Carl-von-Ossietzky
Universität Oldenburg

Fakultät II – Informatik, Wirtschafts- und Rechtswissenschaften
Department für Informatik

Maritime Trajectory Negotiation for n-Vessel Collision Avoidance

Dissertation zur Erlangung des Grades eines Doktors der Ingenieurswissenschaften

vorgelegt von Herrn

Sascha Alexander Hornauer
geboren am 23.04.1983 in Köln

1. Gutachter: Prof. Dr.-Ing. Axel Hahn
2. Gutachter: Prof. Dr.-Ing. Johannes Reuter
Disputation vom 09.06.2016

März 2016

Sascha Alexander Hornauer

Maritime Trajectory Negotiation for n-Vessel Collision Avoidance

ABSTRACT

In the past centuries, technical advances in several fields reduced the risk involved in maritime traffic through improved situation awareness and guidance in critical situations. However, collisions and groundings are still severe problems today, especially with increasing traffic density and speed as well as growing ship sizes. The main reason for collision related disasters is still the human factor, which is periodically confirmed every few years by several studies. Guidance from navigation information and assistance systems provide navigators with situation awareness even in bad visibility, though an increasing number of alarms and separated systems compete for attention. Another aid and challenge at the same time is a domain, where processes are standardised by international regulations for preventing collisions at sea which require a coordinated established response by ships in defined situations. However, misunderstandings and ambiguity in the application of the rules in combination with overwhelming information at the bridge in close-quarter situations lead to error prone ship handling today. At the same time the navigation systems on the bridge try to enable the human crew to find safe and collision free trajectories for all ships at risk instead of developing a common coordinated solution themselves.

One solution is the automatic search for optimal ship trajectories for each ship in a situation, which can be used to steer ships autonomously or to give advice to navigators, integrated in a ship's guidance system. Successful introduction of an assistance system to coordinate and chose an evasive vertical action in the aviation domain serves as a model even though ships are confined to a two-dimensional manoeuvre. Optimal trajectories in the maritime domain must not only allow for minimal resource consumption of ships, on those trajectories, but also ensure safe passage at every point in time. Furthermore, it is beneficial to comply to legal frameworks as the collision avoidance regulations to ease integration of the approach in real world ship handling procedures and allow for unequipped ships to interpret the behaviour of any ship, using the approach.

The search for optimal trajectories in current research focuses often on situations in which full information are available at the position of a global planner. An approach considered unrealistic on the high sea given bandwidth limitations, unknown physical properties of possibly colliding ships and unknown intentions of the human crew as well as incomplete information about the environment.

This thesis investigates decentralised collision avoidance procedures which use a dedicated negotiation system to optimise locally found trajectories according to a global but decentralised performance measure. The system is based on local information which are extended by exchanging trajectories alone in the negotiation with other possibly colliding ships and without explicit additional information sharing. The modelled negotiation finds an initial collision free solution and improves it towards a near-optimal and fair desired outcome of non-colliding trajectories which lead to minimal and equal resource consumption.

As a comparable approach across domains, a Nash Bargaining schema is implemented with a cost metric designed to converge to a Nash Bargaining Solution. In a multi agent framework, agents negotiate according to the procedure trajectories, found by an external path planner, designed to observe the kinematic limitations of the ships and legal constraints. The approach is evaluated in a simulation environment and shows fulfilment of the defined requirements.

Sascha Alexander Hornauer

Maritime Trajektorienverhandlung für die Kollisionsverhütung von n-Schiffen

KURZFASSUNG

In den vergangenen Jahrhunderten ermöglichten es technische Fortschritte auf verschiedenen Gebieten, die Gefahren im maritimen Verkehr zu reduzieren durch verbesserte Lagebilder und Hilfe zur Orientierung in kritischen Situationen. Probleme wie Kollisionen und auf Grund laufen sind jedoch auch Heute noch ein Problem, befördert durch steigende Verkehrsdichte und -geschwindigkeit von immer größeren Schiffen. Dabei ist einer der Hauptgründe für Havarie aufgrund von Kollisionen immer noch der menschliche Faktor, wie stetig in verschiedenen Studien bestätigt wird. Assistenzsysteme und Hilfen zur Navigation bieten Seefahrern ein Lagebild sogar bei schlechter Sicht, während allerdings eine Vielzahl von Alarmen in voneinander getrennten Systemen sehr viel Aufmerksamkeit erfordert. Eine weitere Hilfe sowie Herausforderung gleichermaßen ist das abgestimmte Verhalten in Situationen in der Domäne, welches nach internationalen Kollisionsverhütungsregeln standardisiert wurde. Missverständnisse und nicht eindeutige Auslegung der Regeln in Kombination mit überwältigend vielen Informationen auf der Brücke kann zur fehleranfälligem Verhalten in kritischen Situationen führen. Gleichermaßen versuchen die verschiedenen Systeme auf heutigen Schiffsbrücken die menschliche Besatzung in die Lage zu versetzen, sichere und kollisionsfreie Trajektorien für alle Schiffe in einer kritischen Situation zu finden, anstatt eigenständig ein abgestimmtes Manöver zu entwickeln. Eine mögliche Lösung ist die automatische Suche nach optimalen Schiffstrajektorien für alle Schiffe in einer Situation, welche für autonome Steuerung oder als Teil eines Assistenzsystemes genutzt werden kann. Die erfolgreiche Einführung eines Systems in der Luftfahrt, welches Ausweichempfehlungen abstimmt, dient als Modell, wenn auch Schiffe auf die zweidimensionale Ebene beschränkt sind. Optimale Trajektorien in der maritimen Domäne sind dabei nicht nur Ressourcenschonend sondern müssen auch die sichere Passage von Schiffen garantieren. Des weiteren ermöglicht ein regelkonformes Verhalten eines Systems, gemäß der Kollisionsverhütungsregeln, einfachere Integration in reale Anwendungsszenarien und bietet Schiffen ohne ein solches System die Möglichkeit das Verhalten zu interpretieren. Die Suche nach optimalen Trajektorien in der derzeitigen Forschung konzentriert sich dabei oft auf Situationen, in denen alle Informationen für einen globalen Planer zur Verfügung stehen. Dieser Ansatz wird auf hoher See, mit ungenauen Informationen über die Umwelt als unrealistisch eingeschätzt, da neben Beschränkungen der Bandbreite der Kommunikation die dynamischen Eigenschaften und Absichten anderer Schiffe lokal unbekannt sind. Diese Arbeit erforscht dezentrale Kollisionsverhütung mittels eines Verhandlungssystems welches lokal gefundene Trajektorien global optimiert unter Verwendung eines dezentralen Leistungsmaßes. Das entwickelte System basiert auf lokalen Informationen welche durch die Verhandlung von Trajektorien mit anderen Schiffen erweitert werden ohne explizite weitere Informationen auszutauschen. Die modellierte Verhandlung findet in kurzer Zeit eine initiale kollisionsfreie Lösung und verbessert sie iterativ entgegen einer optimalen und fairen Lösung, welche den Ressourcenverbrauch minimiert und den Bedarf gleich verteilt. Dabei wird ein in verschiedenen Domänen eingesetzter Nash Bargaining Ansatz imple- mentiert welcher mittels der Kostenmetrik entgegen einer Nash Bargaining Solution kon- vergiert. Ein Agent in einem Multiagentensystem verhandelt im entwickelten Verfahren Trajektorien, welche von einem externen Pfadplaner gefunden werden. Dieser Ansatz er- möglicht es die kinematischen Beschränkungen eines Schiffs sowie die Kollisionsverhü- tungsregeln zu berücksichtigen. Das entwickelte System wird mittels Experimenten in einer Simulationsumgebung evaluiert und die Erfüllung der gestellten Anforderungen wird gezeigt.

Contents

1

Introduction

Since over 40 years, the worldwide maritime traffic increased for a variety of purposes and with a variety of means. In between 1970 and 2013 the millions of tons loaded in the international seaborne trade increased from 2605 t to 9548 t (UNCTAD, 2014). Collisions in the maritime domain are still an ongoing challenge, leading often to disastrous repercussions and the loss of lives (Mou et al., 2010).

In the following the problem of maritime collision avoidance is motivated and defined in the context of present rules and customs in a challenging domain. First enhanced measures of collision avoidance are motivated in the context of the versatile and growing maritime traffic. Current navigation and information systems, which are used for collision avoidance are explained and difficulties and limits for full automation of vessels are stated. After a comparison with concepts from the aviation domain, a review of the most relevant existing rules for preventing collisions on sea is given along with a discussion about their implication for a collision avoidance system. A first informal statement of the problem is derived and trajectory optimisation is presented as a possible solution. In the end of the chapter the contribution of this work is stated and the structure of the thesis explained.

1.1 MARITIME NAVIGATION

Maritime traffic is very complex containing many diverse participants and authorities with different intentions and capabilities. Maritime collision avoidance must include all the different moving and stationary objects, under command or drifting afloat, which form the complex maritime domain. In the past, a need arose for laid down international rules, when old fashioned sailing ships became scarce and more and more steam, fuel and finally electrical driven ships became the dominant vessel type. Manoeuvrability improved with less need to use the wind as a mean of propulsion and decreased again with larger sizes and higher load.

In modern days the means to avoid collisions can be based upon many different sources of information. Ships on the open sea or near the shore are connected up to different degrees. Navigation information and assistance systems are based on various sensory information about the environment, which are directly gathered or received from external ships, systems or authorities. Conflicts in interest arise when limited space must lead to detours or waiting times but even though other ships may be seen as antagonists, collision avoidance is always considered a cooperative effort in all observed situation. Collisions between ships happen due to different reasons, often as a consequence of human error, as it was established in the past through thorough investigations, unfortunately often in the aftermath of severe disasters Hetherington et al. (2006).

1.2 DENSELY POPULATED AND CHANGING ENVIRONMENT

Collision avoidance is done against the background of a sometimes unpredictable nature which must be protected and acts sometimes in favour and sometimes against the purpose of the lives involved.

In the world today we differ between international waters on the high sea, an economic zone within 200 nautical miles of solid land, a contiguous zone of 12 nautical miles followed by territorial waters of 12 nautical miles next to the shore. Furthermore we have to discriminate special zones as traffic separation schemes for busy shipping areas (Davidson and Davidson, 1997), inland waters, rivers and lakes. The members of the domain range from very fast and manoeuvrable recreational vessels as yachts, smaller tug boats near a harbour, middle sized training or research vessels up to large oil tankers of a length of over 450 meters Clarkson Research Studies Ltd (1987).

Furthermore different kind of obstacles, static and dynamic exist and have to be avoided. Buoys are used to guide navigation or mark safe waters, underwater cables or other hazards. Landmasses, sand banks, riffs, lighthouses and unmanned ship wrecks are static obstacles which can be noted on a map while flotsam, mines or unmanned vessels have to be detected on the various sensors present on a modern ship's bridge.

In addition, the area near the shore can be expected to be confined further by economic interest. In the German Exclusive Economic Zone (EEZ) alone, the federal maritime and hydrographic agency approved 1137 offshore wind installations in 16 different wind farms (Bundesamt für Seeschifffahrt und Hydrographie, 2015).

1.3 MARITIME INFORMATION AND NAVIGATION SYSTEMS

Depending on the surroundings, different amounts of information about the environment is available on board. Due to the round shape of the earth the horizon is a natural limit for information based on line of sight. Radar waves can be used in situations with bad visibility but do not penetrate landmasses. Satellite or radio-based communication can bridge this gap when the ship is sufficiently equipped. Near to the shore special navigational services can provide additional guidance.

Information about other vessels and their intentions exists as well and can be collected via different means. These information are useful for mariners and collision avoidance planning approaches alike, though each kind of information is not equally easy obtained. The identity, heading, speed and long term intentions of other ships can be supplied via navigation supporting technologies as the Automatic Identification System (AIS), which is a mandatory addition to almost every larger class of ships (International Maritime Organisation, 2004) and sends information using the VHF maritime mobile frequency band. The position and speed can also be locally gathered through the Automatic Radar Plotting Aid (ARPA), especially in situations where communications fail. Information about the voyage plan is already present in the navigational computer of larger ships and could be transmitted to make the intended behaviour known (Porathe, 2012). In harbours or near the shore, Vessel Traffic Services (VTS) are operated, based on guidelines laid down by the International Maritime Organization (IMO) (International Maritime Organisation, 1997). Those services can collect and offer information, traffic organisation and navigational assistance to ships in their dedicated VTS area (Weintrit, 2014).

1.4 AUTONOMOUS BEHAVIOUR EXECUTION

Reliable collision avoidance systems can not only act as assistance to human personnel but are also a prerequisite for any autonomous system. Based on the current state of the art fully autonomous control of ships may be possible in the near future though the development faces hurdles which are encountered in a similar way in the automotive and aeronautic domain. There is an ever growing number of proof-of-concepts from small unmanned drones to larger ships. Semi-autonomous cargo ships are in development which may in the future travel large distances unsupervised and which could be monitored from the shore (Tvete, 2014). However, two of the biggest issues in autonomous driving, on sea or on land, are still the adaptation to human behaviour and legislative questions regarding liability. In the automotive domain, national and international rules apply which were adapted in the past decade to allow for assistance systems to be used by drivers. The current international framework for the automotive domain, which was agreed upon by the UN in Vienna in 1968, (Bundesversammlung Schweiz, 1968), was amended to allow for assistance systems to being used even though still "...Keeping the driver in a superior role is a guiding principle of road traffic regulations ..." (Economic Commission for Europe, 2014). This reasoning can also be found in maritime legislation, where the captain has the ultimate authority and responsibility over its vessel Toremar (1999). In an effort to advance the research possible on autonomous cars, Volvo declared in 2015 that it will "...accept full liability whenever one if[sic] its cars is in autonomous mode ..." (Volvo, 2015).

Autonomous systems which replace direct human control on a ship entirely might emerge in the next decades if the quality of the sensor information and the artificial intelligence, needed to handle ship procedures continue to improve. Due to the international domain in which maritime traffic is conducted, isolated national efforts are ultimately not enough to overcome legal obstacles and international standardisation through the various authorities is necessary for any new autonomous or assistance system. Seamless integration into current maritime procedures, where autonomous systems might have to operate alongside human personnel for a long period of time, can only be realised by following the mandatory procedures laid down as the collision avoidance regulations.

1.5 Regulations for Preventing Collisions

In an effort to standardise earlier national, archaic rules and customs, which existed for several hundred, if not thousand years, international regulations are in place to harmonise behaviour, handle liability questions and define the mandatory equipment to be present on every ship of a certain size. The current used set of regulations, the Conventions on the International Regulations for Preventing Collisions at Sea (COLREG)s date back to an international conference in London in 1972 (Cockcroft and Lameijer, 2003).

