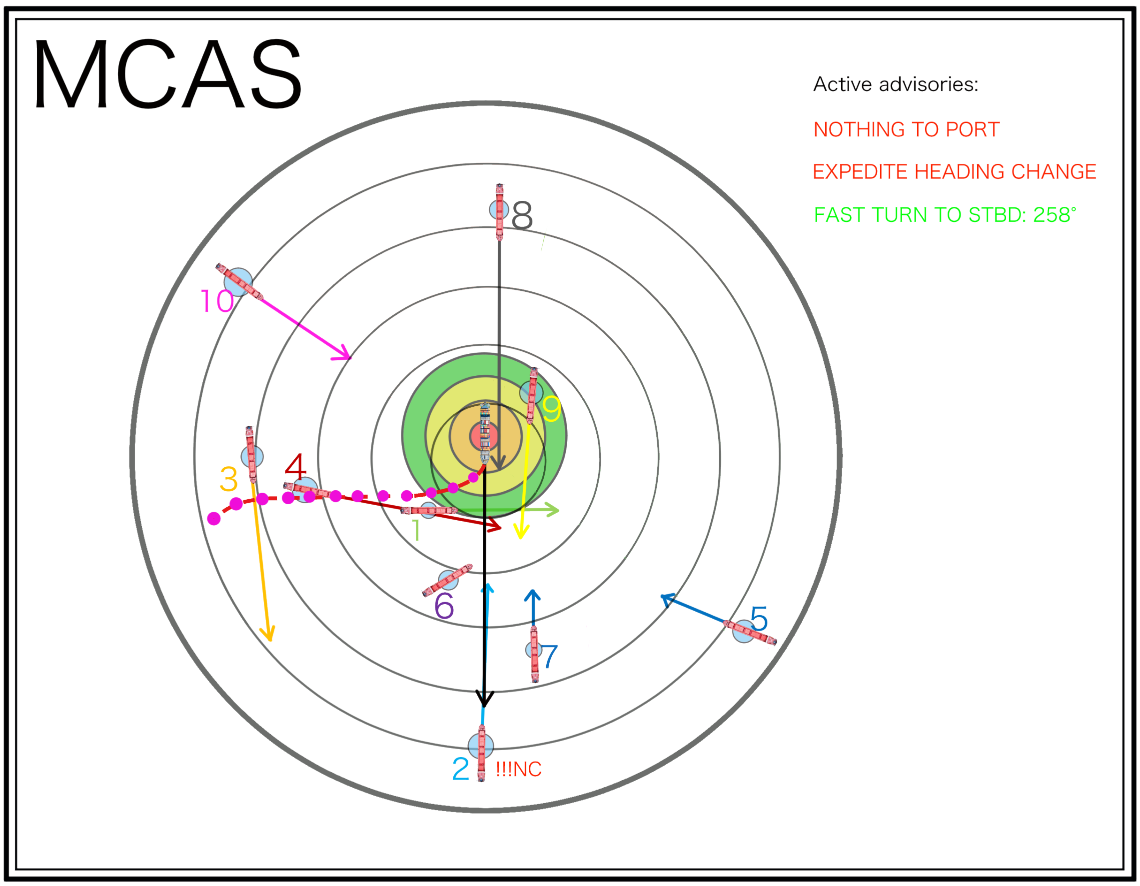


Figure 4.27 – MCAS overview of the optimal trajectory for the Scenario 5

The predictor was able to generate trajectory that is feasible and achievable. The optimal trajectory has been generated by maximizing reward function and trajectory was again successful in averting collision. By continuing simulation, trajectory was bringing us back to the route, but it did take some more time than in the case of scenario 2, as ownship was overtaking target 3 on her starboard side. 100 simulations were conducted again, and results challenged by external disturbances. Results were consistent in all simulations, while collective convergence happened in the range from 9.87 up to 23.3 seconds. Table 4.8 describes external disturbances during simulations.

Table 4.8 – External disturbances Scenario 5

Simulation	Relative Wind Direction (°)	Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Relative Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1-25	0	0	0	0	0	0	0	0
26-50	225	47	247.5	6	263	3	315	6 (period 20 s)
51-75	337.5	33	360	2.5	360	3	292.5	6 (period 20 s)
76-100	62.5	10	67.5	1.25	97	4	22.5	2

Figure 4.28 shows the interacting relationship of own vehicle and target vehicles. In this case, only two targets are recorded, and both targets are active. It is necessary to emphasize that the MCAS screen is now on a different range scale. Each range ring now shows 0.25 NM, instead of 1 NM as shown in other scenarios. There are also other navigational hazards, such are marine buoys and shallow water, that are taken into consideration when generating trajectories. There is very limited action space, which is challenging from the navigational point of view, but beneficial in finding the optimal trajectory as solution space is bounded by observation function. When searching for the optimal solution, predictor has narrow solution space and finds the optimal trajectory faster than in previous scenarios.

Scenario 8

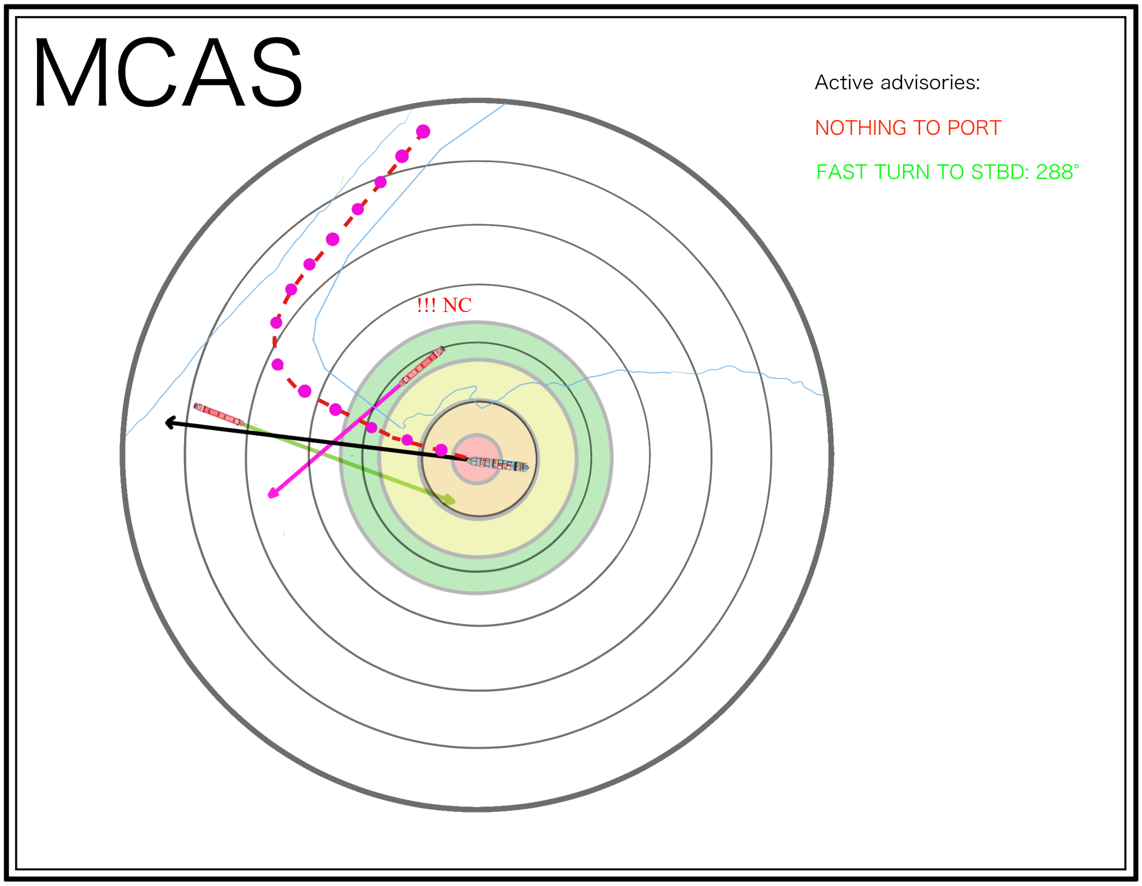


Figure 4.28 – MCAS overview of the optimal trajectory for the Scenario 8

The trajectory generated by the predictor successfully averted collision and brought own vehicle back on the planned route. The optimal trajectory was also feasible and achievable, and it managed to execute the turn as planned utilizing speed of the own vehicle at the beginning of the simulation. Consistent results were achieved after 100 simulations and various external disturbances. Convergence was achieved in the range from 0.63 up to 1.09 seconds. Table 4.9 delivers external disturbances used for the simulated scenario 8.

Table 4.9 – External disturbances Scenario 8

Simulation	Relative Wind Direction (°)	Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Relative Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1-25	0	0	0	0	0	0	0	0
26-50	275	10	297.5	0.5	275	1	297.5	2
51-75	050	21	050	1.25	36	4	320	4
76-100	162.5	10	162.5	1.25	192	4	117.5	2

4.6 Discussion

As a central part of this thesis, Chapter 4 investigated COLREGs compliant collision avoidance model. Initial challenge was implementation of collision regulations into proposed algorithmic approach to resolve close quarter situations. Even though the main task of a collision avoidance system is to resolve target conflicts, if the aim of this study is decision support system, it has to be integrated with all participating entities and has to follow all rules and regulations that are enforced at sea. COLREGs were developed and agreed when integration with sophisticated automated systems was not in focus, so many of the requirements were purposely written vague. This presented a significant challenge for this study, so available resources were utilized to quantify requirements derived from COLREGs in order to shape reward functions. Safety zones around sea surface vehicles was proposed as an additional safety measure to incorporate various sensing, but also intent uncertainties. After assessing individual Rules, COLREGs classification algorithm that is able to provide instant advisories of applicable rules and offer resolution advisories to navigators of own vehicle or equipped targets in the vicinity was proposed. Experimental results showed that COLREGs classification algorithm was able to correctly identify targets and resolution advisories in complex situations where immediate action is required to avoid collision.

Intent uncertainties are consistent among navigators and cause high level of anxiety when navigating international waters. A numeric representation of uncertainties recognized in COLREGs studies conducted with professionals and students was proposed. These quantified representations were used to influence safety zones around target vehicles, which makes safety zone circles dynamic with various radii. The radius of a safety zone directly depends on quality of information navigator has and on intent of target vehicles.

Trajectory generator, or predictor, takes into account situational geometries and utilizes reward function to find optimal solution for a particular collision avoidance case. The crucial part of trajectory generation is carefully designed rewards space in order to prevent own vehicle turning 180 degrees from the planned route in order to "run away" from the problem it faces. Design of rewards space had to be done in a way to equally penalize close-quarters situations and rewards sailing along the planned route.

Collision avoidance algorithm was proposed to resolve collision situations own vehicle would face. In this chapter it was assumed that own vehicle is the only vehicle that is equipped and that trajectories generated for target vehicles (either by the system's generation, or taking input from ARPA), are linear for cases with no ROT recorded, or constant curvature depending on the ROT recorded. All trajectories are updated within the sampling frequency of 30 seconds, which allows ample time to make decisions without losing spatial awareness. Experiments showed that the system coped well with varying number of targets, under different environmental loads, while testing for feasibility during higher level of complexity.

[THIS PAGE INTENTIONALLY LEFT BLANK]

Chapter 5

Dynamic collision avoidance in mixed equipage environments

Chapter 4 introduced a central premise of this thesis by describing the Collision avoidance algorithm where it was assumed that only own vehicle has a decision support system installed onboard. Another concern has to be resolved in this thesis and that is an issue of coordinated collision avoidance in single-threat, multi-threat situations, as well as mixed equipage situations where some of the targets ownship has on its ARPA screen are equipped with decision support systems, while others are unequipped. This chapter offers a solution to ensure coordinated maneuver is done without jeopardizing safety of navigation; however, depending on the number of targets and risk of collision vehicles are forced to a satisficing, rather than an optimal conflict resolution.

Assuming that the proposed Marine Collision Avoidance System (MCAS), as a last-minute collision avoidance aid, is mandatory for sea surface vehicles that navigate in commercial waters, then it is possible to extract benefits of such arrangement directly from the aeronautical domain where similar systems exist for more than two decades and has been shown to significantly improve risk averse behavior and prevents mid-air collisions [Wing et al., 2002].

In situations where two sea surface vehicles meet and risk of collision exists, conflict resolution logic, which includes collision avoidance algorithm, COLREGs classification algorithm, and motion control parameters, would solve the situation for both vehicles utilizing forced cooperation. Many options are available in which vehicle would be a master vehicle in this situation, so this can be simply a higher MMSI number, faster, or slower vehicle, vehicle that is more restricted in maneuverability, or any other priority protocol. Sensor errors and behavioral uncertainties present significant challenges in developing a stable decision support system. In previous chapters of this thesis models that reduce these uncertainties to the lowest possible levels were developed. Only after addressing errors and uncertainties, sensory information are allowed to enter the decision models, so premise stands that the quality of the model output largely depends on the quality of input. Any weakness of the developed model can be identified and rectified in the future, but even the perfect model would not guarantee correct outputs with flawed inputs.

In this chapter, aim is to present challenges and find potential solutions to coordination in mixed equipage environments. One of the main challenges of coordination is to design a protocol where ownship and target will receive advisories that will take into account COLREGs constraints and ensure that there is no maneuver that will increase probability of collision or near miss. Cooperability with other systems already installed on most of the vehicles has to be considered as well. As there is no mandated decision support system installed onboard, it is difficult to offer permanent solutions, but the benefits of having systems similar to the proposed MCAS mandate administrations to initiate communication about finding unified protocols that will allow open platforms where integration with various manufacturers of navigational equipment would be seamless. Multiple targets in restricted waterways is another area of concern which is addressed in this chapter; however, this section begins by introducing an intent-aware collision avoidance model that has a potential of resolving close quarter situations early in the voyage, as well as utilize system’s ability to resolve last-minute conflicts.

5.1 Intent-aware collision avoidance[1]

Reducing uncertainties has been a coalescing objective throughout the thesis. The only sustainable approach to reducing behavioral uncertainties would be cooperation of the sea surface vehicles in the vicinity. Our proposal is that passage plans with dynamic time updates are shared between vessels or to a VTS center as a central point. Sharing between vessels can be achieved by utilizing the existing equipment onboard and designing a specific communication protocol, or via Internet if equipped vessels have navigating bridges connected. One of the software solutions could be the existing software for passage planning and ECDIS updates - Navtor, which keeps a record of all passage plans in the database and all users can select vessels to see their passage plans. Currently, this is limited to owner's fleet, as sharing passage plans to outer owners would be a violation of proprietary information policies; however, the platform exists and all we would require is a software solution without adding new technology onboard ships. Through Navtor NavStation platform, passage plans are visible on ECDIS stations, so it would be easy for navigators to take the advantage of the technology to reduce the uncertainty. At the time of writing of this thesis, legal framework is not allowing for the VTS stations to issue orders, but rather advisories to targets, but the same effect would be achieved in VTS covered areas.

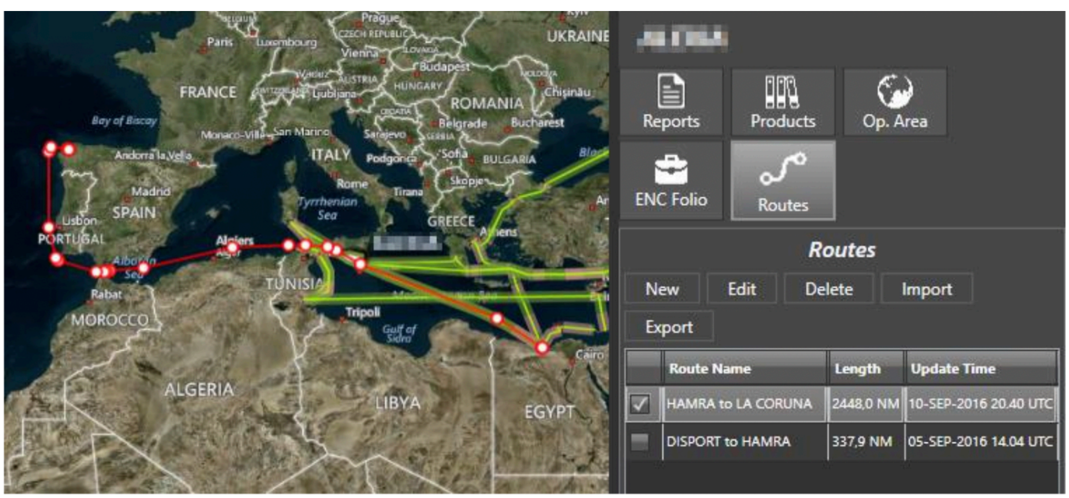


Figure 5.1 – Navtor tracker with routes (Source: Navtor instruction manual, 2018)

---

[1] Similar content to this Section has been published in the article Rudan, I., Francic, V., Valcic, M., and Sumner, M. *Early Detection of Vessel Collision Situations in a VTS Area*, Transport 35 (2) 2020.

In case of the route information sharing between ships, the interval of sharing information and dynamic updates of the root would depend on the technology used to transfer data among the vehicles in range. Exchange of information could be achieved through the Internet and then we could potentially have information for the whole route, which would allow for strategic intent awareness, or if the Internet sharing is not available, a special protocol through existing short-range communication systems onboard could be designed to allow for route information sharing. In the case of the short-range protocols, or tactical intent awareness, the affected vehicles would have a benefit of not only knowing the past tract that is shown on ARPAs or ECDIS units, but also to see the following route with all intended turns and planned speed changes. The other approach is to share information with a VTS center that can keep submitted routes in their database and receive regular updates from vessel intending to pass through their area. In this way, a VTS center can issue early advisories to vehicles with potential collision positions and times to proactively avoid close quarter situations. Regardless of the approach, the main challenge is not technological, but rather legal. Marine industry would need an international standard for developed protocols, quality assurance and audit developments that would guarantee the performance.

When we think of intent, we consider an independent agent that has a capability of reasoning and ability to interact with an environment. When an independent agent interacts with an environment and collects experiences, an agent is using reasoning to make decisions and predict future states, which could be considered as an intent. Merriam-Webster dictionary [2020] defines an intent as "determination to act in a certain way". By defining the intent in this way, we can conclude that passage planning could be used as inference of an intent. Of course, we would not benefit if passage plan is designed with waypoints only and has no motion prediction in the time domain. Therefore, we have to allow for 3D motion prediction (2D position on a plane plus time) in order to allow for potential collision positions of evaluated sea surface vehicles.

Application of intent to motion planning is not a novel concept and has been used extensively in the aeronautical domain [ASAS-TN2, 2008; Brahydt et al., 2005; FAA (NextGen), 2020; Gool and Schröter, 1999; Hoekstra et al., 2002; Ruigrok et al., 2003; SESAR, 2020], as well as in the pedestrian intent motion detection [Bandyopadhyay et al., 2013; Bandyopadhyay et al., 2013; Karasev et al., 2016]. Even though the aeronautical and ground vehicles research literature is valuable resource to study intent further, we have to be aware that it is not possible to simply apply these concepts to our research, mainly because of various technical and legal differences of the two industries. We recognize that maritime industry takes longer time to adapt to technological changes, mainly because the public is less sensitive to marine incidents, especially outside of the oil and gas industry, while admiralty law changes are very slow. Even though many technological advances already happened, and solutions are available, the choice of implementation is left to commercial operators and most of them consider only the immediate economic benefits, for which the new technology is only an additional cost. Without the proper pressure from administrations and the public, new technologies will wait for a scalable maritime disaster to be implemented. On the other hand, administrations have been proactive with introducing technologies that will reduce an environmental impact.

Even though we can draw some insights from the aeronautical domain, we have to note that there are some significant differences that makes a transition to maritime domain challenging. Namely, in regards of the motion planning, aircrafts have a 3rd dimension available, as there is a well-used possibility of maintaining a vertical separation, while sea surface vehicles operate strictly in 2 dimensions on an Earth plane. Also, we have to note that most of the aircrafts have to file their flight plans, which are available to the ground Air Traffic Control (ATC), so there is a central control of collision avoidance and the burden of separation actually lies with the ground ATC, rather than the flight crew. In the maritime domain, commercial ships are also required to develop a passage plan from berth to berth and most of the operators require that passage plans are submitted to the shore operations and then vetted by their marine departments; however there is no requirement to share these passage plans with other vessels or VTS centers. Even though the COLREGs are similar to the right of way the airplanes are following [FAA General Operating and Flight Rules, 2017], the main distinction is in application of these rules where cockpit crew usually relies on ATC to resolve trajectory conflicts and only in rare occasions resolve conflicts themselves, sea surface vehicles' navigating crew are under full obligation to resolve target conflicts autonomously. It is imperative to state how various studies were made to review benefits of intent sharing in the aeronautical domain [Hoekstra et al., 2002;

Ruigrok and Valenti Cari, 2003; Wing et al., 2002], and all of them concluded that sharing intent showed significant improvement of safety and efficiency and none of the reviewed cases showed any negative impacts on efficiency and safety of airspaces.

The maritime research community only recently commenced to research benefits of intent inference to motion planning [IALA, 2020; IMO, 2020]. International Maritime Organization (IMO) and International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA) developed an e-Navigation concept where various projects explore and demonstrate benefits of intent-aware navigation in maritime domain [Billesø, 2015; Lind et al., 2016; Porathe, 2012; Porathe and Shaw, 2012]. Several of these projects explore benefits of having a shore-based services to aid the trajectory conflicts [Borup, 2015; Rihacek et al., 2015; Rudan et al., 2020], while ship-to-ship intent share solutions are scarce.

When we think of intent information that would be shared with other vehicles and VTS centers, we have to consider the amount of data that needs to be promulgated. We have already stated that the leanest model will be the sustainable approach, as it will not require additional computational power. We, therefore, consider two approaches when sharing intent information, which are either sharing only a passage plan with added information about planned speed, or we share a full motion intent where also any deviation from the plan is promulgated by stating an intent of turning and speed change. In both cases we can utilize the meteorological effects on passage plan and determine a sequence on which the dynamic updates would be shared with interested parties.

If we consider situation where intent information is shared with VTS or another vehicle, there are several distinctive layers of information we can transmit:

- Vehicle intent – information that can be obtained by various systems installed onboard; fixed data information, such are length, call sign, type of the vessel (mostly already shared via AIS), with some dynamic components such are destination, current speed, ETAs, steering mode engaged, etc. (also shared via AIS), but also information about the maneuvering characteristics of the vessel, such are minimum and maximum speed, maximum ROT (in order to feed the equipped vessel with enough information to calculate performance, if needed);

- Passage intent - information about the passage and progress on the planned passage including list of all waypoints, expected time reaching these waypoints, data collected from ECDIS where shallow water and static objects present limitation to navigate for that vehicle, weather constraints influencing speed and heading, immediate motion plans (immediate alteration of speed and heading due to collision avoidance or weather routing), with a global objective of interpolation to represent a viable trajectory of the vehicle;
- Utility intent – relevant only to our model of using HMMs to optimize motion planning solutions, so in this category we share rewards structure, utilities and constraints to describe how ownship or target is calculating resolution advisories in case any discrepancies exists so that the ownship can take any potential errors into account when optimizing paths to avoid collision.

In line with the HMM and reinforcement models used in earlier chapters, we can represent intent as a part of a global state vector S_{OV} that contains all other relevant vectors, such are vehicle intent VI_{OV}, passage intent PI_{OV}, utility intent UI_{OV}, etc.:

$$S_{OV} = \begin{bmatrix} s_{OV}(t) \\ VI_{OV}(t) \\ PI_{OV}(t) \\ UI_{OV}(t) \\ \dots \end{bmatrix} \tag{5.1}$$

Similarly, global state vectors of targets are utilized to communicate intent and reduce uncertainty among the sea surface vehicles in the vicinity.

The alternative to intent information sharing is behavioral uncertainty, which is the reality of the present system. Alternative to behavioral uncertainty cannot be intent inference as this could lead to catastrophic consequences. It is better to assume constant speed and heading of a target than to utilize any type of guesswork on what the intent would be. We can, however, quantize the behavioral uncertainty and implement risk remedies in our models, which is exactly what we did in the Section 5.2. Further than that, we could try to estimate environmental influences on targets when predicting their trajectories, but as we do not know their maneuvering characteristics and dynamic parameters of their voyage, we could be incorrect, and that is the reason why we take information from the existing equipment, such are ARPA, AIS and ECDIS that offer better accuracy than estimation or intent inference.

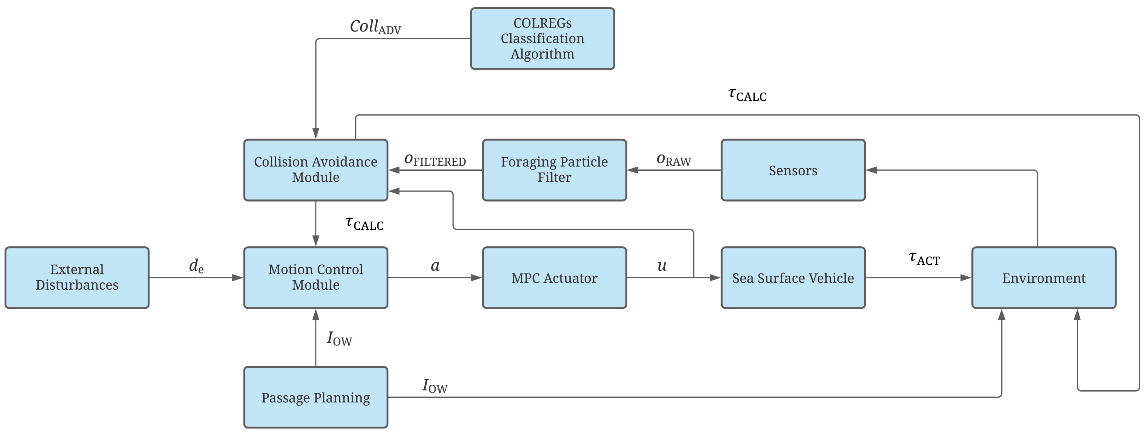


Figure 5.2 – Intent-aware collision avoidance model

Figure 5.2 depicts an intent sharing model where COLREGs Classification Algorithm delivers all necessary information ($Coll_{\mathrm{ADV}}$) to Collision Avoidance Module that contains all collision avoidance algorithms. Sensors collect information from the environment (s_{RAW}) and through FPF filter deliver data (s_{FILTERED}) to Collision Avoidance Module. Collision avoidance module turns data into trajectories (τ_{CALC}) and safety zone information in order to deliver that information to the environment and other equipped vehicles, as well as to deliver it to motion control module. Motion Control Module takes external disturbances (d_{e}) and Passage Planning information condensed in a string of intent data about the own vehicle (I_{OW}) to select actions (a) from the action space and with the aid of an MPC controller turns actions to actuator control inputs. MPC then utilizes actuator data (u) to enforce motion of the own vehicle and then check calculated trajectories and confirm feasibility of the actual trajectories (τ_{ACT}) that are shared with the environment.

Communicating intent is the biggest challenge of introducing this methodology, mainly due to the financial expenditure that would ensure robust, secure and unified system. Developing the communication system is out of the scope of this thesis, however there would be generally two approaches to communicating intent. One approach is a short-time horizon intent communication to vessels in range, distinct conflict resolution services, or VTS centers sharing tactical intents, while the other approach would be sharing a strategic intent with longer time-horizon in mind that is communicated to applicable vehicles, distinct conflict resolution services, or VTS centers.

If we consider a navigating bridge of a modern sea surface vehicle, we notice that most of the data is already in digital form, which makes it easy to relay. Unification of information is of paramount importance to ensure all vehicles are able to understand and exploit information received onboard. This is one of the main reasons why *dCPA* and *tCPA* are preferred to the concept of ship domain. *CPA* related measures are easily quantifiable and transferred to other vehicles, so that uniformly understood metrics is used. We have to ensure that exactly the same logic is used when sharing intent information. In the aeronautical domain, several intent descriptive languages have been developed and agreed upon, but most remarkably Aircraft Intent Description Language (AIDL) [Lopez-Leones et al., 2007; Lopez-Leones, 2008; Vilaplana et al., 2005], where simple phrasing and unambiguous language is used to describe aircraft intent. In the marine domain [Rydlinger, 2015] is the first attempt in standardizing intent communication, where ECDIS is commonly used to phrase intent through waypoint exchange information. Intent communication in marine domain is still in early stages and requires thorough survey of proposed standardization with administrations.

After determining the standardized language, we have to find solution on how to share this information to other vehicles and shore stations. We could utilize the existing equipment or develop new one. In the aeronautical domain, Automatic Dependent Surveillance-Broadcast (ADS-B) is readily available [Barhydt et al., 2005; Barhydt et al., 2004; Hwang and Seah, 2008; Lewis et al., 2012; Mondoloni, 2006; Pasaoglu et al., 2016; Ruigrok et al., 2001; Tarhan et al., 2014; Warren, 2000] and is used to broadcast intent. In the marine domain, the AIS is used to transmit some of the vehicles' information, but there is no intent communication at the present architecture outside of the destination and ETA [IALA, 2016]. However, destination and ETA is not sufficient information to be used for collision avoidance, as it does not provide information about the passage plan and the next waypoint a vehicle is navigating to.

IMO's project of e-navigation was developed to address challenges of digitalization and to bridge the gap in knowledge of human-machine interactions in the marine domain with the main goal of harmonizing "collection, integration, exchange, presentation and analysis of marine information on board and ashore by electronic means to enhance berth to berth navigation and related services for safety and security at sea and protection of the marine environment" [IMO, 2014]. As a response to the e-navigation initiative, a group of researches created the Sea Traffic Management (STM) project that is supported by the European Union [STM, 2019]. The objective of the STM project, also known as MONALISA 2.0, is to establish digital information exchange service between vehicles and shore and to establish a platform that will ensure quality assurance of STM produced modules among manufacturers. With initiatives like that we could witness less reliance on knowledge and ability and more reliance on information sharing. This can aid to achieve more evenly spread-out ability and quality assurance without uncertainty of language barriers.

The STM concept was designed as a service that would resemble the Air Traffic Control (ATC) model where each equipped vehicle would submit their passage plans to the STM controller, who would in turn validate the plan or propose the amendment to the plan so that two vehicles never meet in close-quarter situations. SMT would also issue advisories to the users that along their planned passage, they would be in potentially critical situations with other vehicles. However, there is a significant distinction between ATC and STM operations and it is directly related to the medium used for transportation. While ATC ensures no aircraft meets at the same altitude, in the marine domain we only can assign different paths vehicles can take. Where waterways are wide and deep, this is not a problem, but in busy waterways and narrow channels, we can't assign different paths to vehicles. We could only ensure that different timeslots are available to vehicles and mandate different ETAs to certain waypoints, but from the practical point of view, this would present a significant challenge. Unlike aircrafts, where reducing a thrust or raising an angle of attack would create appropriate drag to reduce speed rapidly, commercial sea surface vehicles require significant time to reduce or increase speed. Even more difficult for marine vehicles is to maintain their speed at constant. As airplane can quickly adjust to meteorological influences with auto/thrust function, sea surface vehicles usually experience many speed fluctuations during a day due to currents, winds, and sea states. Together with the meteorological influences, we have to keep in mind that operating a commercial marine vehicle includes various speed reductions due to restrictions of the power plants onboard; for example, depending on the fuel used, many vessels will have to slow down for several hours every second or third day to clean economizers, so it is hard to expect that at the beginning of the voyage vessels will comply with fixed ETAs on certain waypoints. From the commercial point of view, ETA to destination is usually changed several times per voyage as demanded by Charterers or receiving terminals. In the marine domain, this kind of flexibility is understood and expected. Therefore, more dynamic approach to intent sharing is required for the marine domain.

Having an ATC type service is an idea to resolve many close quarter situations, but we also need a dynamic passage intent sharing system, where navigators would have information about a potential collision situation in a short-term and mid-term time frame with dynamic updates of their progress along the passage plan, so that the time to the potential close-quarter situation can be updated consistently. It is therefore important to have a service where not only passage plan, but also intent is shared with shore services such are STM or VTS, but also directly with vehicles in the vicinity. Ship-to-ship route exchange (S2SREX) ECDIS layers [STM, 2016] have been introduced in the e-navigation research recently, and as a specific ECDIS layer, this service offers a good potential for utilizing our results in a real-world application. S2SREX layer has an embedded rendezvous (RDV) information layer that can calculate predicted meeting points and show them on the ECDIS screen. The major benefit of the STM system is the ability to share passage plan with various stakeholders (shore control centers, charterers, owners, ports and services, pilots, meteorological services, etc). The system provides an option of shore route optimization and adjustments due to weather or port delays. It can also detect deviation from the plan and request is then sent to the sea surface vehicle to verify the deviation. The plan is that STM allows for various layers of services and users are allowed to choose which layers they want to use. For example, LNG tankers often have to circle in holding patterns to manage cargo tank pressure. If deviation from the passage plan would involve warnings received from the shore center for each deviation, this would create additional burden on ship's crew, so LNG tanker can choose not to receive warnings, but only to send updates of the route to the concerned parties.

Finally, as we stated earlier, we are concerned with tactical or "last-minute" navigational situations, while the use of STM is for strategic navigational decision making [Aylward et al., 2020]. If the own ship or the target is not following their broadcasted passage plans, unless there is a manual rebroadcast of the plan, both vehicles will base their decisions on skewed data, which can lead to potential incidents. That is why STM and S2SREX can't be considered as decision support system for collision avoidance. As expressed in [De Vries, L., 2017; Hollnagel, E., 2017], development of decision support system has to be done from the human perspective and according to human needs. It is easy to fall in a trap of technological solutions for technological minds. In their concluding remarks Aylward et al. [2020] confirmed concerns from the International Chamber of Shipping (ICS) that STM approach could lead to overreliance and misinterpretation of the platform and that different kind of approach is required; notably a more "humancentric" development is necessary. When comparing the system's numerical results with human reactions, it was noted that human navigators acted earlier with slightly larger CPA margins.

In this section we consider only scenarios where own vehicle and targets are equipped with motion predictor based on the Chapter 4, so passage plan information and updates received from targets and broadcasted from own vehicle include dynamic progress along the shared passage plans. This means that for the selected targets and while we are monitoring them, we receive updates on the progress along the route including any speed change due to environmental loads, or intentional speed changes. In the next chapter we will consider mixed equipage environments, where equipped targets will have dynamic passage progress updating own vehicle regularly, while unequipped vehicles will not share any intent information with the surrounding vehicles or shore services.

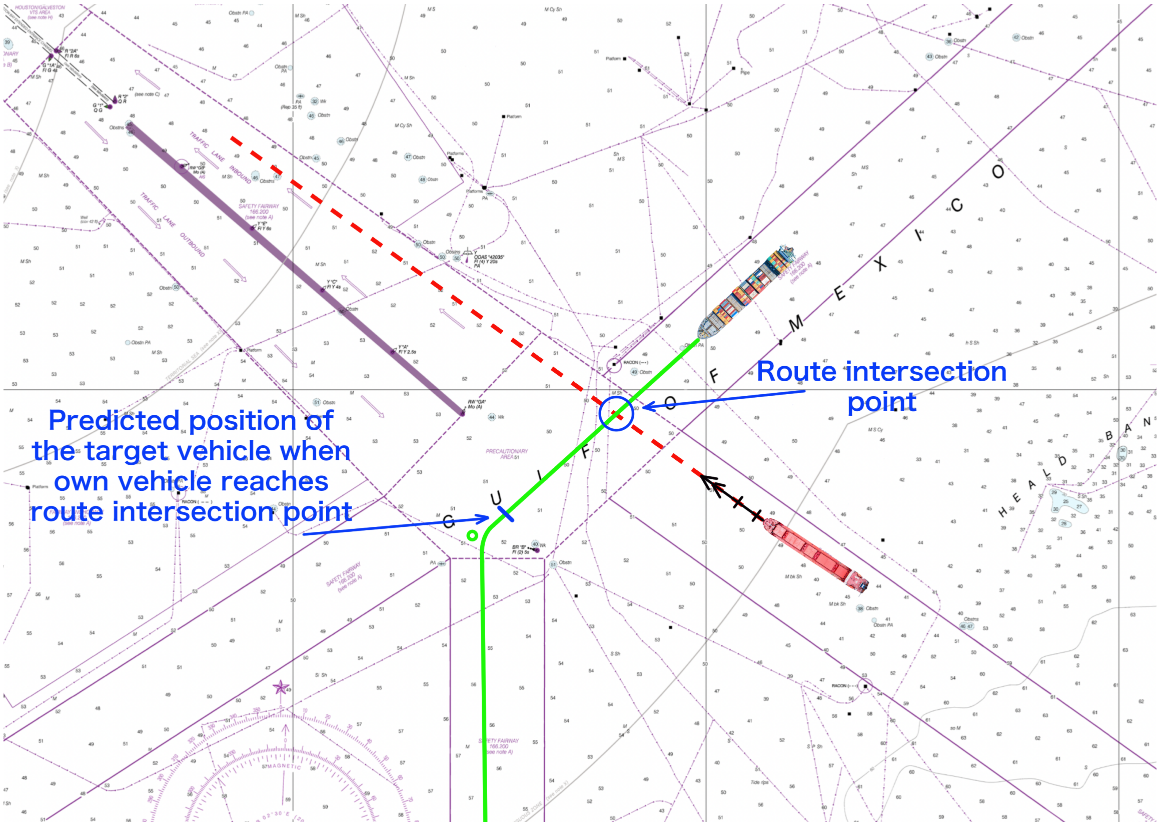


Figure 5.3 – Intent sharing and trajectory generation

As the focus of this thesis is tactical collision avoidance, we assume that the n set of sea surface vehicles $V_1, V_2, \dots, V_{n-1}, V_n$ are located within the 24 miles area observed by the onboard electronic equipment with a special focus on the ARPA range. Each of the observed vessels has a unique passage plan, while the passage plan (PP) of a j-th vessel V_j, $j \in \{1,2,\dots,n\}$, consists of the planned waypoints $w_i^j = \left(\varphi_i^j, \lambda_i^j\right)$, $i = 0,1,2,\dots,m_j$, from the set

$$PP^j = \left\{w_0^j, w_1^j, \dots, w_{m_j-1}^j, w_{m_j}^j\right\}, \tag{5.2}$$

where m_j represents the number of waypoints in the passage plan of the j^{th} vehicle V_j. The passage plan waypoint is considered as a planned position where vehicle changes her course in order to reach her destination. A simplified visualization of passage plans of two random sea surface vehicles V_j and V_k, $k \in \{1,2,\dots,n\}$, $j \neq k$ is presented in Figure 5.4.

Passage plans for vehicles V_j and V_k contain waypoints W_i^j, $i = 0,1,2, \dots, m_j$, and W_q^k, $q = 0,1,2, \dots, m_k$. Considering that the tactical collision avoidance is the focus of this thesis, we concentrate on shorter distances and utilize Legendre's theorem, which states that every spherical triangle can be substituted by a planar triangle as the arcs of spherical triangles are comparatively very small when comparing to the radius of the sphere (Nádenik, 2004). As described in the Chapter 4.3, distances between vessels in our model are within the 24 miles radius, we can utilize planar geometry to compute distances, positions and CPAs utilizing WGS-84 constants. Therefore, we can transform Earth coordinates (φ, λ) to planar coordinates (x, y), where:

$$w_i^j = (\varphi_i^j, \lambda_i^j) \rightarrow W_i^j = (x_i^j, y_i^j). \tag{5.3}$$

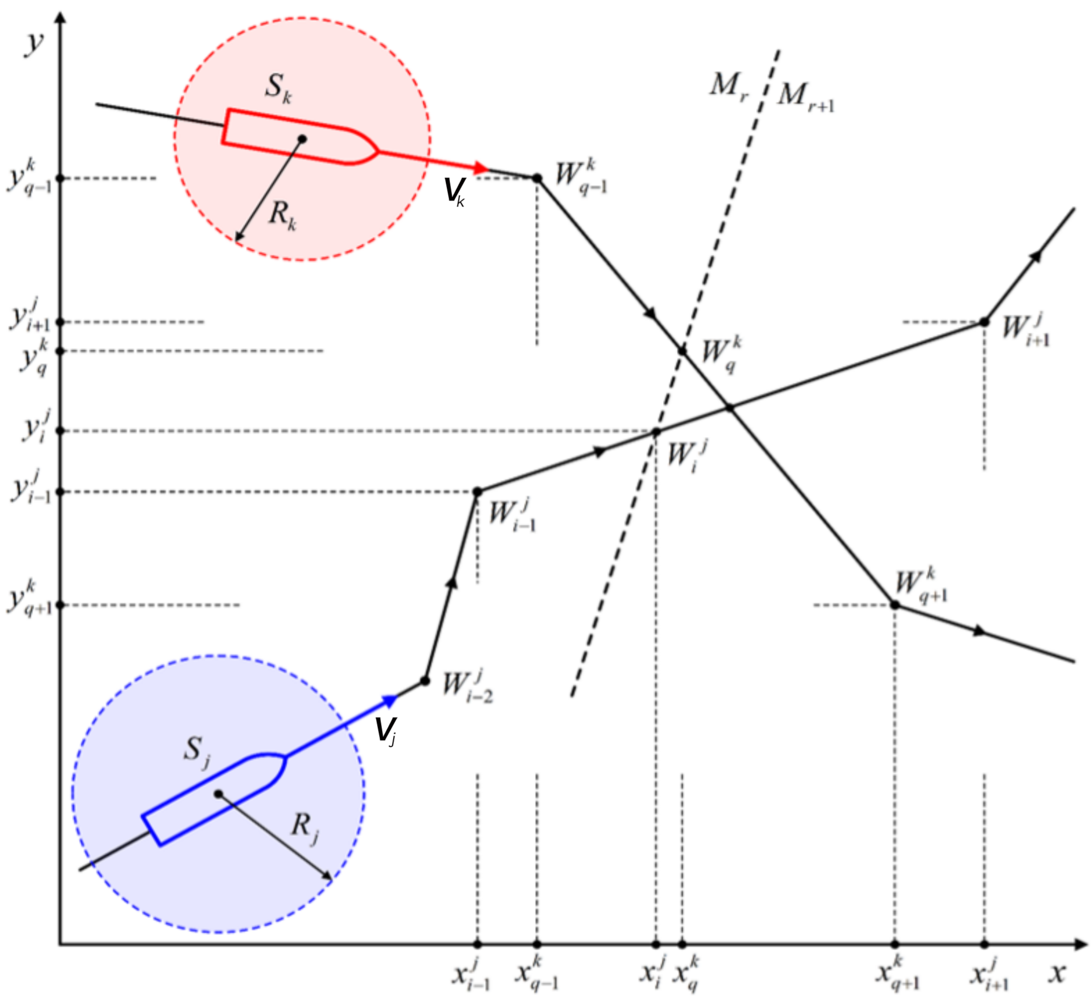


Figure 5.4 – Intent-aware passage planning (Source: Rudan, et al., 2020)

In order to get an accurate estimation of future positions along the passage plan, it is crucial to have precise information about the vehicles' speed. Our approach to estimating precise speed of a sea surface vehicle is to utilize vector algebra in order to utilize various sensory inputs and get the resultant speed and direction of equipped vehicle. The influence of meteorological and oceanological loads on the speed of sea surface vehicles, as well as the speed loss caused by planned or unplanned turns have been described in the Chapter 3 of this thesis. In this way it is possible to have COG, SOG and heading information updated on regular basis and then ownship can use a predictor to compute the meeting point or utilize target's equipment for computation and add graphical and numerical information to the ownship ECDIS and/or ARPA as an additional layer.

For a random vehicle V_j we can utilize position vector form and express a total motion between waypoints W_0^j and W_i^j $\left(1 \leq i \leq m_j\right)$ as

$$\vec{r}_i^{\,j} - \vec{r}_{i-1}^{\,j} = \overrightarrow{W_{\iota-1}W_{\iota}}, \text{ i.e.}$$

$$\vec{r}_i^{\,j} = \vec{r}_{i-1}^{\,j} + v_{i-1}^j \cdot (t_i - t_{i-1}), \tag{5.4}$$

for which the Cartesian vector components are

$$x_i^j = x_{i-1}^j + v_{(i-1,i)x}^j \cdot (t_i - t_{i-1}) \tag{5.5}$$

$$y_i^j = y_{i-1}^j + v_{(i-1,i)x}^j \cdot (t_i - t_{i-1}) \tag{5.6}$$

In line with the dynamic collision avoidance, once again we need to find a tradeoff between the computational cost of high sampling rate and how often we actually need the speed and intent information to be shared. The speed change in most commercial sea surface vehicles is a slow process, so we propose an intent sharing frequency of 30 seconds for most vehicles, but this can be also dynamic by enforcing a rule that whenever a speed changes by more than an agreed percentage, sharing frequency has to be higher, but whenever the speed remain the same, we assume the constant speed for the predictor and doe a handshake check among vehicles in the vicinity every one minute or longer. This is one of the factors that administrations would have to agree upon. In our simulations, we will use sharing frequency of 30 seconds to update vehicles in the vicinity with intent information and dynamic progress along the planned route. Keeping the information sharing frequency in mind, a position vector for a vehicle V_j would take the following form

$$\vec{r}^j(t) = \begin{cases} \vec{r}_0^j + \vec{v}_{0,1}^j(t - t_0^j), & t_0^j \leq t \leq t_1^j \\ \vec{r}_0^j + \vec{v}_{0,1}^j(t_1^j - t_0^j) + \vec{v}_{1,2}^j(t - t_1^j), & t_1^j \leq t \leq t_2^j \\ \vdots & \vdots \\ \vec{r}_0^j + \vec{v}_{0,1}^j(t_1^j - t_0^j) + \cdots + \vec{v}_{i-1,i}^j(t - t_{i-1}^j), & t_{i-1}^j \leq t \leq t_i^j \\ \vdots & \vdots \\ \vec{r}_0^j + \vec{v}_{0,1}^j(t_1^j - t_0^j) + \cdots + \vec{v}_{m_j-1,m_j}^j\left(t - t_{m_j-1}^j\right), & t_{m_j-1}^j \leq t \leq t_{m_j}^j \end{cases} \tag{5.7}$$

The stability of collision avoidance method is of upmost importance when testing feasibility, so it is necessary to apply this methodology not only to one pair of vehicles, but also to test it with multiple targets. Multi-threat analysis is a focal point of interest in the following chapter, so in this section we only verify if the above-described procedure is feasible for a pair of two equipped vehicles in the applicable range. We have selected Wärtsilä (Transas) NTPRO 5000 simulator to extract static and dynamic data, recreate voyages and verify effectiveness of our approach. The ownship remains to be LNG Tanker (Transas LNG 2; Dis. 89634t; 2.31.13.0), while the target is selected to be Feeder (Transas Feeder container ship 1; 1610 TEU; 3.0.33.0). Passage plans for the two vehicles has been reconstructed by using the MarineTraffic information system [Marine Traffic, 2020], where V_{LNG} represents passage plan of the LNG tanker, and V_F represents passage plan for the feeder. We are using W_i^{LNG} and W_q^F to depict waypoints used by LNG and feeder vehicles respectively. We also utilize the Wärtsilä (Transas) NTPRO 5000 ECDIS module to graph the recreated routes and to monitor the progress along the planned route. In our example, an LNG tanker is heading from the new LNG terminal in Rijeka towards the Mediterranean, while Feeder is heading from Trieste to Rijeka.

As both vehicles are initially too far away to transmit the passage plan and intent information directly to each other, vehicles submit their passage plans to shore stations (either VTS center or an MTS). As we assume there is an ECDIS layer available to both vehicles, by clicking on the vehicles name on the ECDIS screen, each of the navigating officers on both vehicles would be able to see the intended passage on their ECDIS screen, as well as the predicted time of reaching the meeting point. Each vehicle can initiate a download of an update to check the progress of the other vehicle along their routes. Until both vehicles are in the VHF range, they have to download updates to verify the progress;

otherwise, prediction of the progress along the route is made by utilizing the last known SOG information received. The predictor will check the last known position and if the vehicle is away from the intended route track, it will use latest SOG information and assign COG that leads the vehicle directly to the next waypoint to calculate meeting point time. If the layer contains information about maneuvering characteristics of the target and desired ROT with wheel over positions, then it can take into account loss of speed due to course change. As vehicles are closing each other in the VHF range, progress and intent information is shared every 30 seconds.