The COLREGs were by 2016 ratified in 156 countries and included in national law (International Maritime Organisation, 2016). In Germany, the national *Seeschifffahrtsstraßen-Ordnung* (SeeSchStrO) defines its specialised application and its exemptions. It contains a few deviations i.e. from Rule 9 b) to d) and Rule 15 and 18, a) to c) in §25 (SeeSchStrO) where contrary to the COLREGs ships in defined fairways have in more situations the right of way (Bundesrepublik Deutschland, 1998). The specific national implementations of the COLREG were not examined in this work when defining requirements for collision avoidance.

Avoiding collisions, groundings and dangerous situations is the key motivation for all legislation concerning traffic separation schemes, lights as well as sound signals and rules concerning the evasive behaviour when a collision seems imminent. They are designed to minimise ambiguity in the behaviour of other, perceived ships, especially with limited situation awareness i.e. during bad weather.

However, due to the nature of current maritime operations, collision avoidance regulations were developed with human seafaring personal in mind, assisted by electronic navigation systems as RADAR or other aids as described in section 1.3. The proposed solution to the trajectory optimisation problem could operate within the collision avoidance regulations, however manoeuvres beyond their strict implementation are also investigated because autonomous ships could benefit from a more lenient procedure.

The most important regulations currently in place are the following, listed in an abridged version and cited from Cockcroft and Lameijer (2003). They impose requirements on any developed collision avoidance system, which can be fulfilled, but will not at all times to achieve a better overall performance and lower resource consumption. To clarify the possible conflicts when implementing a system based on the proposed approach, a traditional vessel with a human crew, assisted by current navigation systems, is seen in contrast with an autonomous ships using the negotiation and optimisation scheme, developed in this work. While details about the proposed procedure will be explained in the next chapters, basic information which are needed to understand the context of the discussion of the rules are that the developed system in this work optimises trajectories for ships by transmitting them to all other in a critical situation to negotiate towards an optimal set of trajectories for all participants and thereby making all intentions known early in the process.

Rule 5 - Look-out *Every vessel shall at all times maintain a proper look-out by sight and hearing as well as by all available means appropriate in the prevailing circumstances and conditions so as to make a full appraisal of the situation and of the risk of collision.*

A full appraisal of a situation would benefit largely by exact information about the intended trajectories of other ships. Earlier work investigated the possible advantages already by "... letting ships exchange routs[sic] by use of the *Electronic Chart Display and Information System* (**ECDIS**) ..." (Porathe et al., 2012b). For autonomous ships this could be translated to the requirement of having sensors for fully audio/visual reconnaissance to replace a human navigator. This might be especially difficult when realising the recognition of flags or blown horns and pipe signals.

Rule 6 - Safe speed *Every vessel shall at all times proceed at a safe speed so that she can take proper and effective action to avoid collision and be stopped within a distance appropriate to the prevailing circumstances and conditions. . . .*

This rule is an example where the different capabilities of humans and computer systems lead to a differing interpretation. A ship, with the exact information about every single trajectory chosen in a group of other ships could steer at full speed in the anticipation of the movements of all other ships. However, even a simpler prediction of the situation could become soon difficult for a human navigator while an autonomous system could always perform on an equally high level. A *safe speed* may be therefore different for a traditional vessel and an autonomous vessel.

Rule 7 - Risk of Collision

...

(b) *Proper use shall be made of radar equipment if fitted and operational, including long-range scanning to obtain early warning of risk of collision and radar plotting or equivalent systematic observation of detected objects.*

...

This calls for navigation information and assistance systems which this approach can be a part of. Even though the long term goal is to investigate a procedure by which ships can be made fully autonomous, in the short term the system may work integrated in an ECDIS as an expert system. This use would work well within current legislation where the captain needs to be in command at all times, however, the approach would have to include the delay and uncertainty introduced by a human crew.

Rule 8 *Action to Avoid Collision*

(a) Any action to avoid collision shall be taken in accordance with the rules of this Part and, if the circumstances of the case admit, be positive, made in ample time and with due regard to the observance of good seamanship.

(b) Any alteration of course and/or speed to avoid collision shall, if the circumstances of the case admit, be large enough to be readily apparent to another vessel observing visually or by radar; a succession of small alterations of course and/or speed should be avoided.

(c) If there is sufficient sea room, alteration of course alone may be the most effective action to avoid a close-quarters situation provided that it is made in good time, is substantial and does not result in another close-quarters situation.

(d) Action taken to avoid collision with another vessel shall be such as to result in passing at a safe distance. The effectiveness of the action shall be carefully checked until the other vessel is finally past and clear.

(e) If necessary to avoid collision or allow more time to assess the situation, a vessel shall slacken her speed or take all way off by stopping or reversing her means of propulsion. A vessel which, by any of these rules, is required not to impede the passage or safe passage of another vessel shall, when required by the circumstances of the case, take early action to allow sufficient sea room for the safe passage of the other vessel. A vessel required not to impede the passage or safe pas- sage of another vessel is not relieved of this obligation if approaching the other vessel so as to involve risk of colli- sion and shall, when taking action, have full regard to the action which may be required by the rules of this part. ...

The observance of good seamanship is a very broad term when one tries to transform collision avoidance rules into machine readable constraints. Ample time can be defined as a function of several parameters as the distance in between ships and their speeds as well as other environmental factors. Rule 8 (b) imposes different constraints on possible trajectories of ships, depending on whether the ship is under command by an autonomous system or a human navigator. For an autonomous system working as proposed, every course alteration is readily apparent, because it is transmitted. Even a succession of small alterations would be known to all ships and might be the shortest and most favourable trajectory under the influence of changing currents and winds. Rule 8 (c) imposes the requirement on the system to change to course first and only reduce speed as a last resort. Rule 8 (d) demands to pass at a safe distance however this may be shorter in fully autonomous systems because there is no need to add a safe margin based on the uncertainty about the behaviour of another ship. Furthermore, in part (e) the further requirement is explained that there are vessels whose passage should not be impeded because they are limited in their manoeuvrability or by their draught. This establishes a hierarchy of vessel types on sea in addition to the other collision avoidance rules.

OTHER RULES Further Rules add specific requirements to any collision avoidance system which will not be discussed in detail here, but given as an overview, presented in table 1.1.

Rule	To be considered
10	Traffic Separation Schemes have specific regulations concerning use of the traffic lane and an inshore traffic zone.
12	Sailing vessels have to consider the direction of incoming wind on the own and other vessel
13	An *Overtaking Situation* is defined with the requirement of keeping clear of the overtaken vessel.
14	Definition of a *Head-On Situation*.
15	In a *Crossing Situation* the give-way vessel, which has the other at its starboard side, should not cross ahead of the stand-on vessel.
16 and 17	Give-way and Stand-On vessels are defined and are required to take early sufficient actions or proceed and refrain from changing course and speed.
18	Special rules for vessels not under command or *wing-in-ground-effect* vehicles
19	Rules for restricted visibility are defined. This rule demands of an autonomous system to exhibit a more cautious behaviour in mixed environments with human navigators in restricted visibility even though the positions and courses of other autonomous vessels may be known.

Table 1.1: Further implications of collision avoidance rules

In conclusion, the COLREGs are the framework which is used in the training of maritime personnel since decades and thus it must be the foundation for any collision avoidance system. This has the advantage that, even though human errors exist and other customs may have unlawfully evolved, previously unknown intentions can be predicted in the context of the COLREGs and they can be used as the basis for an autonomous behaviour (Perera and Soares, 2010). It also means that own intentions can be conveyed to other ships without explicit communication if a behaviour is chosen which conforms to the regulations. They are also the basis for post-disaster investigation which try to make sense of the behaviour of colliding ships and help design traffic separation schemes and passages around built maritime structures. The most applicable COLREGs will be used in chapter 2 to derive requirements for the implementation and the trajectory negotiation designed and evaluated in the remained of this thesis.

1.6 THE MARITIME COLLISION AVOIDANCE PROBLEM

Any procedure which tries to find trajectories for ships has to solve a distributed problem with locally different information which need to be synchronised in a converging way to a safe solution. Ships on the open sea are isolated with sparse means of communication and local authority over their course which makes any central approach difficult. Information can not reach a central planner immediately and in addition, through the distributed and diverse sensors on each ship the perception of the state of the world can differ. Planning a number of trajectories which all ships may follow to avoid a collision decentralised on each ship must yet converge to the same on every ship, or there is no certainty that it is collision free.

A first informal definition of the problem in this work starts with the assumption that no ship intends to deliberately collide with any other ship or solid landmasses, thus excluding cases of wilful interference with maritime procedures. The short term path of a ship, which is intended for a specified duration in time and space, is in the following always called the *Trajectory*. The course of a trajectory is influenced and limited by many different constraints which will be, as all other concepts, formally defined chapter 4.

The *quality* of a trajectory is considered the value on a defined performance measure from the perspective of a single ship or an outside observer. Defining a performance measure is apart from the simple fact that no collision should yield a high quality not straightforward, as manoeuvrabilities, collision avoidance rules and varying resource demands have to be taken into account.

The Problem faced is having to find a set of trajectories for a number of ships with the highest possible quality for every single ship **and** the group as a whole.

A *set of trajectories*, where one trajectory is assigned to each ship, will be referred to in the following as a *solution*. The search for a solution could be done in complete isolation but, since the common interest is established, a way of cooperative problem solving is suggested, developed and evaluated in this work.

While planning a collision free trajectory on the own ship, it is necessary to know the intended trajectories of all the other ships which operate within that area, in the following referred to as *target ships*. To anticipate their intention locally requires more than the long term destination information, which is often available. Factors which influence a trajectory of target ships include but are not limited to their physical model, the capabilities of the crews, currents, winds and the perception of the area from the respective viewpoint. All these information would have to be known locally, or at the point of a central planner, to determine the course of each trajectory up to a satisfying degree. Furthermore, differing information about the geometry of the surroundings and an imperfect perception of other ships, which i.e. could be concealed behind a spit of land, may also determine the possible and desired trajectory of a ship. For the time being communication is considered sparse and the complete exchange of those information, too costly.

In order to overcome this information shortage, preliminary trajectories, planned under local currently available information, can be exchanged which convey a lot more information about a ship than may seem obvious. A chosen and disclosed desired trajectory of a ship contains information about the short term destination, some information about the limits of its manoeuvrability, further implicit information about the preferences of the crew, whether the captain would like to cross port or starboard and finally trivial information as the factual statement that this ship is capable of communicating trajectories, which gives information about the status of the ship's systems.

1.7 Structure of the Thesis

After the description of the domain in this first chapter, where the actors, information sources and the problem of collision avoidance is motivated, the most relevant maritime collision avoidance rules are listed. In the second chapter requirements from those rules and other sources are derived, categorised and prioritised. In the third chapter the work is set in the background of general and maritime collision avoidance after which the problem is formalised in chapter four where the suggested solution is developed. Needed basic concepts as the understanding of a trajectory and the inclusion an external path planner are explained. The chapter closes with the algorithmic description of the solution and a structured development of the quality metrics used to evaluate the approach and define how fulfilment of the requirements can be traced. The design and implementation of the software system is explained in chapter five. In chapter six a thorough evaluation of the performance of the software system in a simulation is shown in different significant settings and the main results are summarised. Chapter seven interprets the findings, draws conclusions for the quality of the procedure and elaborates on the impact on the maritime domain a widespread use of the system would have. In the end topics which could not be pursued in this thesis are described.

The contribution of this work is to design, develop, implement and evaluate a collision avoidance system for the maritime domain. Key features include the guaranteed finding of collision free solutions if they exist, within a predictable worst-case time. Local trajectories will be optimal regarding the current state of the world while the whole procedure converges towards a near-optimal solution in the experimental simulations. This extends work of Waslander (2007) and Blaich et al. (2012) to the maritime domain with n-ships.

2

Requirements of a Collision Avoidance System

The previously considered collision avoidance regulations and demands on a final set of trajectories as fairness and optimal resource allocation are used to define the following requirements. Use cases are presented to identify the different actors which use the system in different ways. In the beginning relevant COLREGs, as selected in section 1.5 motivate normative requirements which determine characteristics of the process of the trajectory search as well as the solution. Further requirements stem from safety demands as well as limitations in the domain, i.e. the need to regard the manoeuvrability of each ship. The last area of requirements about the algorithmic implementation considers not only the possible integration of the developed software in a ship's system but also the use as a general tool to aid research in maritime trajectory optimisation to investigate different cost functions and alternative procedures.

Some of the requirements stem from more than one of the category.

2.1 Requirement Derivation

Three different use-cases exist which can may occur in isolation or all at the same time, depicted in combination in figure 2.1:

Expert System A navigator might use the developed system to generate a safe trajectory to follow in the presence of other navigators who do the same, leading to a safe solution. The navigators which receive a collision warning could query the system for a collision free trajectory and wait for a solution for a predefined time. After a negotiation procedure with all other ships which are equipped with the system and using the available means of communication the system would respond with a coordinated trajectory for the own ship or signal failure to find one in the worst case. The process can be observed and interpreted by a Vessel Traffic Service Operator if the situation takes place in a VTS area. This use case respects the local authority of the crew over its vessel as it only presents solutions which are coordinated with all other systems on other ships, but leaves the decision to ultimately follow the suggestions to the navigator. A realisation would be possible without major changes in the legislation.