Environmental conditions are defined in Table 5.1, where V_{wind} denotes wind speed (knots), β_{wind} is wind encounter angle (º), H_s is significant wave height (m), T_p is wave peak period (s), β_{wave} is wave encounter angle (º), $V_{current}$ is sea current speed (knots),while $\beta_{current}$ is current encounter angle (º). Swell did not play a significant role in this simulation, so it is omitted from this part, but is easily added when required. Directions of environmental loads are adopted from Transas [2011], as shown in Figure 6.5, where u is ship speed in surge direction, v is ship speed in sway direction, and U is total ship speed over ground (knots).

For the purpose of simulation, we have divided the simulated area in 4 meteorological areas M_1, M_2, M_3, and M_4. We have selected arbitrary date of July 01st, 2020 and gathered meteorological data for that date from the Croatian Meteorological and Hydrological Service [DHMZ, 2020], as well as from the Windy meteorological service [Windy, 2020]. As the vehicles progress along the planned passage, they navigate through these meteorological areas, so we experience different environmental impacts on our simulated vehicles. Environmental loads for each of simulated vehicles have been gathered from the Wärtsilä (Transas) NTPRO 5000 [Transas, 2011] simulator by extracting values from a 20 minutes long simulation where average value based on the last 10 minutes of simulation was taken as simulated value in order to allow for simulation stability.

Table 5.1 – Environmental conditions for analyzed meteorological areas M_1, M_2, M_3 and M_4

Met. area	V_{wind} (kn)	β_{wind} (°)	H_s (m)	T_p (s)	β_{wave} (°)	V_{current} (kn)	β_{current} (°)
M_1	22.0	310.0	2.0	6.7	310.0	0.4	120.0
M_2	24.0	345.0	3.0	7.6	345.0	0.6	140.0
M_3	18.0	20.0	2.5	5.9	20.0	0.3	200.0
M_4	14.0	45.0	1.5	4.9	45.0	0.2	195.0

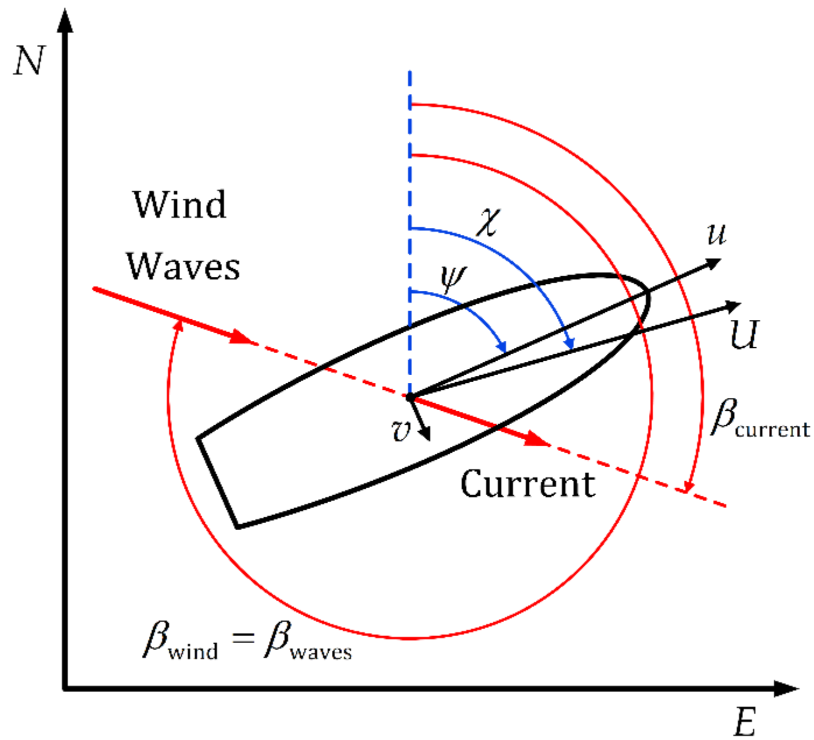


Figure 5.5 – Directions of environmental loads (wind, waves, sea current) with respect to the North axis (N) and directions of the ship with respect to the ship bow (Source: Rudan, et al., 2020)

Wärtsilä (Transas) NTPRO 5000 simulator has been used to simulate the progress of the two selected vehicles with environmental loads. The two following tables, Table 5.2 and 5.3 deliver speeds, courses, headings, distances and times between two adjacent waypoints of the planned route. The graphical presentation of the passage plans is presented in Figure 5.6.

Table 5.2 – Passage plan in time domain for feeder $\boldsymbol{V_F}$ with attainable ship speeds regarding the various meteorological areas and associated environmental loads

WP	Position	Area	Distance (NM)	Speed (knots)	Course (°)	Time (hh:mm:ss)	Duration (hh:mm:ss)
W_0^k	45.6296° N 13.5379° E	M_1	-	15.12	245.9	00:00:00	-
W_1^k	45.5518° N 13.2903° E	M_1	11.39	15.82	229.4	00:45:12	00:45:12
W_2^k	45.3853° N 13.0144° E	M_1	15.21	18.70	167.7	01:42:53	00:57:41
W_3^k	45.1080° N 13.1003° E	M_1	17.12	17.95	145.0	02:37:49	00:54:56
W_4^k	44.8073° N 13.3965° E	M_1 / M_2	21.94	18.03	145.0	03:51:09	01:13:20
W_5^k	44.5331° N 13.6666° E	M_2	20.00	15.94	67.1	04:57:42	01:06:33
W_6^k	44.6236° N 13.9663° E	M_2	14.06	14.13	43.3	05:50:38	00:52:56
W_7^k	44.6538° N 14.0062° E	M_2 / M_3	2.49	12.57	43.3	06:01:12	00:10:34
W_8^k	44.7541° N 14.1386° E	M_3	8.26	12.65	16.4	06:40:38	00:39:26

Table 5.3 – Passage plan in time domain for own vehicle $\boldsymbol{V_{LNG}}$ with attainable ship speeds regarding the various meteorological areas and associated environmental loads

WP	Position	Area	Distance (NM)	Speed (knots)	Course (°)	Time (hh:mm:ss)	Duration (hh:mm:ss)
W_0^j	45.2497° N 14.4453° E	M_4	-	6.11	241.0	02:37:57	-
W_1^j	45.1797° N 14.2670° E	M_4	8.61	14.33	199.6	04:02:30	01:24:33
W_2^j	45.1330° N 14.2435° E	M_4	3.00	14.41	189.1	04:15:04	00:12:34
W_3^j	45.1001° N 14.2370° E	M_4 / M_3	2.01	14.11	189.1	04:23:26	00:08:22
W_4^j	45.0672° N 14.2286° E	M_3	2.00	13.62	189.8	04:31:56	00:08:30
W_5^j	44.9517° N 14.2004° E	M_3	7.02	12.91	201.4	05:02:52	00:30:56
W_6^j	44.7528° N 14.0907° E	M_3	12.82	13.43	191.2	06:02:27	00:59:35
W_7^j	44.6446° N 14.0608° E	M_3	6.61	15.86	166.1	06:31:59	00:29:32

The initial passage plan evaluation of both vehicles is done with the planned speed and it gives us an overview what is the position where a potential for collision exists and at what time would these two vehicles meet at that position. Considering the environmental loads and possible collision avoidance along the planned passage, actual speed will almost certainly differ from the planned one, so as the vehicles get closer, the meeting position and time will change. This is the reason why dynamic updates of the passage progress and intent is important for the intent-aware collision avoidance to work. Once the vehicles are within the acceptable range, our collision avoidance algorithm can find the optimal trajectory to avoid collision between affected vehicles.

The purpose of the initial evaluation of the passage plan is to identify potential collision and near miss situations, so the detailed analysis is not required initially. Only when the vehicles are within the 24 NM range, we engage the full analysis of the potential collision position and time. Once we have the output of the collision avoidance algorithm and the COLREGs classification algorithm we broadcast the intent to equipped vehicles in the vicinity and shore stations (if any in the range).

Each of the vehicles submit their initial passage plan with their planned speed for each segment of the passage. At this moment transition between waypoints is calculated only with the planned speed without taking any speed change due to environmental loads or turns into account. In our case, feeder is doing a short passage from Trieste to Rijeka, while LNG tanker is leaving Rijeka and heading towards the Mediterranean Sea. The feeder is the first one to commence her voyage, so we assign her a time stamp $t_0^F = 00\ h\ 00'00"$, while the LNG tanker leaves Rijeka later that day at the time stamp $t_0^{LNG} = 02\ h\ 37'57"$. For simplicity and visual representation, we use time stamps rather than the actual time of the day. The initial prediction states that two vehicles are going to meet at the position defined as $\overline{W_7^F W_8^F} \cap \overline{W_6^{LNG} W_7^{LNG}}$ at the time stamp $t_{collision}^{F,LNG} = 06\ h\ 19'46"$ where collision risk exists, and predicted CPA is 0.04 NM.

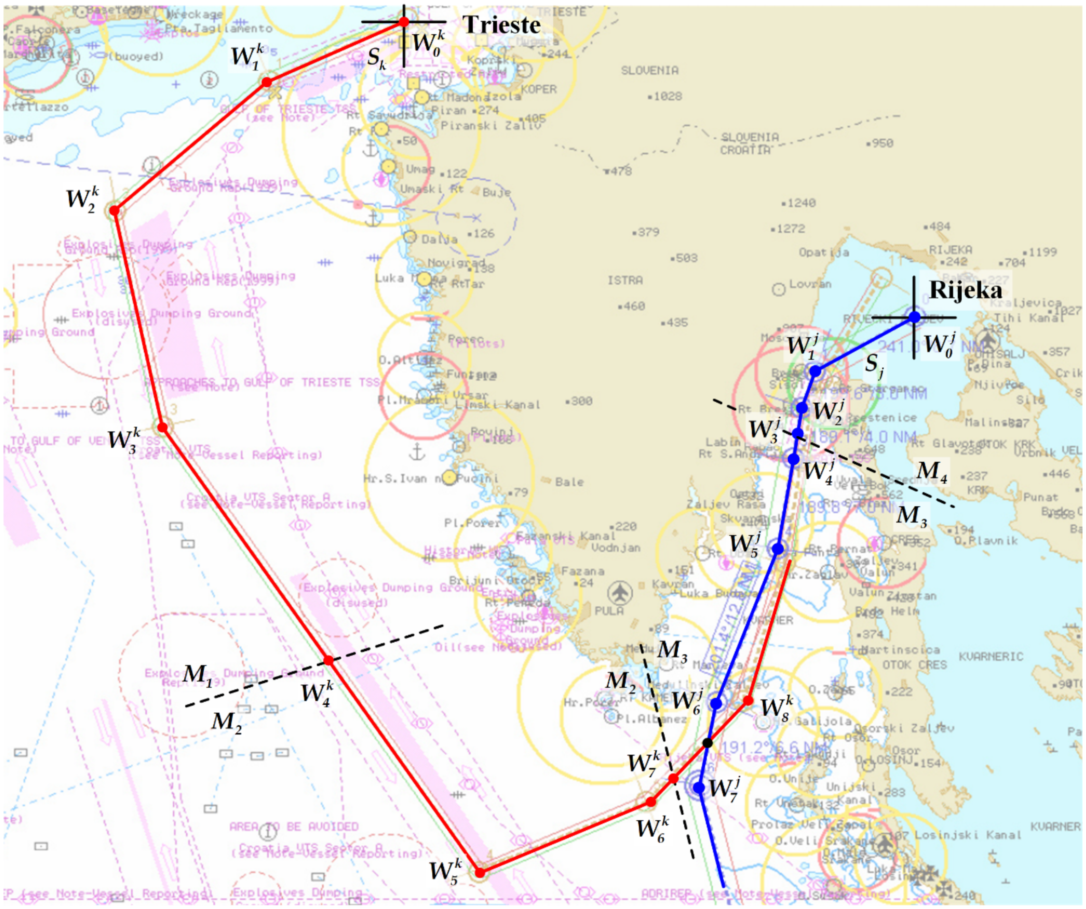


Figure 5.6 – Visualization of passage plans and appropriate waypoints for selected feeder vehicle (S_k) and LNG carrier (S_j) with associated meteo-oceanological areas (Source: Rudan, et al., 2020)

Once the targets are within the range and in proximity, a manual communication setup can be made requesting a vehicle to alter the course in order to avoid collision, which would be beneficial in the case of non-compliant targets, or we could utilize a special communication protocol to automate alerts between targets. In this way we can communicate passage plans, but also, we are able to share intent and direct resolution advisories.

5.2 Coordinated collision avoidance

Once sea surface vehicles are within range, coordinated collision avoidance can take place. Research is still within the intent-aware collision avoidance domain; however, opportunity to engage equipped vehicles is utilized. In this Section focus is on how coordinated collision avoidance can increase safety of navigation. Even though it is important to showcase stability of the model in environments where a single unequipped target is encountered, the reality of modern navigation is that multiple targets will be met at the same time, especially in coastal and restricted areas.

If multiple targets are equipped, the only way to safeguard safety of navigation is to coordinate maneuver recommendations, which requires intent communication and coordination of the collision avoidance systems. How much of the information about state space ownship would be able to share with other targets depends largely on the ability of the communication protocol and compression rate. In order for one vehicle to accurately estimate states in time domain, basic maneuvering characteristics should be shared when initially selecting a target. If a platform is unable to do this, only information on how fast a certain ROT can be achieved would be shared. Building a compatible and cooperative communication system is not a topic covered in this thesis; however, several attempts have been made and are available for further testing on commercial vehicles [Rydlinger, 2015].

One of the first challenges of coordination is the priority ranking. As stated in the previous Section, administrations in the maritime domain have to decide on various rules, including the priority ranking. Logic of urgency is utilized when ranking vehicles in the range. Firstly, any vehicle that has a combination of low CPA and short TCPA is prioritized, as navigating officers on these vehicles have to act immediately to reduce the risk of collision. Secondly, the vehicle that is slower to respond has a priority in decision making. Finally, if vehicles are similar in maneuvering characteristics and there is no priority call from the COLREGs, the vehicle with higher MMSI gets the priority to be the master sea surface vehicle among all equipped vehicles within the range.

Another challenge is to ensure compatibility of coordinated advisories. If ownship faced a head-on situation and affected (in this example two are assumed) vehicles turn to the opposite side (one turns to port, while other turns to starboard), the possibility of collision would increase significantly, and close quarter situations would be hard to avoid. This is often a case when VHF radio units are used and there are poor communication standards enforced which leads to miscommunication and action that is not following the assumed resolution plan. Standard Maritime English should aid in these situations, but often there is a large gap in knowledge and proficiency of mariners meeting each on various waterways. Therefore, the proposed model has to ensure that compatible advisories are given and that COLREGs are utilized to resolve close quarter situations on time and efficiently. This is achieved by assigning a master vehicle that will enforce cooperation to the slave vehicle. In Table 5.4 a set of advisories that could be used to inform navigators visually and aurally about the action and intent of the other vehicles in range are proposed.

Table 5.4 – Advisory set

Name	Description
NOC	NO Conflict
KTCS	Keep The Course and Speed
NTP	Nothing To Port
NTS	Nothing To Starboard
EHC	Expedite Heading Change
TTS ________°	Turn To Starboard – Heading ________°
TTP ________°	Turn To Port – Heading ________°
FTTS________°	Fast Turn To Starboard – Heading _______°
FTTP________°	Fast Turn To Port – Heading _______°
ID-HTS	Imminent Danger – Hard To Starboard
ID-HTP	Imminent Danger – Hard To Port
CL-BTR	CLear – Back To Route
!!!NC	Non-Compliant Target

In case of TTS and TTP advisories, vehicle is required to turn to the heading computed as optimal or satisficing by the Collision Avoidance algorithm. Heading rather than course has been selected, as it is clear for the helmsman to steer heading rather than course. It should be done with the ROT currently active for that vehicle (usually 10 degrees per minute). However, in the case of FTTS or FTPS, vehicle is required to turn to the computed heading as fast as possible and without delay. ID-HTP or ID-HTS requires navigators to turn immediately with an order of hard to port or hard to starboard, as the danger is imminent. It is important to note that in this case a lot of attention is required from the navigators. As vessel turns with a wheel hard over, speed change will be noticeable and, depending on a type, vehicles will transfer significantly, so if vehicles were very close when maneuvering started, navigators will have to correct the turn in order to avoid touching their sterns. When ID advisory is activated, the system tracks COG and SOG every second to determine when to stop the turn, after which a new advisory KTCS will appear to the navigator. Finally, !!!NC advisory informs both master and target vehicles that one of the vehicles is non-compliant. At that moment system tracks the NC target and if no response to the alarm is received, it becomes a vehicle to avoid by others regardless of the COLREGs. Even though not included in this Table, manually initiated advisories in case of emergency could be utilized to alert other vehicles to provide more space, or to answer to some of the requested advisories with UNABLE signal.

5.2.1 Forced cooperation

The proposed model requires a protocol to ensure compatibility and coordination of maneuvers. In order to avoid advisories that lead to significant near-misses or close quarter situations, it is necessary to ensure that after the master vehicle issues intent, slave vehicles have to follow up with compatible maneuvers. Whenever possible, master vehicle will act in the best interest of all participants and advise other vehicles to maneuver in accordance with COLREGs and to increase horizontal separation. In this sense centralistic approach is evident for which information as perfect as possible is needed.

However, in many situations, some aspects of the state space are going to be omitted from the targets in the range and the master vehicle will be required to implicitly predict progress of other targets. So, whenever centralistic approach is not viable, master vehicle will act in its own interest and share the intent with other vehicles together with the maneuvering mode the master vehicle is in. The maneuvering mode will determine what is conditionally allowed for other targets to do in order to avoid close quarter situations. This intent interaction is called forced cooperation structure. Considering the variety of vehicles navigator can potentially meet, uncertainty about ship handling capabilities is evident, namely the ROT and how much time is required before a vehicle actually commences its turn. Achieving uniformity with turning rate is difficult as it depends on the speed the vehicle is doing, draft maintained and environmental loads. It is also known that different speed of turn is achieved with the number of steering gears engaged. Therefore, standard could be enforced where 10 degrees ROT is considered appropriate for all vehicles within the range, so all predictions are done with 10 degrees ROT, mandating navigators to know their maneuvering characteristics and to assume all navigators will initiate turns on time to achieve 10 degrees ROT as soon as possible. As our goal is to minimize the impact of the human factor, this approach should be thoroughly tested before adopting it as standard.

As resolution advisory computation varies due to computational setup (if all algorithms are running concurrently, or there is a parallel computing setup), this approach is to allow for master vehicle to centrally resolve up to certain number of targets (that allows resolutions to be computed within the sample time), while any other targets in the range would get forced cooperation intent information, which will allow for decentralization of decision making with constraints from the master vehicle. The number of targets handled by the own ship is determined within the algorithm itself and can be up to 60 targets at a time. Decentralized vehicle would utilize own resources to find optimal solution. Which of the remaining vehicles would act as a master for the equipped vehicles not involved in the initial group would be determined by the highest MMSI address. The initial master vehicle is selected following the proposed priority ranking, while other vehicles are selected by shortest TCPA and closest CPA to the master vehicle. In case that for any reason slave target issues a resolution advisory first, master vehicle will follow the initial advisory from the slave vehicle and then take over to complete separation. It is necessary to keep in mind that in some situations optimal resolution would not be possible, so time and search could be traded off for the optimal solution, but in the real world it is preferrable that action happens as soon as possible, and that is why satisficing solution is accepted, which is collision prevention, even though it means not conforming to some of the COLREGs, deviating significantly from the original route, or reducing the speed of a vehicle.

Along with the required reward function tuning, forced cooperation requires careful selection of the state variables, while action space is always within the motion control parameters that were defined in Chapter 4. To communicate intent with other vehicles, maneuvering mode has to be defined so that each vehicle is aware on resolution advisory promulgated in the vicinity. Therefore, following modes are possible:

- Mode 1: target does not have any advisory,
- Mode 2: target has an advisory to turn to port,
- Mode 3: target has an advisory to turn to starboard, and
- Mode 4: target has an advisory to reduce her speed.

When the mode is one, there is no advisory, so master vehicle can find an optimal or satisficing solution without constraints. If there is any other mode active with a target, it can be that the target vehicle is master already and own vehicle simply needs to await resolution advisory, or own vehicle has to find an optimal or satisficing solution with constraints received from target vehicles. For example, if mode two is active, target vehicle is turning to port, so ownship is constrained to turn to the side that will further decrease CPA. In case of a head-on situation, if the target vehicle is turning to starboard, ownship should not turn to port, so this will be a constraint. In order to have a full understanding of the situation, COLREGs Classification algorithm has to determine applicable Rules in that particular situation, so then the system will be able to advise appropriately.

Measure to prevent algorithms to frequently "change their mind" needs to be introduced, and there is a requirement to have a contingency in case the other vehicle is not following their advisory. Due to regulatory issues, steering autonomy is not yet feasible, but audible and visual warnings could be introduced. In case a vehicle is unable to follow the advisory due to issues with engine or steering, a warning is sent to all vehicles in the vicinity and all vehicles are then obliged to stay away. Finally, to prevent algorithm to frequently change advisories, reward function will penalize change of advisory for 30 seconds if the TCPA is less than 10 minutes, or 60 seconds if it is more than 10 minutes in order to allow for a target to act upon her advisory.

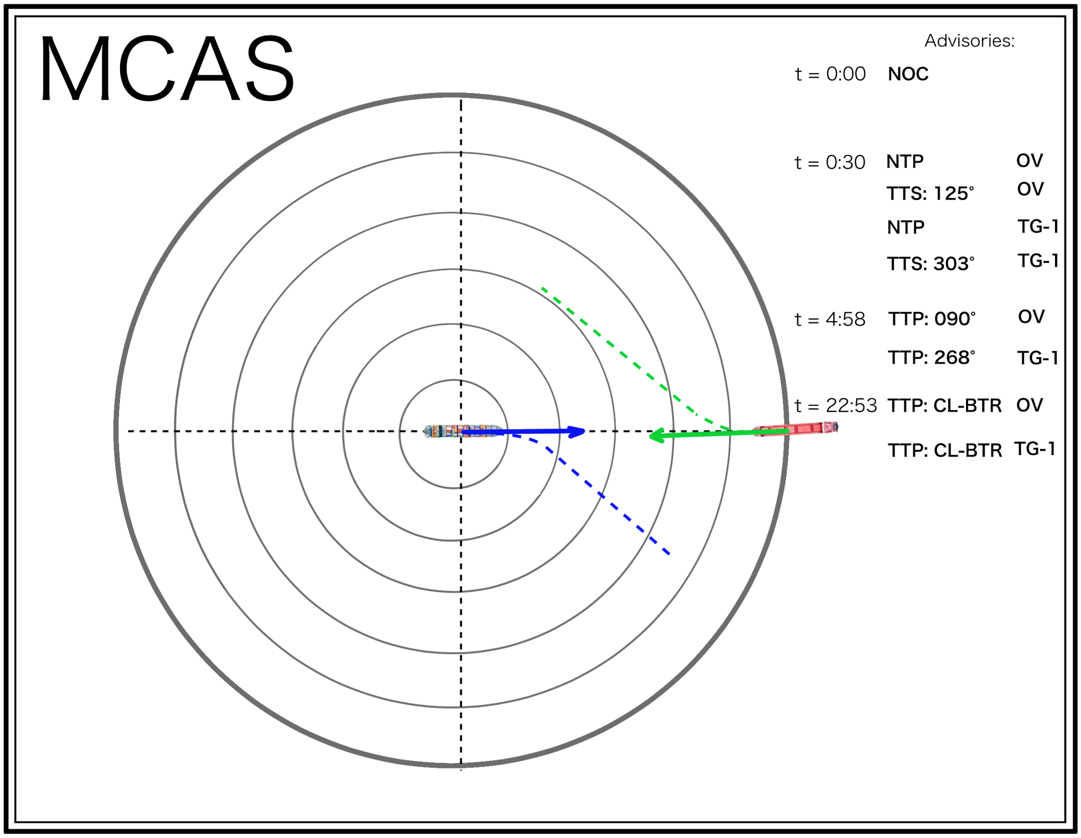


Figure 5.7 – Head-on MCAS advisories – both vehicles equipped

Figure 5.7 shows an example head-on situation where forced cooperation is used. Both vehicles are equipped, and container vehicle is a master vehicle, as it has higher MMSI address. At time stamp 0’ 0’’ vehicles are acquired at a range of 6 NM and both of them are broadcasting NOC advisory, operating in the Mode 1. If manually acquired by a navigating officer, MCAS reads the data from a target vehicle. On the other hand, if a target vehicle was not acquired, vehicle gets acquired automatically and information is broadcasted. Navigator needs to allow for ARPA to complete calculation of safety parameters, but in the meantime the system gets static information (MMSI address, call sign, maneuvering data, etc.) and master vehicle can be selected instantaneously. So, in this example own vehicle is acting as a master vehicle. Ownship is following COG = 090º with SOG = 12 kt, while the target vehicle is following COG = 268.4º with SOG = 11.9 kt. Within the 30 seconds from acquiring the system gets an estimation of CPA = 0.08 NM and TCPA = 14' 28'' and as per the COLREGs Classification algorithm, risk of collision exists with head-on situation.

As mentioned in the Chapter 4, Case Law determined that 35 degrees of course alteration is considered a significant course alteration that signals the intent to other vehicles. There are exceptions, such are narrow channels where 35 degrees would result in exiting channels, but most of the channels are piloted, so special approach is needed. Therefore, proposed algorithm initially searches for 35 degrees course alteration to starboard and verifies if this alteration would be optimal to keep the oncoming vehicle out of the minimum CPA zone. In this example, turning only ownship 35 degrees to starboard and keeping that course until TCPA is 0'0'' would result in CPA = 1.72 NM. Target vehicle receives advisory NTP and TTS 303.4º. By following resolution received form the master vehicle, both vehicles turn 35 degrees to starboard and if they would keep their courses CPA would increase to CPA = 3.29 NM with TCPA = 12' 0".

During the information exchange, designed system receives information about the minimum CPA requested from the target vehicle and if it is higher than the master vehicle's, then the slave's minimum CPA is accepted as an optimization goal. However, there would be times where minimum CPA can't be satisfied, at which the master vehicle's system will find the satisficing CPA to ensure safe distance is achieved. In this example, both vehicles requested a minimum CPA to be 1 NM, so the master vehicle's system is searching for the optimal solution where CPA is 1.1 NM. Master's predictor calculates that CPA = 1.1 NM can be achieved by continuing with evasive headings (35 degrees from the original courses) for 4' 28", after which both vehicles would receive advisories to continue with original courses. Master vehicle will receive TTP 090º, while the target vehicle would receive TTP 268.4º. After TCPA is 0' 00" and vehicles safely pass each other, both vehicles receive advisory CL-BTR, prompting them to go back to their original routes.

5.2.2 Inherent cooperation and robustness

Considering one of the most important traits of MCAS is that it has to follow COLREGs, it is expected that even when there is no actual cooperation, equipped vehicles will have inherent cooperation as individual advisories will follow COLREGs for themselves. The issue remains unpredicted behavior of fishing boats, as their actions are often stochastic in nature and can't be easily modelled. The best way to resolve issue with fishing boats is to enforce different kind of regulations for the zones where fishing is allowed, so that major waterways are reserved for commercial shipping without interference.

For simple scenarios where equipped and non-equipped vehicles meet on open seas, forced coordination would not be required as it would be easy for the system to adhere to the COLREGs without additional constraints of restricted waters. Therefore, even if no forced coordination happens, inherent coordination would be possible due to simpler scenarios and environments. In situations where forced cooperation is required, certain level of inherent cooperation can be maintained if maneuvering characteristics of a target vehicle is known. In that case master vehicle could be allowed to make predictions of target vehicles with higher accuracy, so that we can take advantage of symmetry of the state and action spaces.

Coordinated maneuvers offer safety improvements, especially when safeguarding COLREGs compliance. However, what happens when equipped vehicle does not respond to advisories? In order to maintain robustness of the coordinated maneuvers approach, !!!NC (Non-Compliance advisory) is introduced with a goal of informing other vehicles that one vehicle is not following the advisory. Administrations can discuss appropriate penalty for the non-compliance; however, COLREGs mandate to avoid collision even if deviation from collision rules is necessary. Once the advisory has been issued, the system will allow for 60 seconds for the operator to commence its maneuver, after which an !!!NC advisory will be issued. If in 60 seconds there is no reaction, an NC vehicle will be monitored for its heading, speed and ROT. If the vehicle makes maneuver contrary to the advised, vehicle becomes a high-risk target and advisory is issued to all vehicles that that target vehicle is !!!NC, which will then be a target all other equipped vehicles should avoid.

If target vehicle is !!!NC, master vehicle will endeavor to calculate optimal avoiding trajectory where CPA with !!!NC vehicle will be well within the safe zone radius. If COLREGs compliant optimal or satisficing solution is unavailable, master vehicle will search for evasive action to avoid collision regardless of the particular rules applicable to that situation. Before avoiding collision contrary to applicable collision regulation rules, MCAS will try to check if optimal or satisficing solution exists by reducing the speed of own and other vehicles (if own vehicle is a master vehicle).

5.3 Mixed equipage collision avoidance

In a single equipped environment, collision avoidance situation is solved as an egocentric matter where the best solution for ownship is found, while in the multithreat equipped situations collision avoidance situations are approached by attempting to solve conflict optimally for all equipped vehicles, but if this is not possible, satisficing solution is found. However, in situations where both equipped and non-equipped vehicles are present in a neighboring area, there are additional challenges when master vehicle is making decisions for others. In open water scenarios, all members would have sufficient sea space to maneuver, but multithreat situations rarely occur outside of the confined waterways.

In a situation with highly occupied 6 NM areas, there comes a point when finding an optimal or satisficing solution by a master vehicle becomes computationally intractable. Therefore, if the solution is not available due to computational complexity within the first 30 seconds, forced decentralization is introduced by resolving only 3 NM radius from ownship, while vehicle which has the highest MMSI address and it is outside the 3 NM radius becomes master vehicle while getting updates on vehicles within the first 3 NM radius. In that sense, the vehicle outside the 3 NM radius from the initial master vehicle gets regular updates of the constraints for the vehicles within the 3 NM of the initial master vehicle via intent advisories.

Authors Rosenblatt [2000] and Russel and Zimdars [2003] have shown that the faster and computationally effective way of solving complex state-action systems is fusing the cost or reward functions, rather than fusing state or action spaces. Reward fusion is used to compute the benefit of taking an action based on a state $R^F = (s, a)$ for each target vehicle v in the range $R^F(s_v, a)$. Two possible ways to fuse reward functions are defined, using either a sum function or a max function. The main difference for the two is the goal we wish to achieve. If the goal is to find the optimal global reward for all vehicles in the vicinity, then sum function fits well. On the other hand, if goal is that each vehicle finds optimal reward following the egocentric strategy, then max function offers a better computational performance. Therefore, by defining a function f as a function that combines rewards of all targets in the vicinity, then

$$R^F(S, A) = f\big(R^F(S_1, A), \dots, R^F(S_N, A)\big) \tag{5.8}$$

where N is the number of target vehicles in the range. After finding the optimal fusion reward, an optimal action for the ownship is then computed as the following policy:

$$\pi^F(S) = \operatorname*{argmax}_a R^F(S, A). \tag{5.9}$$

When ownship is resolving multithreat situation centrally, the optimal solution will be the global reward that takes into consideration all vehicles and their individual rewards after all intent information has been shared. At the moment of evaluation, the MCAS model assumes that all equipped vehicles are following computed advisories. If there are any unequipped vehicles in the vicinity, the proposed model assumes they are keeping their current course and speed. ROT is readily available either via ARPA or AIS, but if the predictor is allowed to utilize information about ROT without knowing the intent, false predictions could be received, so rather path updates at each time-step are utilized. Hence, for the scenarios where ownship is resolving the close quarter situations for the equipped vehicles in the vicinity, f is defined with summation strategy:

$$R^F(S, A) = \sum_v R^F(S_v, A). \tag{5.10}$$

The other approach is to find optimal solution to close quarter situation where each equipped vehicle makes egoistic decisions on their own. Even though in this case there is no master vehicle to solve the situation for other equipped vehicles, intent is still shared among the vehicles so the predictors of each vehicle can be aware of the intended passage. Therefore, it is necessary to find a solution where the optimal path is computed taking the best approach to each individual target vehicle in the range and selecting the combination of actions with the maximum reward. With this approach accumulating rewards for each vehicle is avoided, but rather only benefit of ownship considered in relation to the other vehicles in the range (still by following COLREGs):

$$R^F(S, A) = \max_{v} R^F(S_v, A). \tag{5.11}$$

Extending on the works of Rosenblatt [2000] and Asmar [2013], the following theorem is developed:

Theorem 2. *If assumed that the optimal action for each target* $\pi^*(s_1), \dots, \pi^*(s_T)$ *within the predefined horizon is NOC, then the sum and max strategies will also have an optimal action NOC.*

Proof. Considering $\pi^*(s_n) = NOC$, then for all targets n on the horizon,

$$R^*(s_n, NOC) \leq R^*(s_n, a), \;\; \forall\, a. \tag{5.12}$$

⇒ *Sum strategy*

Considering equation (5.12),

$$R^*(s, NOC) = \sum_n R^*(s_n, NOC) \leq \sum_n R^*(s_n, a) = R^*(s, a), \;\; \forall\, a. \tag{5.13}$$

Hence, the optimal policy $\pi^*(s) = \operatorname*{argmax}_a R^*(s, a) = NOC$.

⇒ *Max Strategy*

If, without loss of generality it is assumed that:

$$\pi^*(s) = a \neq NOC, \tag{5.14}$$

then $R^*(s,a) < \underset{n}{\operatorname{argmax}} R^*(s_n, NOC)$. As defined by the f of the max strategy, a target m exists that has the property $R^*(s,a) = R^*(s_m, a)$, and by utilizing (5.12)

$$R^*(s_m, NOC) < R^*(s_m, a). \tag{5.15}$$

Hence, for all targets $n \neq m$ it stands that $R^*(s,a) < \underset{n}{\operatorname{argmax}} R^*(s_n, NOC)$. As from (5.15) it is evident that $l \neq m$, it is assumed without loss of generality that $R^*(s_l, NOC) = \underset{n}{\operatorname{argmax}} R^*(s_n, NOC)$.

From the expression (5.14), the definition of f for max strategy, and expression (5.12),

$$R^*(s_m, a) > R^*(s_l, a) > R^*(s_l, NOC), \tag{5.16}$$

but it was also stated that

$$R^*(s_m, a) = R^*(s, a) < \underset{n}{\operatorname{argmax}} R^*(s_n, NOC) = R^*(s_n, NOC), \tag{5.17}$$

hence, there is a contradiction that ends the proof.

□

If the proposed system manages to issue intent advisories within 30 seconds timeframe, centralistic approach and summation function are used. On the other hand, if the decision process takes more than 30 seconds, the number of equipped and non-equipped vehicles is large and probably there are a lot of constraints in a form of shallow water, land, etc., so egocentric approach and max function are utilized. In the next subchapter, verification of the critical number of targets after which it is necessary to utilize egocentric approach with max function will be conducted.

5.4 Are we better off? – Mixed equipage

Mixed equipage simulations are approached the same way as the egocentric simulations investigated in the previous chapter. Therefore, scenarios 2 and 5 are used to verify if the mixed equipage approach would produce desired effect and if there are any benefits of sharing intent, submitting passage plans and broadcasting advisories. Scenario 8 is omitted from the analysis as it produced relatively same results as in Chapter 4. Own vehicle is selected to be a master vehicle (with the highest MMSI number), so it was the task of the proposed planner to resolve conflicts within the parameter and issue alerts for other vehicles in the vicinity.

Scenario 2

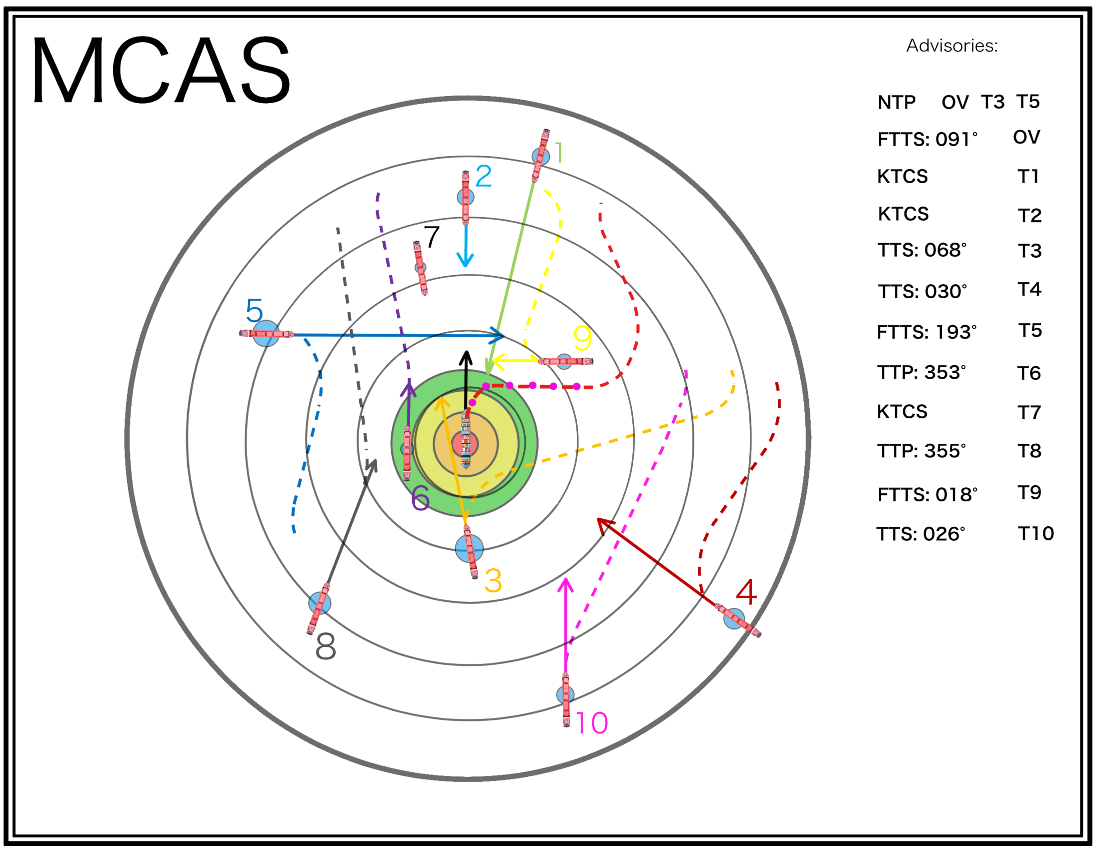


Figure 5.8 – MCAS overview of the equipped optimal trajectory for the Scenario 2

With increasing number of target vehicles, it gets difficult to visualize geometries. Nevertheless, Figure 5.8 depicts simulation of the scenario 2 with no external disturbances (to simplify visualization). Instead of assuming other vehicles are noncooperative and will follow current trajectories, MCAS determines optimal maneuvers to allow for the largest distance between targets. There is a priority to follow COLREGs, but in cases where this is not possible, maneuvering is done to avoid contacts. Simulated cases are complex, and navigators should endeavor not to be in these kind of situations in the real life by acting early and showing intent. Early detection of collision and passage planning aims to resolve conflicts well before the MCAS has to issue advisories.

For the own vehicle, MCAS is urging us to do a fast turn to starboard (utilizing two steering pumps and no restrictions on the ROT). At the same time, it is noticed that MCAS advises targets 1, 2 and 7 to maintain their course and speed and this is due to the fact that sea space around vessels 1 and 2 is limited and turning to starboard would be considered dangerous, while turning to port would create more complexity and it is against the COLREGs relevant Rules. Considering the situation, target vehicle 3 has to do a significant deviation from the intended course and turn to starboard. Similarly, in order to avoid conflicts, target 4 is advised to turn to starboard, as there is ample sea space on her starboard side. Target 5 has created additional complexity as a noncompliant target, but in this case when equipped, target 5 is advised to turn to starboard and allow for conflict resolution. Targets 6 and 8 are advised to turn to port, but not immediately (all targets would get the relevant timestamp of their advised turns). MCAS selects to resolve the lowest CPAs first and then adjusts advisories to other vehicles. In the case of non-equipped vehicles, MCAS fixes current trajectories of non-equipped vehicles and does not allow for any alterations in that iteration. Simulations were done without environmental loads (environmental disturbances are covered in the previous chapter).

Scenario 5

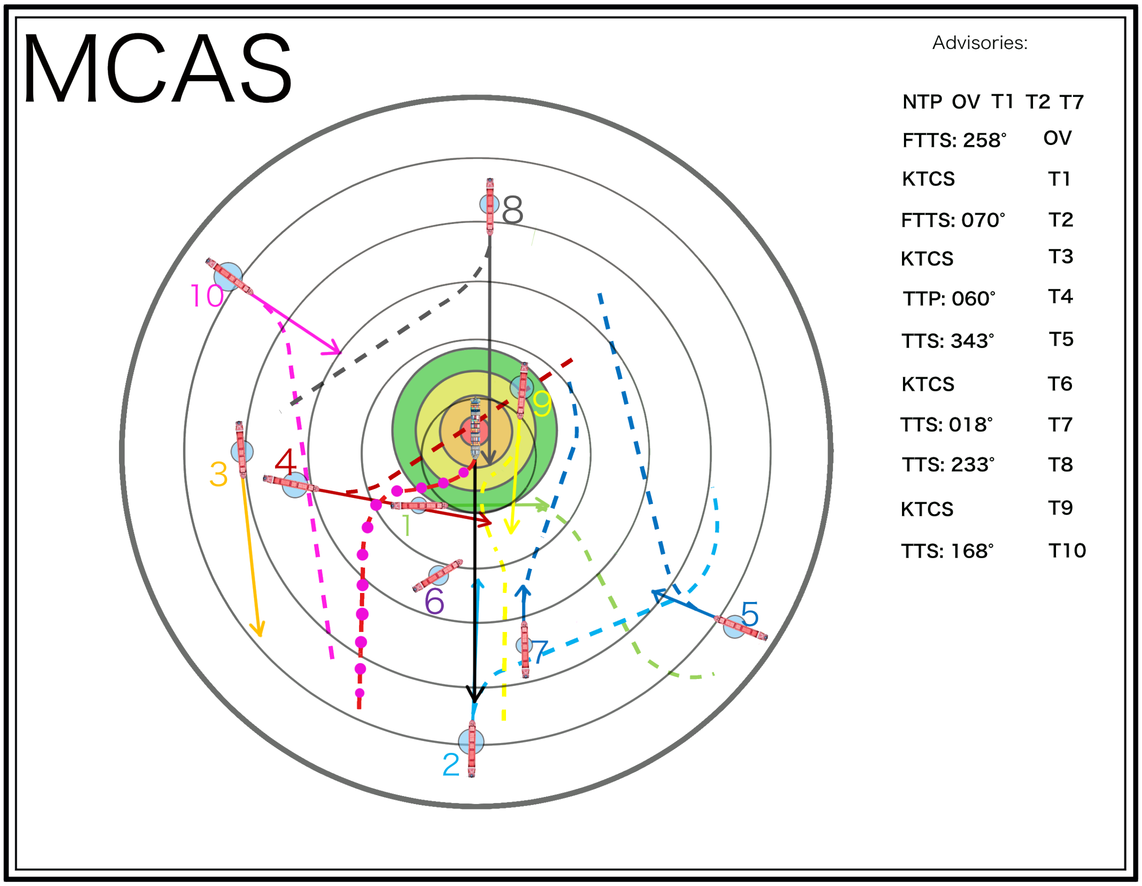


Figure 5.9 – MCAS overview of the optimal trajectory for the Scenario 5

With identical approach to conflict resolution, Figure 5.9 depicts advisories of the complex scenario 5. Own vehicle is again instructed to turn to starboard with full steering power. Target 1 is instructed to keep the course and speed until reaching the turning point at which it is to steer to starboard and avoid targets on its starboard side. Immediate turn to starboard is not safe due to the presence of the target 6. Target 2 has to make more significant turn and overtake vehicles on their starboard side, while targets 5 and 7 are also advised to turn to starboard in order to avoid upcoming traffic and there is enough sea space on the starboard side. Target 3 keeps the course and speed, while targets 8 and 10 are advised to turn to starboard and give greater distance for the critical target 9 that has to do a more complex turn to starboard at a defined time and then turn back to port in order to avoid target 6.

Now it is possible to compare results of scenarios 2 and 5 with intent sharing and without intent sharing where all targets were considered noncooperative. Passage plan sharing is not compared as its main goal is to ensure that navigators do not find themselves in scenarios 2 and 5. Passage plan sharing is the least computationally expensive approach and guarantees that collision risk would be detected early so that the complex situations are avoided. This would particularly work well in confined places, such are Singapore straits or Dover, where adjustment in speed can be advised to vehicles so that they arrive at critical points with lower number of target vehicles.

Table 5.5 – CPA comparison of simulations with no intent sharing and simulations with intent sharing for scenarios 2 and 5

Target vehicle	*CPA – no intent NM*	*CPA – intent NM*
1	0.38	0.52
2	1.94	2.33
3	1.3	1.48
4	2.01	4.21
5	0.96	2.85
6	0.82	0.82
7	2.24	2.24
8	2.32	3.41
9	0.46	0.48
10	3.37	1.83
dRoute	4.24	2.78
tRoute	1:31:46	0:54:42

Target vehicle	*CPA – no intent NM*	*CPA – intent NM*
1	0.38	0.5
2	4.4	4.8
3	1.47	1.84
4	0.33	0.64
5	3.6	4.92
6	1.49	1.44
7	2,8	3.03
8	3.2	2.08
9	0.63	0.8
10	5	2.78
dRoute	4.8	1.38
tRoute	0:50:42	0:19:43

As evident from simulation results and as depicted by Table 5.5 and figures 5.10 and 5.11, it is evident that intent sharing and multi-target coordination outperforms an egocentric approach where own vehicle assumes fixed trajectories of other vehicles and generates optimal trajectory. In both scenarios it is noticeable that for almost all targets CPA is higher when intent information was shared. For the targets 10 (of the no intent sharing experiment), 8 and 10 (of the shared intent information experiment) a lower CPA number is noticed, but even that lower number is well above the SAFE risk zone of the own vehicle, so the general collision situation is improved with the intent awareness.

Also, it is necessary to mention that the intent sharing experiments resulted with lower deviations from the planned route (2.78 NM vs 4.24 NM for the Scenario 2, and 1.38 NM vs 4.8 NM for the Scenario 5). Closely related to the distance from the planned route is also information on how long it took for the own vehicle to return to the original track, and it again showed us that intent sharing experiments outperformed non-intent sharing experiments (0:54:42 vs 1:31:46 for the Scenario 2, and 0:19:43 vs 0:50:42 for the Scenario 5).