Autonomous Interaction The very same scenario can contain autonomous ships which communicate and negotiate their trajectories using the system in cooperation with a human navigator, or use the system as the sole mean for trajectory optimisation when only autonomous ships meet in a critical situation. In the future, if the various difficulties during their development can be overcome, situations may arise where autonomous systems

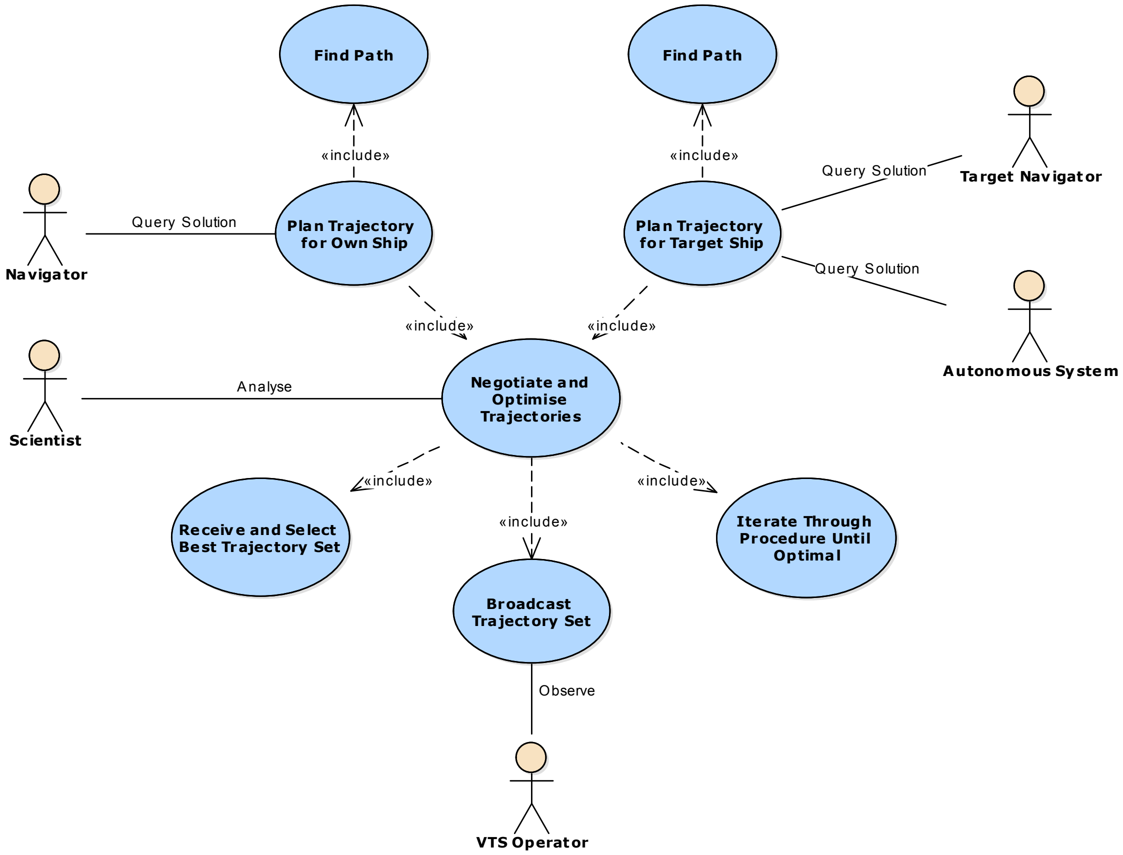


Figure 2.1: Expert System, Autonomous System or Research Tool

and human navigators meet in mixed proportions. Requirements arise therefore to reach a solution which is not only optimal and feasible for autonomous systems but can also be understood and interpreted by human navigators.

Negotiation Research A number of cost-functions, algorithmic details and results of different scenarios are still under investigation. The introduction of passive, uncooperative ships to a collision situation or the use of different path finding methods are of interest for a scientist who works with the system. The algorithmic implementation must produce traceable results and provide easy accessible interfaces and modular classes to be able to change parts of the approach and evaluate the results. The use of standardised and established interfaces support use of the system in combination with a simulation environment and a live test-bed.

2.2 NORMATIVE REQUIREMENTS

The COLREGs discussed in section 1.5 pose requirements which must be fulfilled by either the problem solving process, which is the process which leads to a solution or by the solution itself.

Rule 6 requires to proceed at all times with a safe speed, which means the chosen speed on the trajectories may not exceed the customary safe speed, especially in bad weather conditions:

R_1 (**Safe Speed**) The ship on a trajectory travels with a safe speed, which means it could be stopped within an appropriate distance to other ships under the prevailing circumstances and conditions and fast enough to avoid a collision.

Rule 8 (a) leads to various requirements as collision avoidance in ample time while using course alteration only and keeping an appropriate distance with only apparent course changes:

R_2 (**Ample Time**)
A solution must be found fast enough to avoid a collision.

R_3 (**Action to Avoid Collision**)
The rules 8 and its sub-rules for collision avoidance as stated by the International Maritime Organisation should be obeyed. This is already ensure by some of the other requirements as R_2. Furthermore, the following requirements can be derived from the rule:

R_4 (**Apparent Course** 8(c)) A trajectory should avoid small angles between adjacent edges of a waypoint to make course corrections apparent and lasting for a minimum amount of time or distance covered.

R_5 (**Course Alteration Only** 8(b)) The speed of ships on trajectories should not need to change in order to avoid a collision.

R_6 (**Appropriate Distance** 8(d)) The distance between any two ships on trajectories should at all times be larger than a minimum, safe value.

Other collision avoidance rules are not used to formulate further requirements because in the investigated situations the *Stand-On* and a *Give-Way* vessels are not assigned and both ships try to achieve a better solution. Any ambiguity in the behaviour of ships is avoided through the explicit exchange of their intention through the developed system. However, the *Crossing* and *Head-On* situations of the COLREGs are the basis for the evaluated experiments.

2.3 TRAJECTORY SEARCH AND SOLUTION

Requirements of the trajectory optimisation process are those which concern the finding, exchanging, evaluating and negotiating of trajectories among a number of ships. Collision detection is for the sake of completeness listed even though it is not part, but a prerequisite of this work. Requirements of the solution are those which concern the final outcome of the process, the final set of trajectories for every single ship.

An agreed upon set of trajectories has to be optimal in the sense that it uses the i.e. the available space in an optimal way and allows to traverse in the least time necessary. At the same time the set should also be fair in the sense that the quality of all trajectories should be as equal as possible. The quality entails not only the length of trajectories but also factors as the number of needed turns, because they can incur costs for ships.

R_7 (**Collision-Free Confidence**)
It is possible to judge if a generated solution contains a collision or not.

R_8 (**No collision**)
The final solution should have no remaining collisions.

These are elementary requirements to be able to compare trajectory sets and they motivate the use of a planning approach which does not formulate rules for reactive behaviour but complete time-dependent trajectories in advance. Furthermore, ships have to be able to communicate their intention to other ships:

R_9 (**Common Intention Knowledge**)
Ships should be able to base their behaviour based on the known behaviour of other ships.

Even though safe trajectories for two ships alone would have helped to avoid collisions and groundings in the past the system should be easily capable of finding trajectories for three and more ships. This leads to the requirement of being able to include an arbitrary number of ships in the process, even though the speed has to be evaluated and is expected to degrade significantly with more ships.

R_10 (**N-Ship**)
The solution should be found for at least two ships, and be expandable to an arbitrary number of ships.

Ships which can or will not use the proposed system, but are detected to cross the area, should be avoided.

R_11 (**Passive Ships**)
Ships which do not use the system and traverse the area should be considered and collisions should be avoided.

Exactly one final set of trajectories should be found which means that there is an agreement to use only one set which can not be misunderstood. This one solution set should be found through the process within a certain time and with a known confidence.

R_12 (**Solution Convergence**)
Solutions should converge towards a final solution.

The physical model of the own ship should be used as basis for trajectory generation to be able to find trajectories which can be pursued. This might seem a trivial requirement however, the exact physical model of every ship in a n-ship situation is not expected to be known on every ship because this would demand prior knowledge, an additional model recognition or explicit communication. These are all valid fields of research which are however not part of this work. The physical model used and the received trajectories from other ships are the only basis for the generated trajectories of the own ship and there is no need in the developed system to know the model of any other ship.

R_13 (**Feasibility**)
Trajectories should contain no turns or speeds which the designated ship is not able to pursue due to its manoeuvrability, within a satisfactory margin of error.

The final set of trajectories, one for each ship, should be near an optimal solution, with a maximum distance of ϵ. The distance defined refers to the costs, assigned to sets of trajectories. The optimal solution is a solution which is the most efficient and fair and has therefore the lowest cost according to the defined performance measure.

R_14 (**Highest possible quality**)
The final solution should be within an ϵ-environment near the optimal solution.

For the performance measure, a balance has to be found in between demanding the minimum overall resources and a solution which is equally demanding for every ship, thus considered fair. Even though the next requirement can not be fully achieved the process optimises towards fulfilment of the requirement.

R_15 (**Fairness of Solution**)
All trajectories in a solution should have the same quality.

Furthermore, deviating from an agreed trajectory and thereby endangering other ships with an unforeseen behaviour should be discouraged by finding a *Nash-Bargaining-Solution*, which will be explained in detail later on.

R_16 (**No regrets**)
The agreed upon solution should have the best trajectory for each ship, given that no other ship changes its trajectory. A single ship should not be able to deviate from a solution and achieve a trajectory with a higher quality.

2.4 ALGORITHMIC IMPLEMENTATION

In the critical maritime domain the algorithms have to have certain qualities to ensure their safe use. However, this can only be fulfilled partially because the thorough proof of the algorithmic implementation is beyond the scope of this work. Some requirements stem from the demand to build a usable, reusable, reliable and expandable software.

The medium for the communication is considered limited because every current technology on the high sea as VHF or satellite link suffer from a small bandwidth. Nevertheless, the system is developed having a situation on the high sea in mind where no central authority can or may interfere and ships have to communicate with the limited bandwidth given.

R_17 (**Minimal Communication**)
Information exchange is for all practical purposes limited to a realistic communication between ships. Ships should use as little communication as possible.

R_18 (**Decentralised Method**)
There is no all knowing central instance which receives all information about all ships instantly and there is always a bandwidth limitation and delay using the communication medium.

This includes delays which can stem from the computation time of the procedure, the reaction time of a maritime crew or the delay caused by the ship's manoeuvrability. In the first case the procedure must consider that a solution, planned at a specific point in time must include the delay that planning involves and the subsequent change in position of the planning ship. Reaction time of a maritime crew becomes important when the procedure is used as part of an expert system and stems from human reaction time and inevitable delays caused by a hierarchy of command. After those delays ships may also have an additional delay in between entering commands and the change of the ships behaviour.

R_19 (**Implementation Delay**)
The delay between the end of the planning process and the first possible implementation of an agreed solution has to be considered.

The algorithm in its implementation can not be proven for the used language yet certain guarantees can be given about the developed procedure, abstracted from the concrete implementation. This leads to a safety critical requirement which defines decidability as the ability to judge if the algorithm can theoretically come to a conclusion in worst case scenarios.

R_20 (**Decidable**)
The algorithm should provide information about its termination. It should be possible to asses whether the algorithm terminates or if it is incapable of solving a problem fast enough or in general.

Finally for the use case of the development and evaluation it is important that the course of all variables and functions is traceable.

R_21 (**Traceability**)
All results of the process as the trajectories, the values of the utility functions, the positions of all ships and obstacles have to be vigorously traceable.

No.	Requirement	Category	Functional or Non-Functional
		Essential	
R_2	Ample Time	Trajectory Search	F
R_1	Safe Speed	Trajectory Solution	F
R_3	Action to Avoid Collision	Trajectory Solution	F
R_4	Apparent Course	Trajectory Solution	F
R_5	Course Alteration Only	Trajectory Solution	F
R_6	Appropriate Distance	Trajectory Solution	F
R_7	No Collision	Trajectory Solution	F
R_9	Common Intention Knowledge	Trajectory Search	F
R_18	Decentralised Method	Trajectory Search	F
		Recommended	
R_7	Collision-Free Confidence	Trajectory Search	F
R_10	N-Ship	Trajectory Search & Trajectory Solution	F
R_11	Passive Ships	Trajectory Search & Trajectory Solution	F
R_12	Solution Convergence	Trajectory Search	F
R_13	Feasibility	Trajectory Solution	F
R_17	Minimal Communication	Trajectory Search & Trajectory Solution	NF
R_19	Implementation Delay	Trajectory Search & Algorithmic Implementation	F
		Beneficial	
R_14	Highest possible quality	Trajectory Solution	NF
R_15	Fairness of Solution	Trajectory Solution	F
R_16	No regrets	Trajectory Solution	F
R_20	Decidable	Trajectory Search & Algorithmic Implementation	F

Table 2.1: Requirements Prioritised by Importance

In the domain a number of use cases can already be found which can be mapped to the three generic use cases presented in this chapter. A holistic system which fulfils all the identified requirements is up to the present day unknown though in the domain several approaches exist to solve the problems in isolated applications.

3

Related Collision Avoidance Procedures

Challenges in developing Collision Avoidance Systems for ships were investigated for over three decades. Since information about other ships in an area are often limited due to several factors i.e. bad weather, local prediction of other behaviour is an important prerequisite for collision avoidance. In the past a number of systems were investigated under the assumption of the existence of a central planner, which receives all information instantly while having the authority and means to control each ship in a situation. This leads to a simplified problem which enables statements about the optimality and convergence of planning approaches, but might not be very realistic in the near future without a better connected maritime environment and changed legislation. The introduction of autonomous systems to the domain need reliable collision avoidance, which also conforms to the COLREGs, as an enabling technology. Similar approaches in different domains can help address the various problems faced.

3.1 Early Collision Avoidance Systems

A complete collision avoidance system was implemented and evaluated in depth on the Shioji Maru as early as in 1991, (Iijima et al., 1991). Their rule-based system is aimed towards being used as an autonomous system as well as part of an expert system, integrated in an ECDIS. Collision free straight trajectories can be reached within 3-4 seconds with 2 ships and 10 seconds with 10 ships. Replanning of the trajectory is done at every waypoint where an angle of a straight line is chosen. The resulting path consists therefore of non-continuous lineare edges between waypoints. Other ships are automatically detected by an ARPA. A common intention knowledge as in requirement R_9 is not desired as the system relies on ARPA detection only, which excludes R_18. Exchange of routes or trajectories is not considered. In the examples the manoeuvres are stated to be similar to the manoeuvre of a good officer, hinting but not explicitly stating a COLREG conformity especially of rule 8, as demanded by requirement R_3. Execution and planning times are stated to be sufficient for the task, though no proof is given. Likewise, game-theoretical and fairness considerations are not made and the system defines more a reactive behaviour which can be predicted in the future to receive paths than an off-line planning system.

Later approaches try to incorporate COLREGs as by Lee and Kim (2004), which made a heuristic search to derive a safe trajectory. In their work, Fuzzy Relational Products are used to encode knowledge based rules by domain experts. When an obstacle is detected, sectors around the ship are checked for their collision risk to find candidate solutions for free sectors. The system works also in isolation and no communication is carried out.

3.2 NEGOTIATION OF TRAJECTORIES

In the maritime domain there are very few approaches of using automated negotiation of full, non-trivial trajectory sets for collision avoidance. For that reason the method used is based on approaches from other domains as Waslander (2007) which finds the Nash Bargaining Solution of a decentralised multi-agent coordination problem, shown to be effective in Air Traffic Control and a Multi-Vehicle system. A common goal is to find decentralised Pareto-Optimal solutions for a number of decision makers, as shown by Heiskanen (1999).