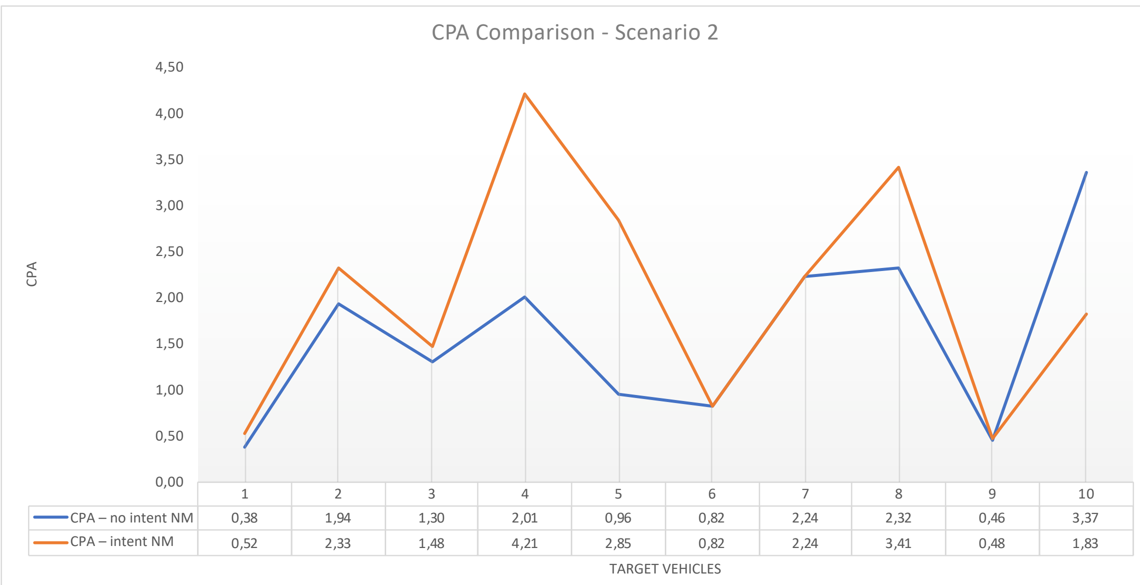


Figure 5.10 – CPA comparison of simulations with no intent sharing and simulations with intent sharing for Scenario 2

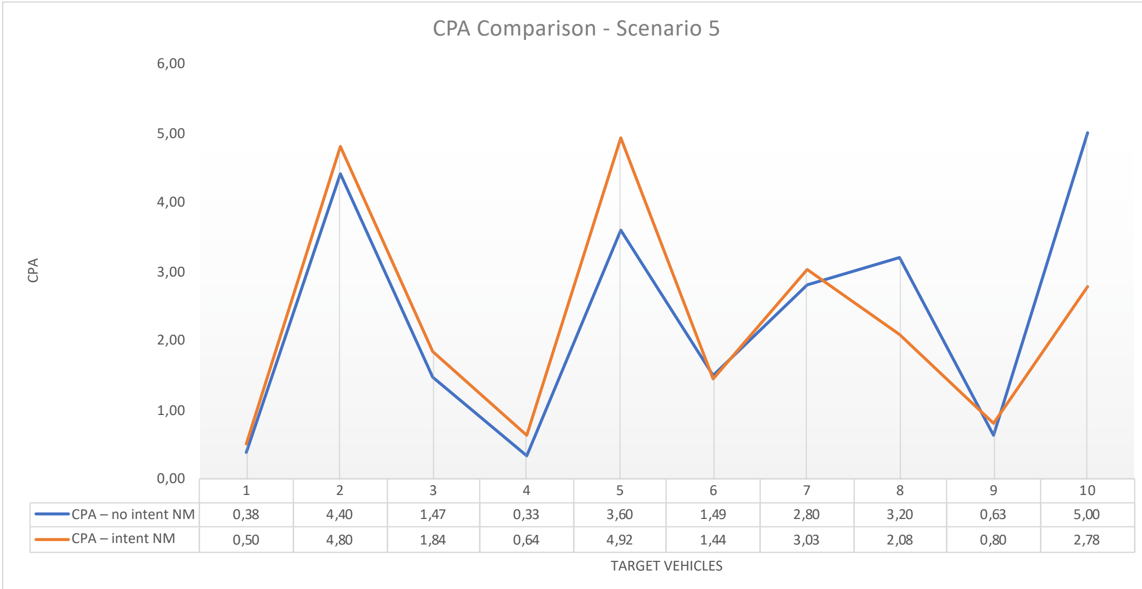


Figure 5.11 – CPA comparison of simulations with no intent sharing and simulations with intent sharing for Scenario 5

Finally, it is possible to compare feasibility performance of all algorithms utilized up to now. Hazard alerting algorithm is also included, even though it is described in the next section. However, hazard algorithm does not contribute to computational complexity as heavy as COLREGs classification algorithm that has been proven to be the most computationally expensive.

Figure 5.12 delivers an overview of computational performance of collision avoidance algorithm for multiple targets, while Figure 5.13 describes the computational performance of all algorithms in the MCAS system working concurrently in order to provide advisories. Complexity increases almost linearly with added number of targets. This points towards the fact that there could be up to approximately 60 targets before the system would not be able to generate trajectories and issue advisories within the sample time of 30 seconds. However, even before target number reach this point, trajectory generation and collision resolution could be allocated to another equipped vehicle for the traffic situation closer to that vehicle and only take inputs from the selected equipped vehicle.

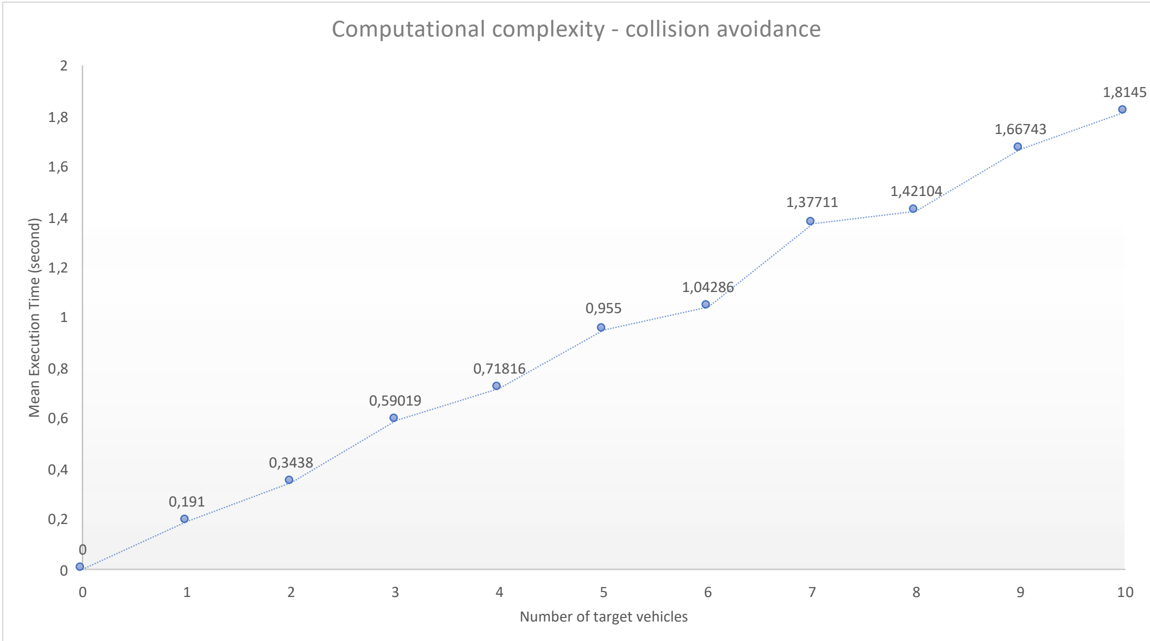
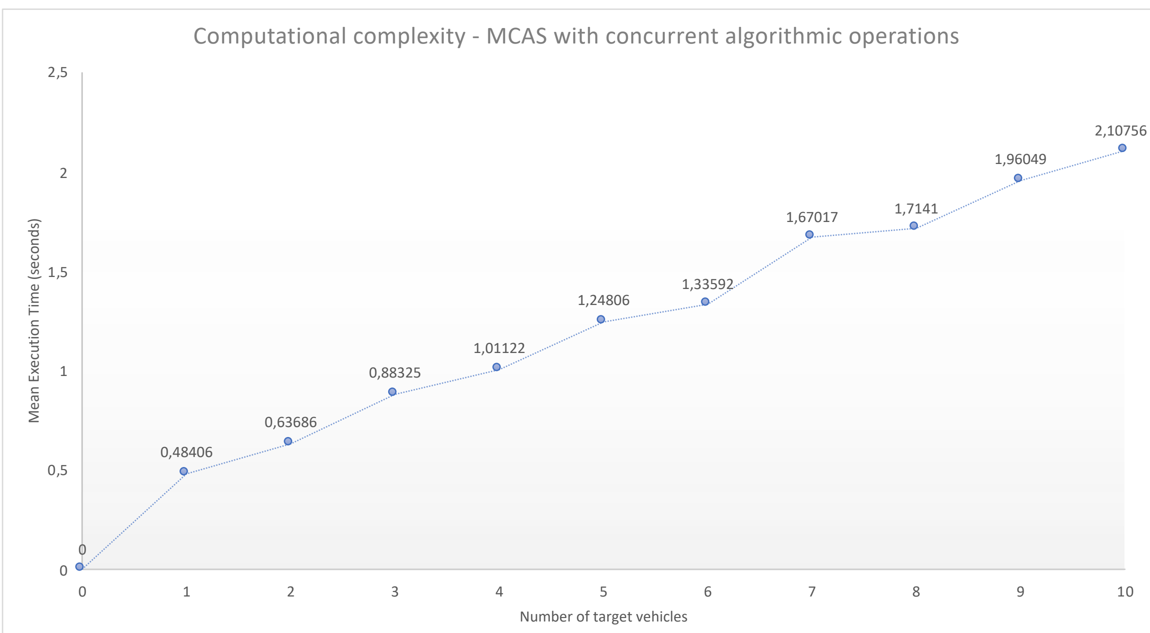


Figure 5.12 – Mean Execution Time for collision avoidance algorithm – isolated performance of only the collision avoidance algorithm



5.13 – Mean Execution Time for all algorithms of the MCAS system working concurrently on one machine (Motion control, Sensor data filter, COLREGs classification, collision avoidance and MHAS)

Environmental loads do not add to complexity as they are computed offline and updated with new experiences. This is the benefit of lookup tables and interpolation of data. That is the reason why decentralization of computation is necessary. This can be achieved by allocating as much as possible to offline computing, fusing data streams, especially reward functions, and utilizing parallel computing.

5.5 Discussion

Dynamic collision avoidance was further investigated in this chapter. Even though egocentric collision avoidance was successful in resolving close-quarter situations, utilization and uncertainty challenges remained unsolved. Possibility of reducing uncertainty with intent information sharing and designing a system that would resolve collision avoidance for multiple equipped and unequipped targets in the vicinity were investigated.

Considering advances in communication systems available on commercial sea surface vehicles, intent-aware collision avoidance is feasible solution to reduce uncertainties. A system of sharing planned passage information is proposed in order to enhance trajectory generations and to have an option of an early collision detection. In that way it is possible to minimize the risk of language barriers and make early avoidance maneuvers, which is highly beneficial in confined and busy areas. Experiment results of intent-aware navigation showed that the system is able to detect collision situations early and that it is possible to have precise information about the position, CPA and TCPA for each target, which is updated within the sampling frequency. Early collision risk detection significantly reduces intent uncertainties and aids to organize traffic flows free of conflicts within confined and busy waterways.

Coordinated collision avoidance is protocol proposed to aid trajectory optimization and risk reduction. Instead of pairwise resolution, a holistic approach is proposed, where traffic situation is monitored within ownship field of view and resolutions made for each equipped target. By sharing resolution advisories, it was possible to establish concise communication with other vehicles in the vicinity and clearly state own intent. Due to the regulatory framework of collision avoidance, resolution advisories are advisories only with an aim of assisting human navigators to make informed decisions on time. Navigators would get both graphical and audio-visual information about the situation and proposed actions to take. An idea of forced cooperation is introduced for the situations where it is clear that there are uncooperative targets and that ownship action is required regardless of the collision avoidance rules.

Experiments results have showed that resolution advisories and intent sharing offer significant reduction in levels of uncertainty, but also improve collision resolutions by enforcing earlier actions to avoid collision, by generating trajectories that benefit all targets in the vicinity, that vehicles are passing with larger CPA values and that own vehicle is deviating less from the planned course. Finally, it is evident from experiment results that even though proposed approach is computationally expensive, it is still possible to resolve close quarter situations. However, improvements in performance could be achieved by exploiting parallel computing power available on commercial sea surface vehicles.

Chapter 6

Maritime hazard alerting with Hidden Markov Models

In this chapter an approach to navigational hazard avoidance alerting system based on Hidden Markov Models is described. Alerting navigators should be initialed when there is a deviation from the action plan, or when the imminent action is required to avoid hazards. Also, it is important that alert should not trigger when the action is in execution. Therefore, if the vehicle is in the process of avoiding collision, alerting about the danger of collision is nuisance and can remove focus from the main task. Hazard alerting is an additional layer to the existing and mandatory equipment onboard commercial sea surface vehicles, so the focus of this study is only on alerting to hazards and taking the dynamic model of vehicle's motion into account. Unlike the MCAS collision avoidance range of 12 NM, Hazard alerting is used to track ranges up to 24 NM in order to allow for navigators to act early without the aid of the decision support system.

6.1 Development of alerting processes

Automatic alerting systems are not novelty in the maritime industry. In order to effectively control operations with minimum human interaction, automation systems were developed that can optimally control processes and protect equipment. This is especially evident through cargo or machinery automation systems, where complexity reached saturation level of human cognitive capacities and requires machines to mine sensing data sets and control processes effectively. An alerting system is capable to trigger an alarm when a certain sensor is out of range, but also perform complex situation monitoring based on wide-ranging data to alert, or even guide, operators by incorporating sophisticated algorithms or machine learning models.

Looking through the prism of stochastic decision processes, the design of an alerting system should take a collection of state measurement as input for the logic or mathematical principle that will decide to alert or not. If the scope of the system is beyond alerting, the design margins will maintain a certain level of automation or guidance for the operator. Commonly, logic is developed by a designer, after which its performance is evaluated and modified accordingly to fit the process better. By taking this approach, designer attempts to predict all possible scenarios in advance. However, more complex approaches use various probabilistic techniques to predict future state trajectories and alert the operator if deemed necessary. Some of the approaches take future uncertainties into account when making alerting decisions. In order to avoid unnecessary alerting, it is imperative to build a strong knowledge base so that alerting requirements are clear. The goal is to model a navigational hazard avoidance alerting system that will take prediction uncertainties into account.

6.1.1 Modeling the navigational hazard alerting process

Figure 6.1 delivers overview of navigational hazard alerting process based on probabilistic methods. Proposed model is based on Kuchar's Unified Methodology for the Evaluation of Hazard Alerting Systems [1995] and is modified to fit the navigational hazard alerting process.

Maritime Hazard Alerting System (MHAS) can be defined as a dynamic system with a state s_t. The MHAS system receives observations o_t from the sensing equipment and delivers actions (in this case the action is the warning sound or message) as an input to the operator a_t. These inputs could be any discrete alert values, or even continuous resolution advisories that the MHAS might provide. The navigator and navigation control are both dynamic systems with corresponding states s_n and s_{nse}.

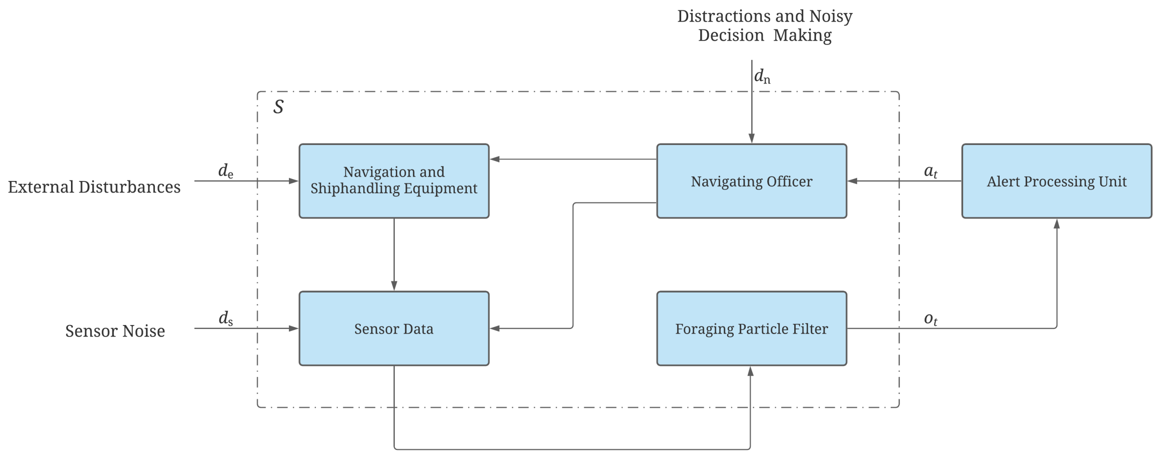


Figure 6.1 – Navigational Hazard Alerting Process

A process state, entailing both operator and navigating process/control, is also a dynamic system with an overall state vector S. The operator makes an observation o_t of the navigating process and performs actions a_t to influence and control the navigating process. Sensors have capability to provide information about the operator as well, depending on the interactivity of the navigating process. Considering that the scope of this study is not autonomous navigation, operators are ultimately controlling the navigating process and are essential part of the process. This is the reason why both navigating process and operator could be considered as the controlled element by the MHAS. It is therefore necessary to bear in mind high level of interactivity when designing the alerting system.

Due to limitations of the presented model, there are possibilities of state dynamics different from what is expected from the model. It is reasonable to expect errors due to sensor noises, distractions and noisy decision making (such are lack of focus, communication disturbances, errors in messaging, lack of knowledge or experience, personal factors, etc.) or external disturbances (such are weather influences, false readings or failure of the equipment). Such errors are presented by corresponding vectors d_{s}, d_{n}, d_{e}, having the same size as the state they represent. Also, it is necessary to take into account that actions of the alerting system could be real-valued vectors, but often are limited to a small set of possibilities and as operators prefer to be alerted in case of rare interventions only, it is necessary to define nominal and deferred action. Nominal action offers an alert immediately, while deferred action a_0 feeds the logic with the information that some state is out of designed range but holds of the actual alerting until the system confirms it with some other value or satisfies time deferral.

Considering that alerting process is defined as probabilistic, knowledge of previous states does not warrant exact prediction of future states. However, if state s is considered at the current time t_c, it is possible to predict future state $s_i, s_j, \ldots, s_n$ with some probability distribution over the states in S, bearing in mind the action trajectory τ_a of an action a_{t_c} at the current time. The future states depend only upon the current state, while previous state before the current one bears no weight of dependence. In other words, the state s exhibits the Markov property [Russel and Norvig, 2003]. It is, therefore, possible to express state dynamics as distribution with transition function f_T conditioned by the initial state and control trajectory:

$$\mathrm{s}(t_i + \Delta t) = f_T(s(t_i, \tau_a)). \tag{6.1}$$

Transition from the current state to some other future state is governed by internal dynamics and stipulated rules. The purpose of alerting system is to prevent incidents and ensure that transition is done flawlessly by influencing the path state s takes in the state space S. However, even though there are guards to prevent incidents, incident probabilities exist along the transition trajectories. In a case of two-dimensional state space, as depicted by Figure 6.2, transition trajectory can lead to two different future states s_1, or s_2. Which path the state will take depends on transition function f_T (transition probability). At the same time, it is necessary to keep in mind that probabilities of incidents exist for each path taken. In the example depicted by Figure 6.2 both τ_1 and τ_2 could be considered safe trajectories to take; however, it is just to assume that trajectory τ_2 bears higher risk as it is closer to the island and there is a higher possibility that something unplanned can happen with severer consequences.

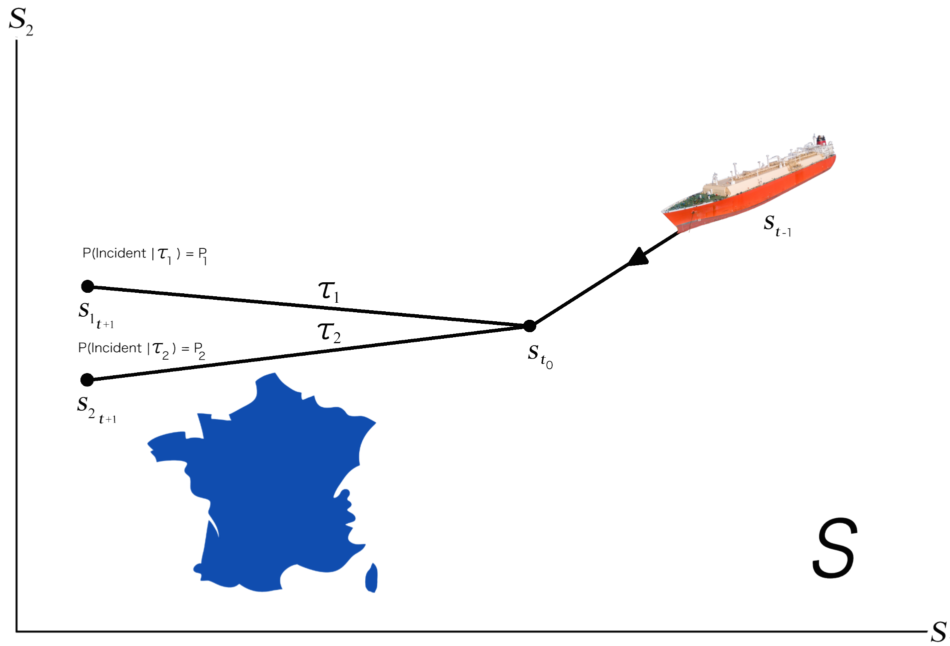


Figure 6.2 – Transition trajectory dynamics

The proposed system has to include a model that specifies these probabilities. The most common approach is to define hazardous area within the state space *S* that assumes the probability of 1 if the trajectory enters the hazardous region, or 0 if it doesn't. Figure 6.3 illustrates this kind of incident model.

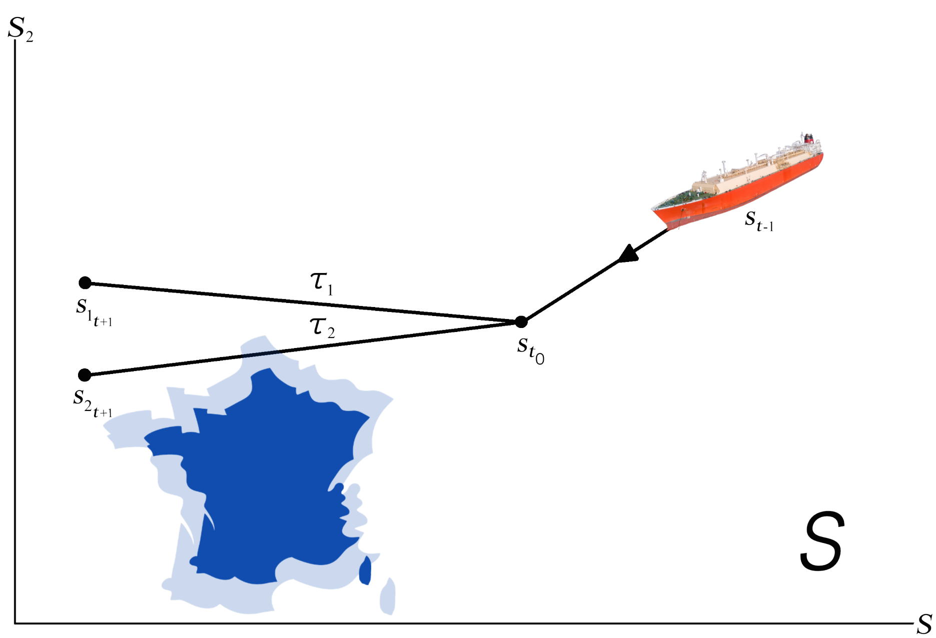


Figure 6.3 – Hazardous area and trajectory dynamics

In the case depicted by Figure 6.3 it is visible that the island also has a shallow water area and that this vessel would run aground if it was taking trajectory 2; therefore the probability of incident with the trajectory 2 becomes 1 and has to be avoided by an early action, but also hazard alerting.

Outside of the simulated space it is hard to expect that the system will have perfect knowledge or complete observability of the current state. Understandably, less uncertainty the alerting system has, more accurate predictions about future state it can produce and deliver more effective decisions. The most efficient way of reducing uncertainties is taking additional observations to enforce learning.

As it is usual with system dynamics, a particular event can affect further development and improvement. A simple example is the infamous case of Exxon Valdez where one of the key components identified in the incident investigation was that the navigator never noticed that the autopilot was steering the vessel when orders were given to the helmsman. Most of the modern vessels have an audio and visual alarm informing operators that the steering is in autopilot mode if the wheel is moved off center. This is a very effective example of reducing uncertainties. Previous chapters depicted how it was possible to decrease uncertainties and increase computational effectiveness. However, often this approach requires a consensus of professionals in determining thresholds for decision making. Alerting is not an exemption, so in order to have an intuitive alerting, professional organizations would need to agree on alerting thresholds for various situations. In this thesis some of the possibilities are offered, but the method is emphasized rather than metrics.

In order to avoid unnecessary alerting, an alerting system has to weigh the cost and utility of each option it might face. For example, if a large vessel is approaching seaport, most likely there will be mandatory pilotage and towage service in force. If the alerting system is unaware of the mode it is operating, navigators might experience unnecessary alerting and collision avoidance resolutions for tugs and pilot boats. However, at the same time there could be a viable hazard in the surrounding that goes unnoticed due to the alarm clutter. One of the possible solutions is to engage maneuvering or underway mode for the alerting system. Even though uncertainty can be reduced by installing an underway and maneuvering switch, there is still a possibility that the human operator fails to engage appropriate mode. A documentary control and appropriate training could further reduce this uncertainty.

In Figure 6.4 a situation in which the vessel is approaching an island is depicted. The vessel has planned and verified the route and it is clear that the passage is safe if turns are done as per the voyage plan. However, if the vessel steers off course and the turn is done late, there is an imminent danger of running aground. If the operator is aware of the island and doing the turn on time, alerting would be unnecessary. Alerting too early can cause complacency in the long run. However, further observations would reduce uncertainty of the operator's awareness. Therefore, the alert has to be raised once reaching the point (a) to ensure the operator executes the turn on time. Considering that the turning radius depends not only on the static characteristics of the vessel, but also on the dynamic influencing factors, such are speed, under-keel clearance (UKC), currents, winds, etc., it is imperative that the point (a) is determined appropriately. There is a clear challenge in finding balance between acting early on uncertain information and delaying the alerting until knowledge about operators' intention, as well as system's operating mode is reinforced in learning. Finding the point (a) is a dynamic challenge, so investigating probabilistic system dynamics is another objective of this chapter.

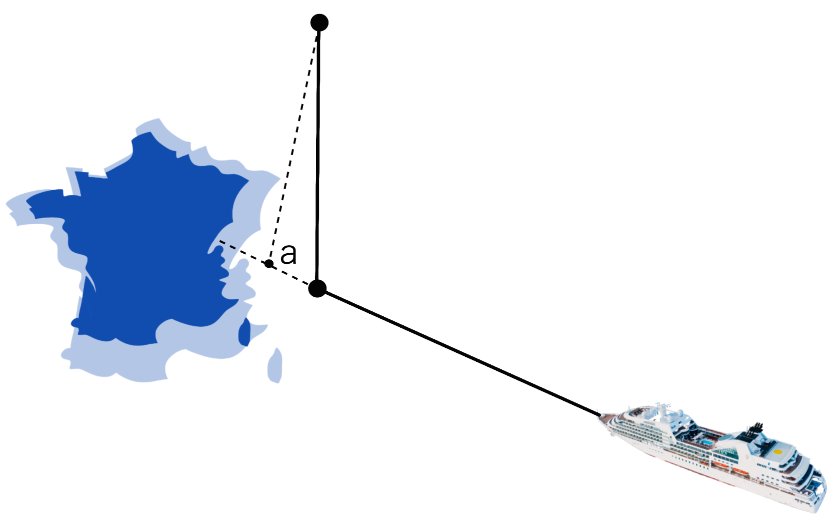


Figure 6.4 – Finding the balance between safe operations and nuisance alerts

The main goal of an effective alerting system is avoiding genuine hazards and associated incidents. As depicted in Figure 6.2, an alerting system has a task of affecting the path s in the space S. Each of the path possibilities (path becoming trajectory in the time domain) has a different incident probability. These incident probabilities should be specified for each possible path at any possible s. As mentioned in Kuchar [1995], the most common approach to preventing scattered hazards and expensive computing is by defining a region or hazard space at which probability for incident will be equal to 1 if the state trajectory enters the hazardous area. This is similar to shallow water contour on ECDIS. If look-ahead vector is set properly, it will alarm the operator that the vessel has a trajectory of crossing the shallow water contour and entering the hazardous area. As objective of this study is to design a last-minute hazard avoidance system, it is necessary to find alerting point at which the operator has to act to avoid incidents.

As stated in the expression (6.1), hazard avoidance has uncertain dynamics and observance of previous states does not guarantee prediction of the future states. The size of the action space largely depends on the definition of utilities and constraints, and modeling of uncertainties. As alerting would be required only when it is necessary, in most of the cases, intuition would be to delay the alert, which can be represented by the action a_0. Therefore, if for example a vessel is approaching a hazard, there is a large set of actions available; however, the realistic option would be to steer around the hazard with minimum impact on the planned path. If the system would propose crash astern maneuver, or turning 180 degrees, it would definitely help to avoid the hazard, but it would not be feasible for the technical and commercial operation of the vessel. It is, therefore, important to define constraints (maneuvering characteristics of the vessel) so that the system engages on time and offers an appropriate resolution.

In Figure 6.5 it is noticeable how probabilistic uncertainties (such are erroneous sensor data) affect dynamic system predictions. Even though $s(t_i)$ is fully observable, $s(t_i + \Delta t)$ is defined by a distribution of possible states due to the prevailing uncertainties. In the depicted process, there is a probability of trajectory reaching the hazard zone and avoiding hazard. Controlling the outcome of probabilistic distribution is of interest with the ultimate objective of hazard free navigation.

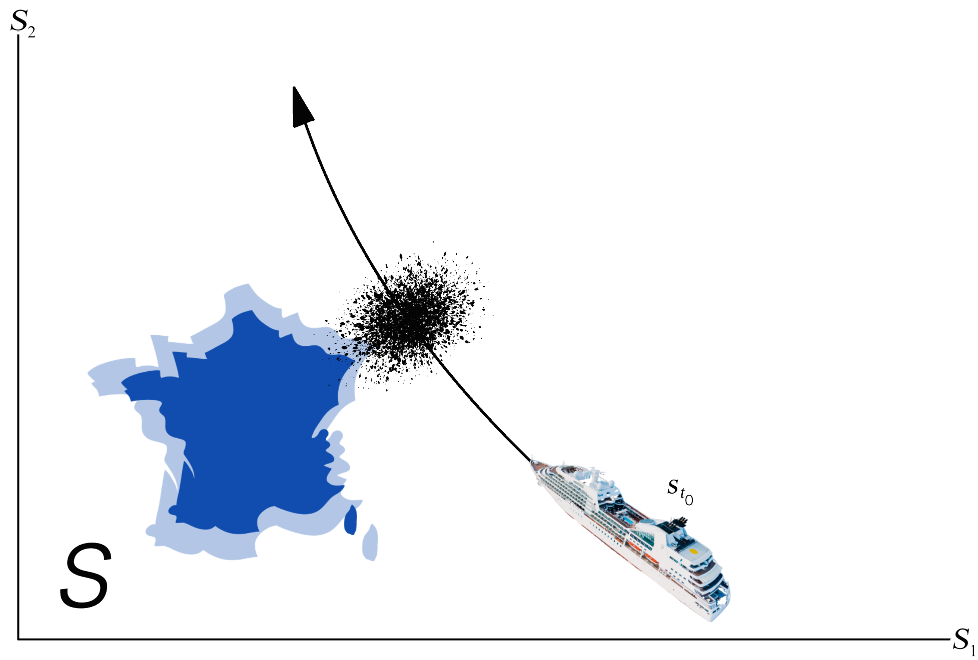


Figure 6.5 – Uncertainty of probabilistic predictions

Having a full observability of the state space is a rare occurrence and usually reserved for only the simplest systems. Once the uncertainty exists the best action is a new and/or additional observation. For example, at the beginning of a new voyage, voyage planning has to be done. If a master or a deck officer never visited a planned port, they might be uncertain of the local requirements. If there is a question on how to book a pilot, navigator can simply take additional observations by going through admiralty publications, visit web pages of the piloting service or consult experienced colleagues. This will significantly reduce the uncertainty navigators experienced at the beginning of the voyage.

If the situation from Figure 6.6 is analyzed, coordinates of an intruding vehicle provides information about the location of the vehicle, but without additional observations, such are azimuth, range, speed through the water, or heading, it would not be possible to determine if the collision potential exists or not. It is possible to further reduce uncertainty of the intruding vehicle by verifying AIS data, calculating relative coordinates, use appropriate sensor filters, etc. In reality it is possible also to experience a total sensor failure, which will add to the uncertainty unless we are able to compensate with a spare set of sensors or additional observations. If any of the sensors is malfunctioning or there is a sensor error significantly obscuring the view, own state space will be defined as partially observable [Russel and Norvig, 2003]. Considering that sensors are used to conduct observations (even the sight of operator could be considered a sensor with a particular sensor error distribution), relationship between the current state and observation of the state can be defined as sensor function, where the function O represents probability distribution over the space of possible measurements O [Russel and Norvig, 2003]:

$$o(t_i) = O\big(s(t_i)\big) \tag{6.2}$$

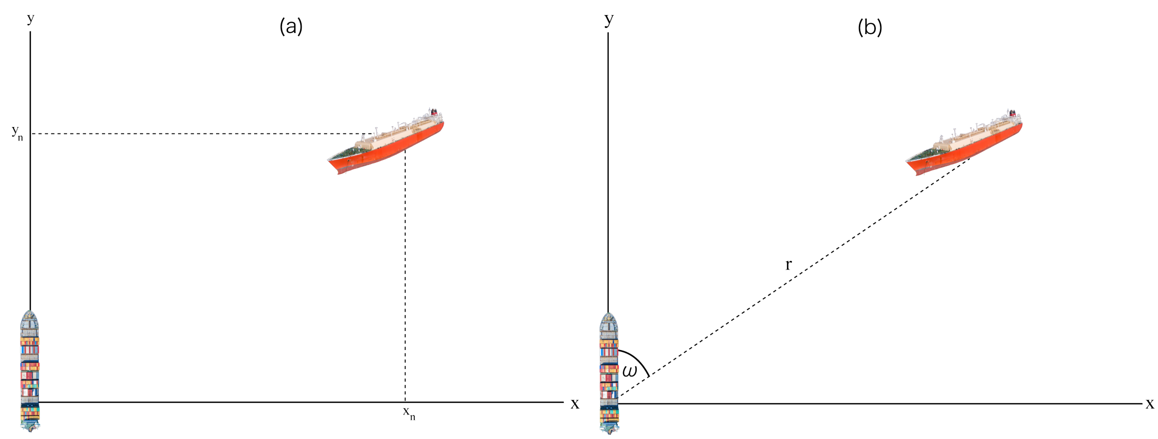


Figure 6.6 – Environment verification: a) target's state space described by position in local reference frame; b) azimuth and range as an observation of a target

6.1.2 Erroneous performance analysis

Even though operators of various system processes would appreciate only relevant and successful alerts, both machine and human controlled systems do experience unwanted alert errors. This may be caused by poor communication, external factors, ambiguous parameters setup, faulty sensors, or similar.

The NHAS should be designed to avoid Nuisance Alerts (NA) and Unsuccessful alerts (UA), and raise only True Alerts (TA), while also recognizing instances of True Negatives (TN) when alert is not required. If there were no incident after the alert is raised, system would be interrupted for no apparent reason and this alert would be classified as Nuisance Alert [Yang and Kuchar, 1997], which can also be classified as false positive (FP). This can lead to diminished productivity of the human operator, increased computing load, or even an indirect incident (trying to avoid collision with safe target and colliding with another target as a consequence). Nuisance Alerts negatively influence alert system dynamics as they can lead to reduced trust and unconformity of an operator. When alert is late or absent and there is no time to avoid an incident, navigator would experience an Unsuccessful Alert [Yang and Kuchar, 1997], which could be classified as false negative (FN). This error can lead to significant consequences; therefore, probability of unsuccessful alert has to be reduced to minimum. Considering subjectivity of the human operators, nuisance alerts are difficult to define. In the ideal world only perfect alerting would exist without unsuccessful and nuisance alerts, but the goal of this section of research is to achieve system with the best balance of all alerting inaccuracies. The performance of the overall NHAS can be described using conditional probabilities taking in consideration both successful and erroneous alerts.

One of the questions raised when dealing with alerting system dynamics is when the alert should occur. This is something that can be resolved in the design stage and one of the options is that any time the state $\boldsymbol{s}$ reaches a pre-defined hazard threshold an alert occurs.

As with other binary classifiers, conditional probability relationship could be visualized by utilizing confusion matrix and System Precision-Recall Characteristic

(SPRC). The SPRC is constructed by connecting all precision-recall points of a monitored system. Recall is plotted on the x-axis, while precision is plotted on the y-axis. The curve is finalized by connecting adjacent precision-recall points with a straight line. Figure 6.7 is an example of SPRC for a pair of successful and unnecessary alerts. Their precision and recall points are calculated and the plotted as the SPRC curve. As much as the SPRC curve is bellied towards the upper right corner (1.0, 1.0) it is more precise and has higher skill in distinguishing between two distinctive groups (successful / unnecessary alert). In the example depicted below, a point on the SPRC curve is a plot of successful alert probability against the unnecessary alert probability at a given time. The ideal place for an alert to occur is in the top right corner.

Recall is calculated by dividing number of True alerts with a sum of numbers of True Alerts and Unnecessary Alerts ($TA/TA + UA$), while Precision is calculated by dividing number of True Alerts with a sum of numbers of True Alerts and Nuisance Alerts ($TA/TA + NA$).

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	True Alerts (TA)	Nuisance Alerts (NA)
	Alert Not raised	Unsuccessful Alerts (UA)	True Negatives (TN)

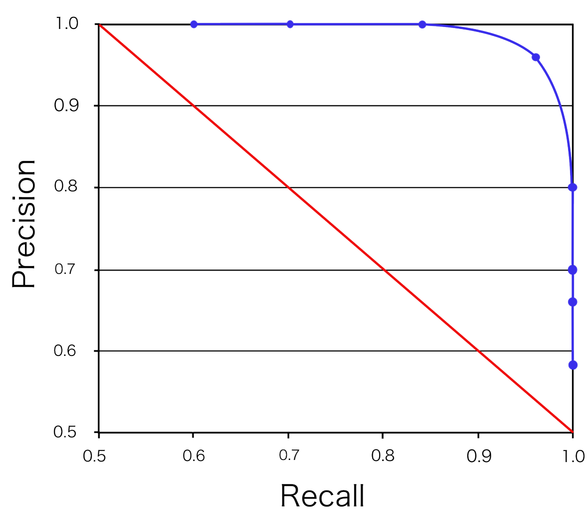


Figure 6.7 – Alert confusion matrix and SPRC example

6.1.3 Observability and approximations

As a potential target vehicle is approaching own vehicle, the alerting system could be reading that the target is maintaining safe distance from the own vehicle. However, at any moment, target could change its course and sail closer to own vessel presenting a hazard. This is an example of the random nature of maritime navigation. It is evident that knowing previous positions (states) of vessels becomes irrelevant once an intruding vessel changes its course and sails closer. Naturally, this example shows either unequipped vehicle, or uncooperative equipped vehicle due to unexpected situation, malfunction or human error. Due to various factors, but mostly because of sensor errors and inaccuracies, observability is restricted and ambiguous. With additional observations, goal is to reduce state uncertainty. Russell and Norvig [2003] define belief state updating with new observations as filtering. As filtering is computationally intensive, it is imperative to reduce ambiguity at design level as much as possible.

Once alerting thresholds have been defined in the design stage, alerting system is operating by constantly scanning for any event that would warrant the alert. If there are no alerts to display, the system is suggesting a deferral action. Determining thresholds is not a straightforward task. It is easy to decide that a certain fixed number will be a threshold value, but it will not satisfy the objective of reducing nuisance alerts. Probability threshold value will have to be adaptive to satisfy the system dynamics and partial observability of the proposed model. In Figure 6.4 a simplistic overview of the alerting system based on probabilistic measurements is depicted. In this example the threshold is set in advance and once the alerting system recognizes need for alerting, the alerting system issued the turn alert, as ownship did not turn on time, but rather late and reached the point a determined by the predictor.

As mentioned earlier in the chapter, there are various noises that affect observability of state s. These noises are not only related to the sensors used to assess the collision situation, but also behavior of operators, which are harder to model than sensor errors. In order to incorporate willingness to follow Collision Regulations (COLREGs), attentiveness of the navigator (setting up the bridge properly, weather conditions, fatigue level of the navigator, etc.) it is necessary to introduce another variable that can be named *Behavior*

Parametric Form (BPF). BPF is used to index a set of variables that have a discrete domain and persist over a period of time. All BPFs could be incorporated as set of transition functions (6.3) which are integrated in trajectories of predictor. As an example, in Figure 6.8 a scenario with uncertain future states is illustrated, where in one mode a vehicle nominally tends to follow the planned route and ROT is actively monitored to slow down the turn on time, while in other mode a navigator loses his/her situational awareness and does not correct the turn on time resulting with running aground after exiting a buoyed channel.

$$s(t_i + \Delta t) = \begin{cases} T_1(s(t_i), \tau_a), & BPF_i = BPF_1 \\ T_2(s(t_i), \tau_a), & BPF_2 \\ T_n(s(t_i), \tau_a), & \dots BPF_n \end{cases} \tag{6.3}$$

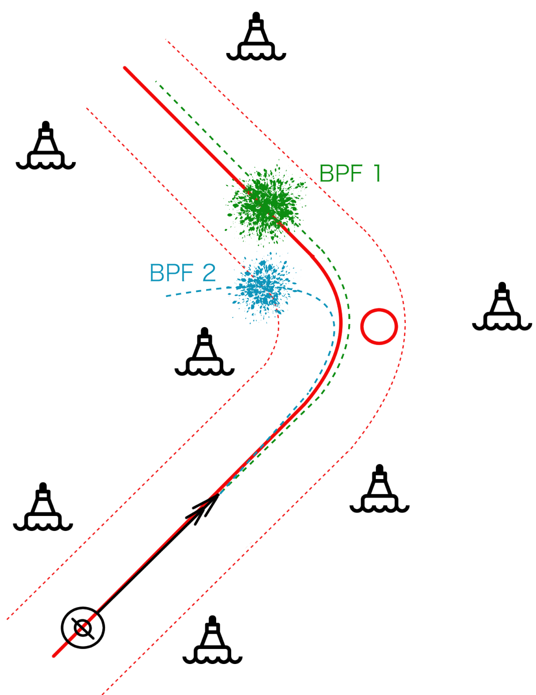


Figure 6.8 – Behavior Parametric Form (BPF) – example of two probabilistic outcomes

BPF variables are included in the state s description for each participating member. As mentioned in previous chapters, equipped members that share their intent will allow for the ownship to have significantly improved observability of targets' states, while with unequipped targets we face partial observability. Some of the BPF variables are directly measurable with intent sharing vehicles (auto-pilot mode, weather conditions, safety cone settings, etc.), while others are difficult to measure directly (attentiveness and fatigue, or proper bridge preparation). It is still possible to infer unmeasurable BPF variables through estimation and influence on the transition function, similarly as when plotting a target on the RADAR plotting sheet and determining target's heading, speed, course, CPA, TCPA, etc.

Once all the parameters have been defined, an alerting system will update itself at each consequent belief state. The most accurate prediction will be at the current state, as any errors brought forward from the current state to predicted states would be larger. An example of this mechanism is depicted in Figure 6.9, which confirms necessity of addressing uncertainties early at the design stage. However, it is necessary to keeš in mind that variance plays a significant role. As the variance is small enough, errors and uncertainties will be easier to control and predict. Larger variances could lead to sparse belief states with misleading alerting and/or guidance results. Therefore, when using a source for updating a belief state, it is important to select sources with tighter data sets, then to select datasets with broad distributions. In this sense, the alerting system would update a belief state by omitting uncertain variables while focusing on information with known properties. Therefore, as far as possible input data has to pass the filtration process and uncertainty reduced with intent sharing.

There are several approaches to handling alerts. After defining and updating belief states, it would be possible to follow the path policy and once the safest path is computed (the path with the maximum allowable probability of an incident where minimum safety level triggers the alarm), ownship would simply follow that path without updating or further planning [Winder and Kuchar, 1999; Samanant et al., 2000]. Incident probabilities are calculated at the time of alert and the path is maintained with that probability as given. Another possibility is to calculate safe path similarly as now, but after some fixed time to allow for the alerting system to update its belief state and compute fresh probabilities. After these probabilities have been updated, a new set of safe paths is proposed until the hazard is clear, incident avoided, and alert suspended. Another upgrade to the alerting logic would be that future belief states are considered when making a decision. This will allow for more informative decision-making; however, it is necessary to find a right balance between computational price and horizon of opportunity. By maintaining a finite horizon, the logic would gather more information with higher precision and less computational expense. Developed path planning and trajectory development is utilized for collision avoidance to determine alerting requirements and alert timings.

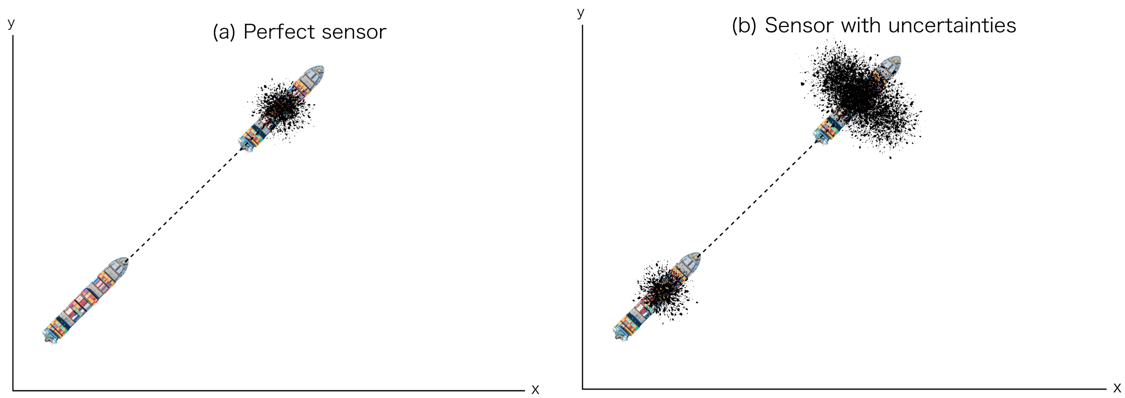


Figure 6.9 – The effect of reduced uncertainties at the present state

6.2 Probabilistic alerting with Q-learning

Unlike the aeronautical sector, maritime transportation never had a separate system for traffic alerting and resolution. Marine RADAR and later ARPA became crucial source of information for collision avoidance. Today, with many other electronic devices available on the bridge, navigators have an overwhelming resource of data to assist them in decision-making. However, incidents still happen and one of the tools that could reduce frequency of incidents could be a "last minute" alerting and collision avoidance system that assists navigators in finding the safest waters during times of increased pressure and congestion. On the other hand, air transportation had an intensive development of various software solutions in the last four decades. From Precision Runway Monitor (PRM) systems where the main task was to separate aircrafts on the final approach, where unnecessary alerts were not considered [Shank, 1994], over the simple-trajectory based alerting with incident proximity criteria [Yang and Kuchar, 2002] all the way until the complex trajectory-based systems where dynamic programming was explored in resolving advisories [Yang and Kuchar, 2002].

Vehicle handling, maneuvering and collision avoidance have been presented in earlier chapters, while the alerting process is the remaining intrinsic part of the hazard and collision avoiding process. Hazard avoidance includes also avoiding stationary objects, buoys and shallow waters and the main goal is to keep the vehicle within the planned route while ensuring that any deviation from the route is safe enough and will not result in near misses or incidents. Many of the commercial vehicles are equipped with ECDIS stations and are able to utilize safety cones, which would, if set up properly, alert navigators of the upcoming hazards. However, this setup will alert anything that is considered hazard and the main issue is the number of alerting navigators get while using this feature. That is the reason the author of this thesis noticed that many navigators tend to switch it off or reduce the distance of the safety cone reach in order to avoid visual and audio alerts. It does not help that each ECDIS station has to be reset individually, so the level of frustration often is very high in the moments of increased stress and pressure due to heavy traffic situations. Therefore, focus is on the last-minute alerting which will prompts users to take actions immediately in order to avoid hazards. It works with already developed dynamic collision avoidance predictors that take into account external disturbances with a modification to take stationary hazards into account, as well as to introduce the third dimension of the vehicle model (vehicle’s draught and available water depth) to avoid shallow waters. Safety zones remain the same as for the collision avoidance algorithm.

In this thesis approach to hazard alerting is separation of dynamic and static hazards. Even though a single algorithm that takes both dynamic and static hazards is designed, approach to conflict resolution is different, so it is necessary to depict reasons for different rewards and alerts. When considering dynamic object, the alerting system should alert when the risk of collision is recognized and when the action has to be taken in order to avoid collision. The former requirement is fulfilled by utilization of ARPA. In proposed algorithm, information from the COLREGs classification algorithm is utilized to determine the risk of collision. However, for the former requirement the system should recognize the time when immediate maneuvering is required to avoid hazards and issue alerts. In the case of static objects and shallow waters, alerting can have two modes: a) alerting when vehicle is not maneuvering as planned (reaching wheel-over position, but no reaction from a navigator), and b) alerting when immediate maneuvering is required to avoid hazards.

Most of the commercial vehicles are obliged to be equipped with various equipment that can aid in alerting to avoid both dynamic and stationary hazards. For collision avoidance both ARPA and AIS can issue alerts about dangerous targets. When approaching shallow water, it is possible to utilize ECHO sounders or ECDIS safety cone to inform agent on time that ownship is approaching hazardous areas. ECDIS safety cone is also effectively used to inform navigators of stationary hazards. However, these alerts often over-alert when navigating close to the hazards. In order to avoid nuisance alerts, approach is to design alerting system that alerts when the system determines action is really required. So, MHAS is working in combination with existing systems and can serve as the last instance of alerting when imminent danger exists, and imminent maneuvering is required to avoid hazards.

Lean approach is utilized again, so required state space includes only the essential state space members that would be sufficient to make alerting decisions. Own state space will consider the situation where alerting is happening for the last-moment situations only. However, any adjustments to the state space could easily be done in the design stage of the finalized product for the industry. Also, it is important to adjust alerting for a specific type of a sea surface vehicle. When determining the time stamp when immediate maneuvering is necessary to avoid collision, shallow water or stationary object, maneuvering characteristics of a vehicle will play an essential role. Smaller and more agile vehicles will be able to react later than fully loaded VLCC. Therefore, integration with the vehicle motion models and algorithms is essential to have precise alerting. Broader type-specific approach could be established, but it would not be as precise as a vehicle-specific solution. In line with the above own state space is as follows:

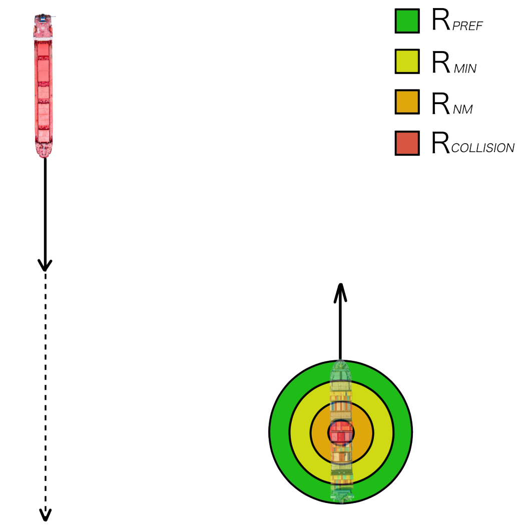
$$S = \begin{pmatrix} n_F,\ e_F, COG, SOG, CPA, TCPA, r_{NM} \\ H_F, ROT_A, CPA_E, TCPA_E, n_{MP},\ e_{MP}. \end{pmatrix} \tag{6.4}$$

Most of the members were described earlier, with an exemption of $CPA_E, TCPA_E$, which stand for Closest Point of Approach and Time to Closest Point of Approach for an equipped target, while n_{MP} and e_{MP} stand for North-East position of the time stamp in which maneuvering has to be initiated in order to avoid incidents. Conservative approach is maintained and this design is focused on avoiding near misses as well as collisions and allisions.

The action space is discrete and low-dimensional:

$$A_{ALERT} = \{a_0, a_1, a_2\} \tag{6.5}$$

When there is no need for alerting the system selects alarm deferral, or a_0. In case of dynamic obstacles, the fact that there is a higher risk of unequipped vehicles acting irrationally is taken into consideration, so an option of alerting earlier is utilized with a pre-alert a_1 warning where navigators get visual warning that the target is potentially dangerous showing that the predictor is showing a possibility of either near miss or incident. In the case of equipped vehicle, pre-alert will still exist, but the system will trigger it only if the equipped vehicle is uncooperative and deviates from the shared intent. Finally, when the predictor determines time stamp requiring immediate maneuvering and own vehicle reaches that time, then imminent maneuvering alert a_2 is issued. The predictor utilizes vehicle motion model, which allows the alerting model to have dynamic input of the alerting timestamp depending of the vehicle's speed and external influences. The imminent maneuvering alert assumes that the vehicle will maneuver according to the selected mode in the vehicle motion model (precise, economy or emergency). The following figures depict various situations with a dynamic obstacle:



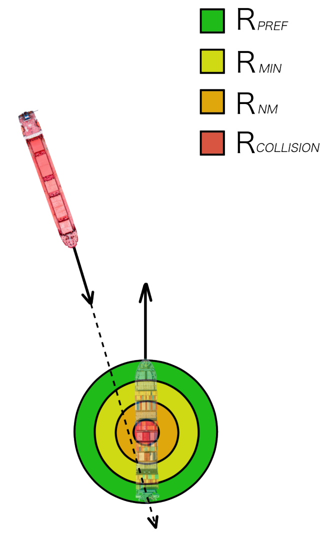
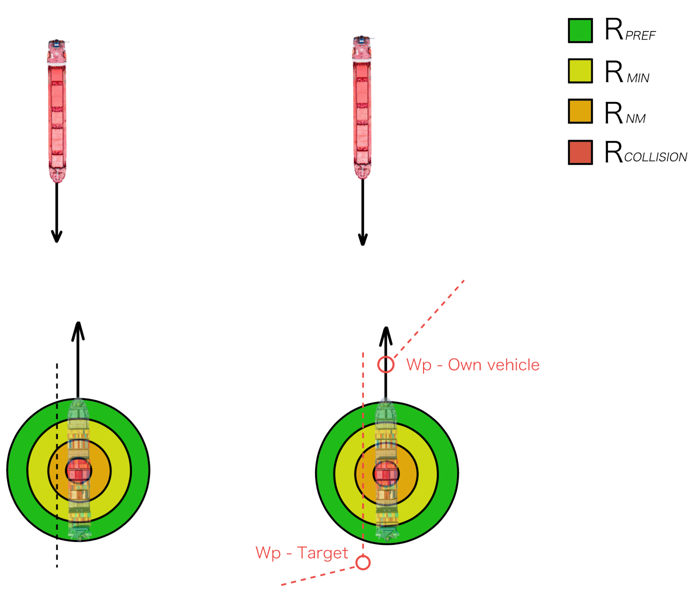


Figure 6.10 (a) – deferred alert equipped and/or unequipped target

Figure 6.10 (b) – pre-alert equipped and/or unequipped target



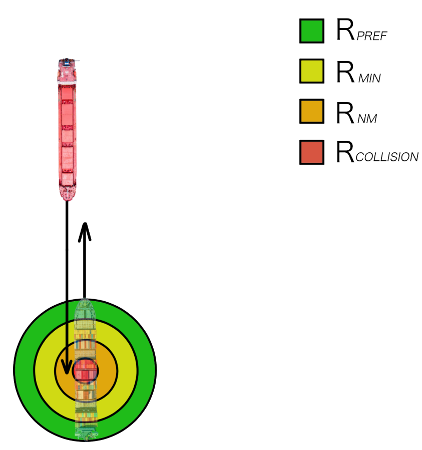


Figure 6.10 (c) – alert for unequipped vehicles; no alert for equipped vehicles

Figure 6.10 (d) – alert is active as vector is already in the R_{NM} area and predictor calculated that imminent action is required

In Figure 6.10 (a), it is noticeable that two vehicles are passing with minimal distance large enough that no alerting is necessary. In this case there is no difference between equipped or unequipped vehicles. In Figure 6.10 (b), pre-alert will signal on the ownship as vectors point out towards the NM safety zone and the intent is to keep the course and speed. There is still no need for the imminent alert as both vehicles can change their intent and alter course to increase distance on their own. In the example 6.10 (c) situation is depicted which makes alerting different for equipped and unequipped vehicles. For the unequipped vehicle uncertainty of an intent has to be assumed, so the system raises pre-alert, while in the case of equipped vehicle where intent is shared, alerting is deferred as the intent of both vehicles is known. However, if one of the vehicles deviates from the shared intent, pre-alert will be activated by the system prompting other vehicle about the deviation. Finally, the example 6.10 (d) shows the situation where own vehicle has reached an imminent alerting timestamp and has to maneuver immediately in order to avoid target vehicle entering the NM safety zone.

Processing fixed objects (for this purpose all objects that are not dynamic are defined as fixed, such are shallow water, land, buoys, etc.) is similar as alerting for dynamic objects; however, there are few differences in utilizing predictor. For fixed objects information available on ECDIS is used to determine position of shallow water, buoys and landmarks. This information is transferred to the occupancy grid developed for the predictor and the main task is then to find a position where maneuvering has to be done in order to avoid the shallow patches and other fixed objects. The logical question is raised: How would the system know what is a shallow water for the vessel? It is visible that human interaction is not completely removed with the DSS systems, so in this case it is necessary to ensure that navigator selects correct shallow water contours and to use accurate draughts. Assumption is made that voyages are prepared professionally with an ultimate goal of safe passage. Therefore, it is imperative to control the process of voyage planning and approval.

Figure 6.11 demonstrates how the system recognizes the need to alert the navigator. The predictor has to have the information from the vehicle motion algorithms in order to find the maneuvering point where immediate action is required. As mentioned earlier, agent chooses to be risk averse, so the system has to find the point that will not allow the shallow water to enter the near miss safety radius of the own vehicle. Therefore, pre-alert will occur once the system recognizes that the shallow water will enter the preferred radius and raise an imminent action alert when reaching a point after which immediate action is required to prevent shallow water contour entering the near miss radius. In this case that means that the predicted waypoint a_1 is refused by the system, but also a_2 because predictor's verification process determined that own vehicle's near miss safety radius is touching the shallow water, so the predicted alteration point a_3 is selected as the last turn and all hazard alerting is set up taking this information into consideration.

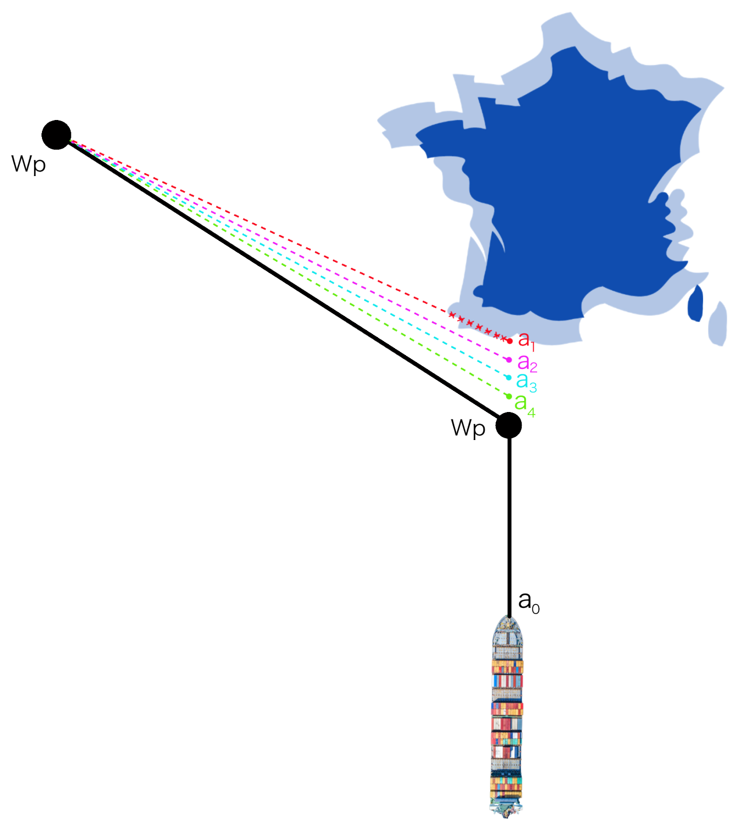


Figure 6.11 – Establishing alteration waypoint positions in order to find the appropriate alerting point

With safety zones and constraints depicted in this way, it is possible to define reward space for hazard alerting:

$$R(S_i, A_j) = \alpha_1 R_1(S_i, A_j) + \alpha_2 R_2(S_i, A_j) \tag{6.6}$$

where

$$R_1(S_i, A_j) = \begin{cases} 10, & \text{if} \quad CPA \leq \mathrm{R}_{NM} \text{ and } TCPA < 30 \text{ min} \\ 0, & \text{otherwise} \end{cases} \tag{6.7}$$

$$R_2(S_i, A_j) = \begin{cases} 10, & \text{if} \quad CPA \leq \mathrm{R}_{NM}, TCPA < 18 \text{ min, and } d_{route} \geq 1\ NM \\ 0, & \text{otherwise} \end{cases} \tag{6.8}$$

$$R_3(S_i, A_j) = \begin{cases} 100, & \text{if} \quad CPA \leq \mathrm{R}_{NM} \text{ and } n_F, e_F = n_{MP}, e_{MP} \\ 0, & \text{otherwise} \end{cases} \tag{6.9}$$

$$R_4(S_i, A_j) = \begin{cases} 100, & \text{if} \quad CPA \leq \mathrm{R}_{NM}, TCPA < 30 \text{ min and } dev_{equip} \\ 0, & \text{otherwise} \end{cases} \tag{6.10}$$

The reward 6.7 assigns reward points for an agent to issue a pre-alert when CPA to a dynamic object is lower than near-miss safety radius and when TCPA is less than 30 minutes away. As discussed earlier, approach is risk averse in assigning numbers to threshold values; however, the appropriate method is to conduct detailed research and organize administrative conferences to agree on threshold values for the whole maritime industry. The reward 6.8 is very similar to the first reward, with an addition of d_{route}, which is used for fixed objects and shallow waters. In this reward the agent is rewarded when pre-alerting for each occurrence when CPA is lower than near-miss safety radius, when TCPA is less than 18 minutes and when a vehicle is more than 1 NM away from the originally planned route. The reward 6.9 is used to reward instances where imminent alerting is required, while reward 6.10 is also used to reward imminent alerting when equipped vehicles are deviating from their shared intent. The maneuvering point MP is calculated by inherent predictor depending on the COG and SOG the vessel is making and adjusted for all external disturbances (heading of a vehicle is disregarded and only COG used). Once a vehicle reaches the MP position (n_{MP}, e_{MP}) and has CPA lower than near-miss radius, imminent maneuvering alert would be raised in order for the navigator to immediately maneuver a vehicle away from a hazard.

It is now possible to outline pseudocode for the Hazard Alerting Algorithm:

Algorithm 7 – Hazard Alerting Algorithm

Input: States S, Action $A_{ALERTING}$

Reward function $R : S \text{ x } A \rightarrow \mathbb{R}$, Discounting $\gamma = 0.1$, ε-greedy factor 0.10,

Learning rate $\alpha = 0.9$.

Output: Display relevant Alerting signal

1 Initialization: $s, Q_0(s, a) = 0$, $a = a_0$.

for every 10 seconds do:

2 Read COG, SOG, CPA and $TCPA$ directly from ARPA, read fused position information $(n_F,\ e_F)$ for the ownship and each target or fixed object and update s

3 If $CPA \leq r_{NM}$, or $TCPA < 30\ min$, or $d_{route} \geq 1\ NM$ proceed with the **step 4**, otherwise return to step 2

4 **for** each equipped target do:

5 extract motion data for the next 30 minutes and store in the buffer $p_{TAR(e)_n}$

6 **end for**

7 **for** each unequipped target do *(motion predictor)*:

8 compute by dead reckoning future positions utilizing present COG and SOG adding 10 seconds for the period of 30 minutes (total of 180 predicted positions) – vector algebra and store predicted positions in the buffer $p_{TAR(u)_n}$

9 **end for**

10 **for** each step of the episode do *(motion predictor)*:

11 compute by dead reckoning future positions utilizing present COG and SOG adding 10 seconds for the period of 30 minutes (total of 180 predicted positions) – vector algebra and store predicted positions in the buffer

12 **for** each of the predicted $p_{OWN}\{t, t + 10, t + 20, \ldots, t + 1800\}$ do:

13 take bearing of the next waypoint (position known from EDIS); if any of the predicted positions accomplishes present leg of the voyage, take bearing of the following available waypoint

14 compute by dead reckoning future positions utilizing present COG and SOG adding 10 seconds for the period of 30 minutes (total of 180 predicted positions) – vector algebra and store predicted positions in the buffer

15 **for** each of the predicted positions $p^{*}{}_{OWN}\{t, t+10, t+20, \dots, t+1800\}$ do:

16 **verify** $CPA > r_{NM}$

17 **if** for any $v^{*}{}_{POS_{t(n)}} = CPA \leq r_{NM}$

18 **then** return $n_{v^{*}_{POS}}, e_{v^{*}_{POS}}$ and **terminate verify**

19 **end for**

20 **end for**

21 utilize TURN Control Algorithm to determine n_{MP}, e_{MP}

22 **end for**

23 Chose a from $Q_{t-1}(s, a)$.

24 **for** each p_{OWN}, $p_{TAR(e)_n}$ and each $p_{TAR(u)_n}$:

25 Take action a, observe R, s'

26 Choose a' from s' using policy derived from Q (ε-greedy)

27 $Q(s, a) \leftarrow Q(s, a) + \alpha[\gamma Q(s', a') - Q(s, a)]$

28 $s \leftarrow s'$; $a \leftarrow a'$

action is deterministic – if R_1 or R_2 is active then $a = a_1$; if R_3 or R_4 is active, then $a = a_2$; otherwise $a = a_0$

29 **end for** when s is terminal

30 **return** Q and go back to step 2

31 **end for** when manually switched off or when “moored” is selected on AIS

32 **end**

Considering that own interest is to determine appropriate alerting for the present situation, the discount factor is $\gamma = 0.1$ as there is higher emphasis on immediate rewards, while maintaining higher fixed learning rate $\alpha = 0.9$, as this problem is relatively deterministic, but it is also necessary to allow for some exploration and to learn from new experiences. The algorithm initializes with no alerting action until the first iteration is completed and action is adjusted to the situation vehicle is in. The algorithm starts from the beginning only after casting off its lines, so for the whole voyage, algorithm would be iterating unless switched off. The ε-greedy policy remains tuned as in (3.74). After the algorithm has initialized, every 10 seconds updates determine if there is need for alerting or not. As per the line 3, if the threshold for alerting is not met, the algorithm terminates and returns to the line 2 for the next 10 seconds iteration in order to preserve computation. From lines 4 to 9 motion predictor is utilized to determine future position of both equipped ($p_{TAR(e)_n}$) and unequipped ($p_{TAR(u)_n}$) targets in the vicinity of the own vehicle. Once the future positions and attitudes of targets is known, algorithm predicts ownship progression taking into account not turning on a next waypoint if there is a waypoint in the next 30 minutes that is observed. For each timestamp, algorithm is taking a bearing to the next waypoint and then predicts progression on that bearing as well. In that way it is possible to determine the point where maneuvering is required to keep the hazard outside of the near-miss radius ($n_{v^*_{POS}}, e_{v^*_{POS}}$). As the vehicle is not able to turn instantaneously, Yawing Control Algorithm is consulted to determine the imminent maneuvering position (n_{MP}, e_{MP}). This is crucial for proposed algorithm, so that it can determine when to issue imminent maneuvering alert a_2. The algorithm iterates with delivery of Q-value and optimal action.

6.3 Encounter scenarios

In this section effectiveness of hazard alerting model is investigated by utilizing several examples of encounters. Examples are comprised of equipped and unequipped dynamic objects, as well as fixed objects and shallow waters.

As described in previous sections, when there is no need for pre-alerting or imminent alerting, the system is in a deferral state. The deferral state assumes that navigators are in control of the process and that only passive monitoring is required. Hazard alerting system is designed as an integrated part of the MCAS, so the motion predictor is of essential importance for the system's stability and accuracy.

It is necessary to reemphasize that the main reason for the hazard alerting system is to reduce the amount of alerting that is currently happening on the navigating bridge and to issue alerts for the situations when the alerting is absolutely necessary. ARPA is already informing navigators about the dangerous targets and proposed COLREGs Classification Algorithm is also determining risk of collision. However, with the MHAS alerting, the biggest benefit is getting an alert when imminent action is required to avoid collision. MHAS is only alerting navigators, while collision avoidance algorithm determines an appropriate helm order. The imminent maneuvering point is a dynamic position, as the hazard alerting algorithm exploits motion control algorithms and takes into account steering modes, depth of the water, etc., so that the proposed turn is achievable. Considering that speeds are relatively slower than in the aeronautical sector, discretization of the time-domain can be maintained. Running hazard alerting algorithm with continuous time space would have marginal benefits for a large computational expense.

Before testing scenarios that were proposed earlier in text, simple examples of own vehicle navigating a narrow channel are considered. Sabine channel is continued as simulation grounds, where own vehicle is surrounded by shallow water and there are many navigational hazards present in the area. In Figure 6.12 situation where own vehicle is approaching the Waypoint 1 is depicted and it is only equipped with ECDIS's Look-ahead feature. What is visible is that alerts are already active as the look-ahead rectangle is wide enough to activate shallow water on each side and every buoy that enters the rectangle.

This presents a challenge for navigators, as many alarms are going off during navigation, so nuisance leads to ignorance. Certainly, it is master's discretion to turn off the look-ahead feature and stop the alarms, but there are no other features available on the navigating bridge to compensate for the missing look-ahead feature. Further, in Figure 6.13 it is noticeable that before own vehicle reaches the turning point, shallow water on the opposite side of the bank is already within the reach of the look-ahead rectangle and then ownship faces an additional issue of getting multiple alarms, including the waypoint alarm and wheel-over position alarm, but with so many active alarms there is a higher chance that navigating officer simply mutes all alarms and disregards what alarms are active. Information overload comes at most critical times of navigation operations.

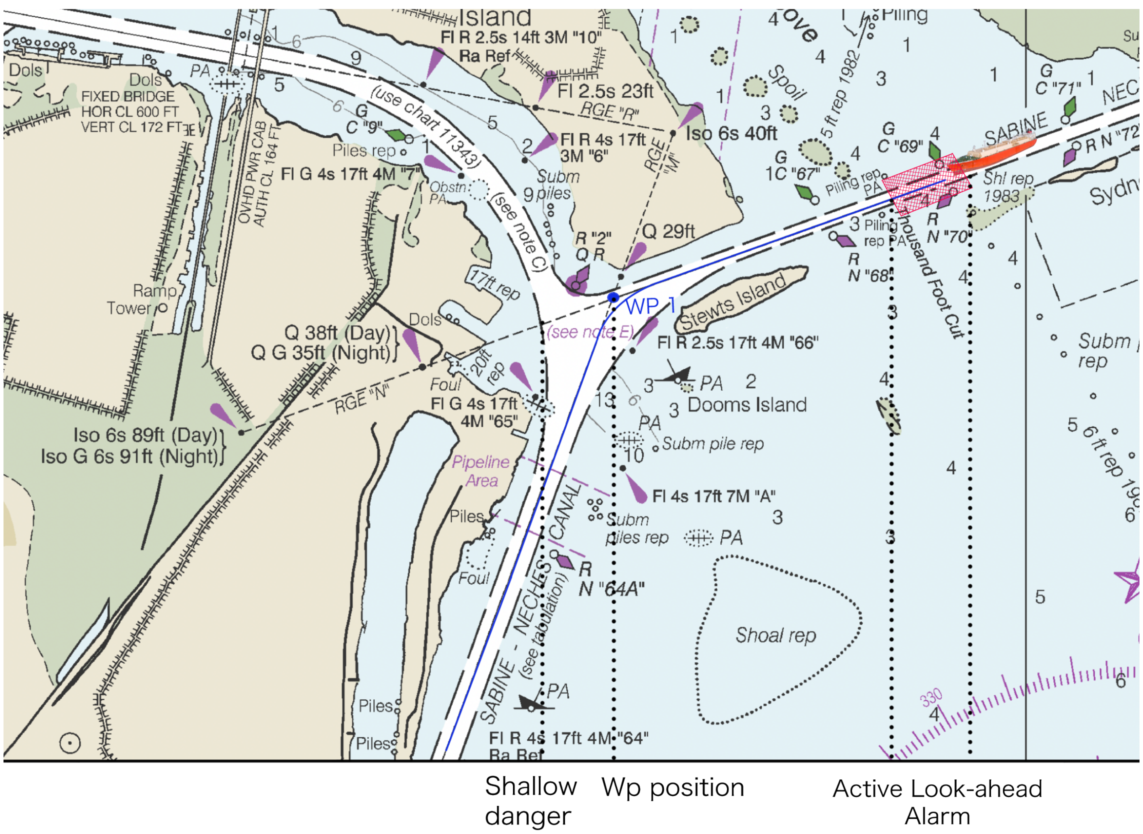
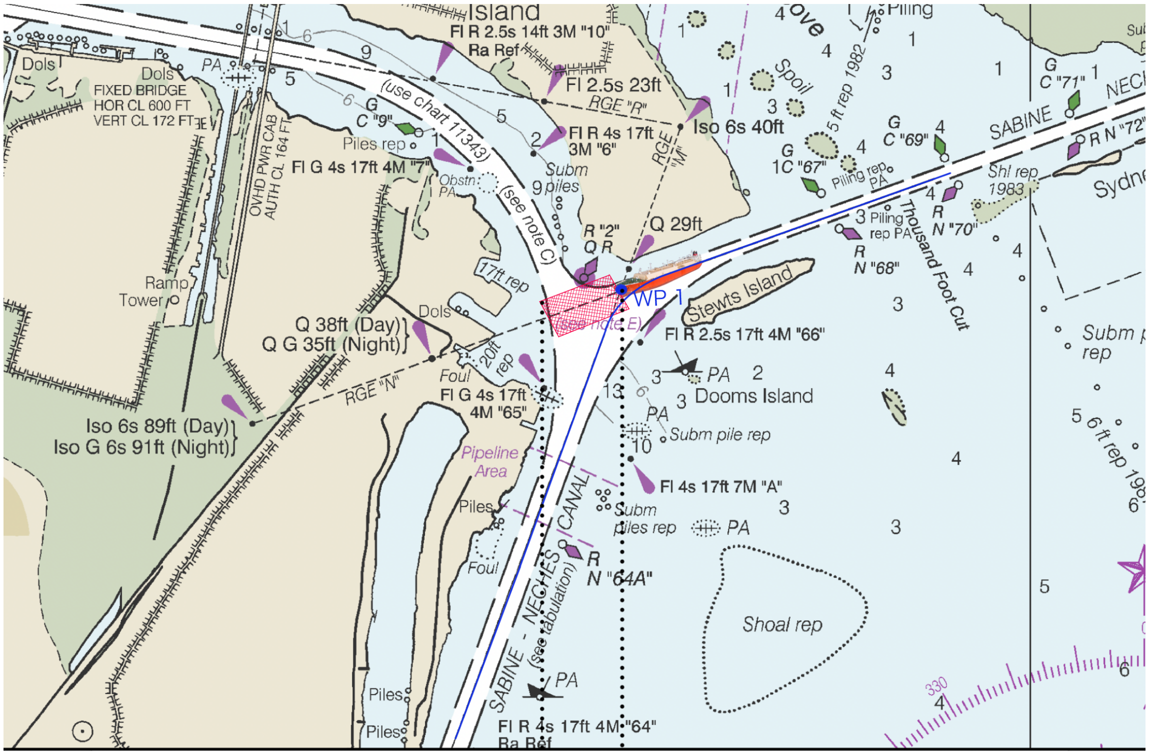


Figure 6.12 – Look ahead hazard alerting 1



Active Look-ahead
Alarm

Figure 6.13 – Look ahead hazard alerting 2

By introducing the hazard alerting algorithm with the predictor described in Chapter 4, the benefit of nuisance alarms elimination is immediately noticeable. The system can work in concurrently with look-ahead option, but even on its own provides a significant level of security. Figure 6.14 depicts the progress of how the predictor verifies the moment where own vehicle has to act in order to avoid hazard, so it issues pre-alerting and then advisory. Utilizing the motion control algorithm, the MHAS system can provide advisories of the helm order and course to steer once the pre-alarm has been activated with the same codes used in the previous chapter.

The blue color is used as regular planned passage line. The green line with waypoints is the predictor that is aligned with the planned passage. It is necessary to note that predictor assigned waypoints with 30 seconds difference and therefore waypoints are sometimes aligned and sometimes not aligned with the actual wheel-over position; regardless, there is a notification to a navigator that the wheel-over position has been reached. Orange line is predictor's trajectory estimation if navigator does not turn at the planned wheel-over position. Predictions are based on own vehicle's maneuvering characteristics utilizing knowledge of current SOG and maximum possible ROT that can be achieved at that speed. Further estimations are blue and red trajectories. Trajectory generation is discrete with 30 seconds interval to ensure computational feasibility. The orange trajectory is still safe for own vehicle. The blue trajectory shows that ownship would still be within the channel, but we do enter the wreck symbol, so it is already considered unacceptable, while the red trajectory clearly shows ownship enters the shallow waters and runs aground. If assumption was made that the planned turn was to be done with 10° ROT, then the planned wheel-over position would overlap with the second green waypoint. Wheel-over positions for the other predicted trajectories are not visible on Figure 6.14 and it is important to state that they utilize maximum ROT for the speed own vehicle achieved at the time of prediction. When own vehicle reaches next waypoint, all predictions are dynamically replanned and would be somewhat different than trajectories in the previous time-step.

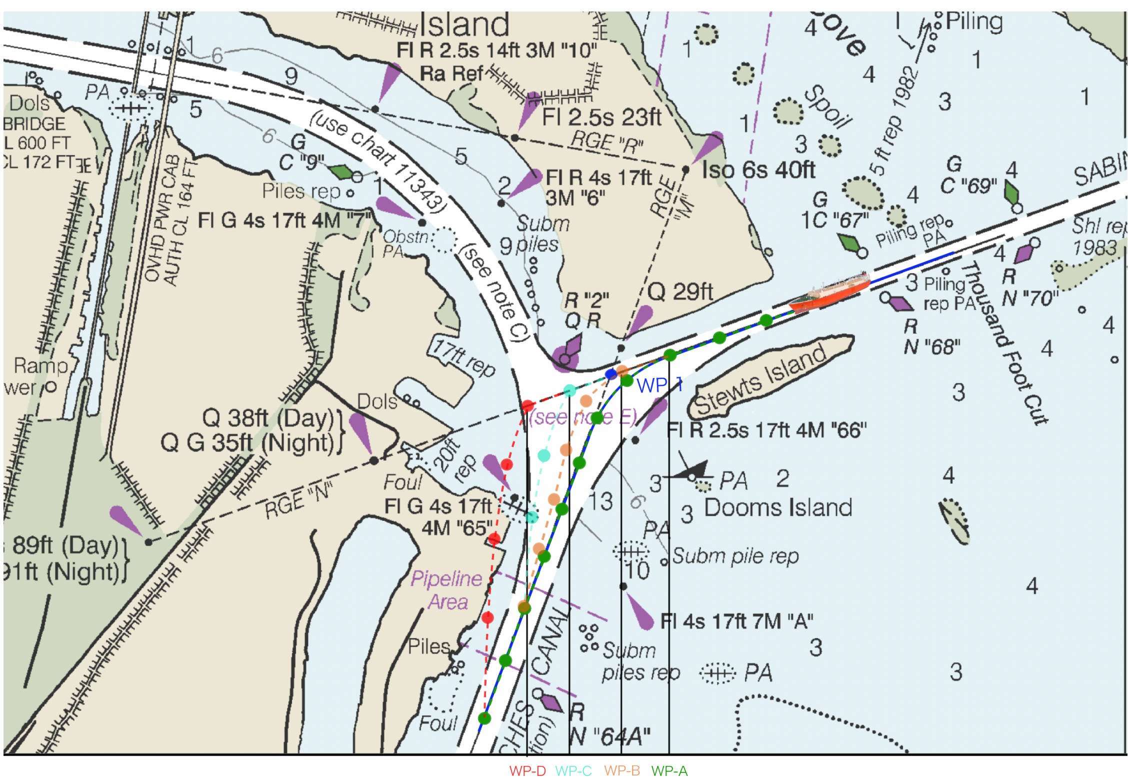


Figure 6.14 – Hazard alerting with predictor

Having the insight of predicted trajectories allows the MHAS system to generate appropriate alerts. At the green wheel-over position navigator received TTP: 200° alert. In presented scenario, simulation was performed with no external disturbances and own vehicle was able to steer the required COG with engine running at Half Ahead and SOG of 9 knots. As TTP alert is issued at the planned wheel-over position, it takes into consideration planned ROT. In order to trigger other alerts, own vehicle was not turned as per initial advisory, but rather kept the initial COG of 260°. 15 seconds after the wheel-over position, EHC alert is received in order to expedite the planned turn. At the position of the WP-A (third green waypoint), next alert is received, FTTP: 200°, which is now to be executed without delay. At the WP-B vessel received alert ID-HTP and NTS, indicating that ownship has reached the point where hard to port helm order is necessary to avoid hazards and that there should be no orders to starboard. Finally, at the WP-C, CAS alert is received indicating that ownship should initiate crash astern maneuver to minimize impact. ID-HTP and NTS alerts remained active. At the WP-D only CAS alert remained active. Alerting progress is presented in Figure 6.15.

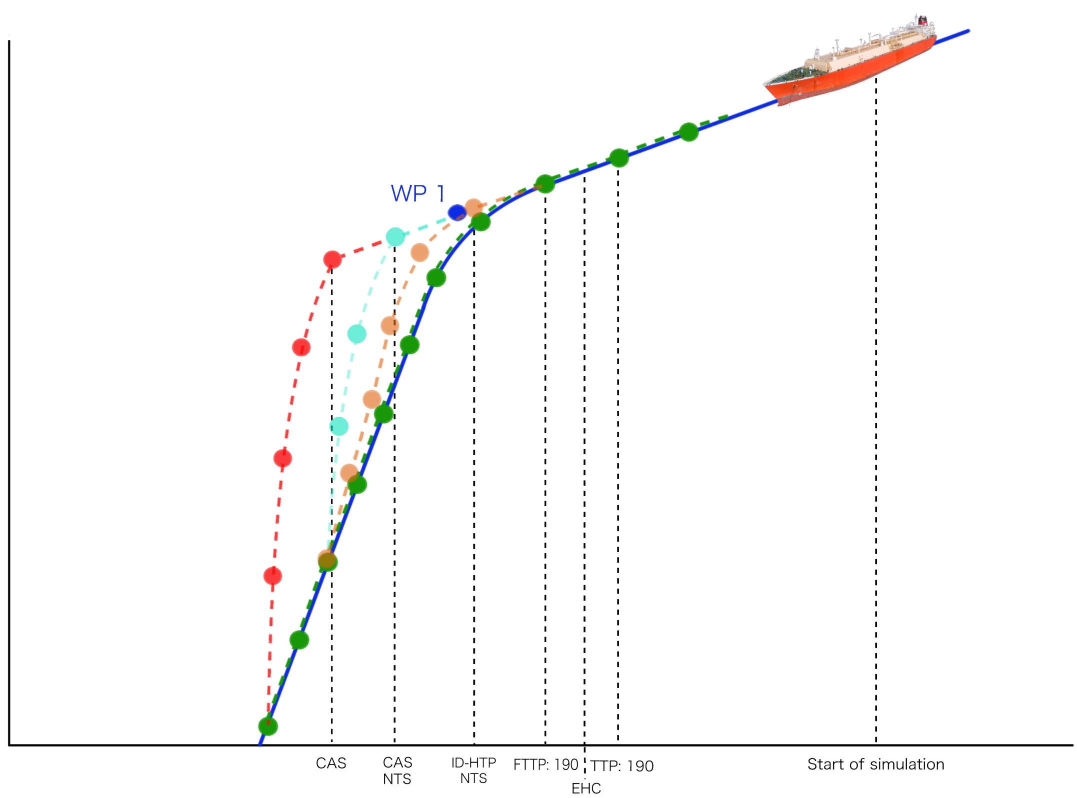


Figure 6.15 – Hazard alerting progress – course alteration

To verify feasibility and benefits of having MHAS system installed onboard, eight scenarios are used as proposed in earlier chapters. For each scenario performance of hazard alerting system is assessed by comparing the results of simulations with MHAS algorithm against the results of simulations with standard ARPA and ECDIS alerting. In each scenario, targets are assessed individually. The situation is monitored as static, so own interest lies in finding out what alerts ownship gets for the situation visible on RADAR and ECDIS screens. Relative geometries are allowed to follow through, without intervening and altering course. Collisions do happen in simulations, but this is required to confirm necessity of alerts. For simplicity it is assumed that there are no external disturbances, but this does not limit the algorithm as it is capable of handling disturbances with an aid of motion control algorithms. Each scenario is simulated for 100 times, and for each pairwise situation, score is entered into the confusion matrix from which precision and recall values are calculated and SPRC curves constructed. Sensor uncertainties are also allowed, so filtering and/or fusion of sensor data is inhibited. Proposed reward function is setup so that alert is required if target trajectories are entering Near Miss radius of 1 NM and the target is less than 3 NM away, or the TCPA is less than 15 minutes. For the turning scenarios and shallow water hazards, alerting is to happen if the vehicle is not following the intended route and reaching the appropriate alerting point determined by the predictor.

The scatter distribution of resulting data is within very dense range, so it is hard to visualize with SPRC curves; therefore, Precision-Recall scatterplots without are constructed without connecting them. However, it is possible to clearly envision the difference between ARPA/ ECDIS alerting and MHAS alerting. It is necessary to keep in mind that scatters that are closer to the upper right corner (1,1) are considered to be better performing with precision and recall score close to 100 %.

Scenario 1 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **324**	(NA) **290**
	Alert Not raised	(UA) **76**	(TN) **310**

Table 6.1 – Scenario 1 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **393**	(NA) **7**
	Alert Not raised	(UA) **7**	(TN) **593**

Table 6.2 – Scenario 1 MHAS CM

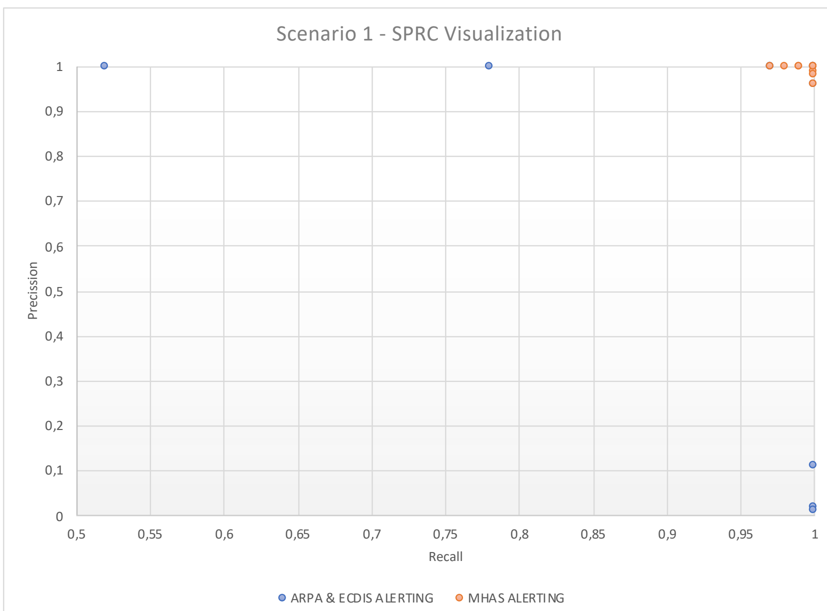


Figure 6.16 – SPRC of the Scenario 1

In the first scenario, 4 targets for which alert was warranted and 6 targets where alert was unnecessary are depicted. It is possible to see from the resulting confusion matrices that MHAS performed comparatively better with lower number of nuisance and unsuccessful alarms. SPRC shows that MHAS alerting is concentrated around the top right, while ARPA & ECDIS alerting had a much wider spread.

Scenario 2 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **392**	(NA) **226**
	Alert Not raised	(UA) **8**	(TN) **374**

Table 6.3 – Scenario 2 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **397**	(NA) **12**
	Alert Not raised	(UA) **3**	(TN) **588**

Table 6.4 – Scenario 2 MHAS CM

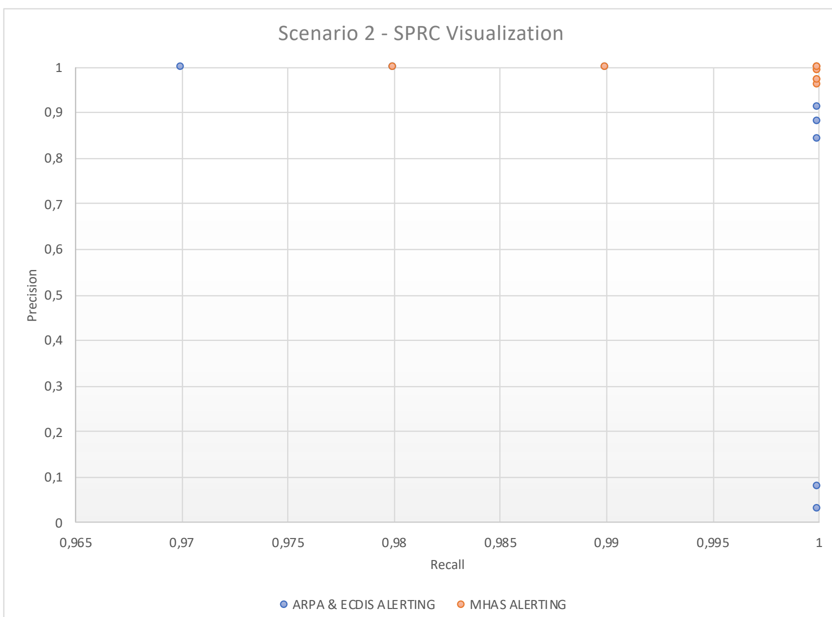


Figure 6.17 – SPRC of the Scenario 2

Similarly, in the second scenario 4 targets for which alert was warranted and 6 targets where alert was unnecessary are illustrated. Even though ARPA & ECDIS alerting performed comparatively better for the True Alert category, it is still noticeable that MHAS outperformed especially in the Nuisance Alerts category. SPRC shows tighter spreads for both categories, but MHAS is still performing better and closer to the upper right corner.

Scenario 3 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **387**	(NA) **180**
	Alert Not raised	(UA) **13**	(TN) **420**

Table 6.5 – Scenario 3 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **395**	(NA) **8**
	Alert Not raised	(UA) **5**	(TN) **592**

Table 6.6 – Scenario 3 MHAS CM

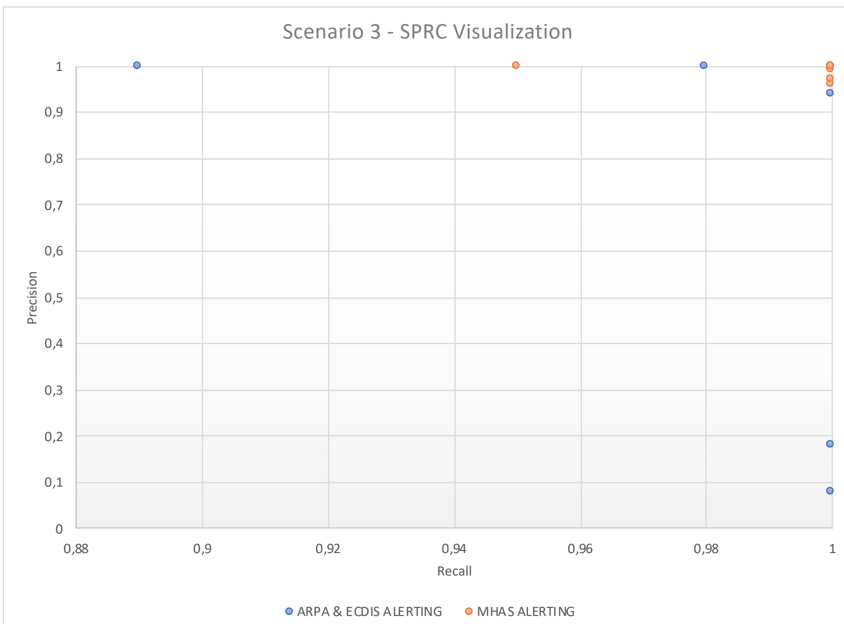


Figure 6.18 – SPRC of the Scenario 3

The third scenario maintains ratio of 4 targets warranting alerts and 6 targets where alerting is not required. Performance of MHAS remains favorable as evident from the confusion matrices. The SPRC visualization confirms MHAS outperforming sole ARPA and ECDIS alerting and the reduction of nuisance alerts is evident in this scenario as well.

Scenario 4 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **296**	(NA) **193**
	Alert Not raised	(UA) **4**	(TN) **507**

Table 6.7 – Scenario 4 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **298**	(NA) **5**
	Alert Not raised	(UA) **2**	(TN) **695**

Table 6.8 – Scenario 4 MHAS CM

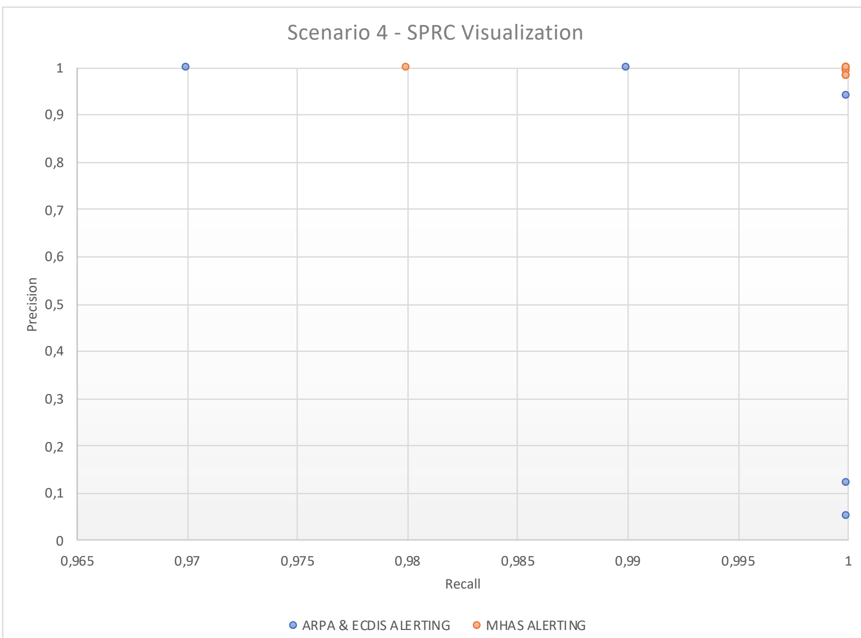


Figure 6.19 – SPRC of the Scenario 4

The fourth scenario has 3 targets that require alerting and 7 targets where alerting would be unnecessary. MHAS continues to outperform, especially in the True Negatives and Nuisance Alerts categories. It is evident that ARPA & ECDIS alerting was designed to err on the negatives side and that the system allows for false positives to maintain safety of navigation. SPRC confirms better performance of MHAS.