Qinyou et al. (2006) put forward the notion of negotiation for the sake of maritime collision avoidance. In their work they develop a negotiation framework which they improve later by considering planned routes of other vessels in Qinyou et al. (2008). Their work considers however only two vessels. It can be seen as a mathematical framework which is capable of calculating the collision risk between two vessels and produce an evading angle as the output. They formalise several behaviours on sea, as the implementation of collision avoidance rules or benevolence, each as factor of the formula. In their negotiation they exchange a number of ship information and the mathematical parameters of their model as well as their collision-avoidance plans.

3.3 EVOLUTIONARY OPTIMISATION

The research on more complex approaches for trajectory optimisation began when computing power became sufficient enough to use procedures based on a huge number of iterative improvements which also include moving target ships. The research of evolutionary methods in the maritime domain is among others marked by the work of Roman Smierzchalski and Zbigniew Michalewicz in Smierzchalski and Michalewicz (1998) and Smierzchalski (1999). They developed a collision avoidance procedure which included other moving ships, which are seen as obstacles and are not active, cooperative parts of the same system. Those moving ships, which are included as polygons, are named *Passive Vessel* in the following. Their work is based on an evolutionary planner from Xiao et al. (1997) to plan a trajectory in a dynamic environment. In a simulation of a two-ship crossing situation the search for a solution converges within 3 seconds while it converges within 28 seconds in a more complex scenario where the own ship needs to avoid landmasses in a slightly confined space with three target ships. An interesting feature is that the shape of a polygon around target ships, which is considered as a safety area, can be defined by the operator of the system and therefore fitted to each situation.

This approach was developed further among others by Roman Smierzchalski himself, Rafal Szlapczynski and Joanna Szlapczynska in various works, which fulfils several requirements:

Requirement	Fullfillment
R_2 (Ample Time)	The original process is stated to be finished in less time than one minute. Depending on the speeds of all ships this may not be fast enough however in Szlapczynski (2011) the author reports of an near optimal solution which is found after 10 to 30 seconds.
R_3 (Action to Avoid Collision)	The original approach provided a ”...simplified supporting of international collision avoidance rules ...” which was improved in later work as described in Szlapczynski and Szlapczynska (2012). In their work they respect rule 8 of the COLREGs by using specialised operators in their optimisation by genetic algorithms (GA).
R_9 (Common Intention Knowledge)	The intention is not clear however course and speed are assumed constant in Smierzchalski (1999) and those information are controlled via ARPA constantly.
R_10 (N-Ship)	Planning is done for all ships in an encounter in Szlapczynski (2011), and sets of trajectories instead of single trajectories are optimised. One of the most important things to note is that sets of trajectories are created as a whole and then genetic operators are applied which will often change more than one trajectory at each generation. In the procedure developed in this work planning will be done in sets with fixed other trajectories and only the own trajectory is changed which preserves collision free sets, once found, as will be described in chapter 4
R_11 (Passive Ships)	The earlier work of Smierzchalski and Michalewicz (2000) tries to find a single trajectory in the presence of *Target Vessels* which are in the authors understanding passive vessel, however, in the more recent work where they optimise sets of trajectories those are intended for a VTS center which would then be in the position of coordinating all ships Szlapczynski (2012).
R_13 (Feasibility)	Special operators ensure feasible trajectories by adding or moving *nodes* in their trajectories, if impossible angles occur.

R_14 (Highest Possible Quality)	In Szlapczynski (2011) turns are implicitly minimised by means of the operators of the GA process. Furthermore, way-loss is considered the standard of comparison to evaluate the quality of a trajectory.
R_18 (Decentralised)	The largest downfall of the presented approaches is the centralised manner in which trajectory sets are optimised so R_18 can not be fulfilled.
R_19 (Implementation Delay)	A 6-minute ”navigational decision time” is assumed in Szlapczynski and Szlapczynska (2012)
R_20 (Decidable)	In Szlapczynski and Szlapczynska (2012) deterministic approaches are abandoned to improve computational time. However, some of the procedures have a complexity of $\mathcal{O}(N)$ where N is the number of cells on the raster-grid of the map. This would make a worst-case prediction possible even though it is not stated.

Especially the latter work of Szlapczynski and Szlapczynska puts forward the notion of searching for a set of trajectories as elements of the solution space instead of single trajectories. The approach developed in chapter 4.3.1 uses a path planner to search for individual trajectories, however the path negotiation and optimisation algorithm uses sets of trajectories, with the same the negative and positive implication for the computation time and size of the search space. Thomas Porathe showed in Porathe et al. (2012b) and Porathe (2012) that it is already feasible and advantageous to exchange planned routes of ships for collision avoidance purposes even though the technical realisation using the AIS system for data transmission may not suffice to reach a desired quality of service in the future. Therefore, the transmission of full sets of trajectories may demand a bandwidth which can not be achieved using a system as AIS alone.

3.4 Traffic Alert and Collision Avoidance System (TCAS)

Autonomous navigation could be realised fully, i.e. on an *Unmanned Surface Vehicle*, or partially on a ship with a maritime collision avoidance system, which could take over or give crucial advice in case of absolute emergency, similar to the Traffic Alert and Collision Avoidance System (TCAS) in the aviation domain (International Civil Aviation Organization, 2006). The system helps to avoid mid-air collisions by warning the pilots of oncoming other aircrafts, coordinate automatically an advised action and presents it to the crew. Similar to the requirements of AIS it is required on all aircrafts over a certain size (European Commission, 2011). When approaching, the system keeps track of all *Intruder Aircrafts* in a range around the own aircraft and extrapolates their probable trajectory. This is presented to the pilot along with a *Resolution Advisory Display* if a collision threat is identified. The resolution is coordinated through a data link with the approaching aircraft, if it also uses a Traffic Alert and Collision Avoidance System (TCAS) system. A great advantage of the avionics domain is the possibility of performing a vertical evasion manoeuvre. For a resolution advise the TCAS system calculates the necessary speed of a descend or climb manoeuvre to achieve a desired minimum vertical separation at the closest point of approach. Since there are other factors which might affect a traffic encounter, TCAS operates only as advisory system, and leave the final responsibility to the pilot (Kuchar and Drumm, 2007). The TCAS system fulfils already most of the necessary requirements and is currently in use. The requirement R_10 of planning for n-ships can be translated to be able to plan for n-aircrafts. The TCAS system has the capability of resolving situations with three or more aircrafts though they are rare. However, simulations indicate that the system might increase the number of near mid-air collisions of the own aircraft with a third aircraft while it is evading a second (Espindle et al., 2009).

3.5 UNMANNED SURFACE VEHICLES

Automatic collision avoidance is one of the enabling technologies for the widespread use of autonomous ships. A categorisation was suggested by Statheros et al. (2008) in three fields: *Mathematical algorithms* which compute evasive trajectories as a function of incoming information, *Soft computing* methods, which represent a branch where, using machine learning or rule design, an inference engine is developed which maps incoming information to outputs and finally *Hybrid Autonomous Navigation Systems*, which are seen as an optimal combination. Lacking a common system and standardisation, current autonomous systems work mainly isolated without communication or cooperation. However, using the COLREGs as a source for deriving evasive manoeuvres enables autonomous systems to handle a situation in a predefined manner without explicit communication (Perera and Soares, 2010). A radar based obstacle detection and avoidance scheme was already successfully tested on a Unmanned Surface Vehicle (USV) by Almeida et al. (2009) where limitations imposed by automatic obstacle detection on a radar became apparent. A catamaran USV developed by Naeem et al. (2012) was used in a simulation with a defined dynamic model to produce trajectories which are not only COLREG compliant but also feasibly, in the sense that they can be followed closely by an autopilot. Their system generates waypoints which should be followed based on a line of sight approach and obstacles in the path area evaded in a COLREG conform way. Evasion is among other methods investigated by use of an A^* algorithm, which is similar to the included path planner, used in this work. The approach is tested in a live-testbed, however, no form of cooperation with other vessels is investigated. In general, Campbell et al. (2012) found the capabilities in situation assessment and planning to be the limiting factors which keep maritime autonomous vehicles from the next step in deployment: the regard of unmanned systems in international legislation. He found that inclusion of the vessel's dynamics, COLREG compliance and the ability to handle unforeseen situations are the most important shortcomings of current collision avoidance systems on USVs.

3.6 Predicting Other Behaviour

Prediction in the context of collision avoidance has several applications. Reliable prediction of the behaviour of other ships is necessary to determine if a collision or a close-quarter situation is expected. During trajectory planning the prediction of the manoeuvrability of other ships could be used to anticipate the reactions of other ships to a locally planned solution or even plan trajectories for other ships as well. In any case, the behaviour of ships which do not take part in a coordinated solution of a situation renders them as dynamic obstacles, which have to be predicted. There are several possible ways based on technology as ARPA, AIS, manual entries by seafaring personal or even external information from a maritime cloud or a VTS to include other trajectories in the process (Weintrit, 2014). From information as the class, length, past trajectories, heading, speed, turning rate and the current environment it would be possible to infer a behaviour by modelling the dynamical model of the ship and predict its trajectory within some margin of error. A number of approaches try to use machine learning approaches to observe trajectories or similar data from the various sources to build a statistical model which predicts future behaviour. Vasquez et al. (2009) develop a growing hidden markov model to learn the parameters and the structure of a model of moving entities at the same time, while adapting to changes in the behaviour. Over a small time horizon they are capable of predicting the intentions of cars in a parking lot, which could be adopted for the maritime domain as well. Mazzarella et al. (2015) extracted traffic patterns from one month of AIS data, which was collected from ships moving in between Gibraltar to Dover and limited to those. They inferred traffic patterns using Constrained Bayesian non-linear filtering to predict the position of ships. Another source for prediction can be done under the assumption that ships generally try to follow the COLREGs and therefore, a certain behaviour can be expected. Kuwata et al. (2014) extend the approach of evading moving obstacles via modelling *velocity obstacles* and evading by obeying the required procedures in Crossing, Overtaking and Head-On situations. Based on the COLREGS, constraints are added to the velocity space. Formalising COLREGs to be used is however not straightforward, and several approaches were made to include the rules with certain ambiguity into a machine readable format (Breitsprecher, 2012). Other ways to predict the probable behaviour of a ship is to first narrow down the possible behaviour. Baldauf et al. (2015) investigated procedures from aviation to develop the ship's *manoeuvring area*, which is the area in which a ship is expected to be, depending on its possible turn rate and speed after a certain time. It could be also used in risk assessment to determine the collision risk before a collision avoidance procedure starts.

4

Negotiation Towards a Fair Solution

In this chapter the problem of finding safe trajectories for maritime collision avoidance is modelled as a cooperative negotiation problem where individual ships do not need to know each others value-function, the rating a ship uses to determine its benefit on an intended trajectory. On sea this is expected to be the default case when different ships with unknown capabilities and intentions meet. All the parts of a trajectory negotiation system are motivated, developed and explained in the following while certain safety properties are shown or required by external components. In the end limits and simplifications are described before a structured approach to evaluate the procedure is applied.

In the beginning, an overview over the whole approach and the sequence of steps in the implementation is given before from section 4.2 on all components are explained in greater detail.

4.1 Steps Towards Optimal Trajectories

The suggested solution to avoid a collision in between n-ships consists of four steps, local path finding, local evaluation and a global negotiation among all ships designed to lead to a common set of near-optimal trajectories, the final agreement. To evaluate the quality and observe if the stated requirements are met, a modular system was implemented as a Java-agent with an interface to an external path planner and a communication medium. Multiple agents were instantiated in a simulation to evaluate the implemented system, which is described in the next chapters.

Under the assumption that a collision is predicted in the near future, all information as position, heading and speed in the current situation are gathered in a first step, seen in figure 4.1. As a statement of intentions, local desired trajectories are exchanged in the second step.

The initial exchange can help to verify that a collision, which could be anticipated earlier by a different system as for example ARPA, will indeed happen if no measures are taken. Using the known trajectories from the first step, an external path planner is queried to find a

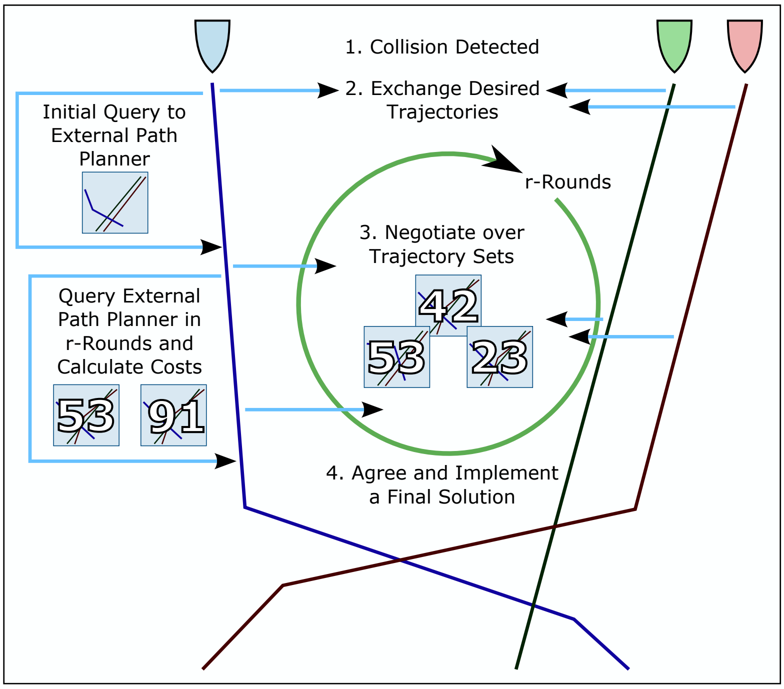


Figure 4.1: Steps of Trajectory Negotiation

path with a number of properties, which include among others:

- Paths are optimal, given the currently known trajectories of other agents.
- The search finishes in a predictable worst case runtime.
- Paths conform to the COLREGs and optimisation is achieved with a domain-specific performance measure regarding the manoeuvrability and other factors as the reduction of needed turns.

After local paths are found, trajectories are generated on their basis, which are simplified approximations. Their course in the set of other trajectories is evaluated according to two performance measures: Local path cost according to the path planner of the single own trajectory and global cost by evaluating the course of the trajectory in the context of the set, in particular measuring the minimal distances kept to other trajectories.

In the third step, shown in figure 4.1, all ships negotiate in several rounds, until no further improvement is detected: A procedure, based on Waslander (2007) was developed, which performs repeated broadcasting, re-evaluating and re-planning of trajectory sets, to be used as bids in the negotiation until the local and global costs of a set of trajectories converges for all ships. Ships hold in this approach the chosen trajectories of other ships fixed and try to plan a new trajectory for the own ship in each set received. When the value of the cost function of consecutive solutions stay within a certain ϵ-distance of each other, a final trajectory set which is agreed on in the last step is found.