Scenario 5 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **418**	(NA) **185**
	Alert Not raised	(UA) **82**	(TN) **315**

Table 6.9 – Scenario 5 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **494**	(NA) **5**
	Alert Not raised	(UA) **6**	(TN) **495**

Table 6.10 – Scenario 5 MHAS CM

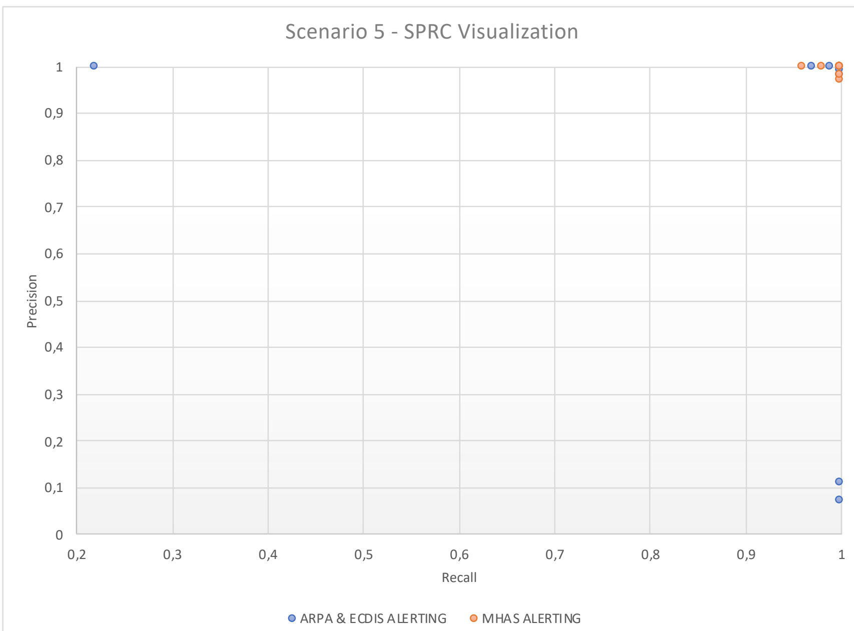


Figure 6.20 – SPRC of the Scenario 5

The fifth scenario has an equal distribution of targets warranting alerts and targets that are considered hazard free. Once again, the results of the experiment show that MHAS performs better in all aspects, but it is particularly successful at reducing Nuisance Alerts. The SPRC of the scenario 5 has shown that MHAS increases safety of navigation with multiple targets within the surveyed area.

Scenario 6 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **97**	(NA) **14**
	Alert Not raised	(UA) **3**	(TN) **386**

Table 6.11 – Scenario 6 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **99**	(NA) **2**
	Alert Not raised	(UA) **1**	(TN) **398**

Table 6.12 – Scenario 6 MHAS CM

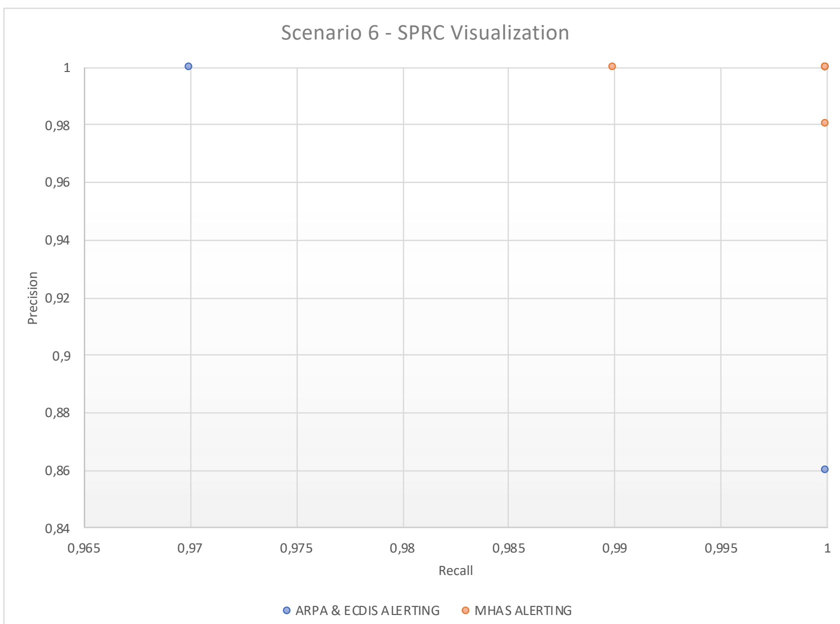


Figure 6.21 – SPRC of the Scenario 6

In this scenario only five targets were tracked, where one target required alerting and others not. Both approaches showed good performance, even though MHAS did provide more stable results. The SPRC depicts MHAS's stability, as the scatter concentration remains in the top right corner of the precision-recall curve.

Scenario 7 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **293**	(NA) **36**
	Alert Not raised	(UA) **7**	(TN) **664**

Table 6.13 – Scenario 7 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **295**	(NA) **1**
	Alert Not raised	(UA) **5**	(TN) **699**

Table 6.14 – Scenario 7 MHAS CM

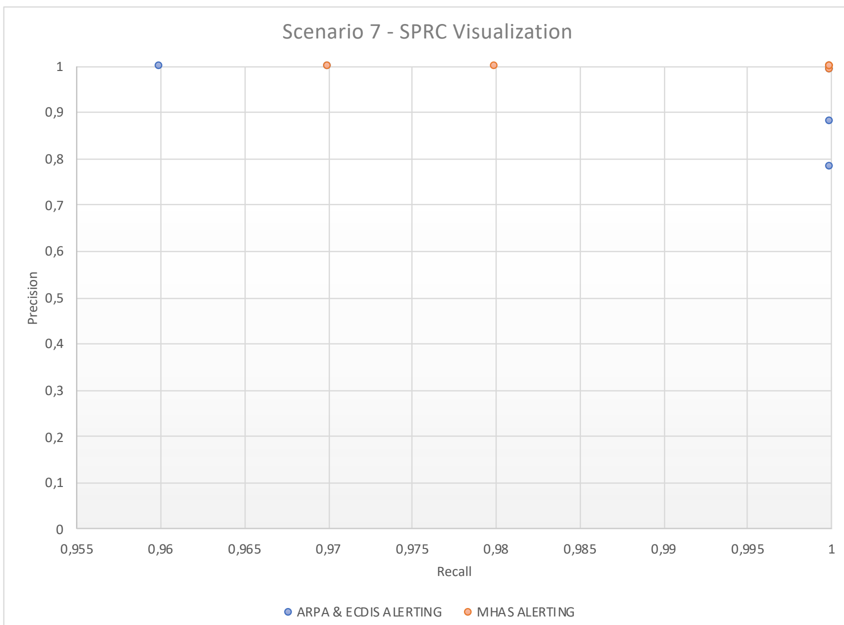


Figure 6.22 – SPRC of the Scenario 7

In this scenario ten targets were tracked, where three targets required alerting while other were alert free. Both approaches showed acceptable performance, even though MHAS did outperform. The SPRC again shows MHAS's stability, as the scatter concentration dominantly remain in the top right corner of the precision-recall curve.

Scenario 8 – Experiment Results

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **200**	(NA) **297**
	Alert Not raised	(UA) **0**	(TN) **3**

Table 6.15 – Scenario 8 ARPA & ECDIS CM

		Actual	
		Alert required	Alert Not required
Simulation	Alert Raised	(TA) **200**	(NA) **1**
	Alert Not raised	(UA) **0**	(TN) **299**

Table 6.16 – Scenario 8 MHAS CM

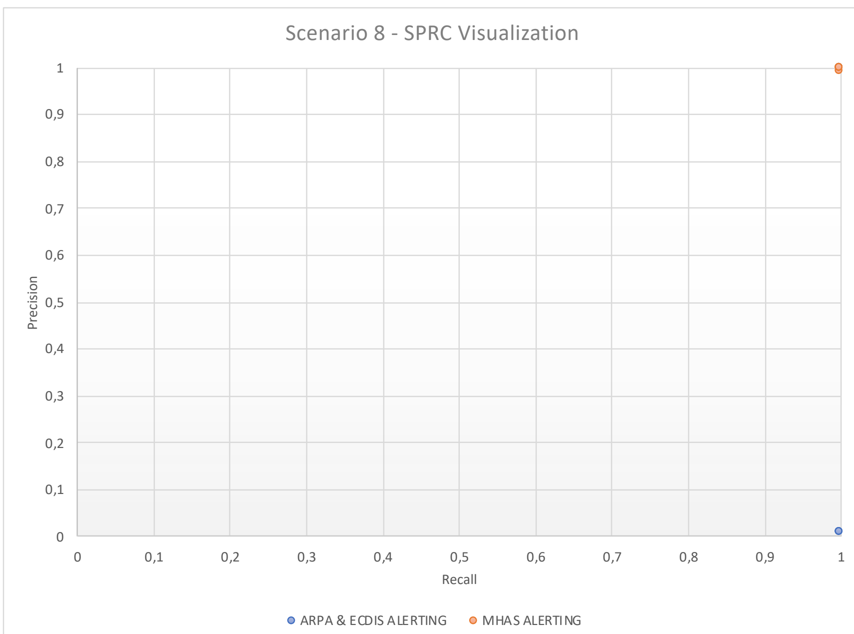


Figure 6.23 – SPRC of the Scenario 8

The eight scenario is a special one as it has only two moving targets and three hazards selected by the author to showcase performance against charted hazards as well. Two of the charted hazards are shallow water, while one is a set of navigational buoys. Both approaches resulted with perfect score for the targets that required alerting. However, ARPA and ECDIS alerting scored poorly on Nuisance Alerts as they were alerting for target vehicle, shallow water and buoys when this was not necessary. MHAS managed to recognize necessity and alerted only when it was required. Failure to recognize true negatives resulted to ARPA and ECDIS scatters distribution to the right lower corner, while MHAS remained concentrated in the top right corner.

Finally, Figure 6.24 depicts combined SRPC for all scenarios and it is easy to notice that MHAS performed better on average than the ARPA and ECDIS alerts, especially when detecting Nuisance Alerts that potentially lead to diminished attentiveness of navigators when performing critical navigational tasks. Current tools we have on navigational bridges are helpful and effective, but the amount of unnecessary alarms could be reduced if systems like MHAS would be utilized onboard commercial sea surface vehicles.

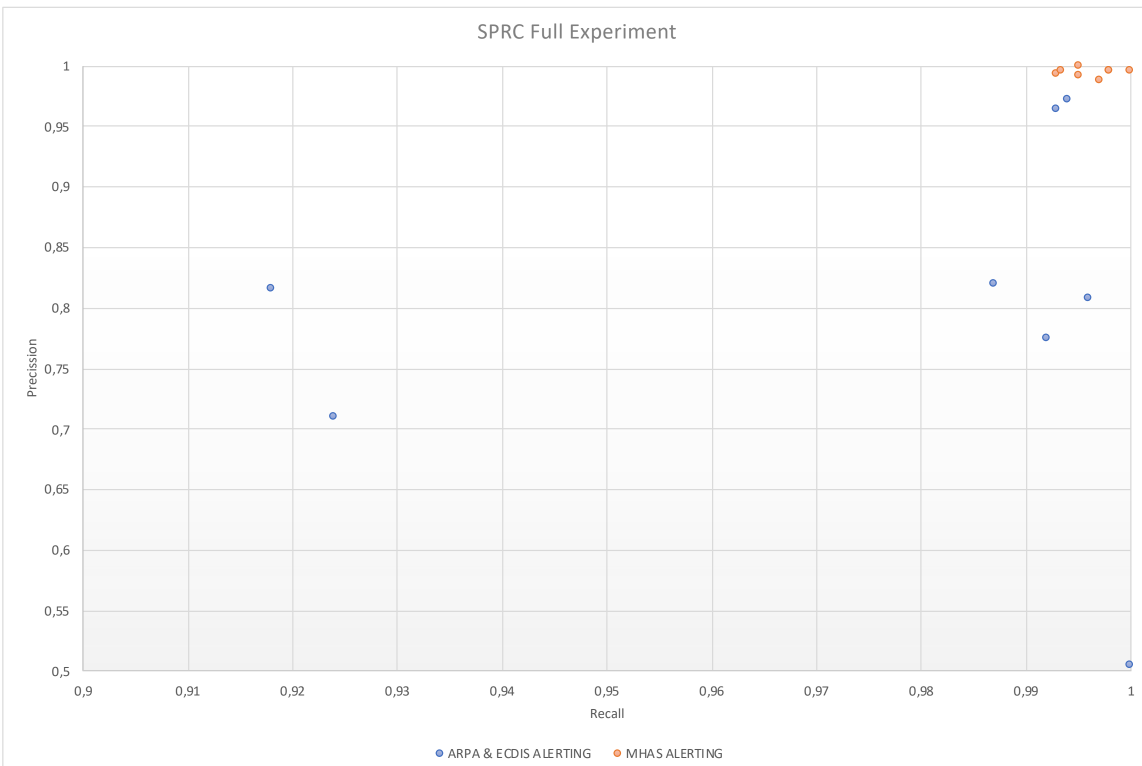


Figure 6.24 – Combined SPRC

6.4 Discussion

Together with filtering algorithms, motion control algorithms, predictors and collision avoidance algorithm, hazard alerting algorithm forms a Marine Collision avoidance and Alerting System (MCAS), which is proposed in the final empirical chapter of this thesis. The remaining problem of dynamic collision avoidance is having a proper alerting system that will inform human navigator of certain occurrence, but only when it is necessary. The level of nuisance alarms onboard commercial sea surface vehicles increased with the introduction of new electronic equipment. Due to the fact that sea surface vehicles use lower quality sensors, alerting protocols are setup to err on the positive side, as it is better to alert than not to alert when alerting is required.

Reasons why navigators face a larger number of nuisance alarms onboard commercial sea surface vehicles is investigated and it is discovered that improvement is needed in information integration and algorithmic design. In the case of autonomous vehicles, nuisance alarms would be potentially a significant problem when used as input signals to stop or correct processes. Therefore, solution is proposed to detect appropriate alerting for collision avoidance that integrates algorithms proposed in earlier chapters. The key component of the proposed Marine Hazard Alerting System (MHAS) is reducing the level of uncertainty for which Behavior Parametric Form (BPF) approach was proposed.

Experiments showed that reinforcement learning was a successful approach and that it is possible to significantly reduce number of nuisance and unsuccessful alerts with stable precision-recall values. It is evident that ARPA and ECDIS alerting lack the finesse to recognize situations where alerts are warranted and when they are not. Proposed MHAS system was able to improve on alerting performance and add another layer of protection for navigators controlling passages of sea surface vehicles.

Chapter 7

Summary, contributions, and further work

7.1 Summary

Technological advances improved safety of navigation throughout the years, but also brought new challenges when navigating sea surface vehicles on commercial routes. Usually described as input data and information overload, being in charge of navigation is a challenging task, especially in busy waterways. In this thesis comprehensive system of algorithmic decision support solutions for navigators is proposed, so that in critical situations navigators can bring informative decisions about their course of actions.

Human element remains the main contributory factor to majority of collision incidents and close quarter situations at sea. Even though better equipment and sophisticated navigational aids are installed today, we can still notice ample number of incidents happening at sea. As a main motivation to conduct this study, robust Marine Collision avoidance and Alerting System (MCAS) is proposed. MCAS is comprised of four modules.

The first module aims to improve input data to the main module, collision avoidance algorithm. Resolution advisories are going to be as good as the input information is, therefore in this thesis an improved nonlinear dynamic state estimator, Foraging Particle Filter (FPF), is proposed. FPF is used to reduce uncertainty and noise of sensing equipment typically carried onboard commercial sea surface vehicles. Fusing of sensing data is also considered. Results of experiments showed that proposed Foraging Particle Filter outperformed existing filters. Having better input signals, it was possible to focus on motion control of sea surface vehicles. One of the objectives of this research was to develop model-free solution that can utilize Hidden Markov Model (HMM) framework to allow for efficient reinforced and imitation learning. Initially, as data is not available when a sea surface vehicle is built, existing methodology of modeling vehicles is exploited, so sea trials are performed in combination with simulations. In this way it was possible to stream initial experience for the system. Several algorithms with specific reward functions have been developed and used to ensure safe, feasible, compliant, and efficient motion control operations. Experimental results have shown that proposed autopilot and auto-telegraph models performed well and were stable under any simulated environmental load. This step was crucial to guarantee feasibility of generated trajectories by collision avoidance system.

Dynamic collision avoidance requires stable trajectory generation that are feasible and compliant. This is the main focus of the second module where collision avoidance regulations are investigated in order to quantify collision avoidance rules. Quantification of COLREGs allowed for rewards and actions space design that would assist predictor to generate optimal trajectories. COLREGs classification algorithm is developed and used to determine traffic situation on the horizon and offer instant uncluttered information about targets and their status. Experimental results showed that classification algorithm made a substantial improvement of the system in general and maintained accuracy levels under navigational and environmental loads. Once the input data was more reliable and traffic overview transparent, focus shifted on collision avoidance algorithm and predictor design. Simulation results showed that the proposed system thrived well and managed to resolve complex situations within the required time to ensure feasibility.

However, if only pairwise situations were resolved, it would not be possible to dynamically adapt to new situations, so the proposed egocentric approach was broadened to include multiple targets. In this study, mixed equipage situations were of particular interest. We continued to utilize HMM framework and reinforcement learning to generate optimal trajectories and avoid close quarter situations. Holistic approach has outperformed egocentric approach, which was evident from the experiment results where it was noticed that the system managed to resolve challenging collision situations with lower deviation from the original course and by returning to the planned route within shorter time period. Uncertainty was further lowered by introducing early collision detection approach where participants shared not only intent, but passage plans as well. This approach significantly reduced risk of collision in navigable areas.

Finally, hazard alerting was investigated as an intrinsic part of collision avoidance. Hazard alerting is not only designed to alert human navigators, but also to generate alerting signals to automated systems and reduce computational complexity of collision avoidance process. Hazard alerting system utilized predictor to determine when alerting is required, which significantly reduced levels of nuisance alerts.

7.2 Contributions

This research is one of the first studies to utilize Reinforcement learning based algorithms to collision avoidance problems in maritime sector. Careful selection of state space members was utilized, discrete action space (as research took only underactuated vehicles into account) maintained, and specific design of reward space provided in order to warrant generation of feasible trajectories.

Nonlinear state estimating variant was proposed that was successful in reducing data uncertainty and filter out sensing noise. This was done by exploiting foraging process in nature, which aided design of solution to particles impoverishment and degeneracy. Foraging Particle Filter was verified during baseline experiments and outperformed some of the common linear and nonlinear filters. FPF was used to design nonlinear passive observer with efficacy and feasibility success.

Trajectory generation and modification solution that utilizes both offline and online computation is introduced to decentralize computational burden. Feasibility of the system was confirmed through simulated studies. Own rewards shaping techniques were used to successfully complete complex collision-avoidance tasks. Considering that increasing number of target vehicles does not increase computational burden exponentially, but rather linearly, it is possible to envision installation of similar systems on commercial vehicles.

Motion control algorithms were designed in order to sustain feasibility of generated trajectories. Motion control system is comprised of heading and course controls, turning control and engine telegraph control. This enabled decrease of the state space size, shaping of reward functions and conjoining action space, so that exploration is discouraged when unnecessary.

Thorough and deep exploration of collision regulations has been made and it showed that only by mutual agreement we can improve navigational tasks and challenge existing collision avoidance rules. It is evident that COLREGs are not aligned with modern shipping industry, and that collision regulations require rewrite. COLREGs classification algorithm has been proposed and it showed that it is highly adjustable and of crucial importance for the stability of the collision avoidance system. COLREGs classification algorithm performed well and enhanced feasibility of the generated trajectories.

Dynamic collision avoidance algorithms were developed and explored within the simulation space. A number of cases were trialed, and our model showed to be a viable option for the commercial use. Various types of uncertainties were considered when making decision about the collision avoidance maneuver. The most prominent inclusion was human operator uncertainties, which were also taken into account and modeled. The proposed MCAS system has shown ability to cope with larger number of obstacles in complex situations and was successful in generating trajectories that avoided conflicts and safeguarded commercial interests of shipowners by rewarding vehicles to return to their planned paths as soon as deemed safe by the algorithm.

An early collision risk detection approach has been investigated together with sharing intent and proposed as a complimentary system that substantially reduces risk of collision. Within this research, intent sharing (short term) and passage plan sharing (long term) have improved performance of predictor and reduced time spent on maneuvering, as well as reduced distance travelled before returning to the planned route.

Together with intent-aware collision avoidance, coordinated collision avoidance was considered and protocols developed to share information with equipped vehicles. Contemplating the fact that equipped vehicles have to operate in areas where unequipped vehicles sail as well, mixed equipage is considered, and collision avoidance algorithm successfully verified for feasibility in mixed equipage environment as well.

A novel methodology for marine hazard alerting based on HMM theory has been proposed and showed considerable improvements over the existing alerting systems mainly by reducing nuisance alerts. Defining the point of "last minute" action was a challenge that can be debated among Organizations, but the approach in this study was to utilize maneuvering restrictions and combine it with passage planning to determine points when pre-alerting and alerting should commence. The proposed model can accept various inputs in order to safeguard alerting necessity and decrease number of unnecessary alerts.

Finally, an improved approach to rewards shaping was proposed and showed good performance within depicted environment. Throughout the exploration and tuning, it was discovered that stable learning and exploration prefers higher penalties than higher rewards. Proposed learning models performed better if they faced a significant penalty than when they faced smaller penalties and larger rewards.

7.3 Further work

It is necessary to do substantial amount of system testing and field validation. However, it is also difficult to endeavor that experimental systems would be allowed for testing on commercial vehicles, but vehicles that sail in controlled environments and near coasts would be perfect candidates for exploration and exploitation. MCAS can be installed onboard test vehicles to work in parallel with existing systems and just collect data and compare computed and optimal decisions for the full length of one voyage.

As it did fit well, shaping and designing rewards were done manually, however, with the implementation of deeper learning techniques, there are solutions to derive rewards from a system when rewards are unknown or there is a necessity of finding more accurate distribution of rewards within observable or latent state space.

It would be beneficial to develop robust protocols that would be used to propagate intent information and to seamlessly share passage plan information with integration to the existing navigational equipment. There is a potential benefit of incorporating vehicle kinematics in collision avoidance optimization function and in this way even further reduce computational loads.

Substantial work is required in the field of collision regulations. It is unclear how would autonomous vehicles cooperate with other commercial vehicles and how should this interaction be regulated.

Presented work is a viable option to control and support decision making without accurate representation of a vehicle's dynamics. However, it requires well designed reward functions, by utilizing experienced professionals to tune reward space. There is another computational way where reward space can be reverse engineered by collecting large amount of data where professionals would show "good behavior" so that parameters could be extracted from the data and used for reward space representation.

In order to make this approach commercially viable, following steps would be required: 1. Building a realistic model for simulation by experienced naval architect, 2. Extracting control parameters from simulator by allowing abstract agent to learn in the simulated environments, 3. Testing learned policies in simulated environments, 4. Confirming learned results on models and 5. Integrating software on commercial sea surface vehicles with trial period, 6. Certification and approvals for full-time use.

Bibliography

[1] Abbeel, P., and Ng, A. Y. Apprenticeship Learning via Inverse Reinforcement Learning. Proceedings of the 21st International Conference on Machine Learning, Banff, Canada. 2004.

[2] Abdullah, A., and Zribi, M. Sensor-fault-tolerant control for a class of linear parameter varying systems with practical examples. IEEE Trans. Ind. Electron., vol. 60, no. 11, pp. 5239-5251, 2013.

[3] Acar, U., Ziarati, R., and Ziarati, M. Collisions and groundings – major causes of accidents at sea. Marifuture papers: 48-51. http://www.marifuture.org/Publications/Papers.aspx. Accessed 20 September 2020. Marifuture, 2008.

[4] Anderson, J. A. An Introduction to Neural Networks, Cambridge, MIT Press, 1995.

[5] Annamalai, A. S., Sutton, R., Yang, C., Culverhouse, P., and Sharma, S. Robust adaptive control of an uninhabited surface vehicle. Journal of Intelligent & Robotic Systems, vol. 78, no. 2, pp. 319–338, 2015.

[6] Antao P., and Guedes Soares C. Causal factors in accidents of high-speed craft and conventional ocean going vessels. Reliable Engineering Systems Safety 93:1292–1304, 2008.

[7] Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. 50, 174–188. 2002.

[8] ASAS-TN2 Consortium. Airborne Separation Assistance System Thematic Network 2 (ASAS-TN2): ASAS application maturity assessment, 2008.

[9] Asmar, M. D. Airborne Collision Avoidance in Mixed Equipage Environments. MS Thesis. MIT Press, Boston, Massachusetts, USA, 2013.

[10] Atia, M., Georgy, J., Korenberg, M., and Noureldin, A. Real-time implementation of mixture particle filter for 3D RISS/GPS integrated navigation solution. Electron. Lett., vol. 46, no. 15, pp. 1083-1084, 2010.

[11] Aylward, K., Weber, R., Man, Y., Lundh, M., and MacKinnon, S. N. "Are You Planning to Follow Your Route?" The Effect of Route Exchange on Decision Making, Trust, and Safety. Journal of Marine Science and Engineering, 8, 280, 2020.

[12] Azad, S. A. The collision regulations through a translator's eyes. The Journal of Navigation, 12, pp. 201-206, 1959.

[13] Back, T. Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, 1996.

[14] Bagnell, J. A., and Schneider, J. C. Autonomous helicopter control using reinforcement learning policy search methods. In International Conference on Robotics and Automation, pp. 1615–1620. IEEE Press, 2001.

[15] Balchen, J. G., Jenssen, N. A., and Sælid, S. Dynamic Positioning using Kalman Filtering and Optimal Control Theory. Proceedings of the IFAC/IFIP Symposium and Automation in Offshore Oil Field Operation, Holland, Amsterdam, pp. 183-186, 1976.

[16] Bandera, C., Vico, F. J., Bravo, J. M., Harmon, M. E., and Baird, L. C. III. Residual Q-Learning applied to visual attention. In 13th International Conference on Machine Learning, pp. 20–27, 1996.

[17] Bandler, W., and Kohout, L. J. Semantics Of Implication Operators And Fuzzy Relational Products. International Journal Of Man-Machine Studies, 12, pp. 89–116, 1980.

[18] Bandyopadhyay, T., Jie, C. Z., Hsu, D., Ang Jr., M. H., Rus, D., and Frazzoli, E. Intention-aware pedestrian avoidance. Experimental Robotics, pp. 903-977, Springer, 2013.

[19] Bandyopadhyay, T., Won, K. S., Frazzoli, E., Hsu, D., Lee, W. S., and Rus, D. Intention-aware motion planning. Algorithmic Foundations of Robotics X, pp. 475-491, Springer, 2013.

[20] Barhydt, R., Kopardekar, P., Battiste, V., Doble, N., Johnson, W., Lee, P., Prevot, T., and Smith, N. Joint NASA Ames/Langley experimental evaluation of integrated air/ground operations for en route free maneuvering. 2005.

[21] Barhydt, R., Palmer, M., and Eischeid, T. Development and evaluation of an airborne separation assurance system for autonomous aircraft operations. 24th International Congress of Aeronautical Sciences, Yokohama, Japan, 2004.

[22] Batalden, B. M., and Sydnes, A. K. What causes 'very serious' maritime accidents? Safety and Reliability – Theory and Application, pp. 3067-3074, Taylor & Francis, London, 2017

[23] Benjamin, R.M. Multi-objective autonomous vehicle navigation in the presence of cooperative and adversarial moving contacts. In OCEANS'02 MTS/IEEE, volume 3, pp. 1878-1885. IEEE, 2002.

[24] Benjamin, R.M., and Curcio J. Colregs-based navigation of autonomous marine vehicles. IEEE/OES autonomous underwater vehicles pp. 32–39, 2004.

[25] Benjamin, R.M., Curcio, J.A., Leonard, J.J., and Newman, P.M. Navigation of

unmanned marine vehicles in accordance with the rules of the road. In Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICFA 2006., pp. 3581-3587, May 2006.

[26] Betts, J. T. Practical methods for optimal control using nonlinear programming, volume 3 of Advances in Design and Control. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2001.

[27] Bevan, G. H., and O'Reilly, J. Automatic lateral emergency collision avoidance for a passenger car. International Journal of Control, 80(11): 1751– 1762, 2007.

[28] Billesø, M. B. ACCSEAS Project – Service Description: Tactical Exchange of Intended Routes. Accessed 18. March 2020. http://www.accseas.eu. 2015.

[29] Bonissone, P. P. A Fuzzy Sets Based Linguistic Approach: theory and applications approximate reasoning indecision analysis. Amsterdam: North Holland, 329–339, 1982.

[30] Borcic, B. Matematicka Kartografija. Tehnicka Knjiga, Zagreb, 1955.

[31] Borenstein, J., and Koren, Y. The Vector Field Histogram--Fast Obstacle Avoidance for Mobile Robots, IEEE Transactions on Robotics and Automation, Vol 7, No 3, pp 278-288, June 1991.

[32] Borup, O. B. ACCSEAS Project – Service Description: Maritime Cloud. Accessed 18. March 2020. http://www.accseas.edu. 2020.

[33] Browning, A. W. A Mathematical-Model To Simulate Small Boat Behavior. Simulation, 56, pp. 329–336, 1991.

[34] Bryant, R. L. Geometry of manifolds with special holonomy: "100 years of holonomy". In 150 Years of Mathematics at Washington University in St. Louis, volume 395 of Contemporary Mathematics, pages 29–38. American Math- ematical Society, Providence, RI, 2006.

[35] Brys, T., Harutyunyan, A., Suay, H. B., Chernova, S., Taylor, M. E., and Nowé, A. Reinforcement learning from demonstration through shaping. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), pp. 3352-3358, 2015.

[36] Bukaty, V. M., and Morozova, S. U. On Determination of the Head-on Situation Under Rule 14 of Colreg-72. TransNav, 4(4), pp. 383-388, 2010.

[37] Burns, R. S., Blackwell, G., and Calvert, S. An automatic guidance, navigation and collision avoidance system for ships at sea. IEE Colloquium on Control in Marine Industry, 3/1-3/3, 1988.

[38] Cannell, W. P. Collision avoidance as a game of co-ordination. The Journal of

Navigation, 34, pp. 220–239, 1981.

[39] Calvert, E. S. Manoeuvres to ensure the avoidance of collision. The Journal of Navigation, 13, pp. 127–137, 1960.

[40] Chang, K. Y., Jan, G. E., and Parberry, I. A method for searching optimal routes with collision avoidance on raster charts. The Journal of Navigation, 56, pp. 371–384, 2003.

[41] Chen, S. Kalman filter for robot vision: A survey. IEEE Trans. Ind. Electron., vol. 59, no. 11, pp. 4409-4420, 2012.

[42] Cheng, X.D., Liu, Z.Y., and Zhang X.T. Trajectory optimization for ship collision avoidance system using genetic algorithm. In OCEANS 2006 - Asia Pacific, pages 1–5, May 2006.

[43] Churkin, V. I., and Zhukov, Y. I. Procedures for ship collision avoidance. OCEANS'98 Conference Proceedings 2, pp. 857–860, 1998.

[44] Coldwell, T. G. Marine traffic behaviour in restricted waters, The Journal of Navigation, Vol. 36, No. 3, pp. 430-444, 1983.

[45] Cook, R. T. A dictionary of philosophical logic. Edinburgh University Press, p. 84, 2009.

[46] Cui, Y., Osaki, S., and Matsubara, T. Reinforcement Learning Ship Autopilot: Sample efficient and Model Predictive Control-based Approach. The 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019). arXIV:1901.07905. 2019.

[47] Darrell, T., and Pentland, A. Active gesture recognition using partially observable Markov decision processes. In ICPR96, pp. 984–988, 1996.

[48] Davis, P. V., Dove, M. J., and Stockel, C. T. A computer simulation of marine traffic using domains and arenas, The Journal of Navigation, Vol. 33, No. 1, pp. 215-222, 1980.

[49] Degre, T., and Lefevre, X. A collision avoidance system. The Journal of Navigation, 34, pp. 294–302, 1981.

[50] Dellaert, F., Fox, D., Burgard, W., and Thrun, S. Monte carlo localization for mobile robots. In Proc. ICRA-99, 1999.

[51] De Vries, L. Work as Done? Understanding the Practice of Sociotechnical Work in Maritime Domain. Journal of Cognitive Engineering Decision Making, 11, pp. 270-295, 2017.

[52] DHMZ. ALADIN - Wind dynamical adaptation for Istra and Kvarner. Croatian

Meteorological and Hydrological Service. Retrieved July 20, 2020, from http://meteo.hr/index_en.php., 2020.

[53] Dias, S. S., and Bruno, M. G. Cooperative target tracking using decentralized particle filtering and RSS sensors. IEEE Trans. Signal Proces., vol. 61, no. 14, pp. 3632-3646, 2013.

[54] Dinh, G. H., and Im, N. The combination of analytical and statistical method to define polygonal ship domain and reflect human experiences in estimating dangerous area. International Journal of e-Navigation and Maritime Economics 4, pp. 97-108, 2016.

[55] Dorigo, M., and Colombetti, M. Robot shaping: Developing autonomous agents through learning. Artificial Intelligence, 71(2), pp. 321-370, 1994.

[56] Doucet, A., Godsill, S., and Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comput.*, vol. 10, no. 3, pp. 197–208, 2000.

[57] Douc, R., and Cappé, O. Comparison of resampling schemes for particle filtering, in Proc. Intell. Symp. Image and Signal Processing and Analysis, pp. 64–69, IEEE, 2005.

[58] EMSA. European Maritime Safety Agency. Annual Overview of Marine Casualties and Incidents 2019. Retrieved from www.emsa.europa.eu on 25.09.2020. Lisboa, Portugal, 2019.

[59] FAA General Operating and light Rules, 14 C.F.R., Chapter 1, Subchapter F, Part 91, 2017.

[60] FAA (NextGen). Federal Aviation Administration. Next Generation Air Transportation System. Visited 19. May 2020. http://www.faa.goc/nextgen/. 2020

[61] Faltinsen, O. M. Sea Loads on Ships and Offshore Structures. Cambridge, U.K.: Cambridge Univ. Press, 1990.

[62] Feith, G. Avoiding the Beast Below. Business & Commercial Aviation. September 9 2002.

[63] Fossen, T. I. Guidance and Control of Ocean Vehicles. New York: Wiley, 1994.

[64] Fossen, T. I. Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles. Trondheim, Norway: Marine Cybernetics, 2002.

[65] Fossen, T. I., and Perez, T. Kalman Filtering for Positioning and Heading Control of Ships and Offshore Rigs. 32 IEEE Control Systems Magazine. 2009.

[66] Gasos, J., Carcia, M. C., and Garcia, R. Fuzzy strategies for the navigation of autonomous mobile robots. Proc. I.F.E.S., pp. 1024–1034, 1991.

[67] Gao, Z., and Ho, D. W. State/noise estimator for decriptor systems with application to sensor fault diagnosis. IEEE Trans. Signal Proces., vol. 54, no. 4, pp. 1316-1326, 2006.

[68] Gao, Z., and Ding, S. X. Actuator fault robust estimation and fault-tolerant control for a class of nonlinear descriptor systems. Automatica, vol. 43, no. 5, pp. 912-920, 2007.

[69] Goodwin, E. M. A statistical study of ship domain, The Journal of Navigation, Vol. 28, No. 3, pp. 328-344 (1975).

[70] Gumaste, A., Singhai, R., and Sahoo, A. Intellicarts: Intelligent car transportation system. In 15th IEEE Local Area Networks and Metro Area Networks (LANMAN), Princeton, NJ, 2007.

[71] Harman, W. H. TCAS: A System for Preventing Midair Collisions. The Lincoln Laboratory Journal. Vol. 2, No. 3, pp. 437-457, 1989.

[72] Hasegawa K. Automatic collision avoidance system for ship using fuzzy control. In: Proceedings of 8th ship control system symposium, pp. 234–258, 1987.

[73] Hauskrecht, M. Planning and Control in Stochastic Domains with Imperfect Information. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1997.

[74] Higuchi, T. Monte Carlo filter using the genetic algorithm operators. J. Stat. Comput. Simul. 59, 1–23. 1997.

[75] Hiraga, I., Furuhashi, T., Uchikawa, Y., and Nakayama, S. An acquisition of operator's rules for collision avoidance using fuzzy neural networks, IEEE Transactions on Fuzzy Systems, Vol. 3, No. 3, pp. 280-287, 1995.

[76] Hiremath, S.A., Van Der Heijden, G. W., Van Evert, F. K., Stein, A., and Ter Braak, C. J. Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter. Comput. Electron. Agr., vol. 100, pp. 41-50, 2014.

[77] Hoekstra, J.M., Ruigrok, R. C. J., van Gent R. N. H. W., Visser J. M. P. C. M., Gijsbers, B., Clari, M. S. V. V., Heesbeen, W. W. M., Hilburn, B. G., Groeneweg, J., and Bussink, F. J. L. Overview of NLR free flight project 1997-1999. Nationaal Lucht en Ruimtevaartlaboratorium – Publications – NLR TP, (227), 2000.

[78] Hoekstra, J. M., van Gent, R. N. H. W., and Ruigrok, R. C. J. Designing for safety: the "free Flight" air traffic management concept. Reliability Engineering & System Safety, 75(2): pp. 215-232. 2002.

[79] Holenstein, A.A., and Badreddin, E. Collision Avoidance in a Behavior-Based Mobile Robot Design, 1991 IEEE International Conference on Robotics and Automation, Sacramento, CA, pp 898-903, April 1991.

[80] Hollnagel, E. FRAM: The Functional Resonance Analysis Method: Modelling Complex Socio-Technical Systems; CRC Press: Boca Raton, FL, USA, 2017.

[81] Howard, R. Dynamic Programming and Markov Decision Processes. MIT Press, Cambridge, MA, 1960.

[82] Hwang, C. N. The Integrated Design of Fuzzy Collision-Avoidance and H_∞–Autopilots on Ships. The Journal of Navigation, 55, pp. 117-136, 2002.

[83] Hwang, I., and Seah, C. E. Intent-Based Probabilistic Conflict Detection for the Next Generation Air Transportation System. Proceedings of the IEEE, 96(12): pp. 2040-2059, 2008.

[84] Hong, X., Harris, C. J., and Wilson, P. A. Autonomous ship collision free trajectory navigation and control algorithms. Proceedings of 1999 7th IEEE International Conference on Emerging Technologies and Factory Automation. ETFA '99. pp. 923–929, 1999.

[85] IALA. International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA). E-Navigation – IALA ASIM. Accessed on 11. June 2020. http://www.iala-asim.org/products-projects/e-navigation/. 2020.

[86] IALA. International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA). IALA Guideline 1082: An Overview of AIS, Edition 2.0, 2016.

[87] Iijima, Y., and Hagiwara, H. Results of collision avoidance manouvre experiments using a knowledge-based autonomous piloting system. The Journal of Navigation, 44, pp. 194–204, 1991.

[88] IMO. International Maritime Organization. E-navigation. Accessed 20. July 2020. http://www.imo.org/en/OurWork/safety/navigation/pages/enavigation.aspx. 2020.

[89] IMO. Guidance on Ergonomic Criteria for Bridge Equipment and Layouts, MSC/Circ. 982. 2000.

[90] IMO. Human Element Vision, Principles and Goals of the Organisation. International Maritime Organisation, Resolution A.947(23). Adopted on 27 November 2003.

[91] International Maritime Organization. The E-navigation Strategy Implementation Plan (SIP), in MSC 94; IMO: London, UK, 2014.

[92] International Regulations for Preventing Collisions at Sea. Archived on 22-Sep-2009. Retrieved from https://www.jag.navy.mil/distrib/instructions/COLREG-1972.pdf on 27-Feb-2020.

[93] Isard, M., and Blake, A. CONDENSATION: conditional density propagation for visual tracking. International Journal of Computer Vision, 29(1), pp. 5-28, 1998.

[94] Ito, M., Zhang, F., and Yoshida, N. Collision avoidance control of ship with genetic algorithm. Proceedings of the 1999 IOOO international conference on control applications pp. 1791–1796, 1999.

[95] Iwasaki, H., and Hara, K. A fuzzy reasoning model to decide the collision avoidance action. Journal of the Japanese Institute of Navigation, Vol. 75, No. 2, pp. 121–129, 1987.

[96] James, M. K. Modelling the decision process in computer simulation of ship navigation. The Journal of Navigation, 39, pp. 32–48, 1986.

[97] Jamshidi M., Vadiee N., and Ross T. (Eds) Fuzzy Logic and Control -- Software and Hardware Applications. Prentice-Hall, Englewood Cliffs, NJ, 1993.

[98] Jing, X., Michalewicz, Z., Lixin, Z., and Trojanowski, K. Adaptive evolutionary planner/navigator for mobile robots. Evolutionary Computation, IEEE Transactions, 1, pp. 18–28, 1997.

[99] Jing, L., Vadakkepat, P. Interacting MCMC particle filter for tracking maneuvering target. Digit. Signal Process. 20, 561–574. 2010.

[100] Jingsong, Z., Wang, F., and Wu, Z. L. The development of ship collision avoidance automation. The Journal of Navigation, Vol. 45, No. 1, pp. 107–113, 1992.

[101] Jones, K. D. Application of a manoeuvre diagram to multi-ship encounters. The Journal of Navigation, 27, pp. 19–27, 1974.

[102] Kalami Heris, S. M., and Khaloozadeh, H. Ant Colony Estimator: An intelligent particle filter based on ACO_R. Engineering Applications of Artificial Intelligence 28, pp. 78-85. 2014.

[103] Kao, S. L., Lee, K. T., Chang, K. Y., and Ko, M. D. A fuzzy logic method for collision avoidance in vessel traffic service, The Journal of Navigation, Vol. 60, No. 1, pp. 1-15, 2007.

[104] Karasev, V., Ayvaci, A., Heisele, B., and Soatto, S. Intent-aware long-term prediction of pedestrian motion. Proceedings of the International Conference on Robotics and Automation (ICRA), 2016.

[105] Kemp, J. F. Behaviour patterns in encounters between ships. The Journal of Navigation, 26, pp. 417-423, 1973.

[106] Khatib, O. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots, The International Journal of Robotics Research, Vol 5, No 1, pp. 90-98, 1986.

[107] Kingma, D. P., and Ba, J. Adam: A method for stochastic optimization. ArXiv preprint arXiv: 1412.6980. 2014

[108] Kochenderfer, M. J. Decision Making Under Uncertainty. MIT Press, Boston, Massachusetts, 2015.

[109] Koren, Y., and Borenstein, J. Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation, 1991 IEEE International Conference on Robotics and Automation, Sacramento, CA, pp 1398-1404, April 1991.

[110] Kosko, B. Neural networks and fuzzy systems. Prentice-Hall. 1992.

[111] Kosko, B. Fuzzy Thinking: The New Science of Fuzzy Logic. Hyperion. June 1, 1994.

[112] Kuc, R., and Barshan, B., Navigating Vehicles Through an Unstructured Environment With Sonar, 1989 IEEE International Conference on Robotics and Automation, Scottsdale, AZ, pp 1422-1426, May 1989.

[113] Kuchar, J. K. A Unified Methodology for the Evaluation of Hazard Alerting Systems. MIT Aeronautics and Astronautics PhD thesis, Cambridge, MA, ASL-95-1, 1995.

[114] Kuchar, J. K., and Yang, L. C. A Review of Conflict Detection and Resolution Modeling Methods. IEEE Transactions on Intelligent Transportation Systems, Vol. 1, No. 4. December 2000.

[115] Kurniawati, H., Hsu, D., and Lee, W. S. SARSOP: efficient point-based POMDP planning by approximating optimally reachable belief spaces. In Proceedings in Robotics: Science and Systems, 2008.

[116] Kwok, N., Fang, G., and Zhou, W. Evolutionary particle filter: re-sampling from the genetic algorithm perspective. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), pp. 2935–2940. 2005.

[117] Lee, Y. I., and Kim, Y. G. A collision avoidance system for autonomous ship using fuzzy relational products and COLREGs. Intelligent Data Engineering And Automated Learning Ideal 2004, Proceedings. Berlin, Springer-Verlag Berlin, 2004.

[118] Lee, H. J., and Rhee, K. P. Development of collision avoidance system by using expert system and search algorithm. Journal of International Shipbuilding Progress 48, pp. 197–212, 2001.

[119] Lefeber, E., Pettersen, K. Y., and Nijmeijer, H. Tracking control of an underactuated ship. IEEE transactions on control systems technology, vol. 11, no. 1, pp. 52–61, 2003.

[120] Lewis, F. L. Optimal Estimation: With an Introduction to Stochastic Control Theory, John Wiley & Sons, 1986.

[121] Lewis, T. A., Phojanamongkolkij, N., and Wing, D. J. The effects of limited intent

information availability on self-separation in mixed operations. Integrated Communications, Navigation and Surveilance Conference (ICNS), IEEE, pp. B6-1, 2012.

[122] Lin, H. S., J, X., and Michalewicz, Z. Evolutionary algorithm for path planning in mobile robot environment. IEEE World Congress on Computational Intelligence, Proceedings of the First IEEE Conference, pp. 211–216, 1994.

[123] Lind, M, Hagg, M, Siwe, U., and Haraldson, S. Sea Traffic Management – Beneficial for all Maritime Stakeholders. Proceedings of 6th Transport Research Arena, Warsaw, Poland, 2016.

[124] Lisowski, J. A simulation study of various approximate models of ship dynamics in collision avoidance problem. Foundations in Control Engineering, 10, pp. 176–183, 1985.

[125] Lisowski, J., and Smierzchalski, R. Assigning of safe and optimal trajectory avoiding collision at sea. In Proceedings of 3rd IFAC Workshop Contr. Appl. Marine Syst., 346–350, 1995.

[126] Liu, Y. H., and Shi, C. J. A fuzzy-neural inference network for ship collision avoidance. Proceedings of the 4th international conference on machine learning and cybernetics, Guangzhou, China, 2005.

[127] Liu, Z., Zhang, Y., Yu, X., and Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. Annual Reviews in Control, vol. 41, pp. 71–93, 2016.

[128] Liu, J., Zhou, F., Li, Z., Wang, M., and Liu, R. W. Dynamic ship domain models for capacity analysis of restricted water channels. The Journal of Navigation, 69, pp. 481-503, 2016.

[129] Lopez-Leones, J., Vilaplana, M.A., Gallo, E., Navarro, F. A., and Querejeta, C. The Aircraft Intent Description Language: A key enabler for air-ground synchronization in Trajectory-Based Operations. 2007 IEEE/AIAA 26th Digital Avionics Systems Conference, pp. 1.D.4-1-1.D.4-12, 2007.

[130] Lopez Leones, J. Definition of an aircraft intent description language for air traffic management applications. PHD Thesis. University of Glasgow, 2008.

[131] Macrae, C. Huma factors at sea: common patterns of error in groundings and collisions. Maritime Policy & Management: 35(01), 21-38. 2009.

[132] MAIB. Bridge Watch-keeping Safety Study. Safety Study 1/2004, Southampton, MAIB. 2004.

[133] MarineTraffic. MarineTraffic: Global Ship Tracking Intelligence, AIS Marine Traffic. Retrieved July 11, 2020, from https://www.marinetraffic.com., 2020.

[134] Martinez, A., Tunstel, E., and Jamshidi M. Fuzzy Logic Based Collision Avoidance For a Mobile Robot. Robotica, 1994.

[135] Mataric, M. J. Reward functions for accelerated learning. In Proceedings of the Eleventh International Conference on Machine Learning. Morgan Kaufmann, 1994.

[136] Mayne, D. Q., Rawlings, J. B., Rao, C. V, and Scokaert, P. O. M. Constrained model predictive control: Stability and optimality. Automatica 36, pp. 789-814, 2000.

[137] McCulloch, W., and Pitts, W. A Logical Calculus of Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics 5 (4): pp. 115–133, 1943.

[138] Merriam-Webster. Definition of INTENT. Visited on 23. June 2020. https://www.merriam-webster.com/dictionary/intent., 2020.

[139] Mertz, C., McNeil, S., and Thorpe, C. Side collision warning systems for transit buses. In IV 2000, IEEE Intelligent Vehicle Symposium, October 2000.

[140] Merz, A. W., and Karmakar, J. S. Collision avoidance systems and optimal turn manoeuvres. The Journal of Navigation, 29, pp. 160–174, 1976.

[141] Miele, A., Wang, T., Chao, C. S., and Dabney, J. B. Optimal control of a ship for collision avoidance manoeuvres. Journal of Optimization Theory and Applications 103, pp. 495–519, 1999.

[142] Mnih, Volodymyr, Kavukcuoglu, Koray, Silver, David, Rusu, Andrei A, Veness, Joel, Bellemare, Marc G, Graves, Alex, Riedmiller, Martin, Fidjeland, Andreas K, Ostrovski, Georg, et al. Human-level control through deep reinforcement learning. Nature, 518(7540), pp. 529–533, 2015.

[143] Minsky, M., and Papert, S. An Introduction to Computational Geometry. MIT Press, 1969.

[144] Mitrofanov, O. An anti-collision indicator. The Journal of Navigation, 21, pp. 163–170, 1968.

[145] Mohovic, Dj., Mohovic, R., and Baric, M. Deficiencies in Learning COLREGs and New Teaching Methodology for Nautical Engineering Students and Seafarers in Lifelong Learning Programs. The Journal of Navigation, 69, pp. 765-776, 2016.

[146] Mondolini, S. Flight Object – A Recommendation for Flight Script and Trajectory Description, 2006.

[147] Moravec. H. P. Sensor Fusion in Certainty Grids for Mobile Robots, AI Magazine, pp 61-74, Summer, 1988.

[148] Moravec, H. P., and Elfes, A. E. High resolution map from wide-angle sonar. Proceedings of the IEEE International Conference on Robotics and Automation. Los

Alamitos, California, 1985.

[149] Morrel, J. S. The physics of collision at sea. The Journal of Navigation, 14, pp. 163–184, 1961.

[150] Mukai, M., Harada, Y., Murata, J., Kawabe, T., Nishira, H., and Deguchi, Y. An automotive collision avoidance control based on a feasible set. In ICCAS-SICE, pages 2164–2168, August 2009.

[151] Nádeník, Z. Legendre Theorem on Spherical Triangles. Research Institute of Geodesy, Topography and Cartography, 50(36), 41–48, 2004.

[152] Naeem, W., Irwin, G. W., and Yang, A. Colregs-based collision avoidance strategies for unmanned surface vehicles. Mechatronics, vol. 22, no. 6, pp. 669–678, 2012.

[153] Newman, J. N. Marine Hydrodynamics. Cambridge, MA: MIT Press, 1977.

[154] Ng, A. Y., Harada, D., and Russell, S. Policy invariance under reward transformations: Theory and application to reward shaping. In Proceedings of the Sixteenth International Conference on Machine Learning, pp. 278-287, Bled, Slovenia, 1999. Morgan Kaufmann.

[155] Ng, A. Y., and Russell, S. Algorithms for inverse reinforcement learning. In Proceedings of the Seventeenth International Conference on Machine Learning, 2000.

[156] Ng, A. Y., Kim, H. J, Jordan, M. I., and Sastry. S. Autonomous helicopter flight via reinforcement learning. In NIPS, 2003.

[157] Nordwall, D. TCAS More 'Foolproof' Than Generally Recognized. Aviation Week & Space Technology, July 15 2002.

[158] NTSB – National Transportation and Safety Board. Marine Accident Report. Collision between US Navy Destroyer Fitzgerald and Philippine-Flag COntainer Ship ACX Crystal. NTSB/MAR-20/02, 2020.

[159] O'Hara, C. Friendlier Skies. Federal Computer Week. FCW Media Group. March 30 1998.

[160] Paquet, S., Tobin, L., and Chaib-draa, B. Real-time decision making for large POMDPs. In: Advances in Artificial Intelligence (LNAI 3501), pp. 450–455, 2005.

[161] Park, S., Hwang, J.P., Kim, E., and Kang, H.-J. A new evolutionary particle filter for the prevention of sample impoverishment. IEEE Trans. Evol. Comput. 13, 801–809. 2009.

[162] Pasaoglu, C., Akcam, N., Koyuncu, E., Tarhan, A. F., and Inalhan, G. Collaborative Intent Exchange Based Flight Management System with Airborne Collision

Avoidance for UAS. Journal of Intelligent & Robotic Systems, 84(1-4): pp. 665-690, 2016.

[163] Patterson, D. W. Artificial Neural Networks: Theory and Applications, Singapore, Prentice Hall, 1996.

[164] Peng, Z., Wang, D., Chen, Z., Hu, X., and Lan, W. Adaptive dynamic surface control for formations of autonomous surface vehicles with uncertain dynamics. IEEE Transactions on Control Systems Technology, vol. 21, no. 2, pp. 513–520, 2013.

[165] Phillips, E. H. Aviation Week: Global Teamwork Called Key to Improving Aviation Safety. Aviation Week & Space Technology. July 16 2001.

[166] PIANC. Capability of ship maneuvering simulation models for approach channels and fairways in harbors. Report of Working Group No. 20 of Permanent Technical Committee II, Supplement to PIANC Bulletin No. 77, pp. 49, 1992.

[167] Pietrzykowski, Z. Ship's fuzzy domain - a criterion for navigational safety in narrow fairways, The Journal of Navigation, Vol. 61, No. 3, pp. 499-514, 2008.

[168] Pietrzykowski, Z., and Uriasz, J. The ship domain - a criterion of navigational safety assessment in an open sea area, The Journal of Navigation, Vol. 62, No. 1, pp. 93-108, 2009.

[169] Pineau, J., and Thrun, S. High-level robot behavior control using POMDPs. In AAAI Workshop Notes, 2002.

[170] Pineau, J., Gordon, G., and Thrun, S. Point-based value iteration: An anytime algorithm for POMDPs. International Joint Conference on Artificial Intelligence (IJCAI), pp. 1025–1032, Acapulco, Mexico, 2003.

[171] Porathe, T. Transmitting intended and suggested routes in ship operations: cognitive off-loading by placing knowledge in the world. Work, 41 (Supplement 1): pp. 4873-4878, 2012.

[172] Porathe, T. Safety of autonomous shipping: COLREGS and interaction between manned and unmanned ships. Proceedings of the 29th European Safety and Reliability Conference, 2019.

[173] Porather, T., and Shaw, G. Working with the human element: Human Factors and technical innovations from EfficienSea and on to ACCSEAS. Chalmers Publication Library (CPL), 2012.

[174] Psarafits, H.N, Cardis, P., and Desypris, N. The human element as a factor in marine accidents. www.martrans.org Retrieved on 24 August 2020. Martrans, 2000.

[175] Puterman, M. L. Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons, Inc., New York, NY, USA, 1994.

[176] Randløv, J., and Alstrøm, P. Learning to drive a bicycle using reinforcement learning and shaping. In Proceedings of the Fifteenth International Conference on Machine Learning, 1998.

[177] Rawson, A., Rogers, E., Foster, D., and Phillips, D. Practical application of domain analysis: port of London case study. The Journal of Navigation, 67, pp. 193-209, 2014.

[178] Rihacek, C., Lutz, T., Weinert, B., Bolles, A., Nielsen, K., and Jensen J. K. EfficienSea2 – Conceptual Model of the Maritime Cloud. Accessed 18. May 2020. http://efficiensea2.org. 2020

[179] Ristic, B., Arulampalam, S., and Gordon, N. Beyond the Kalman Filter: Particle Filters for Tracking Applications. Artech House. 2004.

[180] Rommelfanger H. J. Multicriteria decision making using fuzzy logic. In: Bezdek J, Hall LO (eds) Proceedings of conference of the North American fuzzy information processing society, Pensacola Beach, FL, USA, pp 360–364, 1998.

[181] Rosenblatt, J. K. Optimal selection of uncertain actions by maximizing expected utility. Autonomous Robots, 9(1), pp. 17-25, 2000.

[182] Ross, S., Pineau, J., Paquet, S., and Chaib-draa, B. Online planning algorithms for POMDPs. Journal of Artificial Intelligence. Res. 32, pp. 663–704, 2008.

[183] Rothblum A. M., Wheal D., Withington S., Shappell S. A., Wiegmann D. A., Boehm W., and Chaderjian M. Key to successful incident inquiry. In: Proceedings 2nd international workshop on human factors in offshore operations (HFW2002), Houston, TX, pp 1–6, 2002.

[184] RTCA, Inc. Minimum operational performance standards for traffic alert and collision avoidance system II (TCAS II) airborne equipment. Technical Report RTCA/DO-185A, SC-147, Washington, DC, June 2008.

[185] Rubin, D. B. Using the SIR algorithm to simulate posterior distributions. In Bayesian Statistics 3. Oxford Univ. Press, 1988.

[186] Rudan, I., Francic, V., Valcic, M., and Sumner, M. Early detection of vessel collision situations in a VTS area. Transport, pp. 1-19, 2019.

[187] Ruigrok, R. C. J., Tessier, C., Imbert, N., Farges, J. L., Hoffman, E., Nuic, A., Shaw, C., Bouassida, M., Platt, P., and Magill, A. INTENT Project Scope and Relevant Results of other Projects, Activities and Initiatives. Technical Report D1-1, National Aerospace Laboratory, NLR, 2001.

[188] Ruigrok, R. C. J., and Valenti Clari, M. S. The impact of aircraft intent information and traffic separation assurance responsibility on en-route airspace capacity. 5th FAA/EUROCONTROL ATM R&D Seminar, 2003.

[189] Russel, S., and Norvig, P. Artificial Intelligence: A Modern Approach. 2nd ed. Prentice Hall. Pearson Education, Inc, NJ. 2003.

[190] Russel, S. J., and Zimdars, A. Q0decomposition for reinforcement learning agents. International Conference on Machine Learning, 2003.

[191] Rutkowski, G. Determining Ship's Safe Speed and Best Possible Speed for Sea Voyage Legs. Transnav, 10(3), pp. 425-430, 2016.

[192] Rydlinger, A. Mona Lisa 2.0 – STM Voyage Exchange Format and Architecture. Accessed on 11. April 2020. http://stmvalidation.eu. 2020.

[193] Sælid, S., Jenssen, N. A., and Balchen, J. G. Design and Analysis of a Dynamic Positioning System based on kalman Filtering and Optimal Control. IEEE Transactions on Automatic Control, TAC-28 (3): 331-339, 1983.

[194] Saksida, L. M., Raymond, S. M., and Youretzky, D. S. Shaping robot behavior using principles from instrumental conditioning. Robotics and Autonomous Systems, 22(3-4), pp. 231-249, 1997.

[195] Samanant, P., Jackson, M., Haissig, C. M., and Corwin, B. CASPER/AILS: An Integrated DGPS/ADS-B Airborne Alerting System For Closely Spaced Parallel Approaches. IEEE Plans, Position Location and Navigation Symposium, pp. 57-65, March 13-16, 2000.

[196] SESAR. Single European Sky ATM Research Joint Undertaking. Visited 18. May 2020. http://www.sesarju.eu/. 2020.

[197] Simon, D. Optimal state estimation: Kalman, H infinity, and nonlinear approaches. John Wiley & Sons, 2006.

[198] Simmons, R., and Koenig, S. Probabilistic robot navigation in partially observable environments. In Proceedings of the International Joint Conference on Artificial Intelligence, pp. 1080–1087, 1995.

[199] Skinner, B. F. The Behavior of Organisms. D. Appleton & Company, 1938.

[200] Skjong, R., and Mjelde, K. M. Optimal evasive manoeuvre for ship in an environment of fixed installations and other ships. Identification and Control, 3, pp. 211–222, 1982.

[201] Smierzchalski, R. Evolutionary trajectory planning of ships in navigation traffic areas. Journal of Marine Science and Technology 4, pp. 1–6, 1999.

[202] Smierzchalski, R., and Michalewicz, Z. Modeling of ship trajectory in collision situations by an evolutionary algorithm, IEEE Transactions on Evolutionary Computation, Vol. 4, pp. 227-241, 2000.

[203] Smith, A.F.M., and Gelfand, A.E. Bayesian statistics without tears: A sampling-resampling perspective. American Statistician 46(2), pp. 84-88, 1992.

[204] Smith, T., and Simmons, R.G. Heuristic search value iteration for POMDPs. Proceedings of International Conference on Uncertainty in Artificial Intelligence (UAI), 2004.

[205] Statheros, T., Howells, G., and McDonald-Maier, K. Autonomous ship collision avoidance navigation concepts, technologies and techniques, The Journal of Navigation, Vol. 61, No. 1, pp. 129-142, 2008.

[206] Stentz, A. Optimal and efficient path planning for partially-known environments. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, pp. 3310-3317, vol. 4, May 1994.

[207] Stitt, I. P. A. The COLREGs – Time for a Rewrite? The Journal of Navigation, 55, pp. 419-430, 2002.

[208] STM. Sea Traffic Management Validation Project: Final Report; STM Authority: Göteborg, Sweden, 2019; p. 106.

[209] Su, C. M., Chang, K. Y., and Cheng, C. Y. Fuzzy Decision on Optimal Collision Avoidance Measures For Ships in Vessel Traffic Service. Journal of Marine Science and Technology, Vol. 20, No. 1, pp. 38-48, 2012.

[210] Sutton, R. S., and Barto, A. G. Reinforcement Learning: An Introduction. MIT Press, 1998.

[211] Swedish Club. Claims at a Glance 2019. Retrieved from www.swedishclub.com on 24.09.2020. Gothenburg, Sweden, 2019.

[212] Szlapcynski, R. A new method of ship routing on raster grids, with turn penalties and collision avoidance. The Journal of Navigation, 59, pp. 27–42, 2006.

[213] Szlapczynski, R. A Unified Measure of Collision Risk Derived from The Concept of a Ship Domain. The Journal of Navigation, 59, pp. 477-490, 2006.

[214] Szlapczynski, R., and Szlapczynska, J. An analysis of domain-based ship collision risk parameters. Ocean Engineering, 126, pp. 47-56, 2016.

[215] Szlapczynski, R., and Szlapczynska, J. Review of ship safety domains: Models and applications. Ocean Engineering, 145, pp. 277-289, 2017.

[216] Tam, C. K., Bucknall, R., and Greig, A. Review of collision avoidance and path planning methods for ships in close range encounters, The Journal of Navigation, Vol. 62, No. 3, pp. 455-476, 2009.

[217] Tarhan, A. F., Koyuncu, E., Hasanzade, M., Ozdemir, U., and Inalhan, G. Formal

intent based Flight Management System design for unmanned aerial vehicles. International Conference on Unmanned Aircraft Systems (ICUAS), pp. 984-992, 2014.

[218] Tong, G., Fang, Z., and Xu, X. A particle swarm optimized particle filter for nonlinear system state estimation. In: IEEE Congress on Evolutionary Computation, 2006 (CEC 2006), pp. 438–442. 2006.

[219] Tran, L. D., Cross, C. D., Motter, M. A., Neilan, J. H., Qualls, G., Rothhaar, P. M, Trujillo, A., and Allen, B. D. Reinforcement learning with autonomous small unmanned aerial vehicles in cluttered environments. Aviation Technology, Integration, and Operations Conference, p. 2899, 2015.

[220] Transas. Description of Transas Mathematical Model. Version 02.08, Transas Ltd., 2011.

[221] Ugurlu, O., Yildirim, U., and Basar, E. Analysis of grounding accidents caused by human error. Journal of Marine Science and Technology, Vo. 23, No. 5, pp. 748-760, 2015.

[222] Uosaki, K., and Hatanaka, T. Evolution strategies based particle filters for fault detection. In: IEEE Symposium on Computational Intelligence in Image and Signal Processing, 2007 (CIISP 2007), pp. 58–65, 2007.

[223] Uosaki, K., Kimura, Y., and Hatanaka, T. Evolution strategies based particle filters or state and parameter estimation on nonlinear models. In: Congress on Evolutionary Computation, 2004 (CEC2004), vol. 1, pp. 884–890, 2004.

[224] van Gool, M., and Schröter, H. PHARE Final Report. 1999.

[225] Vilaplana, M. A., Gallo, E., Navarro, F. A., and Swierstra, S. Towards a formal language for the common description of aircraft intent. 24^{th} Digital Avionics Systems Conference, volume 1, pp. 3.C.5-3.1.-9 Vol. 1, 2005.

[226] Vonk, E., Jain, L. C., and Johnson, R. P. Automatic Generation of Neural Network Architecture Using Evolutionary Computation. World Scientific, 1997.

[227] Wagenaar W.A., and Groeneweg J. Accidents at sea: Multiple cause and impossible consequence, International Journal of Man-Machine Studies, 27, 1987.

[228] Wang, N., Meng, X., Xu, Q., and Wang, Z. An intelligent spatial collision risk based on the quaternion ship domain. The Journal of Navigation, 63, pp. 733-749, 2010.

[229] Wang, P., and Gao, R.X. Markov Nonlinear System Estimation for Engine Performance Tracking. Journal of Engineering for Gas Turbines and Power, 138 (9), pp. 091201-1-091201-10, 2016.

[230] Warren, A. Trajectory prediction concepts for next generation air traffic

management. 3rd USA/Europe ATM R&D Seminar, pp. 171, 2000.

[231] Wiewiora, E., Cottrell, G., and Elkan, C. Principled methods for advising reinforcement learning agents. In Proceedings of International Convention of Machine Learning, pp. 792-799, 2003.

[232] Williams, G., Wagener, N., Goldfain, B., Drews, P., Rehg, J. M., Boots, B., and Theodorou, E. A. Information theoretic MPC for model-based reinforcement learning. International Conference on Robotics and Automation (ICRA), pp. 1714–1721, 2017.

[233] Williamson, T. Vagueness. Routledge, London, p. 258, 1996.

[234] Wilson, P. A., Harris, C. J., and Hong, X. A line of sight counteraction navigation algorithm for ship encounters collision avoidance. Journal Of Navigation, 56, pp. 111–121, 2003.

[235] Winder, L. F., and Kuchar, J. K. Evaluation of Vertical Collision Avoidance Maneuvers for Parallel Approach. AIAA, Journal of Guidance, Control and Dynamics. November-December, 1999.

[236] Windy. Windyte S.E. Meteorological service. Retrieved July 12, 2020, from https://www.windy.com., 2020.

[237] Wing, D., Barmore, B., and Krishnamurthy, K. Use of Traffic Intent Information by Autonomous Aircraft in Constrained Operations. American Institute of Aeronautics and Astronautics, 2002.

[238] Wolf, T. B., and Kochenderfer, M. J. Aircraft collision avoidance using Monte Carlo real-time belief space search. Journal of Intelligent and Robotic Systems, pp. 277-298, 2011.

[239] Wylie, Captain F. J. Mathematics and the collision regulations. The Journal of Navigation, 15, pp. 104–112, 1962.

[240] Xianyi, Y. A Neural Network Approach to Real-Time Collision-Free Navigation of 3-D.O.F. Robots in 2D. ICRA, pp. 23–28, 1999.

[241] Xiaolong, L., Jinfu, F., Qian, L., Taorong, L., and Bingjie, L. A swarm intelligence optimization for particle filter. In: The Seventh World Congress on Intelligent Control and Automation, 2008 (WCICA 2008), pp. 1986–1991. 2008.

[242] Xu, B., Chen, Q., Wang, X., and Zhu, J. A novel estimator with moving ants. Simul. Modell. Pract. Theory 17, 1663–1677. 2009.

[243] Yang, L.C., and Kuchar, J.K. Prototype conflict alerting system for free flight. Journal of Guidance , Control and Dynamics. AIAA. Vol. 20, No. 4, July-August 1997.

[244] Yang, L. C., and Kuchar, J. K. Performance Metric Alerting: A New Design Approach for Complex Alerting Problems. IEEE Transactions on Systems, Man and Cybernetics—Part A: Systems and Humans, Vol. 32, No. 1. January 2002.

[245] Yavin, Y., Frangos, C., Miloh, T., and Zilman, G. Collision avoidance by a ship with a moving obstacle: Computation of feasible command strategies. Journal Of Optimization Theory And Applications, 93, pp. 53–66, 1997.

[246] Yavin, Y., Miloh, T., and Zilman, G. Parametric Study Of Ship Maneuverability In Laterally Restricted Waters – Stochastic-Control Approach. Journal Of Optimization Theory And Applications, 85, pp. 59–74, 1995.

[247] Yavin, Y., Zilman, G., and Miloh, T. A Feasibility Study Of Ship Maneuverability In The Vicinity Of An Obstacle – A Stochastic-Control Approach. Computers & Mathematics With Applications, 28, pp. 63–76, 1994.

[248] Yin, S., Zhu, X., Qiu, J., and Gao, H. State Estimation in Nonlinear System Using Sequential Evolutionary Filter. IEEE Transaction on Industrial Electronics, 63(6), pp. 3786-3794. 2016.

[249] Zadeh, L. A. Fuzzy sets. Information and Control, Vol. 8, pp. 338–353, June 1965.

[250] Zeng, X. Evolution of the safe path for ship navigation. Applied artificial intelligence 17:2, pp. 87–104, 2003.

[251] Zekic, A., Mohovic, Dj., and Mohovic, R. Analysis of the level of knowledge and understanding of regulations for preventing collisions at sea. Scientific Journal of Maritime Research 29, pp. 143-149, Rijeka, 2015.

[252] Zheng, Y., and Meng, Y. Swarming particles with multi-feature model for free-selected object tracking. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008 (IROS 2008), pp. 2553–2558. 2008.

[253] Zhu, X., Xu, H., and Lin, J. Domain and its model based on neural networks. The Journal of Navigation, Vol. 54, No. 1, pp. 97-103, 2001.

[254] Zhuo, Y., and Tang, T. An intelligent decision system to ship anti-collision in multi-ship encounter. Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China, pp. 1066-1071, 2008.

[255] Zhu, J., Xu, B., Wang, F., and Wang, Z. A real-time moving ant estimator for bearings-only tracking. In: Tan, Y., Shi, Y., Tan, K. (Eds.), Advances in Swarm Intelligence, vol. 6145. Springer, Berlin/Heidelberg, pp. 273–280. 2010.

Appendix A

Reward Shaping Experiment

In order to maintain simplicity in envisioning benefits of reward shaping, a gridworld domain was developed where a floating object has a shortest path objective from the origin to the goal waypoint. In this case it is a simple shortest path challenge, without any collision avoidance, but with stationary object along the way. Uniform depth without shallow water patches has been assumed. The domain is depicted as a 7x10 grid-world. Each cell has a dimension determined by the size of the surface vehicle taken as an example (200 x 200 meters). To maintain simplicity, assumption is made that the action space comprises of only 4 compass directions (N, E, S, W) with possible movement of 1 step. As the agent receives -1 reward for each step it takes, additional penalties are added in order to avoid areas of an obstruction. Therefore, for cells closest to the shallow water the agent receives -2 reward, while -100 for hitting an island. It is also assumed that the agent has an information about distance from the goal and present position that is received by various sensors (sensing accuracy is not considered in this case). As it will be visible in the following sections, a certain percentage of navigating officers do not always follow collision regulations, so this is taken into account by assuming for the purpose of this test that in 90 % of cases navigators will sail in the intendent direction, while in 10 % navigation will be random. If the agent tries to sail out of the grid, he will remain in the same location.

To effectively test this approach, simplified navigation on a grid is utilized. In order to effectively estimate $V^*_{MDP}(s)$, sensory information about distance to the goal waypoint are utilized, as well as the knowledge about the position of the sea surface vehicle and the goal waypoint. This information is enough to aid the learning process of minimizing cost to reach the goal type of problems. Excluding all environmental influences, goal is to steer the vehicle from the present position towards the desired goal waypoint. As stated earlier, in this example input control and restrictions are irrelevant, so it is assumed that the vehicle can select any of the 4 compass directions instantly at any time-step point, which correspond to the size of the grid cell. Therefore, in each time step, own agent selects a direction in which it will steer the sea surface vehicle until it reaches the waypoint. The agent is mimicking decision delivery of a navigating officer and goal is to show the difference of making decision with and without expert's input (shaping) of reward function. Desired performance is to take as less as possible time-steps to reach the waypoint (assuming constant speed and excluding earth curvature). Reward function is initially set up as -1 for each time-step, as aim was to collect least number of penalty points until reaching the waypoint.

As the premise is 90 % chance of selecting appropriate step towards the waypoint, there is still 10 % chance of acting randomly (a thorough research has been made in thesis to actually determine real-world percentages). When acting randomly, the agent selects any of the actions from the action space, so there is an equal chance of selecting any of the eight available compass directions, including the one towards the waypoint. Therefore, if guessing the performance of the optimal policy, it would be that at each time step agent takes approximately 0.9 distance per time step, which brings us to the relation that it would take $d/_{0.9}$ of steps to reach the waypoint, where d is the radar distance from the origin until the waypoint. So, the estimate of the value function and reward shaping function $\Phi(s)$ is set to be $\Phi_0(s) = \hat{V}_{MDP}(s) = -\,d/_{0.9}$. However, as the aim is to incorporate knowledge that traces quality of actions taken, it is also necessary to trace behavior of the decision maker (human or the artificial agent). Real-world challenges of sailing seas, such are traffic, external influences, voyage planning, etc. are disregarded in this experiment. In the real-world instance it would be necessary to introduce an action tracking that would probably be tied to the tracking of the planned route and penalize any deviation from the planned route. However, in this case, agent is free to explore the whole domain. That is why it is necessary to track the action selection performance by assigning progressively increasing penalties for selecting an action that is further away from the current direction. For the initial action a_0, no penalties are assigned $a_0 = 0$. Certainly, it is possible to valorize actions in a better way but for the depicted experiment progressive penalties would suffice. In order to implement progressive penalties, $\Phi(a)$ is defined as:

$$\Phi(a) = \begin{cases} -0.8 & \text{if } (a' - a) = \pm 45^{\circ} \\ -0.6 & \text{if } (a' - a) = \pm 90^{\circ} \\ -0.4 & \text{if } (a' - a) = \pm 135^{\circ} \\ 0 & \text{if } (a' - a) = \pm 180^{\circ} \\ -1 & \text{otherwise} \end{cases} \qquad (A.1)$$

Finally, it is possible to incorporate both shaping functions by adding them up:

$$\Phi(s, a) = \Phi(s) + \Phi(a) \qquad (A.2)$$

Uncertainty is not implemented in the shaping function of an action space solely because it is already covered in the value of the state space, where distance from the goal divided with the 0.9 factor is taking that uncertainty into account. Therefore, in this case $\Phi(s, a) = {}^{d}/_{0.9} + \Phi(a)$, where $\Phi(a)$ depends on the conditional relationships described in (A.1). In conclusion, the estimate of the value function and reward shaping function $\Phi(s, a)$ is $\Phi_0(s, a) = \hat{V}_{MDP}(s, a) = -\left({}^{d}/_{0.9} + \Phi(a)\right)$. Tuning of hyperparameters for this training is: 0.10-greedy exploration, no discounting $\gamma = 0.9$, and learning rate $\alpha = 0.4$. Higher learning rate will allow the algorithm to rely strongly on recently observed events, while higher discount factor implies that the algorithm takes into consideration compounded experience of the agent. For this experiment ε-greedy exploration fits well, but the appropriate approach to exploration and exploitation needs to be tuned to each problem an analyst is trying to solve.

As mentioned earlier, this experiment is designed as a simple gridworld where own sea surface vehicle is allowed to move in 4 directions (in order to simplify Q-values representation): N, E, S, and W. The goal is to reach the goal state from the initial state by avoiding the fixed obstacle and to stay away from the obstacle with at least one gridworld's square distance. Figure A.1 represents the gridworld used in the experiment, while Figure A.2 represents the Q-values after 500 training episodes. As seen from Figure A.2, each field of the gridworld is divided into 4 triangular areas representing 4 possible actions own vehicle can take. The number in each of the triangle represents a potential value of taking that action, so in the red triangle it is possible to see negative values, while the dark green is the optimal action for that field. Other colors represent various values of taking other action than the optimal one, and this is due to randomness and inherent stochasticity of the process.

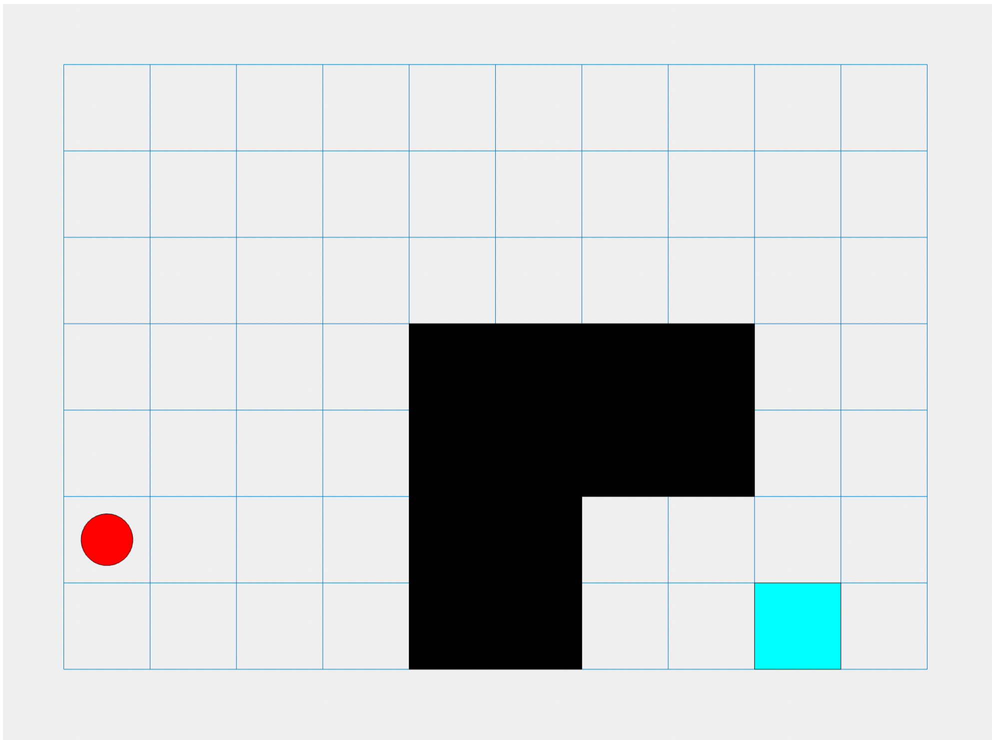


Figure A.1 – Gridworld experiment layout

The goal of this training was to find the optimal policy that would take own vehicle from the field [1, 6] (red dot on Figure A.1) to the goal field [9, 7] (blue square on Figure A.1). Therefore, if we would look at colored map of Figure A.3, it is noticeable that the optimal policy contains following actions to take: E, N, N, E, N, N, E, E, E, E, E, E, E, S, S, S, S, S, W. The optimal policy is depicted on Figure A.3.

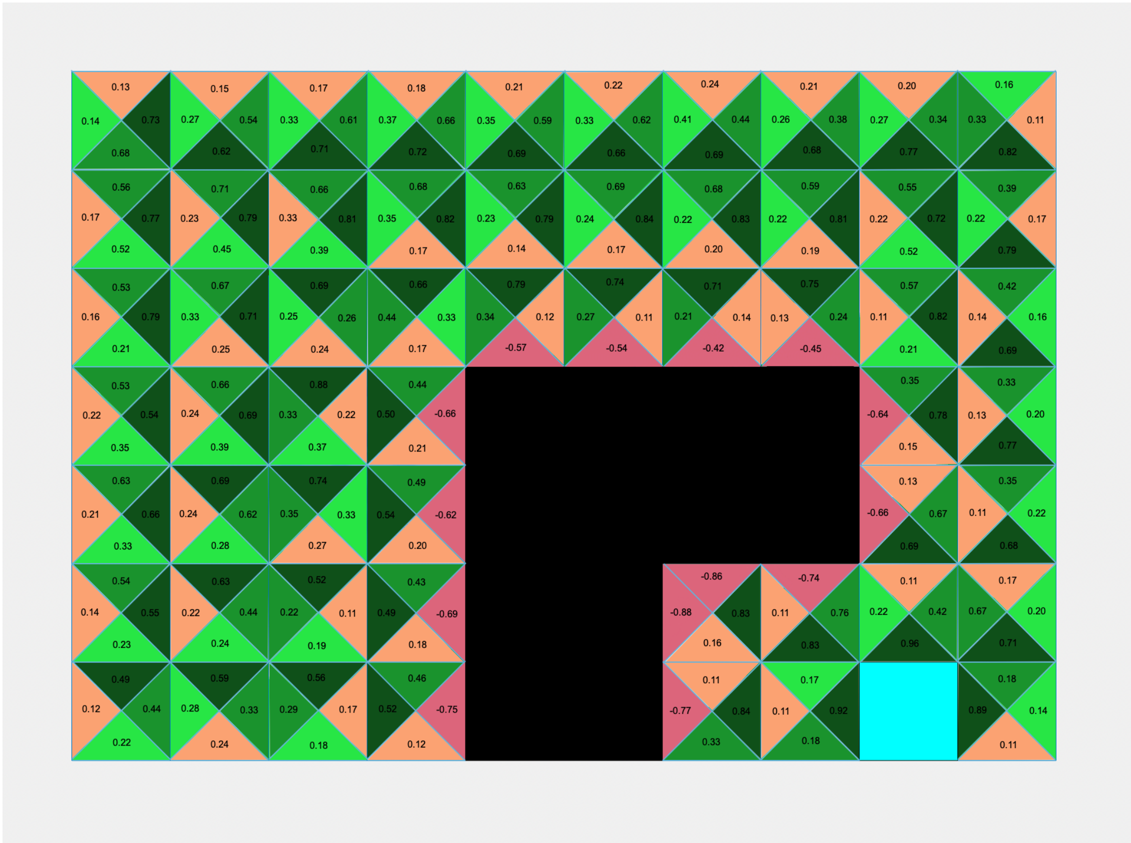


Figure A.2 – Q-values representation after 500 training episodes

As the exploration is allowed and considering inherent stochasticity, Q-values are visible in all fields of the gridworld. In case that agent finds itself in any of the fields, it will find the highest value and follow that direction when it is exploiting the previous knowledge or explore to some another field in order to discover if higher rewards await somewhere else. Finally, allowing for adequate training episodes, learned policy will allow for the artificial agent to know exactly which path to take from the origin to the destination as long as there are no changes along the way.

We are mostly interested to see if the rewards shaping offers any benefits over the standard approach; therefore, artificial agent was trained with and without the rewards shaping. Figure A.4 shows the progress of training and comparison of both approaches.

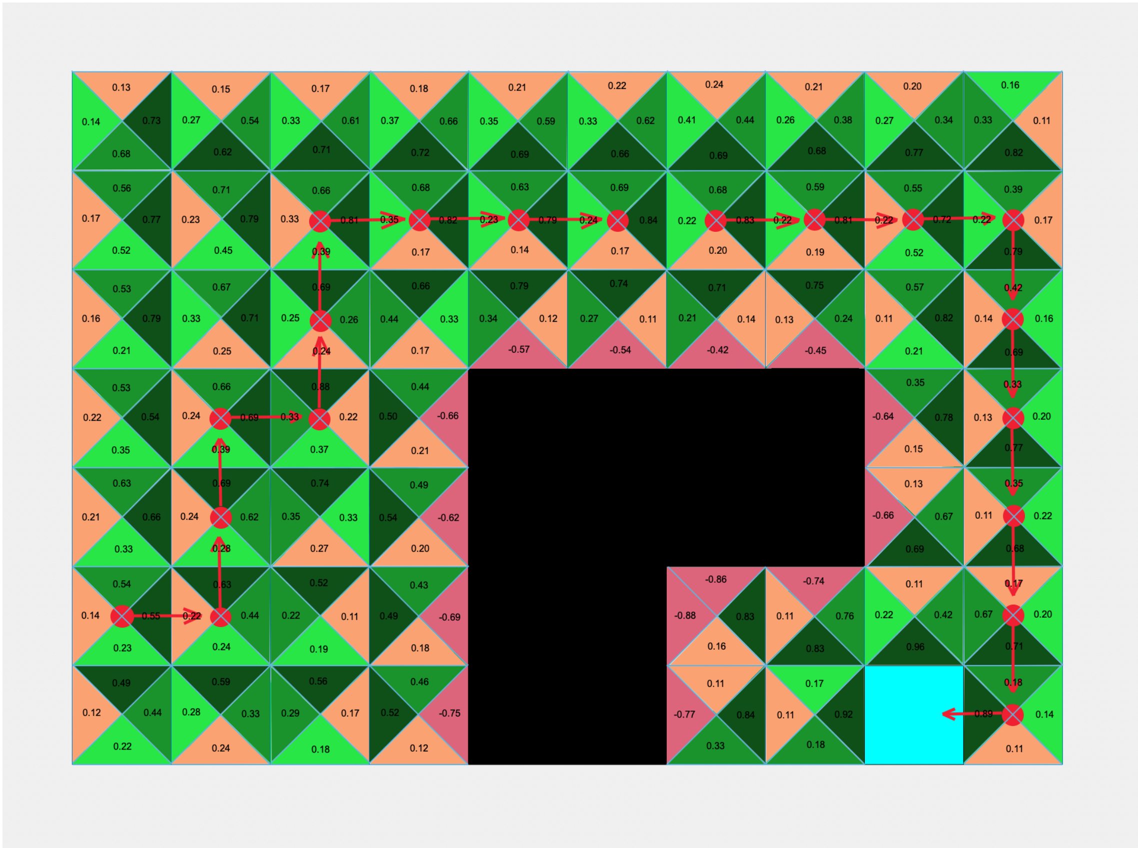
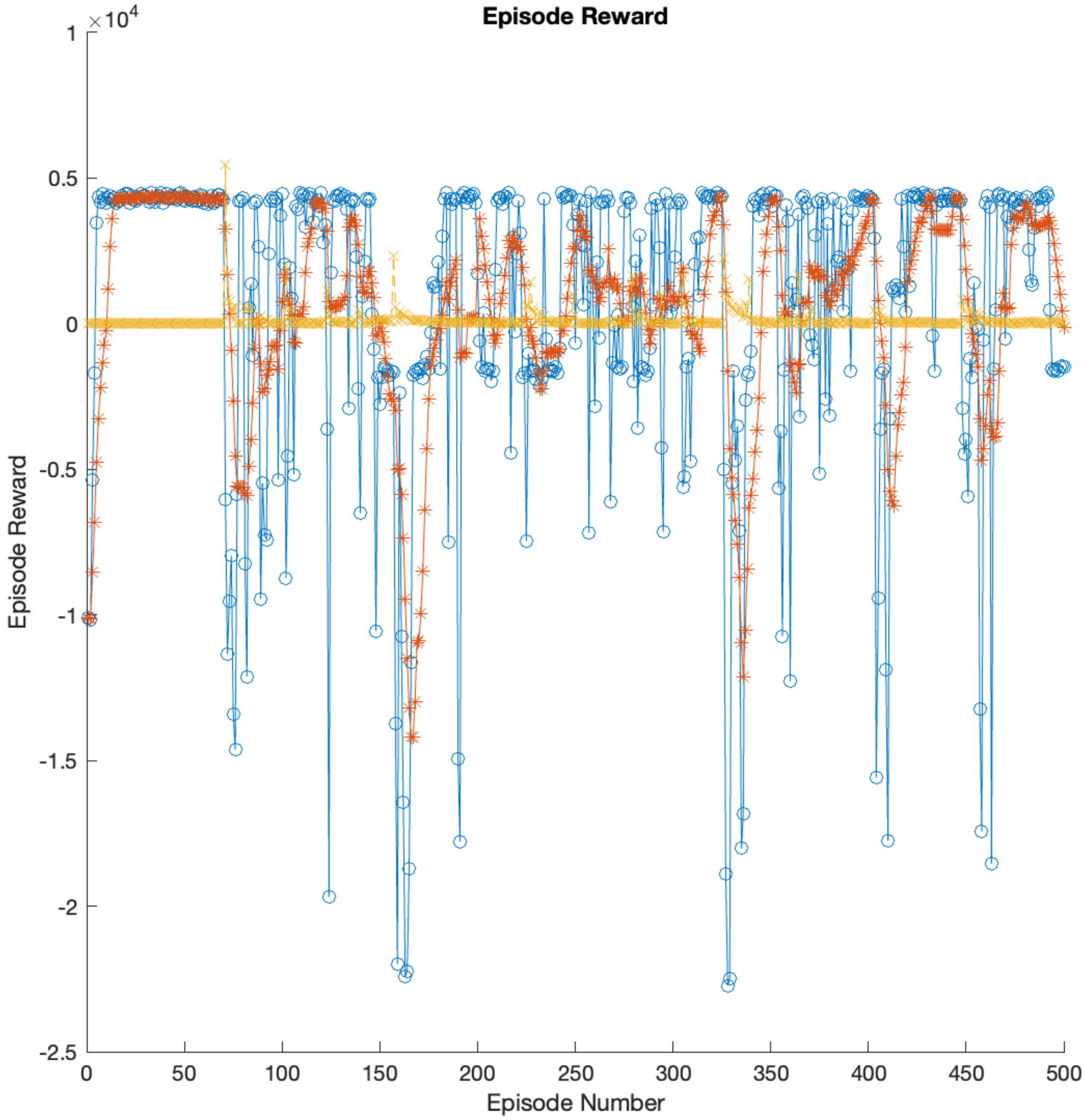


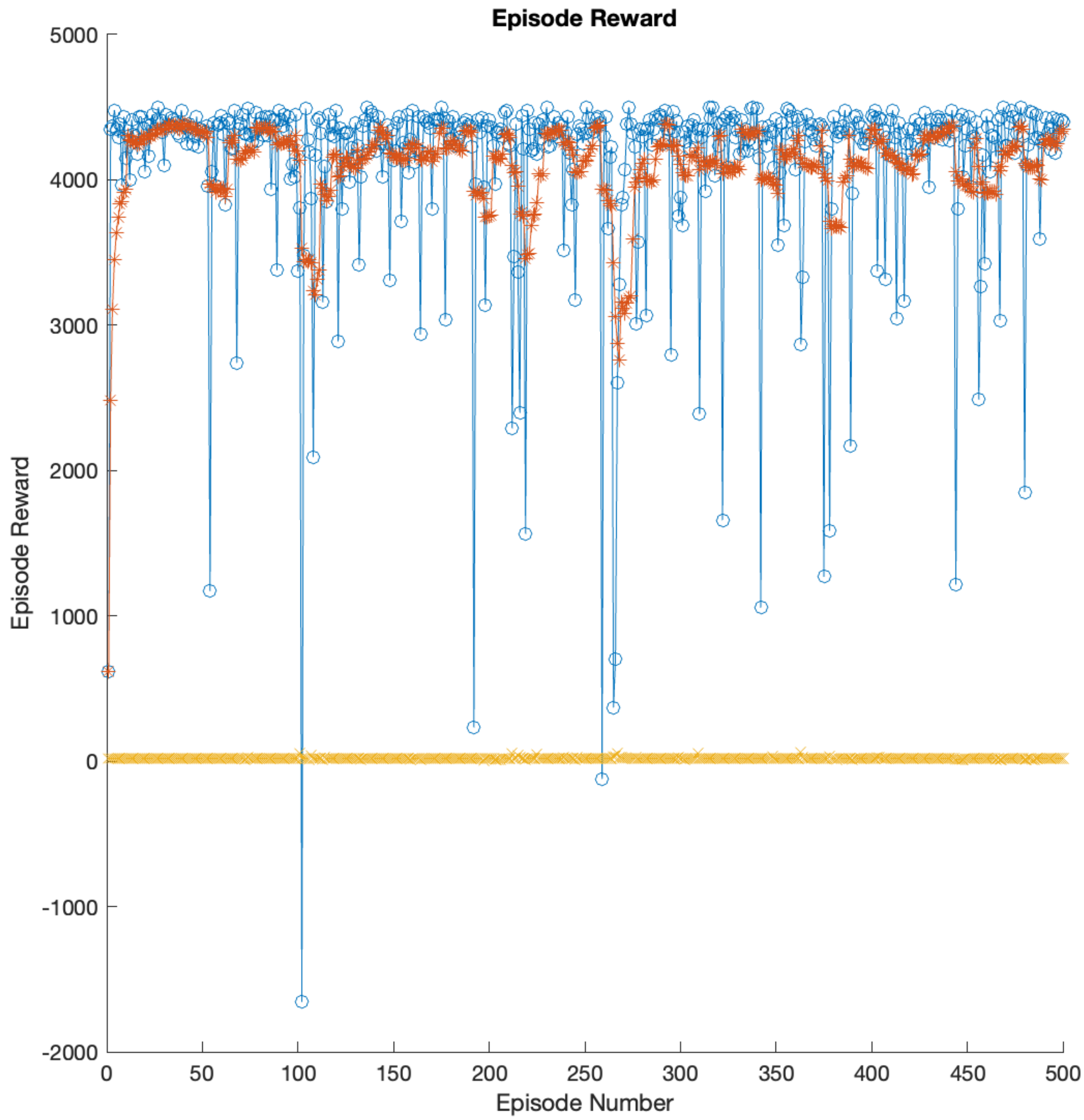
Figure A.3 – Optimal policy after the sea surface vehicle training session

In the case of training session without rewards shaping (a), we can notice that there is a higher intensity and frequency of exploration with negative spikes well below -2000 points. On the other hand, when training with rewards shaping (b), we can notice that training is more stable with lower intensity exploration spikes and higher cumulative rewards. Rewards shaping also improves the exploration, as it penalizes exploration that offers no benefits. For example, if we have to steer own sea surface vehicle in a certain direction, there is a little benefit to check the state space that considers steering in an opposite direction from the direction we wish to steer. Therefore, rewards shaping clearly improves convergence of the training and maintains the focus of the main task.