Some alterations to the original approach were made to add safety-critical properties at the expense of additional computation time: After a predictable runtime the procedure either fails or finds a solution and after a first collision free solution, which is found very fast, no further collisions in consequent negotiation rounds are present.

The approach is modelled as a multi-agent optimisation problem, where one agent represents exactly one ship so the term will in the following be used synonymously. A software prototype consisting of negotiating agents was developed, quality metrics to judge the procedure were derived and experiments planned, simulated and evaluated to measure the performance.

In the remainder of this chapter, a short background of game theory is given to explain the design of the cost functions in its context. Path finding is defined in contrast to trajectory generation, exchange and negotiation. It will be described how a fair solution can be reached, how fair is to be understood in this context and how the process optimises towards a *Nash Bargaining Solution* which discourages deviation from a final agreement. In order to judge the feasibility of the overall process, convergence properties and requirements are discussed and limitations are stated. The algorithmic implementation is explained in section 4.6 with some special cases before in the next chapter 5 the whole framework is introduced. Finally, it is described how the quality of the approach can be measured using a structured approach to generate measurable performance indicators which will in chapter 6 be used to control the fulfilment of the requirements. This approach extends the work of Waslander (2007) to the maritime domain and uses a dedicated path planner from Blaich et al. (2012) for feasible path-generation to be able to include complex constraints as i.e. fulfilment of the COLREGs. The algorithmic procedure is kept as closely to Waslanders approach as possible to be able to compare the results. However, the *ship vehicle cost* functions are based on the external cost-function from the path planner and a different distance-relation for the *constraint penalty function* is used.

4.2 THE GAME OF COLLISION AVOIDANCE

The origin of game theory stems from the field of economic and decision theory but current concepts are heavily applied to classical problems in the domain of artificial intelligence as well. First investigations were done by John von Neumann in his work *Theory of Games and Economic Behaviour* in 1944 (von Neumann et al., 1944). Over the past 70 years it gained importance in economics, law and computer science, applied to analyse and structure the search for a solution in a huge variety of problems concerning more than one decision maker. A problem in game theory is modelled in terms of participating *players* which perform *actions* based on *information*. They gain a certain *utility* value from the *outcome* of the game. A part of game theory investigates especially negotiations, with *cooperative* or *uncooperative* behaviour, *offers*, varying *strategies* in several *rounds* of a game.

There are many different ways in which collision avoidance can be modelled as a game. In the following a collision free set of trajectories, one for each ship in a situation, is seen as the outcome of a round of bidding in a negotiation. Each trajectory set is associated with costs for each player, in this case each single ship. The costs are calculated based on penalty functions, which will be defined in the following.

4.3 PATH PLANNING AND TRAJECTORY GENERATION

The concept of *path planning* for the individual ship is separated from *trajectory generation and negotiation* and *optimisation* for a group of ships. To a *path* is referred to in the following in context of the external component, which is used to find an exact, collision free and feasible time-dependent function of the position of a single ship in a known static environment, presented to the path planner. Its generation can be influenced but is ultimately out of control of the negotiation component. A *trajectory* is the representation of a path, reduced in complexity to improve communication and computation. It is defined in section 4.3.7 by its waypoints and a constant speed and is negotiated with all other ships in an area.

4.3.1 THE PATH PLANNER

Finding a path *offline*, in a fixed and known environment, or *online*, where reactive behaviour leads to a reasonable but not always predictable solution, is very well understood and investigated in the domain of artificial intelligence. From the perspective of the own ship it is possible, using *Path-Finding* methods, to plan a collision free trajectory very fast when all needed information are available. Path planning is used in many different domains where approaches, which fulfil certain requirements, can guarantee to find a solution if one exist, find the best solution if it exists and to a certain degree can even be limited in terms of time or calculation steps needed to find a solution. These properties are important on a ship because mariners must know as early as possible if an assistance system or an autopilot is going to produce a feasible solution or if alternative measures to avoid a collision become necessary.

The creation of a path planner is out of the scope of this work and its implementation is considered to be replaceable. However, due to a cooperation with the Hochschule Konstanz Technik Wirtschaft und Gestaltung (HTWG) in Konstanz, Germany, this work was created relying on the path planner described in Blaich et al. (2012) where detailed information about it can be found. In general the path planner is included as a module and all information reaching the path planner are locally available or part of the negotiation process. These information include a kinematic model of the own ship and a locally implemented mechanism to avoiding violation of the better part of the COLREGs.

Feasible paths which regard the manoeuvrability of a ship are generated based on the kinematic model to structure the search in a computationally favourable way. Every ship has a very distinct manoeuvrability which changes with factors as the load and age of a ship and even with pollution of the ship's body from mussels. External factors as currents, winds and in a sense the capabilities of the crew determine the way in which speed, position and heading can be changed. The succession of possible states of the world, which a ship can occupy are limited by its kinematic model and hence a path planning algorithm is able to work in a limited *configuration space* to produce feasible results, a term used in robot motion planning to encode the states and possible transformations which can be applied to a robot (Lavalle, 2006) and (Lozano-Pérez, 1983).

4.3.2 Fast Grid Based Collision Avoidance (Blaich et al., 2012)

The path planner by Michael Blaich is integrated to find collision free paths for single ships which are dynamically feasible and regard the COLREGs via implementation of a *ship's domain*. In his work he is able to search for the optimal path in a grid representation of a given environment with other given paths as obstacles.

A path in his work is defined by a number of waypoints and a constant speed to travel along while the segments in between waypoints are interpolated by parametric Bézier curves, to create trajectories. The grid representation of the *configuration space* stores additional information about the orientation and lateral speed of ships to support the design of an action space more fitting for ship motion planning. The grid representation is based upon an approach by Szlapczynski (2006) who tried not only to optimise according to the length of a path, which is common in path optimisation, but also according to the number of turns which slow navigating ships and are inconvenient for navigators.

Costs of directional changes are included in the path cost by the grid design which accounts for the direction in which a path is planned through adjacent cells. In the grid, he defines a *region of reachability* and a *T-neighbourhood* of cells, which a ship can traverse considering its possible speed and position changes. This reduces the number of cells a path planner has to include when searching for a path and at the same time it ensures dynamically feasible trajectories.

Through an interface based on the Inter-Module Communication Framework (IMC) by Martins et al. (2009), the path planner is queried from the negotiation component with trajectory sets as parameters. These trajectory sets contain the own path, which is used as a desired trajectory of the own ship, and other trajectories which are used as obstacles. If other paths are not given, but instead only positions, speeds and headings of other ships are known to the path planner, his approach is able to use a probabilistic model to infer the future probable position of ships with a slowly growing uncertainty. The prediction is based on *Constant Velocity* or *Constant Turn Rate and Velocity* models.

After the query a specialised A^*-algorithm searches based on a *cost-to-come* function and a heuristic, designed to suit the demands of ship motion planning. Other trajectories are included in the grid as occupied cells. The used cost-function includes a penalty for course changes, the distance to an obstacle, represented as an occupied cell, and deviation from the default surge speed. The heuristic used is the euclidean distance, which is not only admissible but also consistent. This ensures not only optimality but in its design produces also in a deterministic way the same solution under the same circumstances. In coordination with the negotiation component, this property is preserved in the negotiation in the first sequential round, which will become important in section 4.6.1 where it is described how a collision free initial solution is guaranteed.

With the *cost-to-come* function, the approach is able to plan paths which have a realistic and safe distance from obstacles as shore-lines and other trajectories, minimise needed turns and the distance travelled. The destination of the trajectory might not be used if the trajectory is longer than a defined sensor range of the own ship. In that case the trajectory is planned to the intersection of the sensor range with the desired trajectory.

The path planner reveals also information about its costs it assigns to found trajectories and these costs will be referred to in the following sections as the *ship vehicle cost*-function, a name to indicate the understanding in the context of the work of Waslander (2007).

4.3.3 Collision Avoidance Regulations

COLREG conformity is achieved using unevenly shaped ship domains, which are safe areas around each ship to be kept clear of other ships. Ship domains are used in collision avoidance planning since decades and different approaches emerged where circular domains with enlarged segments or an offset of the position of the ship from the center is used, to reach a desired behaviour (Goodwin, 1975) and (Davis et al., 1980).

Blaich et al. (2012) uses the concept of unevenly shaped ship domains to favour the generation of trajectories, compliant with rule 8 of the COLREGs and requirement R_3. Ship domains are used around the own and other vessel in the area, while planning a trajectory to achieve safe distances and favour paths which cross stern and port of each ship. Constant speed on all trajectories is used in coordination with the negotiation component, to ensure that changes in heading of a ship take precedence over changes in speed, satisfying requirement R_5. The approach from Michael Blaich is capable of finding trajectories with speed changes on segments of the trajectories, this is yet not used and not stated as a requirement for an arbitrary path planner.

In the final stages the found paths are simplified to waypoints using an algorithm based on Ferguson et al. (2005) and control points are calculated, to form a Bézier curve. It enables the smooth transition in between segments of the paths. Time parametrisation of the final trajectory is performed by linear interpolation of the times, stored in the waypoints.

The work of Michael Blaich enables the generation of collision free optimal paths in a predictable time in the sense that a maximum runtime is defined, after which the algorithm signals failure. This is very useful because it enables a worst case prediction if it is impossible to solve a collision situation on sea when the algorithm needs its predicted maximum time. Alternative measures may be taken immediately if after a possibly failed trajectory search there would be no time to avoid a collision. Safe use of the trajectory search can be guaranteed if there would be time for alternative measures after a failed trajectory search in a worst case scenario[*].

The inclusion of the path planner is designed to allow for other path planners to be used because other approaches, as described in chapter 3.3, may be based on a more detailed dynamical model of the ship or outperform the used approach computationally. Improved COLREG observance or paths, planned based on more information about the environment as shallow waters or traffic separation schemes, may be used in the future. In the current implementation, the path planner is queried in each round of a negotiation by the own ship

---

[*] Failed in this context refers to about 6% of the cases of the non-deterministic search in Michael Blaich's work where no solution could be found at all or any case where more than 2 seconds are needed for a trajectory search, defined as the worst case time.

for a collision free path while providing known trajectories of other ships, given as obstacles. A desired trajectory is used to state the intention of the own ship in the very first round, and the path planner will try to deviate as little as possible.

4.3.4 Extension to the n-Ship Scenario

Without a central planner the found trajectory has to be coordinated with all other ships in an area. To this end it is broadcasted within a neighbourhood of the own ship, where all other ships are then able to plan an improved trajectory for themselves, based on the now common known intentions. In a negotiation based on offers of trajectories and counter-offers over several rounds, trajectories may initially still collide and there may be alternating sets of trajectories over several rounds which would be the dangerous equivalent of two pedestrians walking towards each other, being unable to decide on which side to evade. To structure the exchange of trajectories to a procedure, which can be proven to produce a solution, if one exists, and which converges towards an optimal solution, without a central planner, a decentralised negotiation is used.

4.3.5 Cooperative Collision Avoidance (Waslander, 2007)

The algorithm suggested by Waslander (2004) and later refined and evaluated in different scenarios in Waslander (2007) was developed to optimise full paths of individual decision makers, which are modelled up to their control inputs. In his work he suggests a procedure for cooperative collision avoidance i.e. for rovers in a rover testbed and for Unmanned Aerial Vehicles in a simulation.

He developed a decentralised negotiation schema based on earlier work of Inalhan (2002), in which vehicles value trajectory sets based on their local cost functions, which converge towards a global cost optimum. The decentralised algorithm finds an *efficient and equitable* solution, which solves the problem of balancing a solution which has the highest overall quality, but may be unfair towards single vehicles, and a solution where all vehicles benefit in the same way from the system's resources, but which might not distribute all of the available resources. In the maritime domain this balance is often off when a ship is designated as a *give-way* vessel and other ships have to follow long evasive trajectories to conform to the COLREGs. Similar COLREG-conform solutions may not only be unfair but also not use the available sailing-space as good as it were possible if all ships in a situation adjusted their course slightly. This is one of the key motivations in this work for searching also for solutions which might not conform to all of the COLREGs .

In the work of S. Waslander, a set of aerial vehicles have colliding desired trajectories on a two dimensional plane. Even though he considers specifically quadrotors in the *The Stanford Testbed of Autonomous Rotorcraft for Multi Agent Control* (STARMAC) (Hoffmann et al., 2004), they only try to evade in two dimensions which is a common restriction for UAV motion planning (Richards and How, 2002). The vehicles have two control inputs, roll and pitch or speed and angular velocity in his second application scenario using rovers. He defines a trajectory as a number of time-dependent positions of a vehicle j as $x_j(k) \in \mathbb{R}^s$ in s dimensions over $k \in \mathcal{K}$ constant timesteps[†]. Vehicle dynamics are modelled as a function which determine the course of the trajectory dependent on the control inputs $u(k)$ at time k: $x_j(k) = f_j(x(k-1), u(k))$. The trajectories are followed using the control inputs $u(k)$, which are chosen based on a receding horizon control procedure with the later developed augmented cost function. Desired trajectories are denoted x_j^d. The way in which costs are assigned to a trajectory is investigated through the work with several different cost metrics. While in the work of Michael Blaich the cost-function considers penalties for several factors over its course in a grid, the *vehicle cost function* J_j from Waslander only penalises deviation from a desired trajectory quadratically in every timestep, leading to his definition

$$J_j(x_j) = \sum_{k \in K} w_j(k) ||(x_j^d(k) - x_j(k))||_2^2 \tag{4.1}$$

with an weighting factor w_j.

In contrast, in this work the course of the trajectory is not under the control of the negotiation component, thus the cost can not be minimised directly by optimising control inputs u. A desired trajectory will be defined and transmitted to the path planner which also tries among other factors to minimise the deviation from this desired trajectory, but it also includes the other trajectories as basis for the cost calculation, which are not a part of Waslanders *vehicle cost function*. When in section 4.4 cost functions are adjusted to the maritime problem, the *cost-to-come* function from Michael Blaich is used as a substitute for the *vehicle cost* from Waslander, and the term used to emphasise the origin is, as explained, the *ship vehicle cost function*.

For the cooperative collision avoidance problem a nash bargaining cost metric can be used, which Waslander defined based on his vehicle cost as $J^{nb} = \sum_{j \in J}(-\log(d_j - J_j))$ where J is a global cost metric and d_j a *disagreement* point, a cost expected when the vehicles do not come to a conclusion.