Episode Reward
$\times 10^4$
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
Episode Reward
0
50
100
150
200
250
300
350
400
450
500
Episode Number

(a)



(b)

Figure A.4 – Training experiments with (a) representing training without rewards shaping and (b) with rewards shaping with the 7x10 gridworld

Appendix B

Motion Control and Autopilot Experiment Results

Table B.1 – Training scenarios with external disturbances

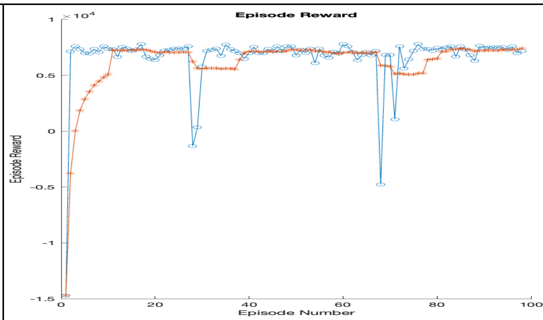
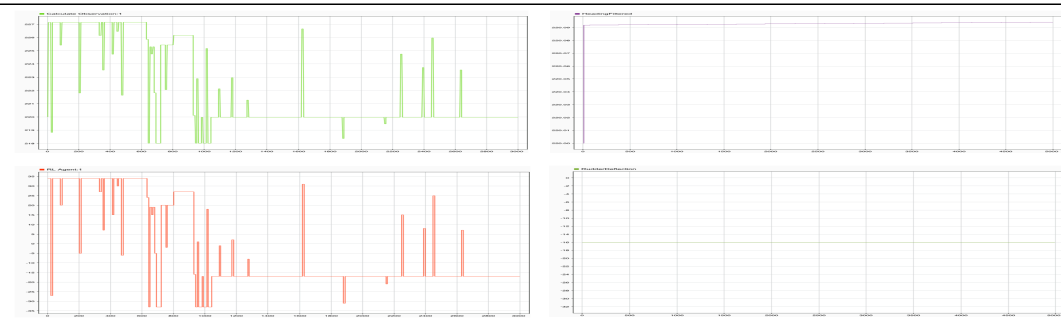
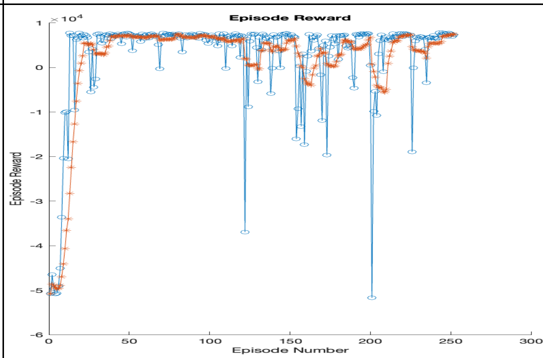
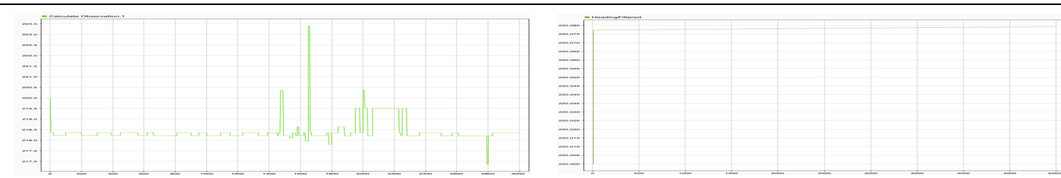
Scenario	Relative Wind Direction (°)	Relative Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1	67.5	21	45	2.5	74	2	67.5	4
2	0	10	22.5	1.25	0	1	22.5	2
3	22.5	33	0	4	57.5	4	0	6
4	45	47	67.5	6	83	3	135	6 (period 20 s)
5	90	10	112.5	1.25	94	2	90	2
6	112.5	63	90	14	100	1	180	6
7	135	21	135	2.5	121	4	45	4
8	157.5	33	180	4	180	3	112.5	6 (period 20 s)
9	180	21	157.5	2.5	239	1	157.5	6
10	202.5	47	225	6	260	2	270	4
11	225	21	202.5	2.5	266	3	292.5	2

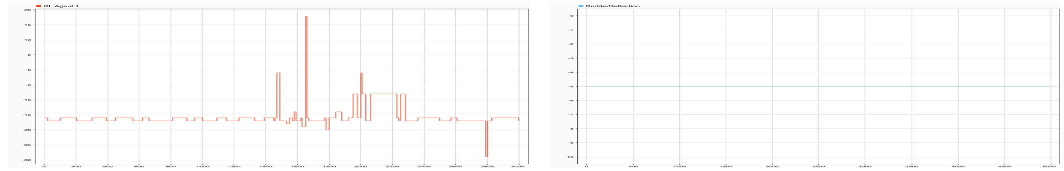
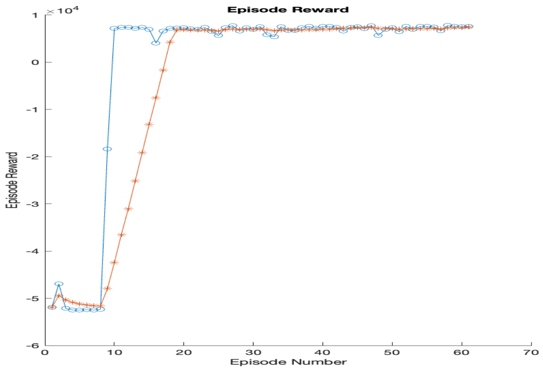
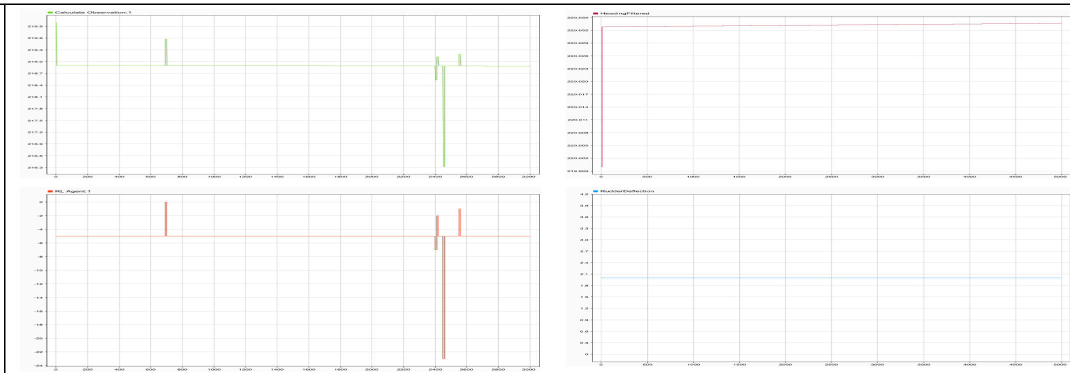
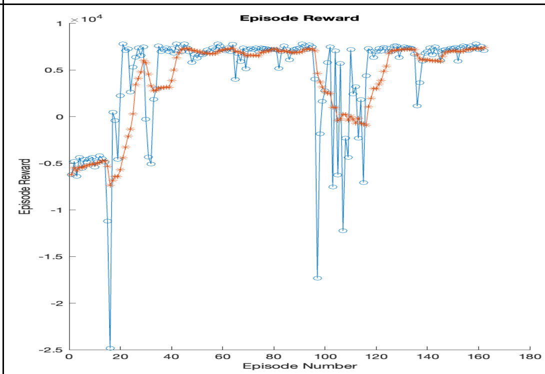
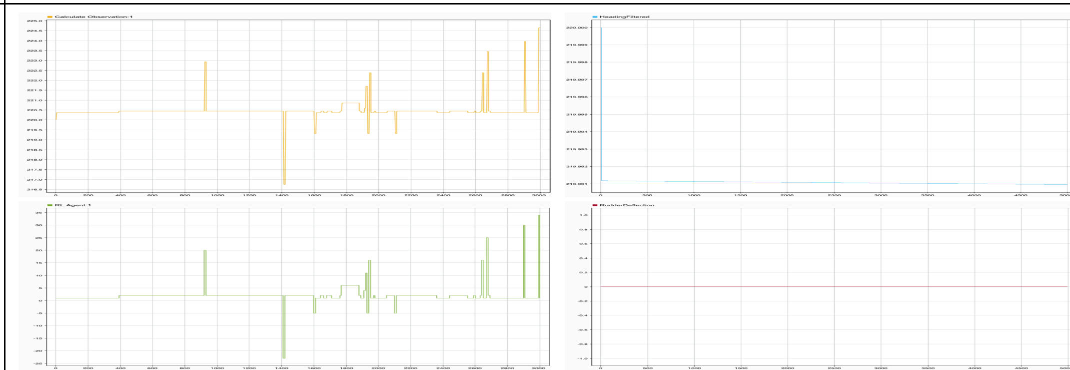
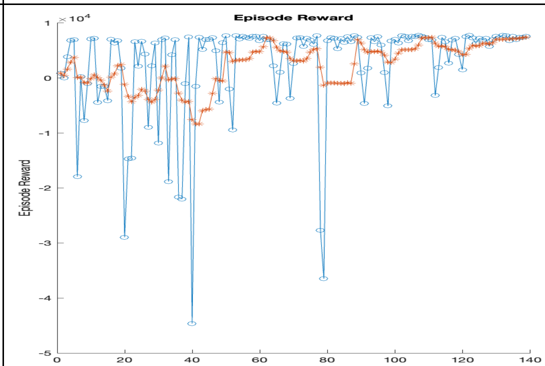
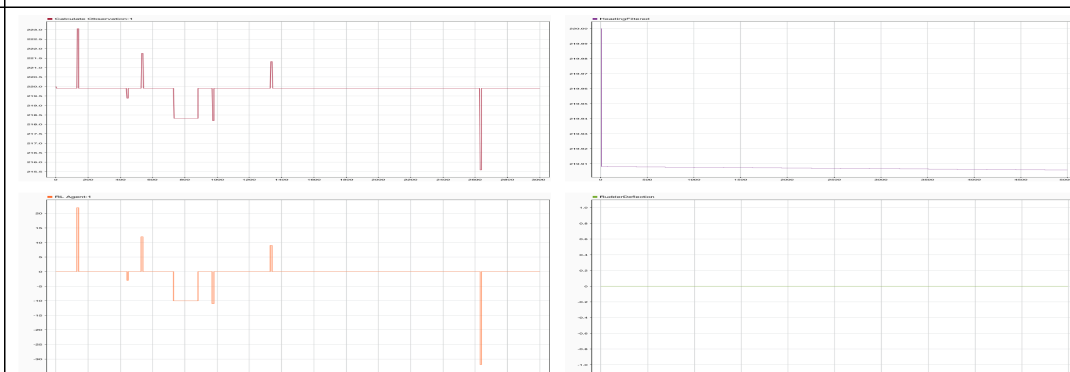
12	247.5	10	247.5	1.25	277	4	202.5	6 (period 20 s)
13	270	33	292.5	4	286	2	315	2
14	292.5	47	270	6	302.5	3	247.5	4
15	315	21	315	2.5	83	1	337.5	6
16	337.5	33	337.5	4	100	4	225	20

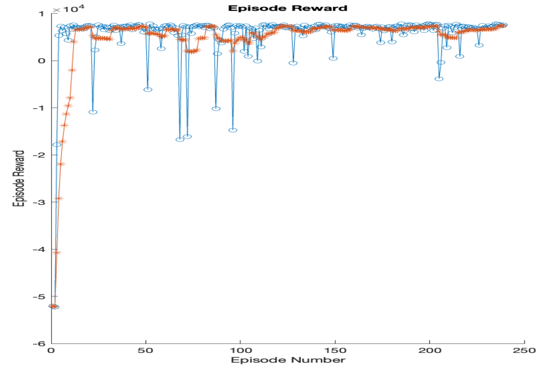
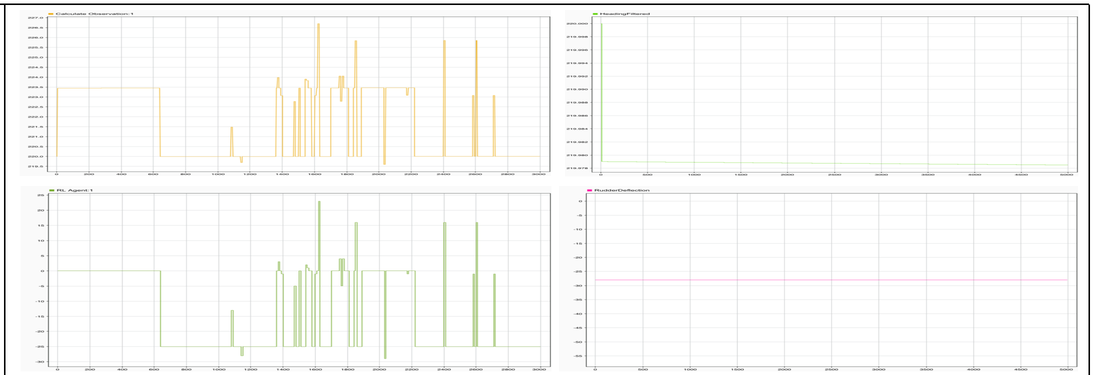
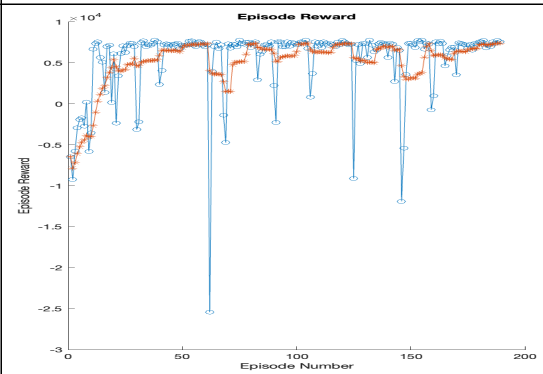
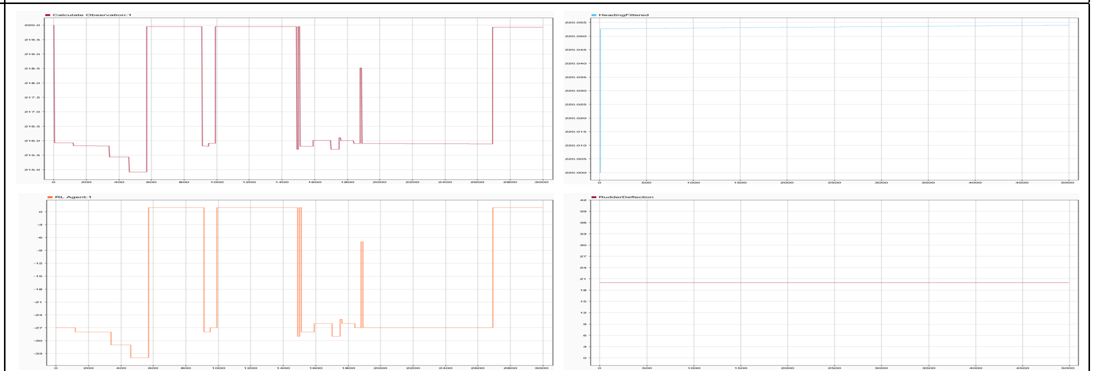
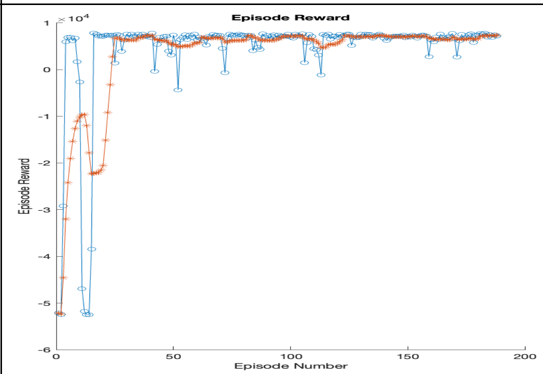
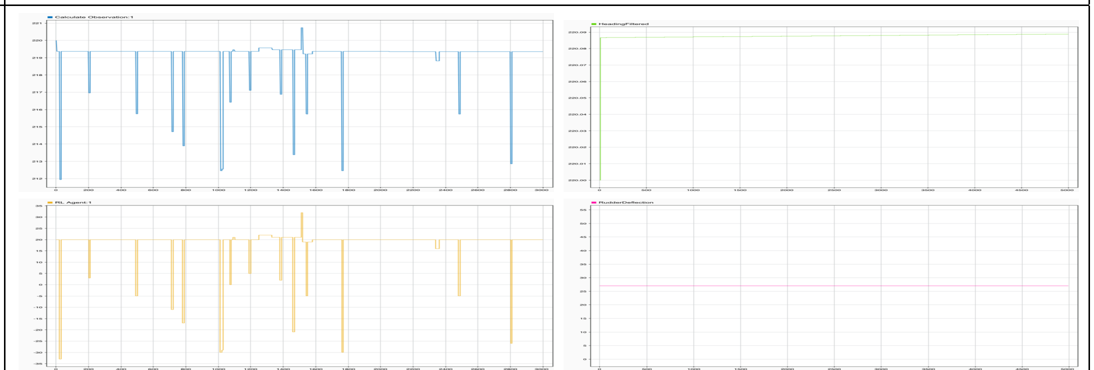
Table B.2 – Simulation scenarios with external disturbances

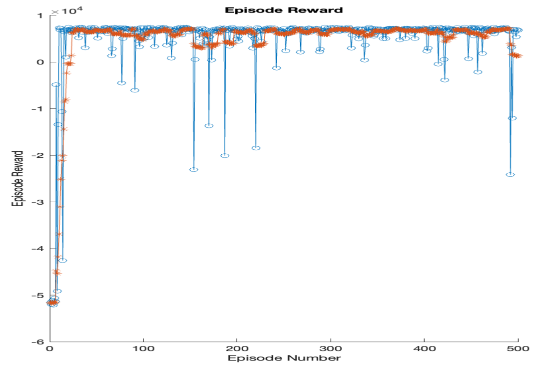
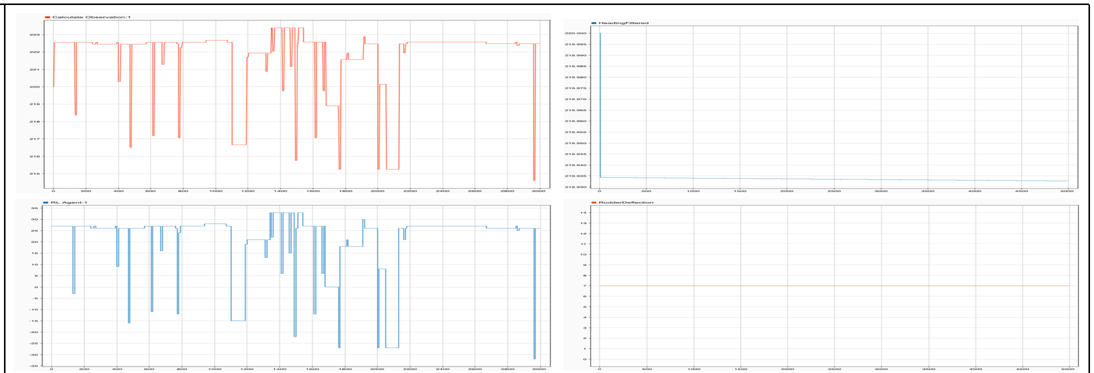
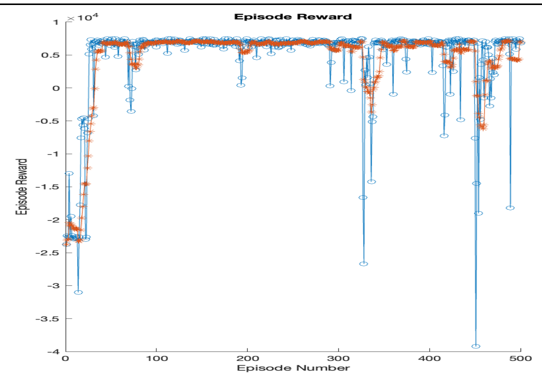
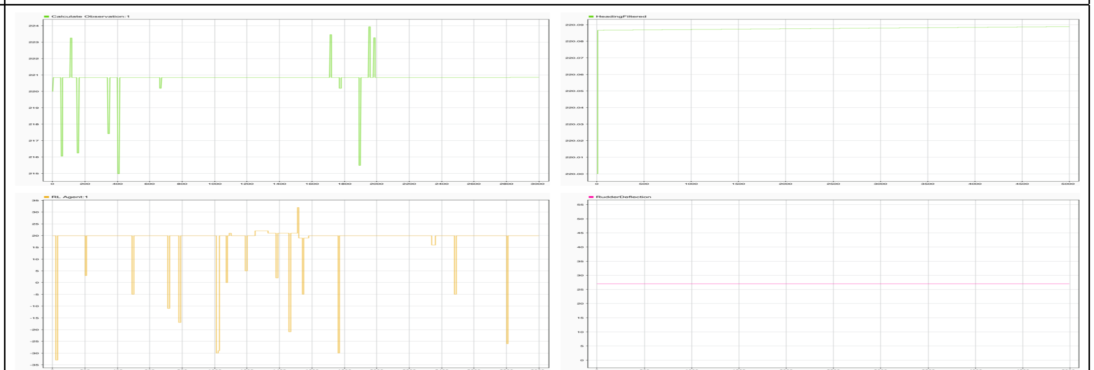
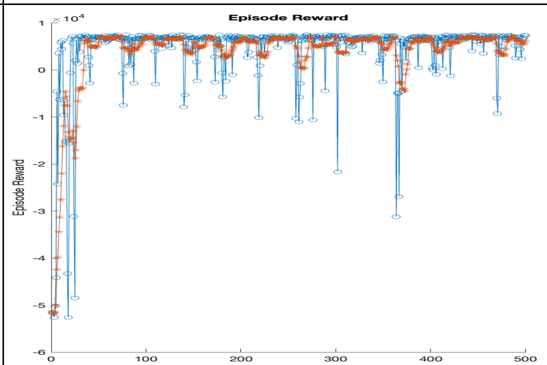
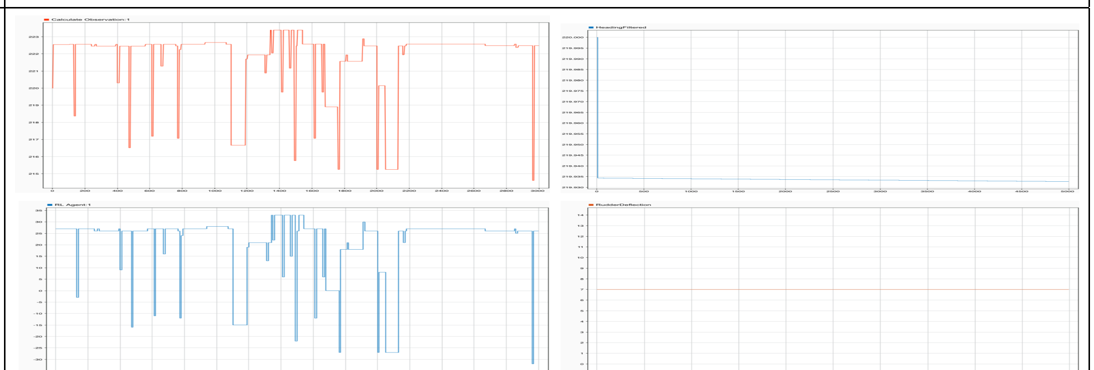
Simulation	Relative Wind Direction (°)	Relative Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1	060	32	045	4	152	3	257	1
2	282	24	277	3	269	2	323	2
3	185	48	163	5	345	3	090	4
4	358	9	013	2	047	1	272	6
5	307	54	329	6	197	2	82	4

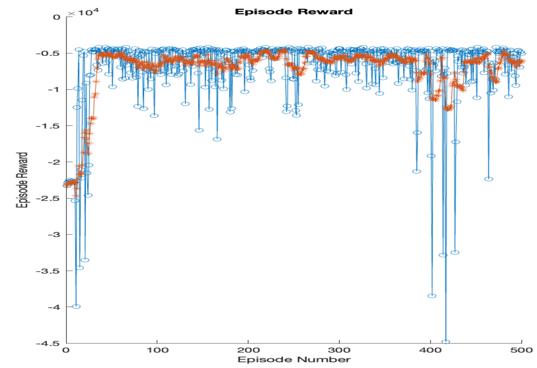
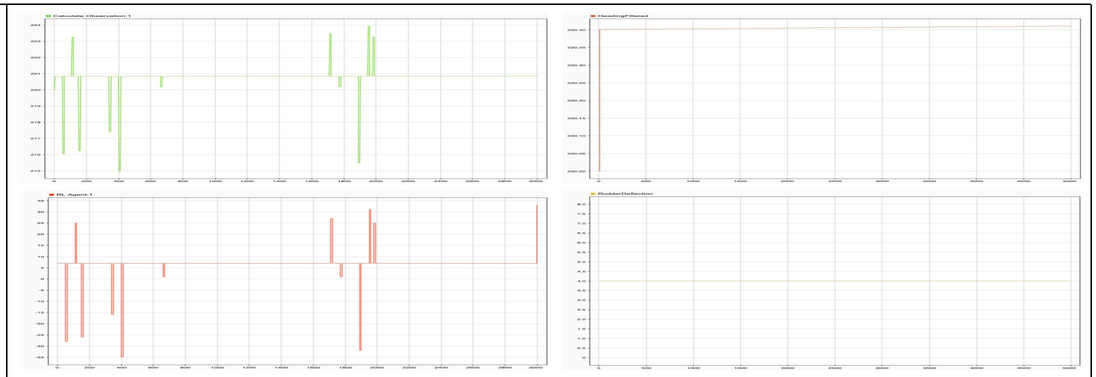
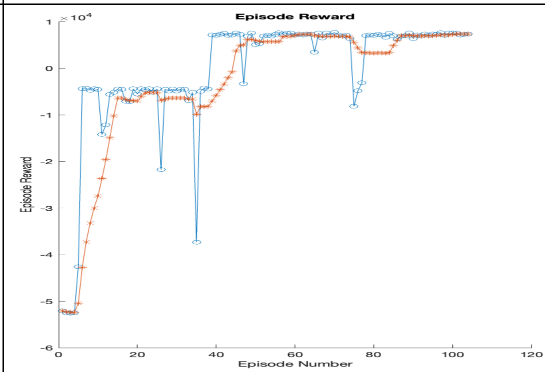
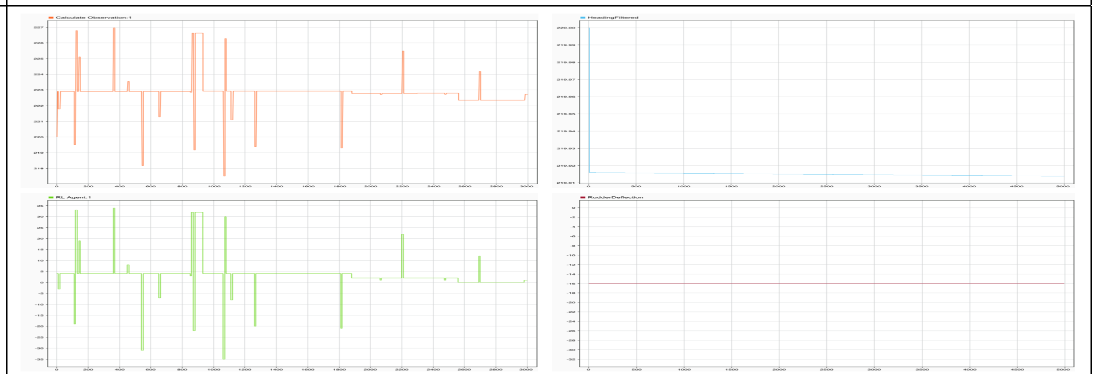
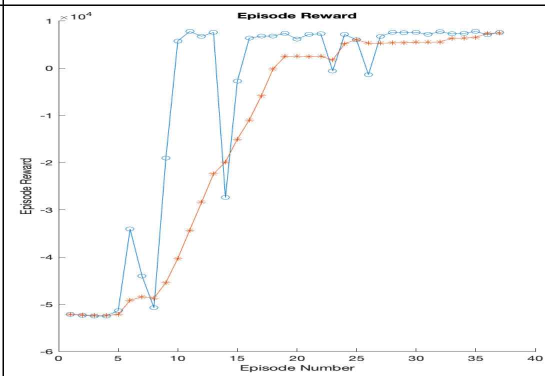
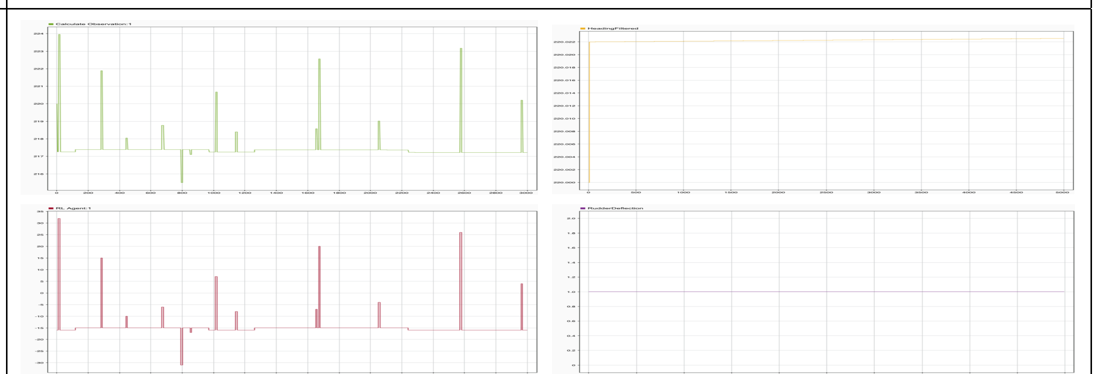
Table B.3 – Training progress and results for Heading control

1		
2		

		
3	Episode Reward Episode Number Episode Reward	
4	Episode Reward Episode Number Episode Reward	
5	Episode Reward Episode Number Episode Reward	

6	Episode Reward Episode Reward Episode Number	Calculate Observation:1 HeadingFiltered RL Agent:1 RudderDeflection
7	Episode Reward Episode Reward Episode Number	Calculate Observation:1 HeadingFiltered RL Agent:1 RudderDeflection
8	Episode Reward Episode Reward Episode Number	Calculate Observation:1 HeadingFiltered RL Agent:1 RudderDeflection

9	 Episode Reward Episode Number	
10	 Episode Reward Episode Number	
11	 Episode Reward Episode Number	

12	Episode Reward Episode Reward Episode Number	
13	Episode Reward Episode Reward Episode Number	
14	Episode Reward Episode Reward Episode Number	

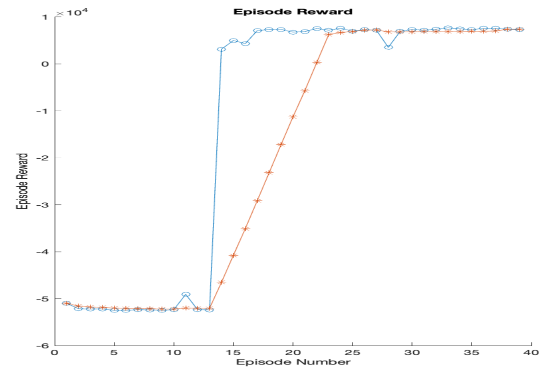
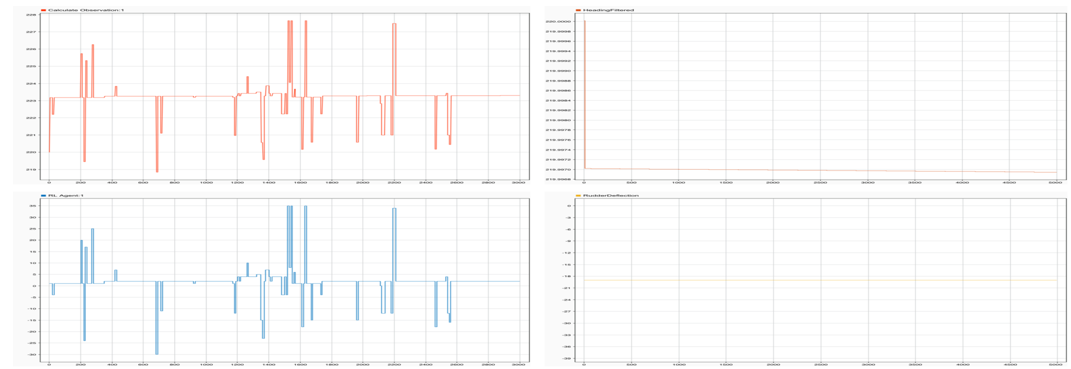
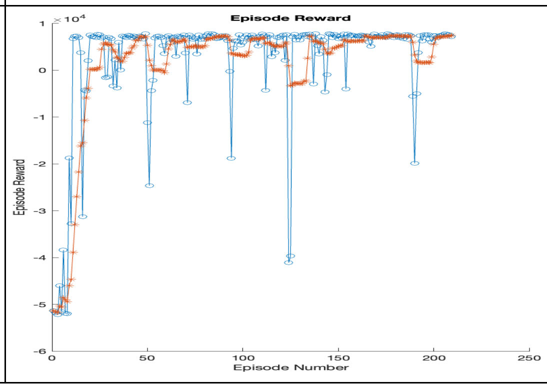
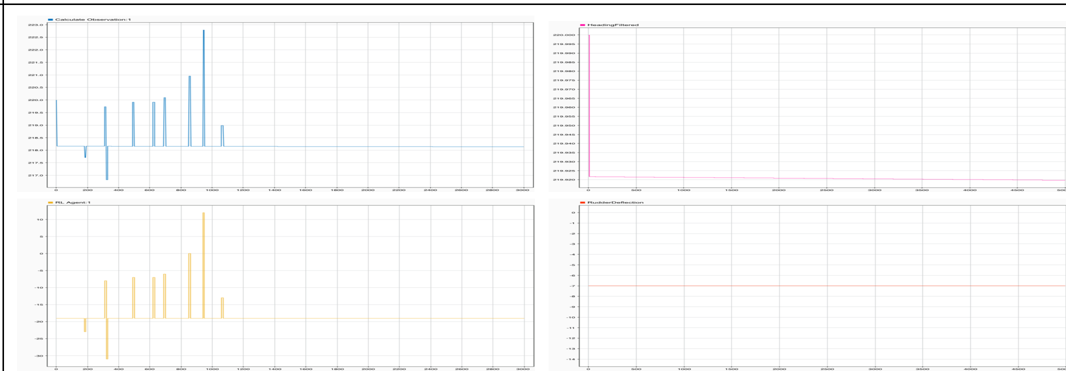
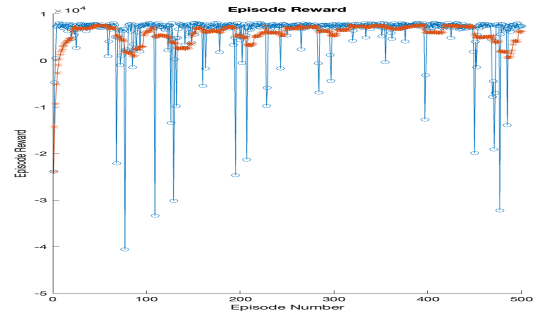
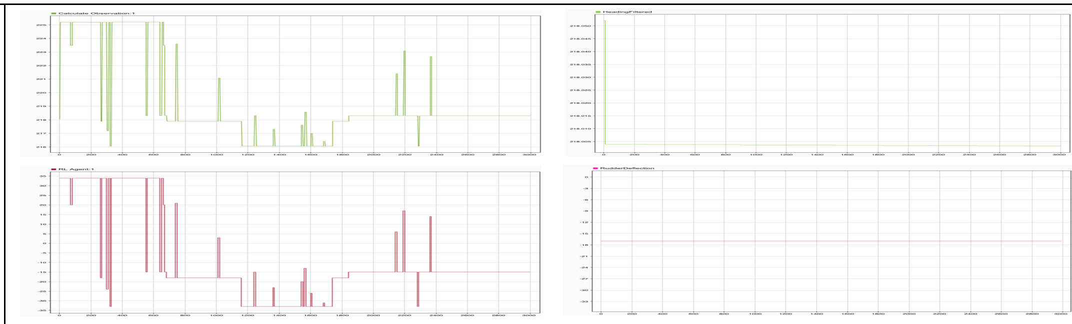
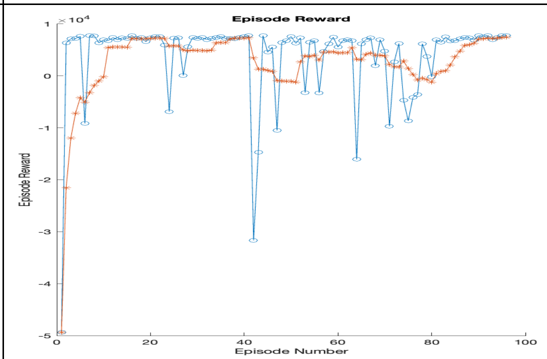
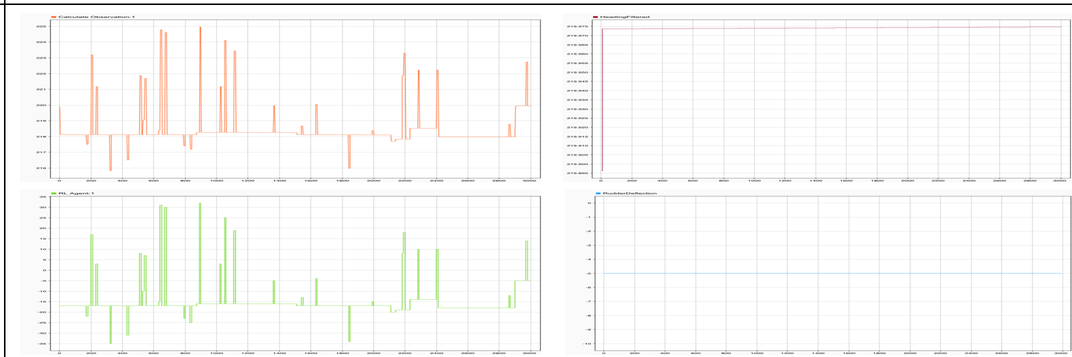
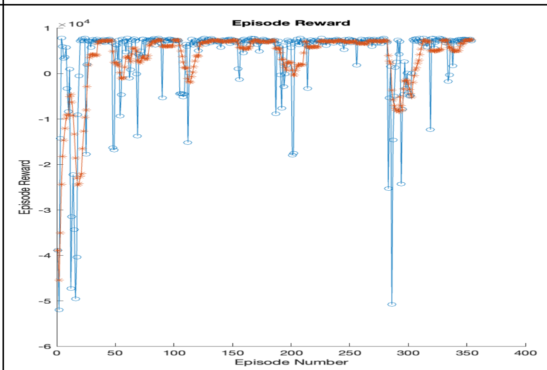
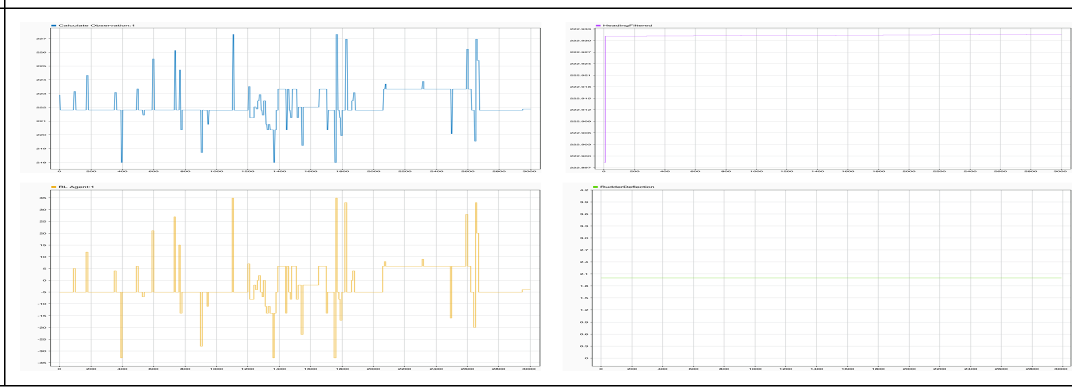
<table>
<tr>
<td>15</td>
<td>
</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>
</td>
<td></td>
</tr>
</table>

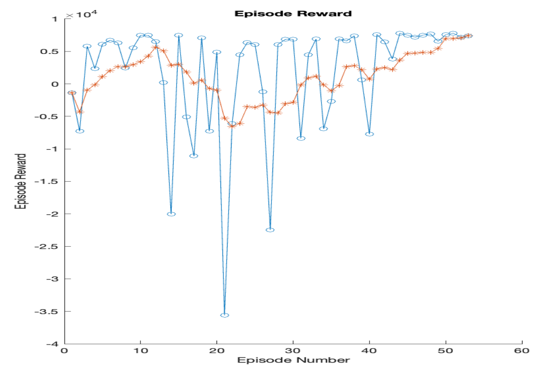
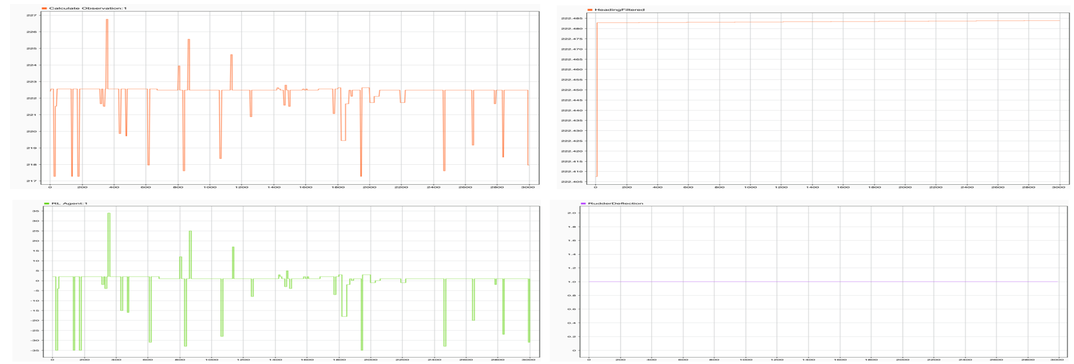
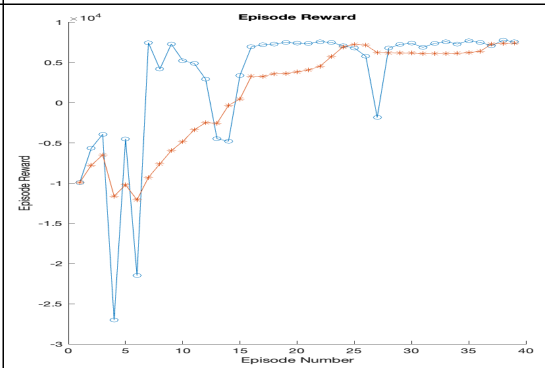
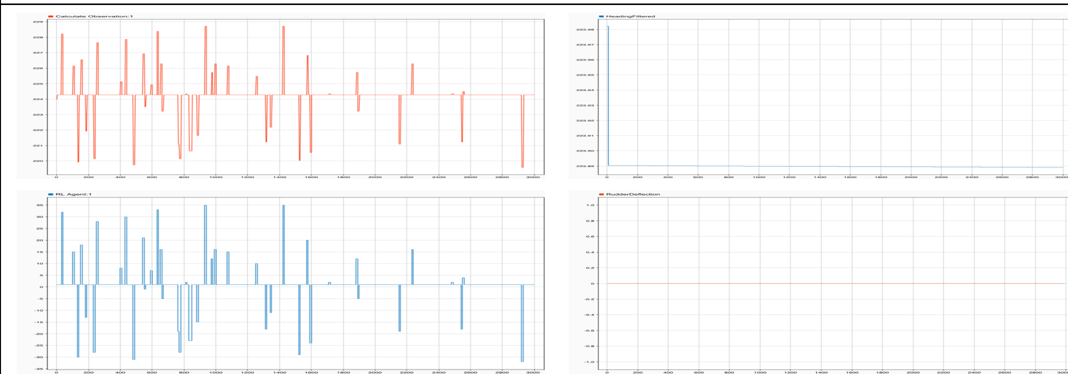
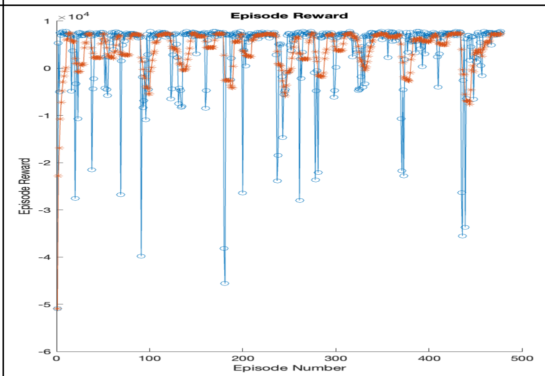
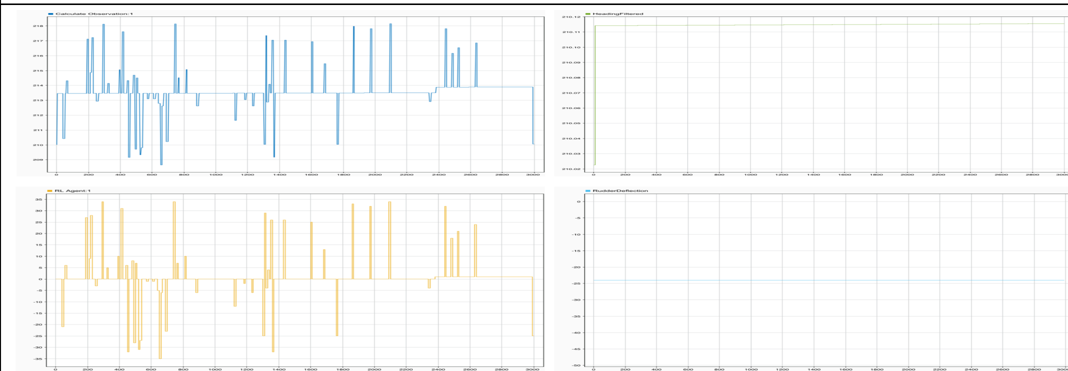
Table B.4 – Feasibility test for 16 scenarios (left) and 5 simulations (right)

Scenario	Set Heading (°)	Achieved Heading (°)	Deviation (°)	Learning Time (s)	Execution Time (s)
1	220	220.1	0.1	439.6	0.002
2	220	220.1	0.1	1083,4	0.001
3	220	220.0	0	268,57	0.001
4	220	220.0	0	723,26	0.002
5	220	219.9	0.1	640,72	0.003
6	220	220.0	0	1086,3	0.001
7	220	220.0	0	880,37	0.002
8	220	220.0	0	902,68	0.001
9	220	220.1	0.1	2225	0.002
10	220	220.1	0.1	2279,7	0.001
11	220	219.9	0.1	2163,1	0.001
12	220	220.4	0.4	2220,2	0.002
13	220	219.9	0.1	470,36	0.001
14	220	220.0	0	181,74	0.002
15	220	220.0	0	187,96	0.004
16	220	219.9	0.1	932,52	0.006

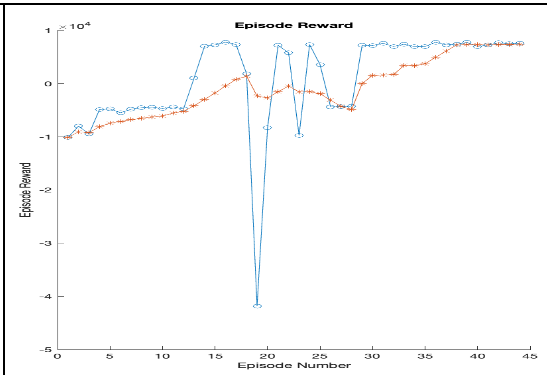
Simulation	Set Heading (°)	Achieved Heading (°)	Deviation (°)	Learning Time (s)	Execution Time (s)
1	220	220.0	0	n/a	0.002
2	220	220.0	0	n/a	0.001
3	220	220.1	0.1	n/a	0.001
4	220	220.1	0.1	n/a	0.002
5	220	219.8	0.2	n/a	0.003

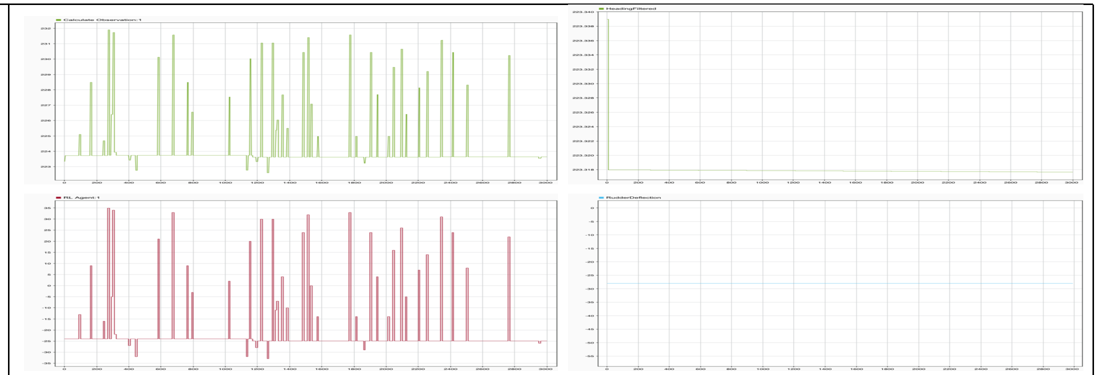
Table B.5 – Training progress and results for Course control

1		
2		
3		

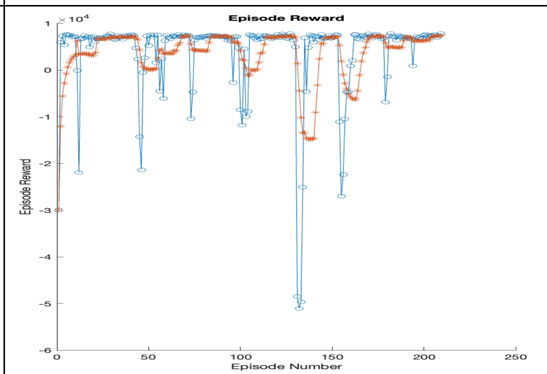
4	Episode Reward Episode Reward Episode Number	
5	Episode Reward Episode Reward Episode Number	
6	Episode Reward Episode Reward Episode Number	

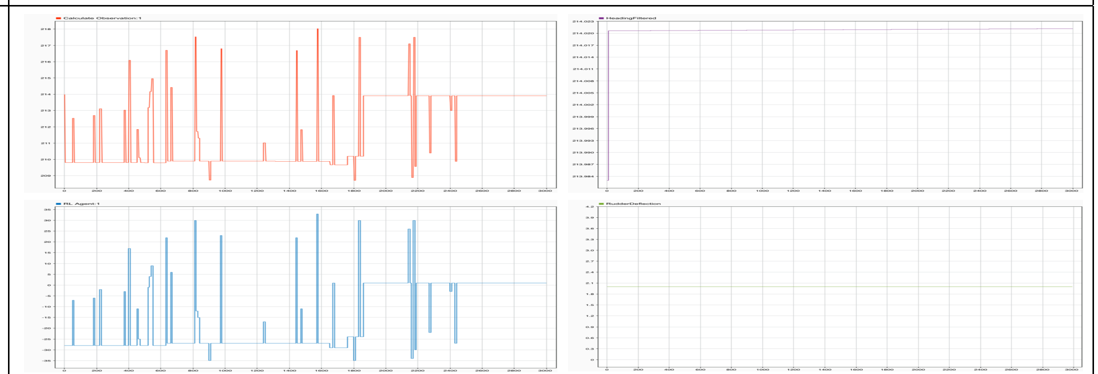
7


Episode Reward
Episode Reward
Episode Number

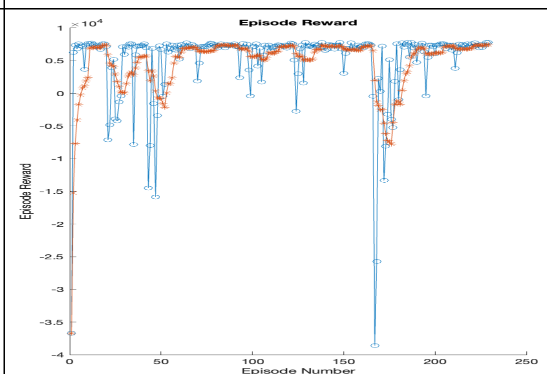


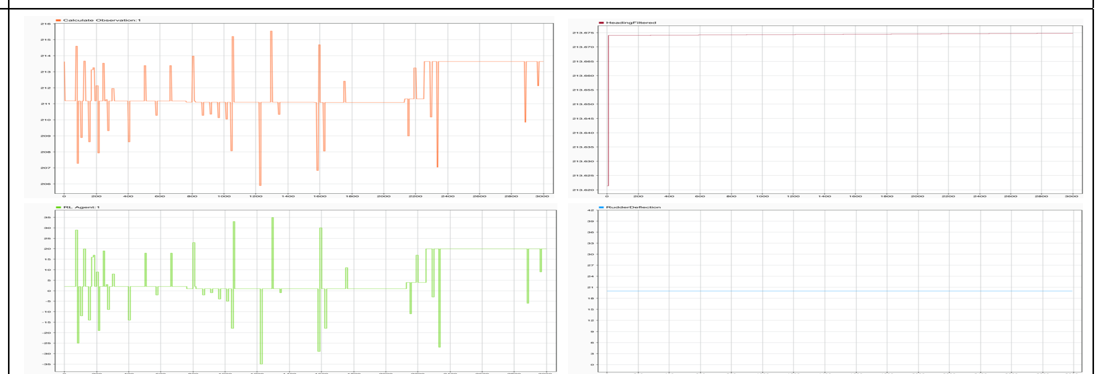
8

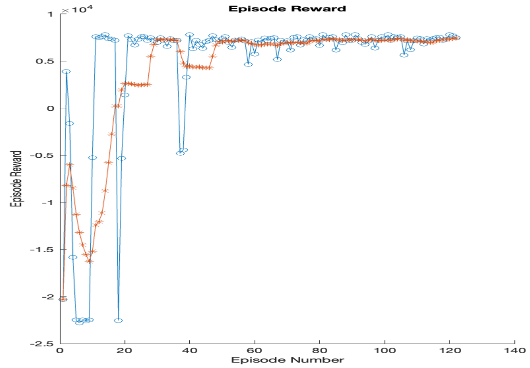
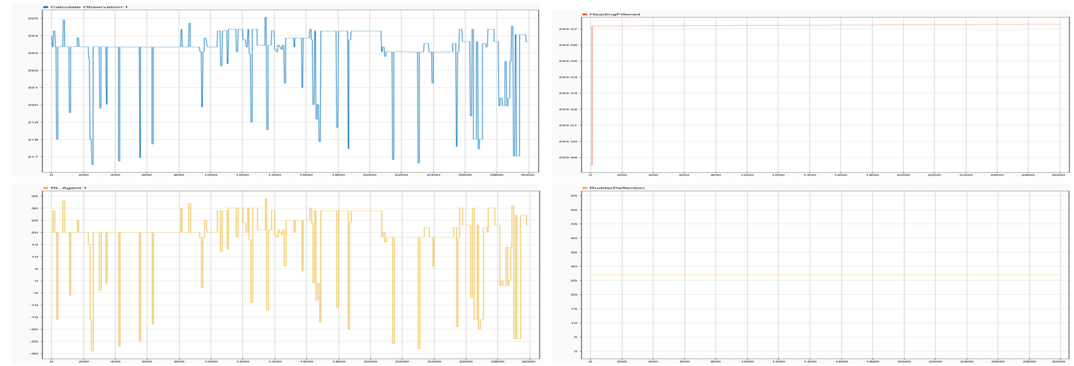
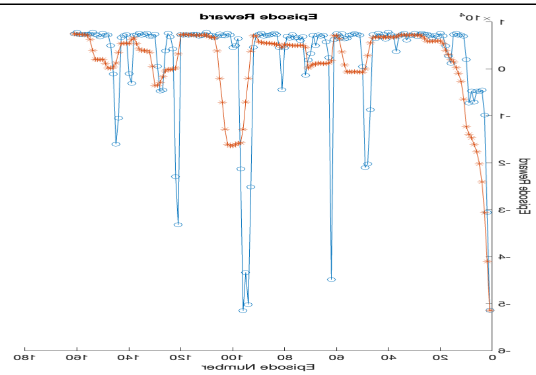
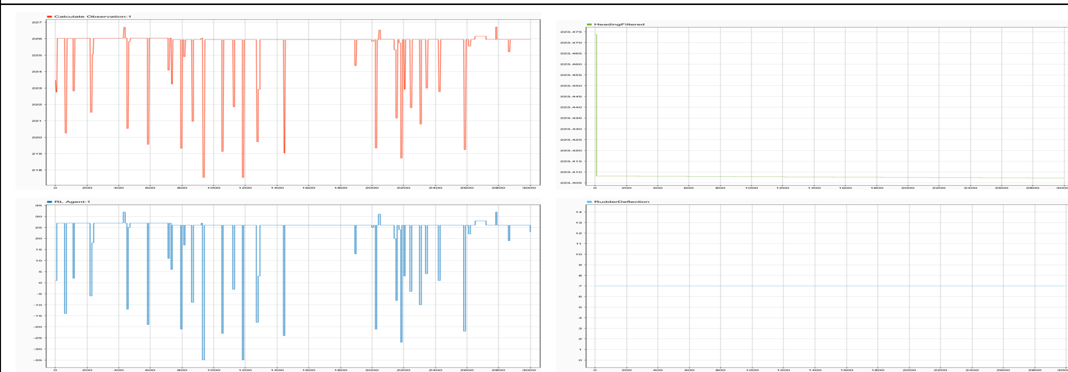
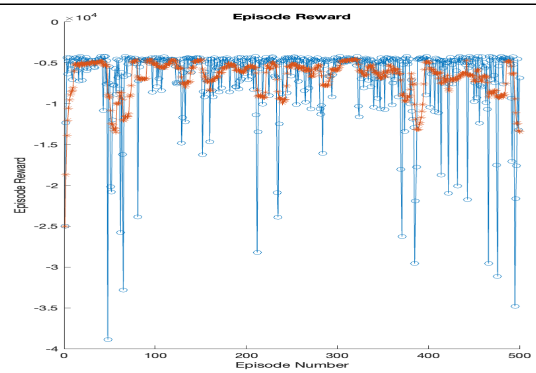
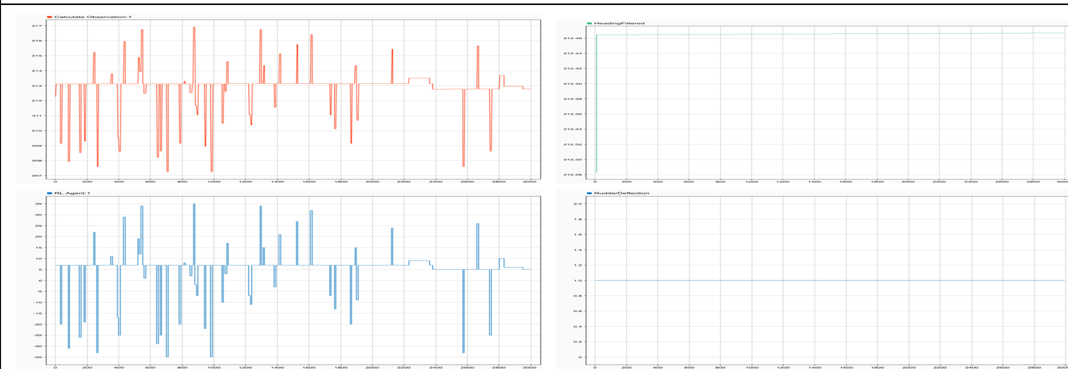

Episode Reward
Episode Reward

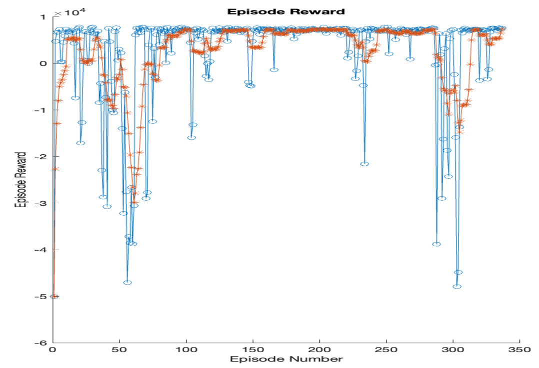
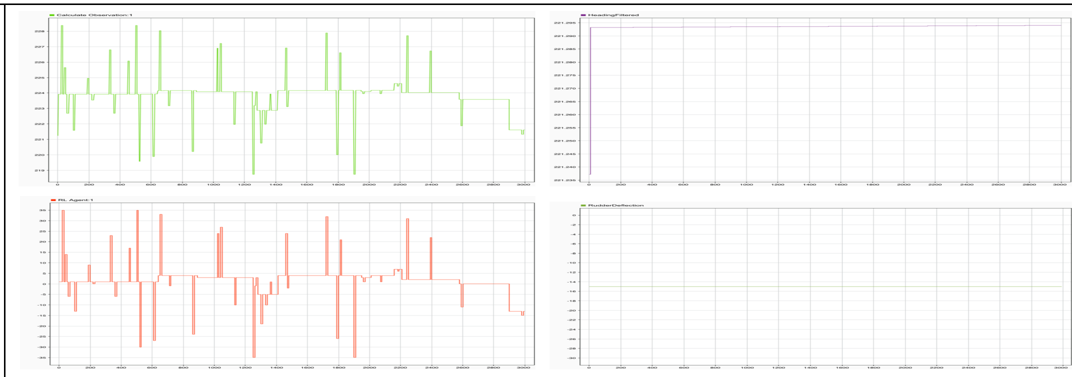
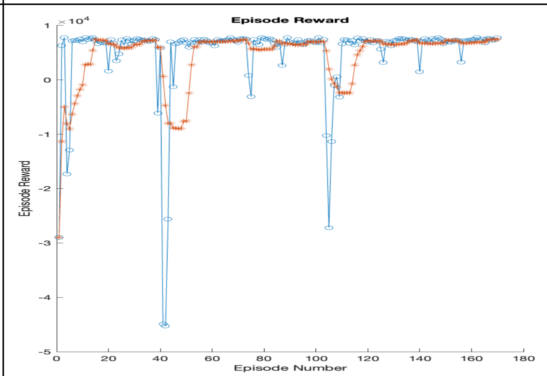
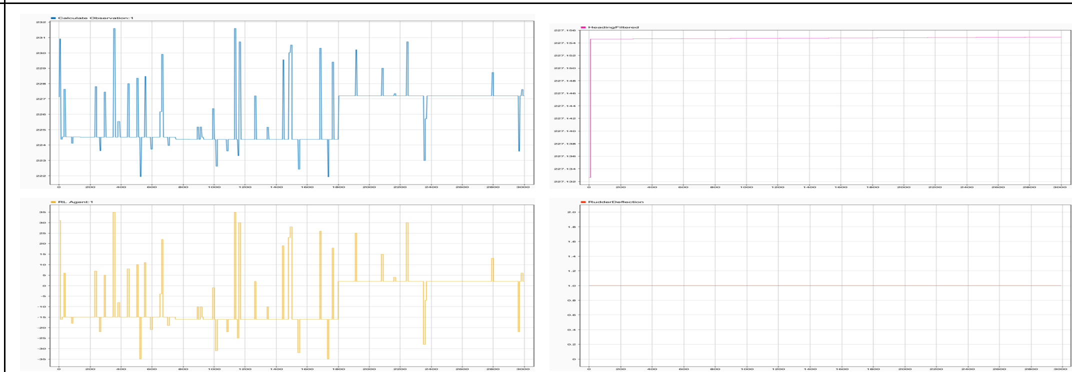
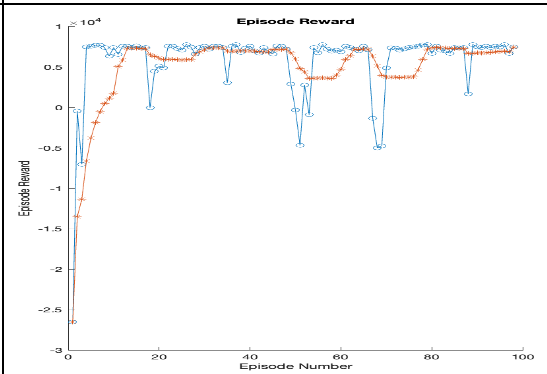
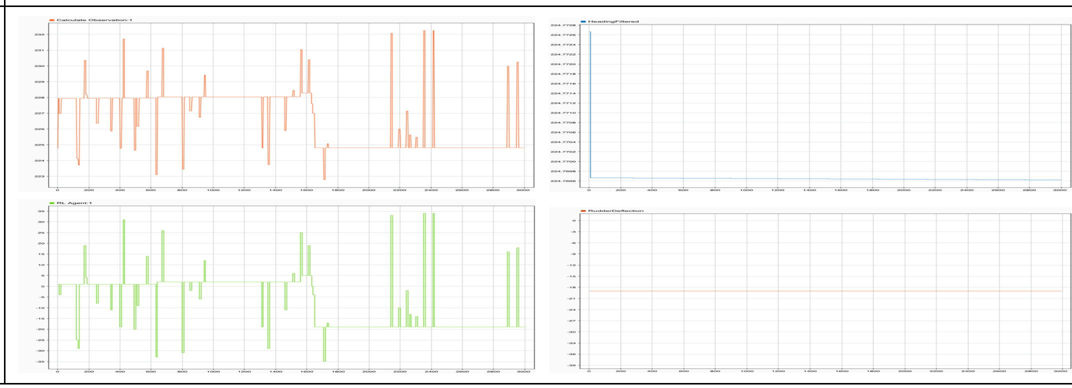


9


Episode Reward
Episode Reward
Episode Number



10	Episode Reward Episode Reward Episode Number	
11	Episode Reward Episode Reward Episode Number	
12	Episode Reward Episode Reward Episode Number	

13	Episode Reward Episode Number Episode Reward	
14	Episode Reward Episode Number Episode Reward	
15	Episode Reward Episode Number Episode Reward	

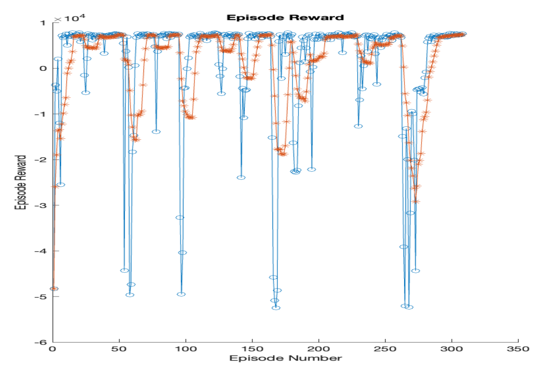
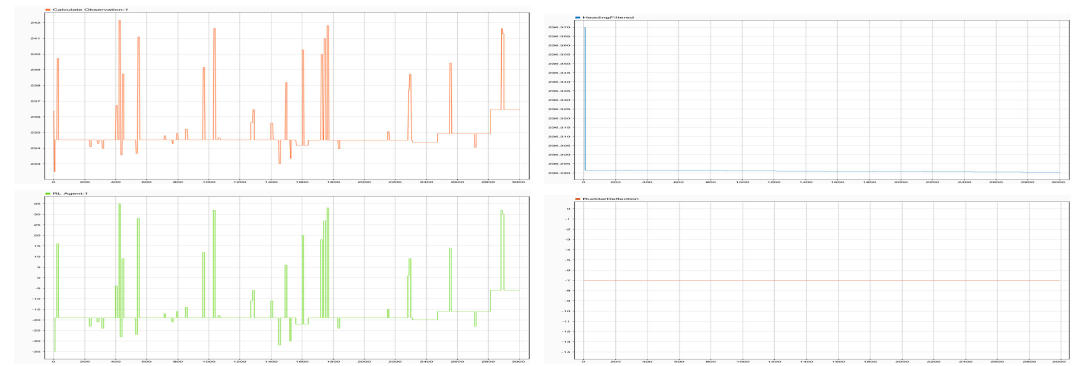
16		

Table B.6 –– Feasibility test for 16 scenarios (left) and 5 simulations (right)

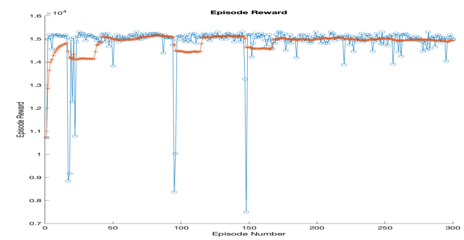
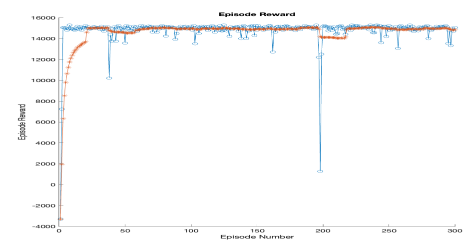
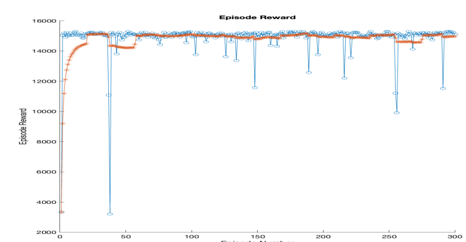
Scenario	Set Course (°)	Achieved Course (°)	Deviation (°)	Learning Time (s)	Execution Time (s)
1	220	218.9	0.1	2252,1	0.003
2	220	220.0	0	437,36	0.001
3	220	222.3	1.3	1573,6	0.003
4	220	220.5	0.5	237,06	0.001
5	220	220.9	0.9	180,26	0.002
6	220	220.0	0	2103,9	0.003
7	220	220.3	0.3	194,45	0.004
8	220	218.9	1.1	942,16	0.006
9	220	220.7	0.7	1011,2	0.003
10	220	220.2	0.2	565,72	0.002
11	220	220.4	0.4	700,44	0.001
12	220	220.0	0	2197,8	0.001
13	220	219.7	0.3	1574,4	0.008
14	220	220.1	0.1	795,16	0.005
15	220	220.8	0.8	452,72	0.003
16	220	220.3	0.3	1416,9	0.005

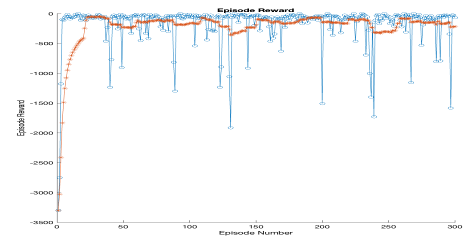
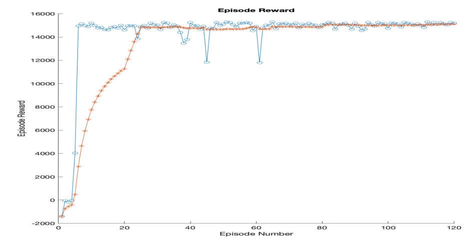
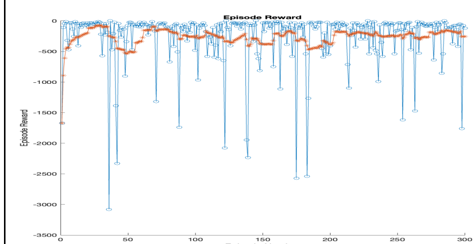
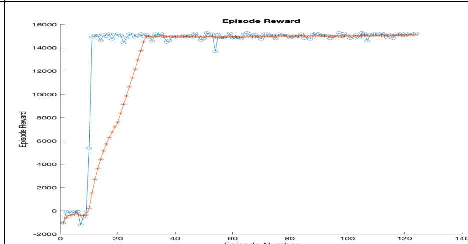
Simulation	Set Course (°)	Achieved Course (°)	Deviation (°)	Learning Time (s)	Execution Time (s)
1	220	220.1	0.1	n/a	0.003
2	220	219.9	0.1	n/a	0.002
3	220	220.4	0.4	n/a	0.006
4	220	219.6	0.4	n/a	0.001
5	220	220.2	0.2	n/a	0.002

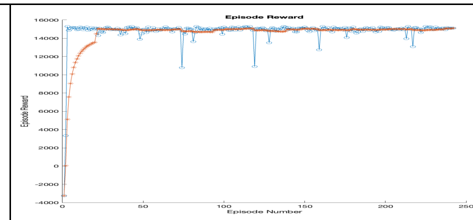
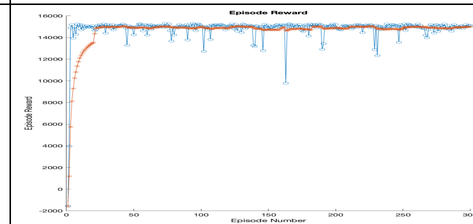
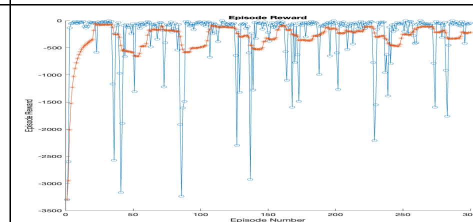
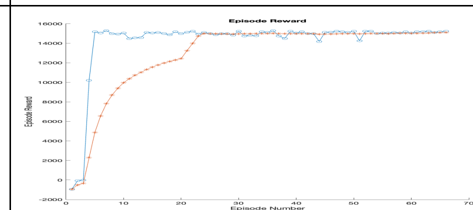
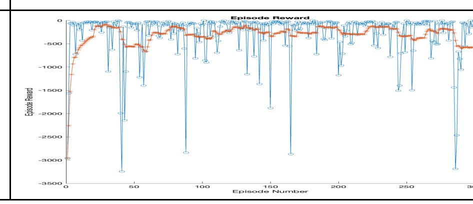
Table B.7 – Scenarios for auto throttle algorithm training

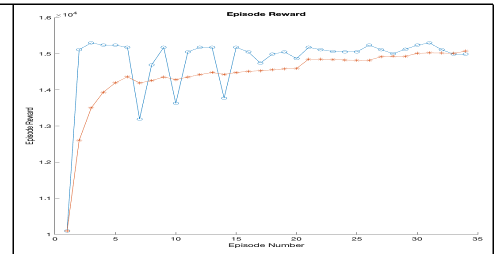
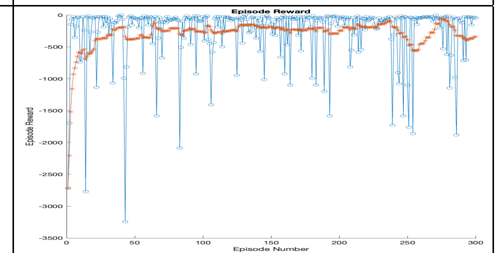
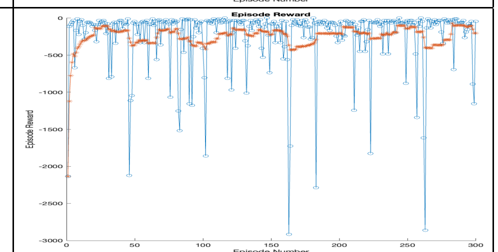
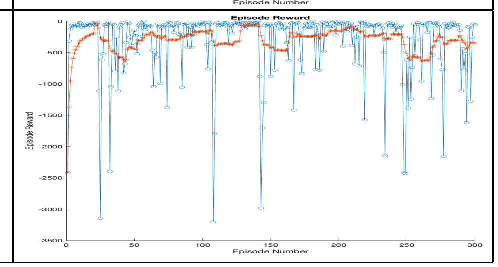
Scenario	Set Speed (kt)	Relative Wind Direction (°)	Relative Wind Speed (kt)	Relative Wave Direction (°)	Significant Wave Height (m)	Relative Current Direction (°)	Current Speed (kt)	Relative Swell Direction (°)	Swell Height (m)
1	18	67.5	21	45	2.5	74	2	67.5	4
2	13.3	0	10	22.5	1.25	0	1	22.5	2
3	4	22.5	33	0	4	57.5	4	0	6
4	11.8	45	47	67.5	6	83	3	135	6 (period 20 s)
5	20.4	90	10	112.5	1.25	94	2	90	2
6	7.2	112.5	63	90	14	100	1	180	6
7	12.4	135	21	135	2.5	121	4	45	4
8	8.6	157.5	33	180	4	180	3	112.5	6 (period 20 s)
9	19.5	180	21	157.5	2.5	239	1	157.5	6
10	-4.5	202.5	47	225	6	260	2	270	4
11	16.8	225	21	202.5	2.5	266	3	292.5	2
12	5.5	247.5	10	247.5	1.25	277	4	202.5	6 (period 20 s)
13	14.4	270	33	292.5	4	286	2	315	2
14	-5.8	292.5	47	270	6	302.5	3	247.5	4
15	10	315	21	315	2.5	83	1	337.5	6
16	21. 3	337.5	33	337.5	4	100	4	225	6 (period 20 s)

Table B.8 – Auto-telegraph training and simulation results

Scenario	Set Speed (kt)	Sea trial RPMs for desired speed (no external disturbance)	Longitudinal external influence on speed (kt)	Achieved Speed (kt)	Achieved RPMs	Training Duration (seconds)	Training Visualization
1	18	69	+1.38	17.98	65	1248.5	
2	13.3	52	+0.76	13.46	52	1251.8	
3	4	23	+3.19	3.19	0	1251	

4	11.8	49	+0.21	11.41	47	1257.5	
5	20.4	86	-0.04	20.26	86	452.64	
6	7.2	29	-0.45	6.35	29	1308	
7	12.4	51	-1.73	12.57	56	539.76	

8	8.6	38	-2.90	8.50	48	1122.6	
9	19.5	83	-0.01	19.49	83	1114.6	
10	-4.5	-29	-0.39	-4.29	-29	1412.3	
11	16.8	66	-0.03	16.97	67	280.85	
12	5.5	23	+1.16	6.36	23	1230.5	

13	14.4	56	+1.06	14.26	53	145.97	
14	-5.8	-46	+2.50	-3.75	-46	1246.7	
15	10	38	+0.52	9.52	38	1392.4	
16	21. 3	89	-1.24	20.06	89	1118.6	

Appendix C

COLREGs Classification Algorithm

In this Appendix an algorithm for COLREGs classification problem is delivered. COLREGs classification algorithm focuses on own vehicle's and targets' attitude, so course and speed through water are significant to determine which Rule will be appropriate for each collision situation. It is necessary to keep in mind that the algorithm is developed with decision support model in mind where interaction with human navigators is still required, while it has to be modified if it would be used for autonomous navigation.

Algorithm C – COLREGs Classification Algorithm

Input: H_F, *COG, CTW, SOG, RPM, STW*, n_{OV}, e_{OV}, h, D, *ECDIS info*,

$T_{1,2,\ldots,n}(H_T, COG_T, CTW_T, SOG_T, STW_T, n_T, e_T, AIS_T, dCPA_T, TCPA_T, R_T, \theta_T, BCR)$.

Output: Display relevant COLREGs Rules

Every 10 seconds do:

  Rule 6

  for *RPM* = NAV FULL:

    if visibility < 3 NM:

    display: RULE 6 – CONSIDER SLOWING TO MANEUVERING FULL

    end if

    if visibility < 3 NM, and $R_T \leq 3$ NM:

    display: RULE 6 – SLOW TO MANEUVERING FULL

    end if

    if ECDIS look-ahead (safety cone) encounters safety contour alarm, and/or obstacle alarm (user selects look ahead parameters):

display: RULE 6 – SLOW TO MANEUVERING FULL

end if

if any of the following conditions are true: $1.5 < h/_D \leq 3.0$;

$(n_{OV}, e_{OV}) => TSS$, narrow channel, or safety fairway:

display: RULE 6 – CONSIDER SLOWING TO MANEUVERING FULL

end if

if any of the following conditions are true: $1.5 < h/_D \leq 3.0$;

$(n_{OV}, e_{OV}) => TSS$, narrow channel, or safety fairway **and** $n_T > 4$,

where $R_T \leq 3$ NM:

display: RULE 6 – SLOW TO MANEUVERING FULL

end if

if $h/_D \leq 1.5$:

display: RULE 6 – SLOW TO MANEUVERING FULL

end if

end for

***Rule** 7*

Read $T_{1,2,\ldots,n}(dCPA_T, TCPA_T, R_T, \theta_T)$

for each T_n and until T_n is cancelled:

store θ_{T_t} every 60 seconds

end for

for each T_n do:

if $dCPA_T < CPA_{MIN}$ and $TCPA_T < 30\ minutes$:

display: RULE 7 – COLLISION RISK EXISTS

end if

end for

for each T_n:

if $R_T < 6\ NM$ and $\theta_{T_{t-1}} \pm 2° < \theta_{T_t} < \theta_{T_{t-1}} \pm 2°$:

display: RULE 7 – COLLISION RISK EXISTS

end if

end for

Rule 9

if $N_{OV}, E_{OV} = ECDIS\ Narrow\ Channel$:

display: RULE 9 – VESSEL IN A NARROW CHANNEL

end if

Rule 10

Read AIS for vessel type and read vessel type entry by navigators:

if $N_{OV}, E_{OV} = ECDIS\ \ TSS$:

display: RULE 10 – VESSEL IN A TSS

end if

if $n_{OV}, e_{OV} = ECDIS\ \ TSS$ and vessel type = vessel restricted in her ability to maneuver engaged in an operation for the maintenance of safety of navigation:

display: RULE 10 – TSS EXEMPTION – PROCEED WITH CARE

end if

if $n_{OV}, e_{OV} = ECDIS\ \ TSS$ and vessel type = vessel restricted in her ability to maneuver engaged in an operation for the laying, servicing or picking up of a submarine cable:

display: RULE 10 – TSS EXEMPTION – PROCEED WITH CARE

end if

Rule 13

verify information extracted from water stabilized RADAR

for each T_n:

for T_n with bearing from own ship $112.5° \leq \theta_T \leq 247.5°$:

if $R_T < 3\ NM, STW < STW_T,$ and $CPA \leq CPA_{PREF}$:

display: RULE 13 – OVERTAKEN BY T_n – KEEP COURSE AND SPEED

end if

end for

end for

for each T_n:

read $T_n(H_T, CTW_T, STW_T, n_T, e_T, dCPA_T, TCPA_T, R_T, \theta_T)$

for n_{OV}, e_{OV} in the sector behind the T_n ($247.5° \leq \theta_{OV} \leq 112.5°$):

if $R_T < 3\ NM, STW > STW_T$ and $CPA \leq CPA_{PREF}$:

display: RULE 13 – OVERTAKING T_n – GIVE-WAY VESSEL

end if

end for

end for

Rule 14

verify information extracted from water stabilized RADAR

for each T_n:

read $T_n(H_T, CTW_T, STW_T, n_T, e_T, dCPA_T, TCPA_T, R_T, \theta_T)$

for n_{OV}, e_{OV} in the sector ahead of the T_n ($354° \leq \theta_{OV} \leq 006°$) and T_n with bearing from own ship $354° \leq \theta_T \leq 006°$:

if $R_T < 6\ NM$, and $CPA \leq CPA_{PREF}$:

display: RULE 14 – HEAD-ON

end if

end for

end for

Rule 15

verify information extracted from water stabilized RADAR

for each T_n:

read $T_n(H_T, CTW_T, STW_T, n_T, e_T, dCPA_T, TCPA_T, R_T, \theta_T, BCR)$

if $247.5° < \theta_{T_n} < 354°$, $R_T < 6\ NM$, $BCR > 0$, and $CPA \leq CPA_{PREF}$:

display: RULE 15 – T_n CROSSING BOW FROM PORT – STAND-ON

end if

if $247.5° < \theta_{T_n} < 354°$, $R_T < 6\ NM$, $BCR \leq 0$, and $CPA \leq CPA_{PREF}$:

display: RULE 15 – T_n CROSSING STERN FROM PORT – STAND-ON

end if

if $006° < \theta_{T_n} < 112.5°$, $R_T < 6\ NM$, $BCR > 0$, and $CPA \leq CPA_{PREF}$:

display: RULE 15 – T_n CROSSING BOW FROM STARBOARD – GIVE WAY

end if

if $006° < \theta_{T_n} < 112.5°$, $R_T < 6\ NM$, $BCR \leq 0$, and $CPA \leq CPA_{PREF}$:

display: RULE 15 – T_n CROSSING STERN FROM STARBOARD – GIVE-WAY

end if

end for

Rule 16

if Rule 13 = GIVE-WAY, Rule 15 = GIVE-WAY, or Rule 18 = GIVE-WAY:

display: RULE 16 – GIVE-WAY VESSEL

end if

Rule 17

if Rule 13 = STAND-ON, Rule 15 = STAND-ON, or Rule 18 = STAND-ON:

display: RULE 17 – STAND-ON VESSEL

end if

Rule 18

for all T_n:

Read manual inputs from navigators or AIS

if own vehicle = WIG-IN-GROUND (WIG) operating on the water surface:

refer to own vehicle = POWER DRIVEN UNDERWAY

end if

if own vehicle = POWER DRIVEN UNDERWAY and T_n = NOT UNDER COMMAND, or T_n = RESTRICTED IN HER ABILITY TO MANEUVER, or T_n = ENGAGED IN FISHING, or T_n = SAILING VESSEL:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle = SAILING VESSEL and T_n = NOT UNDER COMMAND, or T_n = RESTRICTED IN HER ABILITY TO MANEUVER, or T_n = ENGAGED IN FISHING:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle = ENGAGED IN FISHING and T_n = NOT UNDER COMMAND, or T_n = RESTRICTED IN HER ABILITY TO MANEUVER:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle ≠ NOT UNDER COMMAND, or own vehicle ≠ RESTRICTED IN HER ABILITY TO MANEUVER and T_n = CONSTRAINED BY HER DRAUGHT:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle = CONSTRAINED BY HER DRAFT and T_n ≠ NOT UNDER COMMAND, or T_n ≠ RESTRICTED IN HER ABILITY TO MANEUVER:

display: RULE 18 – NAVIGATE WITH CAUTION

end if

if own vehicle = CONSTRAINED BY HER DRAFT and T_n = NOT UNDER COMMAND, or T_n = RESTRICTED IN HER ABILITY TO MANEUVER:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle = SEAPLANE:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

if own vehicle = WIG-IN-GROUND (WIG) taking off, landing or flight near surface:

display: RULE 18 – KEEP OUT OF THE WAY OF T_n

end if

end for

Rule 19

for $visibility \leq 3\ NM$:

display: RULE 19 – CONSIDER INCREASING CPA RADII TRESHOLD

if own vehicle = POWER DRIVEN UNDERWAY and $R_T \leq 3$ NM:

display: RULE 19 – SLOW TO MANEUVERING FULL

end if

if hearing fog signals in the clockwise sector [270°, 180°] and $CPA \leq CPA_{PREF}$:

display: RULE 19 – REDUCE SPEED TO MINIMUM

end if

end for

Rule 8

for Rule 7 = TRUE and Rule 16 = TRUE, then for each T_n:

if $R_T < 6\ NM$:

display: RULE 8 – MANEUVER EARLY AND APPARENTLY

end if

end for

end

Classification algorithm is separate from the collision avoidance algorithm and the main function is to utilize vehicles' water geometries and to determine appropriate attitude for accurate COLREGs Rule determination. Once the appropriate Rules have been classified, the collision avoidance algorithm exploits Rules classification as constraints, which are then used for reward design and shaping.

Even though most of the variables have been already described earlier in text; in order to aid clearer understanding, clarification is offered: H_F – filtered heading of the own vehicle taken from the gyro compass and filtered by the FPF, *COG* – Course Over Ground for the own vehicle taken from the GPS or radar, *CTW* – Course Through Water taken from the radar, *SOG* – Speed Over Ground taken from the GPS or radar, *RPM* – Revolutions Per Minute taken from the engine speed indicator directly, performance measurement monitoring, or conning display, *STW* – Speed Through Water taken from the speed log or radar, n_{OV} – GNSS north position of the own vehicle (can be also fused information), e_{OV} – GNSS east position of the own vehicle (can be also fused information), h – draught of the own vehicle taken from the loadicator computer or manual input in order to verify safe waters, *D* – depth of water, *ECDIS info* – various ECDIS available information, especially position of safe waters, fixed obstructions, temporary notices, TSS, narrow channel, and other relevant information needed for safe navigation. When tracking a new target T_n, following information is of interest: H_T – heading of a target (not filtered) taken from the radar, COG_T – Course Over Ground for a target taken from the radar, CTW_T – Course Through Water of a target taken from radar, SOG_T – Speed Over Ground of a target taken from radar, STW_T – Speed Through Water taken from radar, n_T – GNSS north position of a target, e_T – GNSS east position of a target, AIS_T – various Automatic Identification System information of a target taken from the AIS receiver, $dCPA_T$ – distance to Closest Point of Approach (usually called simply a CPA) of a target in relation to own vehicle taken from the Automatic Radar Plotting Aid (ARPA), $TCPA_T$ – Time to the CPA of a target in relation to own vehicle taken from ARPA, R_T – Range of a target taken from radar, θ_T – bearing of a target taken from radar, θ_{OV} – bearing of the own vehicle from the perspective of a target (calculated after acquiring new target), and BCR – Bow Crossing Range taken from ARPA, which can be positive (bow crossing), or negative (stern passing).

Nomenclature

Symbol	Description	Page no.
t	Time step	39
s	State $s \in S$	39
S	State space	39
a	Action $a \in A$	39
A	Set of actions	40
o	Observation $o \in O$	40
O	Set of observations (sensory information)	40
T	Test signal	41
R	Real incident situation signal	41
I	Additional distress information signal	42
Y	Some random occurrence	45
H	Hidden (unobservable) state space	48
h	Unobservable state $h \in H$	48
R_a	Real-valued reward	50
γ	Discount factor $(0 < \gamma \leq 1)$	50
H	Horizon	50
π	Policy	51
π^*	Optimal policy	51
$V^*(s)$	Utility of a state s	55
$Q^*(s, a)$	Utility of a Q state	55
Ω	Observation function	60
b	Probability distribution over the state space S (belief state)	61
η	Normalizing factor	61

R	Reward function	61
τ	Belief state transition function $\tau \in B$	61
B	Set of belief states	61
α	Learning rate $(0 < \alpha \leq 1)$	69
x_t	State to be estimated	76
z_t	Measurement	76
u_t	Control in the system dynamics	76
$f_x(\cdot)$	Process function	76
$f_z(\cdot)$	Measurement function	76
$f_u(\cdot)$	Control function	76
v_{x_t}	State noise	76
v_{z_t}	Measurement noise	76
v_{u_t}	Control noise	76
PN	Number of particles	77
q	Importance distribution	77
$w_t^{[j]}$	Importance weight	77
$\widetilde{w}_t^{[j]}$	Normalized importance weight	77
$\delta(\cdot)$	Dirac delta measure	78
NP_{eff}	Effective sample size	78
$fit_x^{[j]}$	Fitness function	80
v_k	Measurement noise covariance	81
z_t	Newest observation	81
$\varpi^{[i]}$	Random number in the interval $[-1,1]$	81
$NP_t^{[j]}$	Number of scions for the parent particle $x_t^{[j]}$	82
d_t	Independent zero-mean Gaussian noise	84
e_t	Independent zero-mean Gaussian noise	84
n	North position	95
e	East position 95	95

ψ	Yaw	95
u	Surge	95
v	Sway	95
r	Yaw rate	95
$\boldsymbol{R}(\psi)$	Rotation matrix of yaw	95
$\boldsymbol{\tau}_{control}$	Vector of control force	96
$\boldsymbol{\tau}_{wind}$	Vector of wind force	96
$\boldsymbol{\tau}_{waves}$	Vector of wave control	96
X	Surge force	96
Y	Sway force	96
N	Yaw moment	96
$\boldsymbol{M}_{RB}$	Rigid-body mass matrix	96
$\boldsymbol{C}_{RB}(\boldsymbol{\nu})$	Skew-symmetric Coriolis-centripetal matrix	96
$\boldsymbol{M}_A$	Positive-definite hydrodynamic added mass matrix	96
$\boldsymbol{d}(V_{rc}, \gamma_c)$	Current and damping factors	96
x_g	Longitudinal center of gravity relative to the body-fixed frame	97
m	Mass of a sea surface vehicle	97
I_z	Moment of inertia about the z-axis of the body-fixed frame	97
$X_{\dot{u}}$, $Y_{\dot{v}}$, $Y_{\dot{r}}$, $N_{\dot{v}}$, $N_{\dot{r}}$	Added-mass coefficients	97
u_c , v_c	Components of the current velocity in the body-fixed frame	97
γ_{rc}	Angle of the current relative to the bow of the sea surface vehicle	97
$C_{X_c}(\gamma_{rc})$	Surge nondimensional current coefficient	97
$C_{Y_c}(\gamma_{rc})$	Sway nondimensional current coefficient	97
$C_{N_c}(\gamma_{rc})$	Yaw nondimensional current coefficient	97

LOA	Length Over All	98
A_{F_c}	Frontal projected areas of the submerged part of the hull	98
A_{L_c}	Lateral projected areas of the submerged part of the hull	98
ρ	Density of the water	98
$\boldsymbol{D}$	Linear damping	98
$\boldsymbol{b}$	Slowly varying bias / vector of non-modeled external forces and moments caused by the effects of wind, waves and currents related to the earth-fixed frame	98
$\boldsymbol{x}$	Constant matrix that gives a transmission between the input and the thrust	98
$\boldsymbol{u}$	Control inputs	98
$\boldsymbol{M}$	Inertia matrix	99
ρ_a	Density of air	100
A_{F_w}	Frontal projected wind area	100
A_{L_w}	Lateral projected wind area	100
V_{rw}	Wind speed	100
γ_{rw}	Relative wind direction	100
V_w	Speed of the wind	100
β_w	Direction of the wind relative to the earth-fixed frame	100
$\boldsymbol{T}$	Diagonal matrix of positive bias time constants	100
$\boldsymbol{\phi}$	Diagonal matrix scaling the amplitude of $\boldsymbol{\omega}$	100
$\boldsymbol{\omega}$	Vector of zero-mean Gaussian white noise	100
$\boldsymbol{\xi}$	Wave force state vector	101
$\boldsymbol{\Omega}, \boldsymbol{\Sigma}, \boldsymbol{\Gamma}$	Constant matrices of appropriate dimensions	101
$\boldsymbol{\eta}_{\omega}$	Wave induction motion	101
ζ_i	Relative damping ratio	102
w_{0i}	Dominating wave frequency	102
K_{wi}	Wave intensity parameter	102

$\boldsymbol{\nu}$	Zero-mean Gaussian white measurement noise	103
$\boldsymbol{u}$	Actuator measurements	103
$\boldsymbol{A}$	State (system) matrix	105
$\boldsymbol{B}$	Input matrix	105
$\boldsymbol{E}$	Process noise amplitude matrix	105
$\boldsymbol{H}$	Output matrix	105
h	Sample time	105
$\mathcal{T}$	Transition function	117
F	Shaping reward function	117
Φ	Shaping potential	117
Q^*_{MDP}	Optimal Q-function of the native MDP	118
ε	Greedy factor $(0 \leq \varepsilon \leq 1]$	123
β	Random number $(0 \leq \varepsilon \leq 1]$	123
H	Heading	149
H_d	Desired heading that is selected by a navigator	149
H_F	Filtered heading	149
ROT	Rate Of Turn	149
R_d	Rudder deflection	149
M_{AP}	Autopilot modes	149
P_n	Pumps number	149
C_d	Desired course	155
COG	Course Over Ground	155
H_R	Requested heading	155
S_R	Speed requested	155
SOG	Speed Over Ground	155
RPM	Revolutions Per Minute	155
W_S	Wind speed	155
W_D	Wind direction	155

$Curr_S$	Current speed	155
$Curr_D$	Current direction	155
S_{STATE}	Sea state	155
S_D	Sea waves direction	155
SW_H	Swell height	155
SW_D	Swell direction	155
M_{AP1}	Autopilot modes	155
M_{AP2}	Track modes	155
r	Planned turn radius that is extracted from a passage plan	171
VRM	Variable Range Mark	171
ROT_D	ROT demanded	171
	ROT_A – ROT actual	171
d_{LOG}	Log depth	171
s_t	Testing state	175
s_{ED}	State with expert demonstration	175
Σ	Covariance responsible to measure the influence of demonstrated state-action pairs	175
B_{ED}	Buffer of expert demonstration	176
B_{EX}	Exploitation buffer	176
B_{BEST}	Eest of the exploitation and expert demonstration buffers	176
$B_{SIMULATION}$	Simulation buffer	176
SOG_R	SOG requested	184
SOG_A	SOG actual	184
RPM_R	RPM requested	184
RPM_A	RPM actual	184
B_{BM}	Boiler burner mode	184
R_{COLL}	Collision safety zone radius	202

R_{NM}	Near Miss safety zone radius	202
R_{MIN}	Minimum safety zone radius	202
R_{SAFE}	Safe safety zone radius	202
e_{XP}	Expansion factor	241
$coll_{freq}$	Collision frequency	241
f_{ACTS}	Incorrect answers in the ACTS questionnaire factor	241
r_{EXP}	Expanded radius of the safety zone	241
r_{SEL}	Selected radius prior expansion	241
$\mathbb{t}$	Trajectory	247
$\mathbb{T}_T$	Set of target trajectories	247
$\mathbb{T}_{OWN}$	Own vehicle trajectories	247
O_H	Hazard information	247
$\mathbb{t}_R$	Random initial own vehicle trajectory	247
$\mathbb{t}_P$	Planned route	247
CPA	Closest Point of Approach	248
φ	Latitude	249
λ	Longitude	249
x	Cartesian coordinate abscissa	249
y	Cartesian coordinate ordinate	249
z	Cartesian coordinate applicate	249
a_{elip}	Semi-major axis of the Earth reference ellipsoid	249
e^2	First eccentricity squared	249
S_{OV}	Global state vector	279
s_{OV}	State vector of own vehicle $s_{OV} \in S_{OV}$	279
VI_{OV}	Vehicle intent	279
PI_{OV}	Passage intent	279
UI_{OV}	Utility intent	279
V	Sea surface vehicle	285

w	Waypoint	285
PP	Passage Plan	285
W	Set of waypoints	286
$\vec{r}$	Position vector	287
R^F	Rewards fusion	303

List of Figures

List of Tables

List of Algorithms