[†] All defined formulæ can be found in (Waslander, 2007, pp. 69-78).

The second important part of the final cost function is a penalty function[‡]

$$P_j(x_j, \{x_i\}_j) = \sum_{i \in J_j} \sum_{k \in K} \max \left(0, R^2 - ||x_i(k) - x_k(k)||^2\right)^{\gamma} \tag{4.2}$$

which is used to penalise any trajectories in a set closer than a certain radius R at a timestep k with a parameter γ to ensure continous differentiability. $\{x_i\}$ denotes the trajectories in the *neighbourhood* of the own vehicle, for a number of i neighbours. This is similar to the understanding of the *other* trajectories in this work. Due to the summation the penalty will be given at all neighbouring points around time k where the trajectories are too close. As will be described in detail in section 4.4, in this work only the distance between the closest points of approach in between two trajectories is used to calculate the penalty given.

In a last step Waslander derives the *augmented cost function*, J_j^P which is adopted in this work with changes to the contributing cost functions as will also be explained later in the chapter. The original function is defined as:

$$J_j^P(x_j, \{x_i\}_j) = \beta_j J_j^*(x_j) + P_j(x_j, \{x_i\}_j) \tag{4.3}$$

where β is a convergence parameter which tends to zero and is used to decrease the value of the vehicle cost function J_j^* over all negotiation rounds. Other cost metrics were evaluated in the original thesis but in this work only nash bargaining costs are regarded leading to $J_j^* = J^{nb}$.

Finally, to produce trajectories Waslander solves the track-keeping problem of choosing the appropriate control inputs in addition to the problem of optimising the augmented cost function. It is approached by solving the local optimisations of the receding horizon control problem with Matlab's fmincon solver MathWorks (2016). The approach developed in the following is not considering the track-keeping problem, yet the dynamic model of the vehicle is part of the path planner and the Bézier curves are used to transform the found path to a trajectory.

Waslanders approach was designed to be applicable to vehicles with both holonomic and non-holonomic dynamic constraints, which is important for an application on a ship, which has 6 degrees of freedom but can not directly control its position along each dimension. One important downside of his approach is that, even though it can be shown to converge to a solution when the problem is formulated as a convex optimisation problem, the *decentralised* cooperative problem *"...is formulated as a nonconvex optimization, it is not possible*

[‡] (Waslander, 2007, p. 70 eqn. (4.6))

to guarantee that the solution achieved will be optimal, nor is it possible to ensure that if a solution exists, the algorithm will achieve it …"[§]. This problem is approached by solving a relaxed problem in a first round of the negotiation to generate an initial collision-free solution, which will then be improved in further negotiation rounds. This is a compromise in between the proposed *sequential form* from Waslander (2004) and the *multi-threaded* form, with an initial collision-free solution to provide safety critical guarantees for the maritime domain.

In the original approach the augmented cost function is optimised by ensuring that the penalty cost decrease strictly over all rounds. This leads in combination with the implementation as a control law to trajectories, initially planned very close to each other with high penalty costs for violated constraints. They are then strictly optimised towards a Nash Bargaining Solution with no penalty costs. Due to the coordination with the path planner in this work collision-free trajectory sets converge towards fair, collision-free and efficient sets in a time sufficient for the maritime domain with the advantage of producing only collision free sets after the initial solution. The penalty function is re-interpreted as violation of penalising soft-constraints and will, opposed to Waslanders approach, not strictly decrease.

The definition 12 of the final *augmented cost function* used in this work, explained later in the context of the other modified functions, contains the parameter β to decrease the weight of the external ship vehicle cost function while increasing the weight of the penalty cost function thereby shifting the objective of the selection process from finding locally favorable trajectories to finding a global fair solution.

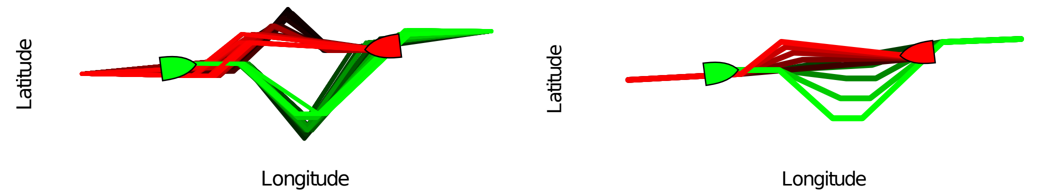
4.3.6 COLLISION DEFINITIONS AND ACTION ALTERNATIVES

The definition of a *collision* in Waslander (2007) is that no vehicle enters the safe radius R around any other vehicle as a hard constraint. The optimisation performed gradually moves from straight line trajectories, which receive a penalty depending on their course inside the radius, towards collision free trajectories. This definition differs in the following in the sense that a collision is considered having ships closer than their physical size, which is idealised to a certain, smaller radius σ. The radius R from Waslander is used in the penalty function to structure the search for an optimal trajectory set, however a distance shorter than R but larger than σ is not considered a collision in this work. This change is motivated by the fact that the path planner will immediately find collision free trajectories and the procedure will after a certain point only generate collision free trajectory sets, according to the *σ-collision* definition just given. Details about preserving collision free sets can be found in

---

[§] (Waslander, 2007, p. 76)

section 4.6.1. Since the first rounds will be already collision free but containing unequally distributed trajectories with varying quality, the penalty of falling short of the radius R will be used for two purposes: to be able to compare the final results to the original approach and to speed up the process of equalising the quality of the trajectories through better spatial dispersion. In a sense the area inside R can be seen as a potential field because of its linear gradient in between R and 0. Trajectories may during path planning and in the final result stay closer than R without considering the solution as containing a collision, thus interpreting it as a soft constraint. The two different ways in which the negotiations produce trajectory sets can be seen in figure 4.2. On the left side of the figure the behaviour as observed in the experiments is shown, where trajectory sets found at the beginning are dark and get lighter in each round. It can be seen that the course of the trajectories converge near the final solution in an non-uniform way. On the right side a schematic of the convergence behaviour in the work of Waslander is given where the trajectories strictly converge outwards from colliding to non-colliding trajectories. Note that the trajectories on the left are shown for two ships however the experiment was conducted with five ships.



(a) Non-Uniform Converging Collision Free Sets

(b) Strictly Converging Sets With Initial Collisions

Figure 4.2: Different Ways of Convergence

4.3.7 TRAJECTORIES, WAYPOINTS AND EDGES

In contrast to Waslanders trajectories, which are defined over a range of discrete points in time with a constant step size and a step counter, the trajectories in this work are defined as a number of waypoints with a constant speed at continuous points in time. A point in time is added to the definition of a waypoint for several practical purposes. It is used in the next definition of edges to implicitly define a direction, when edges are seen in isolation of a full trajectory and trajectories can be defined by linearly interpolating the points in time and space of all waypoints. Note that in the following definitions, trajectories and edges have a defined speed too because either the interface specification with the path planner requires it or the way in which the penalty function is calculated based on the closest point of approach of two edges benefits from it.

Definition 1 : A **waypoint** w is a tuple, representing a 2-dimensional coordinate at a specific point in time s.

$$w = (x, y, s) \in \mathbb{R}^3 \text{ with } x, y, s \in \mathbb{R}$$

The actual implementation of a waypoint is defined as an offset in meters in the *North-East-Down* (NED) reference frame to a coordinate in the WGS 84 system. The NED is defined as a local coordinate system on a plane, tangent to the spherical world. For each trajectory, defined in the following, the plane is fixed at a specific reference latitude and longitude value and when trajectories are compared to each other, their meter-wise distance is calculated with the same reference coordinate. Because of the curvature of the earth, points further away from the reference point are slightly inaccurate, which is in the context of the evaluation however limited due to the maximum trajectory length in between 600 and 900 meters and the limited distance of two trajectories in the observed scenarios. For the remainder of this chapter, this simplification will enable calculating distances as euclidean distances. Even though close-quarter situations for ships with speeds over 50 knots may already start at 4 nautical miles ahead, (Hilgert, 1983), yet this can be measured and corrected in scenarios which stretch over several nautical miles (Duvenhage and Nel, 2007). Due to the design of the path planner, which in its current form is configured for smaller recreational ships, this correction was not performed in this work but the results are considered transferable to more extended scenarios with the appropriate correction.

One definition needs to be made before defining the constraint penalty function in section 4.4.2 on the distance of two edges from a trajectory in between waypoints:

Definition 2 : An **edge** e_{jk} is a linear connection in between exactly two waypoints w_j and w_k with a speed v. The direction is implied by the points in time of the two waypoints.

$$e_{jk} = (w_j, w_k, v) \in \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R} \text{ with } j \neq k$$

A trajectory is understood as the idealised, intended time-dependent position which a ship tries to maintain as close as possible with a constant speed. This important limitation enables a definition based on a number of waypoints and the constant speed instead of a function with time as a parameter. Trajectories are in this context defined in a non-differentiable way, which would lead to a deviation when following the course of the trajectories at the transitions over waypoints. However, the trajectories used and negotiated are only the basis for the external path planner which may add smooth transitions before the final implementation through a ship.

Definition 3 : A **trajectory** t is a number of $l \in \mathbb{N}^+$ successive waypoints w_i where $i \in \{1, \ldots, l\}$ and a constant speed $v \in \mathbb{R}$.

$$t = (w_1, \ldots, w_l, v) \in \underbrace{\mathbb{R}^3 \times \mathbb{R}^3 \times \ldots \times \mathbb{R}^3}_{l} \times \mathbb{R}$$

The desired trajectory of an agent a, x_a^d, usually a straight line connecting the start and the destination of a ship, is known to each agent controlling its ship from the very beginning. All other trajectories are either the result of a query to the path-planner or incoming broadcasts from other agents and thus implicitly results from their queries to their path planner. The path planner and the trajectory negotiation process communicate other trajectories and own desired trajectories using *Trajectory Sets*:

Definition 4 : A **trajectory set** T is a number of $n \in \mathbb{N}$ trajectories, exactly one for each of n agents:

$$T = \{t_1, \ldots, t_n\}$$

A collision-free trajectory for the own ship can be found by querying the external path-planner with a desired trajectory and an optional trajectory set with information about other trajectories:

Definition 5 :

A **path-search** p_a of an agent a with a trajectory set T^{a}

$$p_a : (T^{\text{a}}) \rightarrow (t^{\text{a}}_{\text{cf}}, c^{a})$$

is the generation of a collision free trajectory for that agent t^{a}_{cf} and associated costs $c^{a} \in \mathbb{R}$, where:

T^{a}	contains the set $\overline{T^{\text{a}}}$ of all other trajectories which may be empty and the own trajectory $t^{\text{a}}_{\text{own}}$
$\overline{T^{\text{a}}} := \{t_i \mid t_i \in T^{\text{a}}, t_i \neq t^{a}_{\text{own}}\}$ $T^{\text{a}} := \overline{T^{\text{a}}} \cup \{t^{a}_{\text{own}}\}$	
$w_s, w_d \in t^{a}_{\text{own}}$	While the own trajectory t^{a}_{own} contains at least two waypoints w_s and w_d, the position of the ship and the immediate destination.

Here the $\{t_{\text{own}}\}$ notation is used to denote the set with the own trajectory of agent a as its only member. In order to form a trajectory set from a path and because it will be useful later in the extension of properties to all rounds, a trajectory-search is defined:

Definition 6 :

A **trajectory-search** f on a trajectory set T^{a} replaces the own trajectory t^{a}_{own} with a collision free trajectory t^{a}_{cf}, found by a *path-search* p_{a} in that set.

$$f\,(T^{\text{a}}) = \overline{T^{\text{a}}} \cup \{t^{\text{a}}_{\text{cf}}\} \text{ with } p_{\text{a}}(T^{\text{a}}) = (t^{\text{a}}_{cf}, c^{a})$$

After the definition of a trajectory set it is possible to compare different trajectory sets, based on two cost functions which are explained in the following:

4.4 COST OF TRAJECTORIES AND SETS

Global and local optimisation can only be performed according to a defined performance measure. A cost function assigns a numerical value to a trajectory set and imposes an order onto the space of all trajectory sets which is used to encode desired quantifiable qualities of a single trajectory and the combination of all. The number of turns, which should be kept minimum at sea, can be part of the cost function as well as the length of a trajectory, or even the direction if the cost function is used to avoid planning an trajectory against a strong current or heavy winds. To assign higher costs to unfavourable trajectories, Waslander (2007) uses a combined *augmented cost function* for decentralised optimisation in which individual decision makers optimise their part of the global cost function by minimising local cost functions. He shows that the local optimisation converges towards a global optimum, which is a combination of individual cost minimisation and global constraint satisfaction. The used constraint cost function penalises infringement of the safe radius R as shown in section 4.3.5, which is the only part of the function based on other trajectories. The individual distance-wise deviation from a desired trajectory is the only base for the *ship vehicle cost function*. The resulting augmented cost function can be used for each agent to combine costs for the individual agent and the group as a whole, weighted by the factor β which increases the influence of constraint satisfaction over the course of the trajectory negotiation.

In contrast, in this work the cost function for a single trajectory is calculated externally by the path planner. Individual costs for the own ship may be calculated differently among ships as i.e. turns may be more costly than detours. Small ships with less draught may also account the distance to obstacles as landmasses less costly. Therefore, in the following an individual ship may instead of choosing parameters on a fixed cost function use an arbitrary path planner in the procedure, as long as it fulfils the requirements. The implemented and described path planner in is used in this work as a reference implementation and to formulate general requirements on suitable path planners for to be used in the negotiation. Fulfilment of these requirements will determine which statements can be made about the optimality and completeness of the negotiation process.

4.4.1 SHIP VEHICLE COST FUNCTION

The used path planner, explained in section 4.3.1 combines several penalty functions which contribute to the cost-to-come function in the specialised A* algorithm which are hidden to the trajectory negotiation. Only the final calculated cost of a trajectory is available through an interface and in the following referred to as the *ship vehicle cost function* for agent a. In contrast to the work of Waslander, it assigns costs to a single trajectory by also considering information about other trajectories as well. Nevertheless, since the modified cost functions are based on their work, the notation in the following is based on Inalhan (2002), Blaich et al. (2012) and (Waslander, 2007, pp. 68-72):

Definition 7 : The **ship vehicle cost-function** J^{a} of agent a is calculated externally and assigns a cost c^{a} to a trajectory set T^{a}, given a trajectory t^{a}, based on an external performance measure of that agent.

$$J^{a}(T^{\mathrm{a}}) = c^{a}$$

Note that especially in the beginning without prior knowledge the case $T^{\mathrm{a}} = \{t^{a}_{\mathrm{own}}\} \cup \overline{T^{\mathrm{a}}}$ with $\overline{T^{\mathrm{a}}} = \emptyset$ is explicitly allowed.

Apart from the individual ship costs, the concept of penalising infringement of a safe area around the ship is also realised within a radius R, yet by adding a scaled cost at the closest points of approach in between the own trajectory and every other trajectory:

4.4.2 CONSTRAINT COST FUNCTION

All found trajectories should contain not only no collisions but also enable the ships to keep a safe minimum distance at all times. Trajectories which are planned too narrow are therefore penalised using a function, which adds a cost for each edge e_i of any other trajectory, which is too close to an edge e_a of the own agent. The minimum distance of the two closest points of approach between two edges is calculated and costs are incurred proportionally to a desired distance R. The radius R and distance-wise penalty is analogous to Waslander (2007), however, since the trajectory definition differs, the cost is calculated based on points on the linear edge:

Definition 8 : For two edges e_i, e_j their closest points, where their distance is minimal, will be written as $p^i_{CPA}(e_i, e_j)$ on e_i and $p^j_{CPA}(e_j, e_i)$ on e_j. The **distance of the closest points of approach** $d_{CPA} : (\mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}) \times (\mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}) \to \mathbb{R}$ is the distance between the unique closest points of approach of two edges:

$$d_{CPA}(e_i, e_j) = ||p^{\mathrm{i}}_{CPA}(e_i, e_j) - p^{\mathrm{j}}_{CPA}(e_j, e_i)||$$

This scaled distance is used to penalise infringement of the safety radius R. A parameter $\gamma \geq 2$ is chosen to keep the penalty function continuously differentiable.

Definition 9 : The *penalty cost function* $P_a : T^{\mathrm{a}} \to \mathbb{R}$ gives the constraint penalty cost for a given trajectory of an agent t^a, in a trajectory set T^{a}, containing i edges relative to the set of all other trajectories $\overline{T^{\mathrm{a}}}$ containing j edges on all trajectories.

$$P_a(T^{\mathrm{a}}) = \sum_{x=1}^{i} \sum_{y=1}^{j} \left(max\Big(0, R^2 - d_{CPA}(e_x, e_y)\Big)^{\gamma} \right),$$

where $e_i \in t^a \wedge e_j \in t$

4.4.3 PARETO OPTIMALITY

In the space of possible solutions, many can be found where every ship can be closer to its objective, better off than before according to the cost functions. However, at some point every increase in quality for one ship must result in the disadvantage of another because at that point the resources, in this context i.e. the available space on sea , would be already allocated exhaustively. All solutions which are of that kind lie on the *Pareto-Optimal Frontier*. In the work of Waslander a Pareto Optimal Solution is only defined on a convex centralised optimisation problem and for the decentralised problem the frontier is calculated in a centralized way to evaluate the solution. Central problems, as the arbitrary ship vehicle cost function which lies within the domain of the path planner and a changed penalty function in addition to a non-convex problem statement make it impossible to guarantee that the Pareto Optimal Solution can be reached. However, all experiments in chapter 6 converge very close towards the theoretical cost minimum, in the absence of a collision which can be calculated through the design of the experiments and the cost function. A solution of a trajectory planning process on a Pareto-Optimal frontier may use the available space as good as possible, or avoid turns and obey the COLREGs as closely as possible and will be considered *efficient*. However, with regard to the individual ship, the outcome may not be distributed as *fair* as possible. Achieving a *fair* and *efficient* solution which assigns the *best* possible set of trajectories as evenly as possible to each ship is not straightforward. Individual ships could change their behaviour after an agreed solution, to pursue a better trajectory at the expense of other ships. In practise this could lead to unintended egoistic behaviour once it would seem that the overarching goal of collision avoidance is reached, which must be discouraged by the assurance that a fair situation can not be improved by single deviation. One fair solution could be one in which all trajectories have the same length or the same number of necessary turns. This solution would lead to the same demand of time and space of every ship, however, changing a course is not an equally demanding task for every ship. Speed and manoeuvrability differ among ships, making it more costly for some ships to change a course or follow a longer trajectory. In Waslander (2004) a fair solution is considered one in which: ”*...each agent receives an equal amount of the excess in the system, or incurs an equal amount of cost.*”. Therefore, fairness in the negotiation will be traced in the evaluation by comparing if the cost-functions, which include all factors as turns and lengths, converge towards the average costs of the group as a whole since the set-up will be chosen to incur exactly the same cost for every ship in an ideal symmetric *fair* solution. This will be done by choosing the same type of ship with the same properties as speed and manoeuvrability and using trajectories of the same size with the same ship domain to model the COLREG compliance.

4.4.4 NASH BARGAINING

In 1951, John Forbes Nash investigated decision problems, where the outcome heavily depends on the decision of every single planner (Nash, 1951). He defined a *Nash Equililbrium*, a solution after which it becomes unrewarding for a single decision maker to individually change their strategy to improve their individual gain. This concept can be used to reach trajectory sets which are not only fair but will be also accepted by maritime personal because it can be assured that alternative trajectories would not be collision free or would contain detours. The cost metric which is used is based on the Nash Cost Metric, adopted by Waslander for the decentralised problem, which is defined in the following. A necessary definition in this context is the negated cost which the agents try to maximise, their *utility* for a final set of trajectories which is defined analogous to Muthoo (1999):

Definition 10 : The *utility* $U_a : T^{\,a} \rightarrow \mathbb{R}$ for an agent a is the quality of the found trajectory in context of its trajectory set on the agent's performance measure. Here it will be simply defined as the negated ship vehicle cost function $J^{\,a}(T^{\,\mathrm{a}})$.

$$U_a(T^{\,\mathrm{a}}) = -J^{\,a}(T^{\,\mathrm{a}})$$

4.4.5 DISAGREEMENT POINT

A disagreement point $d_a = U_a(T^{\text{failed}})$ is the final *utility* for an agent if all agents fail to reach an agreement. It will be assumed to be no better than any feasible solution and is for practical purposes defined in this context, though not intuitively as the worst case solution of an actual negotiation. It will instead be defined in the evaluation as the cost of a hypothetical rectangular trajectory, which has the maximum *length* and *width* of the available *area*.

The course of this most unfavorable trajectory would be the outer rectangular rim of the defined area in which trajectory negotiations take place. Finding the optimal size of an area is beyond the scope of the thesis and as a sensible choice the length and width were set to a fixed value, approximately three times the size of the Chebyshev distance of the most outer ships under examination. Theoretically, in a small area a collision-free trajectory with numerous turns could be planned which would result in a higher cost-evaluation than d_a and therefore to an undefined logarithm in the following definition 11. This was in the evaluation solved programmatically by setting a fixed worst case bargaining cost, in case the argument became negative, even though this was never observed and is highly unlikely because the path planner plans always feasible, collision free and optimal trajectories. The disagreement point can now be used to define the Local Nash Bargaining Cost Function:

4.4.6 LOCAL NASH BARGAINING COST FUNCTION

Definition 11 : The *Local Nash Bargaining Cost*-Function $J_a^{nb} : T^a \rightarrow \mathbb{R}_+$ is the following local penalty function for an agent a.

$$J_a^{nb}(T^{\,a}) = -log\,(d_j - J_a(T^{\,a}))$$

It was shown by (Waslander, 2007, pp. 31-32) that the global cost function for the centralised optimisation problem can be separated into local cost functions and that the central problem converges towards a unique *Nash Bargaining Solution* if the local cost functions J^{nb} all converge towards a minimum. Now it is possible to use the Nash Bargaining Cost Function with the ship vehicle costs from the path-planner and the edge-based constraint penalty function to define the augmented cost function, analogous to Inalhan (2002):

4.4.7 AUGMENTED COST FUNCTION

Definition 12 : The *Augmented Cost Function* $J_a^{\mathrm{P}} : T^{\,a} \rightarrow \mathbb{R}$ with the local nash bargaining cost is defined as:

$$J_a^{\mathrm{P}}(T^{\,a}) = \beta_a J_a^{\mathrm{nb}}(T^{\,a}) + P_a(T^{\,a})$$

The parameter β is reduced towards zero during the negotiation. In the evaluation, the initial offset of β was chosen according to the determined domains of both functions and designed to lead to an equal contribution of both functions to the augmented function approximately at half the number of trials of the maximum allowed. Optimal selection of the offset and domain of the penalty weight parameter is still an open problem and the chosen way a compromise between fast convergence of the augmented cost function and thorough exploration of the configuration space.

After all cost functions are defined, it is possible to define the global problem which has to be solved by the decentralised negotiation: A number of n agents $a_i, i \in \mathbb{N}$ are individual decision makers, one on each ship. set of trajectories $T_i^a \in \mathbb{R}^m$ is assigned.

Definition 13 : The global problem to be solved by n agents a_i is to find the optimal trajectory set T^{opt} in the set of all trajectory sets $T^{\text{opt}} \in \mathbb{T}$ where

$$\sum_{i=1}^{n} J_{a_i}^{\text{P}}(T^{\text{opt}}) \leq \sum_{i=1}^{n} J_{a_i}^{\text{P}}(\mathrm{T}) \,\forall\, \mathrm{T} \in \mathbb{T}$$

4.4.8 Proof of convergence

A remaining question is whether the negotiation can diverge. Divergence would mean in this context that the chosen solutions from all agents in consecutive rounds always differ by an augmented cost value of more than a chosen ϵ and alternate between two or more possible solutions. In this case the negotiation would not end since the process is algorithmically designed to finish and pick the best solution from the last broadcasted set of trajectory sets once the costs of two solutions in consecutive rounds are less than or equal to ϵ.

Even though it can be observed in the current evaluation that the process converges towards one solution, the proof presented, which will show that solutions will never diverge, has to rely on further assumptions and requirements which would make it harder to keep the requirement chosen in this work that the ships do not know each others value functions. Possible properties of a negotiation with a different path planner are shown in the following, however they rely on a limitation on the way in which the vehicle cost function J can be designed, even though the used and evaluated path planner works without this restriction.

Further requirements of the path planner

Let Θ be the set of all agents, participating in the negotiation.

Definition 14 : After a trajectory search $f_a(T) = T'$ on an arbitrary trajectory set T the ship vehicle costs of its result $J(T')$ will not only decrease monotonically for the local cost function of an agent a $J^{\,a}(T') \leq J^{\,a}(T)$ but for all agents.

$$f_a(T) = T' \Rightarrow J_i(T') \leq J^{\,a}(T) \forall i \in \Theta$$

This definition could be achieved by the path planner by excluding factors from the calculation of its cost function which may increase the costs based on other trajectories. The actually used path planner calculates the proximity of the own to other trajectories and allows for other paths to be within the own ship domain but associates costs depending on the level of intrusion into the domain. This would have to be changed to a behaviour where there is a minimum distance to other trajectories which is the same during the negotiation over all ships. Then it can be achieved that no trajectory is planned in the proximity of another trajectory so that it will not change its local value function. However, as a downside this ship domain size would have to be known over all ships beforehand and therefore either communicated, mapped according to certain specifications as the ship size which are easily obtainable or fixed regardless of other factors over all ships.

A further requirement is, due to the large configuration space and due to the continuous cost functions, held most of the times however theoretically a path planner with a discretised cost function and a coarse grid representation could assign exactly the same costs to very different solutions. Therefore the following assumption is made about the vehicle cost function of the path planner for all sets of trajectories from the set of sets of trajectories $\mathbb{T}$.

Definition 15 : For the final part of the proof seen in the following, an injective cost function is required so that convergence towards a final cost value also means convergence towards one final set of trajectories.

$$\forall T, T' \in \mathbb{T} : J(T) = J(T') \Rightarrow T = T'$$

Convergence of the augmented cost function

For the proof two arbitrary consecutive rounds r and r_{+1} will be examined, in which the trajectory sets with the lowest augmented cost over all ships are denoted T and T'.

As seen in definition 12, the augmented cost function of an agent a, $J_a^{\mathrm{P}}(T^{\mathrm{a}})$ is the result of adding the weighted results of ship vehicle and penalty cost function, $\beta \cdot J_a^{\mathrm{nb}}(T^{\mathrm{a}})$ and $P_a(T^{\mathrm{a}})$ calculated on a trajectory set T^{a}. The penalty function P_a is defined to be 0 iff pairwise, for all trajectories in T^{a}, the summed distances at their closest points of approach $d_{\mathrm{CPA}}(T^{\mathrm{a}})$[§] are larger than the defined range R and it will increase monotonically when trajectories are planned closer together.

For any agent $i \in \Theta$ a trajectory set T_i was selected to be broadcasted locally in an arbitrary round because it has the lowest local augmented costs $J_i^P(T_i)$. After a broadcast event all agents exchange their sets, leading to a set of trajectory sets on each ship $\mathbb{T}_i$.

When a new trajectory is planned locally on each ship i and in each of n trajectory sets $T_1^i, \ldots, T_n^i \in \mathbb{T}_i$ this leads to the new set of trajectory sets $T_1^{i\prime}, \ldots, T_n^{i\prime} \in \mathbb{T}^{i\prime}$ on each ship and since $J_i(T^{i\prime}) \leq J_i(T^i)$ for every set because of definition 14 all the new trajectory sets in $\mathbb{T}^{i\prime}$ will have a lower or equal value for the ship vehicle function $J_i(T^{i\prime})$. This is now repeated in each round and even though the next set to be broadcasted will be chosen on the basis of the augmented cost function, regardless of the chosen set all ship vehicle costs of all sets will decrease monotonically arbitrary close towards a final value of $J_i(T^i) \rightarrow J^{\text{min}}$ given enough rounds $r \rightarrow \infty$ regardless in particular of the behaviour of the penalty function. Finally, because of definition 15 convergence towards constant minimum ship vehicle costs also means convergence towards a final trajectory set on each ship $T^{i\ \text{final}}$. Since the penalty cost function is only dependent on the distances at the closest points of approach $d_{\text{CPA}}(T^{i\ \text{final}})$ will stay constant when the final trajectory set $T^{i\ \text{final}}$ is approached. Therefore the augmented cost function will also converge on each ship as the vehicle cost function is converging. In the end, through the broadcast, the set with the lowest individual augmented cost function on any ship will dominate as the selected final solution.

This will however say nothing about the speed of the convergence since the penalty cost function could theoretically lead to choosing the set with maximum ship vehicle costs in each round therefore slowing the convergence. For the used path planner no proof could be produced that would show that there are no circumstances in which the negotiation, without a monotonically decreasing ship vehicle cost function, can not diverge so it remains an open problem. However, in reality the first sequential solution will always be created before the actual negotiation and an implemented component on a ship's bridge can use it in case of diverging solutions detected after a fixed timeframe.

4.5 DEVELOPMENT TOWARDS A COLLISION AVOIDANCE SYSTEM

The solution to the problem in definition 13 lies in finding algorithmically the optimal balance in between an efficient and a fair solution. The algorithmic procedure for the three stages of the algorithm is shown in the following; sending initial desired trajectories, plan-

---

[§] The term $d_{\text{CPA}}(T)$ will be used as abbreviation for the sum of all distances of the closest points of approach of all trajectories in a trajectory set T, since d_{CPA} was previously only defined on edges.

ning not in parallel but in a defined order and finally entering a parallel negotiation until all cost functions are near their interdependent optimum. When applied in a real-world scenario a number of safety properties of the procedure are necessary in the critical domain, which will be shown in section 4.7.1. In the end some restrictions on real-world application are given.

4.6 ALGORITHMIC PROCEDURE

In the beginning as shown in algorithm 1, each agent receives position, heading and speed information about other agents which will take part in the negotiation. Its own position and desired destination are used to query the path planner for an initial straight line trajectory to observe eventually present obstacles, apart from other ships or their trajectories as landmasses or buoys and plan the desired trajectory around them. After each ship knows its possible best trajectory in the absence of other trajectory information, it broadcasts this *desired trajectory* to other ships in the area. The received desired trajectories of all ships are merged to the first initial trajectory set, usually with several collisions. To be able to find an initial collision free solution every time, even if it is unfavourable, a sequential round is used, shown in algorithm 2. This requires that in a fixed sequence the first agent starts with an ASTARREQUEST to its path planner, with the just merged trajectory set as a parameter to inform the path planner of all other desired trajectories. If the agent is not the first in the sequence it continues to the sequential round where it waits for its turn. Note that the pseudo code is heavily simplified to improve readability and some terms are ambiguous, since a TRAJECTORYSET usually refers to the set of all trajectories while it is used in line 8 of algorithm 3 in the min calculation to refer to the costs of the set after the evaluation according to the cost function.

4.6.1 SEQUENTIAL PRE-ROUND, ALGORITHM 2

In order to guarantee finding a trajectory set with collision free trajectories in every round and thus fulfil the requirement R_7, even though it might not yet be optimal in the sense that is strikes the perfect balance in between fair and efficient, a first sequential round is introduced. In this extra round before the negotiation starts the ships plan their trajectories not in parallel but in a strict order. The source of the order is a technicality and easily done via hash methods. It could be determined based on the ship's MMSI in addition with the current date or call name but should in any case contain an element of random chance combined with an unique element for every ship to avoid duplicate hash values and change the

Algorithm 1 Sending desired initial solution

```
Initialise Agent
Receive Position and Heading of each Ship
Query Path Planner for Initial Straight Line Trajectory
Plan Desired Trajectory to Destination
if All Other Desired Trajectories Received then
    Establish Sequence Among Ships
    if Own Ship is the First in the Sequence then
        Create Merged Set with all Desired Trajectories of all other Ships
        Query Path Planner for Own Trajectory based on Merged Set
        Receive and add Own Trajectory to Merged Set which is collision free in that set
        Broadcast complete Merged Set as Preferred Solution
        Continue with Sequential Round
    else
        Start Sequential Round
    end if
end if
```

order in every situation because if the procedure is stopped before an optimal solution is reached, the initial order has an influence over the quality of the final trajectories. A ship which always plans its trajectories last or first may have an advantage, depending on the procedure. The first ship to plan a trajectory after a collision was detected, usually finds itself in a large free state space with straight trajectories of all other ships, colliding in the same, or nearly the same point. The last ship however can usually keep its course because every other ship before planned a collision free trajectory around the last one left. In general bargaining situations are known to have a *First Mover Advantage* (Sutton, 1986) and Waslander (2004) also found the advantage of the last agent to plan. In the implementation the AgentID is used for the order in which the agents take part in the first round.

4.6.2 Further Rounds (Parallel Negotiation) Algorithm 3

Before the normal round-wise negotiation can start all agents have to use the last trajectory set, the Full Solution, found by the very last agent in the sequence, as a basis. The last agent broadcasts its solution to all other ships, as shown in line 11 of algorithm 2, which all agents then use as basis for a new request to the path planner component. Once they receive their answer the Parallel Planning starts, shown in algorithm 3 which is iterated

Algorithm 2 Sequential Round

```
if Own ship is first in the sequence then
    CurrentSet← previously created Preferred Solution
else
    CurrentSet← received solution
end if
if Sequential Solution is received then
    if Own ship is next in the sequence then
        Plan a new Own Trajectory on the basis of the CurrentSet
        Merge planned collision free Own Trajectory with CurrentSet to Preferred Solution
    end if
    if Own Ship is the very last in the sequence then
        Broadcast Preferred Solution as Full Solution to each ship
        Start Parallel Planning Round
    else
        Broadcast Preferred Solution as Sequential Solution to the next agent in the sequence.
    end if
end if
```

over all rounds. The augmented cost function of the temporary Best Solution which is chosen at each round, converges after some rounds within an ϵ distance of its previous solution. Even though this would terminate the algorithm at a point where in a live scenario an implementation of the behaviour should follow, this is not used as criteria to end the algorithm in the evaluation. Since the convergence behaviour in combination with the path planner leads to differing convergence speed and for the evaluation the behaviour after such a convergence gives insight into the stability of the solution, the process was ended after 30 rounds. The number is chosen because either a convergence was achieved in observations or the procedure would need to much time and parameters had to be changed.

4.7 Properties of Trajectory Negotiation

A number of properties of the procedure are of particular interest in the safety critical maritime domain. The procedure should be able to supply information in case it will not find a solution, or only in an unknown time. In that case alternative approaches may be used, which do not try to plan full trajectories, or an immediate speed correction could be per-

Algorithm 3 Full Planning

```
if All FULL SOLUTIONS are received from each ship then

    for all FULL SOLUTIONS in round $r$ do
        Remove OWN TRAJECTORY from FULL SOLUTION, thereby creating
      the set OTHER SOLUTIONS
        Query the PATH PLANNER for a new collision free OWN TRAJECTORY,
      based on the OTHER SOLUTIONS.
    end for

    for all Newly Planned $n$ TRAJECTORY SETS from the PATH PLANNER,
      $\{$TRAJECTORYSET$[i] \leftarrow$ TRAJECTORYSET$[1]$, TRAJECTORYSET$[n]\}$ do
        Evaluate TRAJECTORY SET$[i]$ according to the cost-function to find
      the best solution
        BEST SOLUTION $\leftarrow$ min(TRAJECTORYSET$[i]$, BEST SOLUTION)
    end for

    Decrease convergence parameter $\beta$ in cost function
    Increase Round Number $r$

    Broadcast BEST SOLUTION as next FULL SOLUTION
end if
```

formed, giving the procedure the needed time.

In the following assumptions are made, about performance requirements which the current path planner fulfils and which must be guaranteed by any other path planner, which might be used in the future. After the hard assumptions a number of performance descriptions of the currently used path planner are given which determine the quality of the current procedure, but can be different for other path planners.

Since the used path planner from Blaich et al. (2012) is based on an A^* - algorithm with a consistent heuristic it is guaranteed that it is optimal, meaning it will always find the path with the lowest way-cost if it exists. Any search for a path with a fixed set of trajectories as obstacles can therefore expect to retrieve the *best path*, according to the ship vehicle cost function. Furthermore, the path planner contains an artificial limit of $s_{\max} = 2$ seconds, after which a search for a solution will be considered failed. This makes it possible to decide after a fixed time if a solution can be found at all. Since the configuration space is limited by the size of the grid-representation of the environment, a worst case time could always be calculated which is however, due to the huge configuration space, beyond 2 seconds and since in the investigated scenarios too long to avoid a collision.

Assumption 1 : There is a known point in time, s_{max} after which a collision free path is found or the path-search fails.

The path search on a set T is used to find a new own trajectory for a ship a, which is then used to exchange the previous own trajectory in that set. This can not ensure that the trajectory set is itself collision free, meaning that no other trajectories collide, but it will not introduce a new collision in collision free sets and remove any present collisions of the own trajectory in the original set. This is a fulfilled requirement of the used path planner, which is formulated for an arbitrary path planner as assumption:

Assumption 2 : Let T^{a} be the trajectory set of the own ship which contains the own trajectory t^{a}_{own}. If T^{a} is collision-free or only trajectory t^{a}_{own} collides with any other trajectory, then the resulting set T'^{a} of a *trajectory-search* $f\left(T^{\mathrm{a}}\right) = T'^{\mathrm{a}}$ on that set is also collision free.

Now it can be shown that the negotiation procedure does not introduce new collisions after any round in which all agents had only collision free trajectory sets to consider.

4.7.1 COLLISION FREE SETS

Each ship a has in each round r a number of n trajectory sets $\{T_i{}^{\mathrm{a}}\} := \{T_1{}^{\mathrm{a}} \ldots, T_n{}^{\mathrm{a}}\}$

In each round new trajectory-sets $\{T_i{}^{\mathrm{a}}\}$ are received, trajectory-search is performed on all to find new own trajectories in each set:

$$\{T_{\mathrm{A}}{}^{\mathrm{a}}\} := \bigcup_{i=1}^{n} f\left(T_{\mathrm{i}}{}^{\mathrm{a}}\right) \tag{4.4}$$

Finally the best one according to the penalty functions is chosen, leading to the single selected trajectory set

$$T_{\mathrm{S}}{}^{\mathrm{a}} = \min_{T\in\{T_{\mathrm{A}}{}^{\mathrm{a}}\}} \left(J_a{}^{\mathrm{P}}(T)\right) \tag{4.5}$$

on each ship.

In the round after the sequential round, which will be defined as $r = 3$,[¶] each ship has exactly one initial selected and collision free trajectory set, T_S^{I}, which is after the final broadcast, as seen in section 4.6.1, the same on each ship. Since in the parallel planning phase, the only actions performed on each ship are conducting a trajectory-search, selecting the best trajectory set, increasing the round number and broadcasting it, the course of the trajectories are only changed by the trajectory-search. After the sequential round, T_S^{I} is collision free and because of Assumption 2, every repeated trajectory-search for all rounds $r \geq 3$ will also be collision free.

This property is important as this makes it possible to end the negotiation at any point after the second round and still have safe trajectories for all ships. Furthermore, since each round has an artificial time limit before the search is considered to have failed, this can be extended to each round and the whole process making it possible to calculate the time negotiating over a certain number of rounds will take in the worst case if the procedure does not fail. On sea this makes the statement possible if in a concrete situation it is possible to negotiate i.e. 10 rounds, a number shown in the experiments to achieve a very good solution, or if there would be in the worst case not enough time left.

4.7.2 Predictable Fail-Time

It is possible to give an estimate for the earliest time after which the trajectory negotiation can be considered to have failed or produced a feasible trajectory set: The runtime of broadcasting a trajectory set is referred to as s_{bc}, the time to calculate the penalty-cost function is denoted as s_{penalty} and the runtime of the path planner s_{path}, which entails the time needed to calculate the ship vehicle cost function. Since a trajectory-search as defined in definition 6 is just the simplification of a path to form a trajectory in a neglectable runtime, compared to the underlying path-search, the runtime of a trajectory-search is used synonymously with the runtime of the path search.

In the initial round all ships broadcast their desired straight line trajectories in parallel without any cost-function calculation, leading to the initial trajectory set at a time $s_{\text{initial}} = s_{\text{bc}}$[‖]. The sequential round after the first round defines an order on all ships, in which each one plans a path, converts its into a trajectory, merges it with already known trajectories and sends its to the next ship in the order until the last ship broadcasts the final set to be the basis for all ships in the rest of the negotiation. In that way n ships reach the first feasible

---

[¶] (Because 1 is the desired trajectory exchange and 2 is the sequential round)

[‖] If the communication bandwidth differs, s_{bc} is considered the longest time a ship needs to perform a broadcast.

solution after

$$s_{\text{sequential}} = n \cdot (s_{\text{bc}} + s_{\text{path}}) + s_{\text{initial}} \tag{4.6}$$

Based on the evaluated value s_{path} of the path planner from Blaich et al. (2012) this leads to the following minimum times, after which the procedure can be considered to have produced a solution or failed. Some fast path search queries could be observed to finish in $40\ ms$ while the average times are marked in blue in the table. Blaich found for two ship encounters an average runtime of $s^{\text{avg}}_{\text{path}} = 67\ ms$ with a standard deviation of $\sigma = 54\ ms$ and for five ships an average runtime of $s^{\text{avg}}_{\text{path}} = 122\ ms$ with a standard deviation of $\sigma = 82\ ms$. The hard runtime limit of the path planner of $2\ s$ would lead to a time of $20\ s$ after which it is decidable if the process finds a feasible solution or not. Shown in the following are the minimum, average and maximum times according to the standard deviation with hypothetical values for 10 ships, if the values were the same as for 5 ships even though this was never part of an evaluation. As worst cases, the times for 500 ms and the fail time of 2 s is also calculated.

	$s^{\text{avg}}_{\text{path}} - \sigma$	$s^{\text{avg}}_{\text{path}}$	$s^{\text{avg}}_{\text{path}} + \sigma$	$s_{\text{path}} = 500$ ms	$s_{\text{path}} = 2$ s
2 Ships	26 ms	134 ms	242 ms	1 s	4 s
5 Ships	200 ms	610 ms	1.02 s	2.5 s	10 s
10 Ships	500 ms	1.5 s	2.7 s	5 s	20 s

Table 4.1: Expected Minimum Runtimes for an Initial Solution Depending on the Path Planner Runtime

It could be observed in the evaluation, which follows in chapter 6, that the runtime for the negotiation and broadcast was on average twice to four times as long as the path planner runtime in the full parallel mode. These considerations lead to a worst case runtime of one minute until it is decidable if a feasible solution can be found in a 10 ship scenario, if every path planner needs almost the maximum allowed time. On average, times below 5 seconds for 10 ships could be observed.

4.7.3 BÉZIER INACCURACY

The trajectories defined here are representations with straight line connections in between waypoints. The external path finding component adds smooth transitions between edges to account for the physical model of the ship. In its current form it uses Bézier-Splines to account for the natural ship movement in between two segments of a path where a change in heading is needed. These additional path information are discarded in the negotiation component. It is inevitable that some information from the path planner are lost during an approximation where only the points of the Bézier-Splines are used in the negotiation component to create simpler trajectories, which connect the *waypoints* of the trajectory through straight edges. However, this leads to consequences for the penalty cost function only which assigns costs independent of the path planner and might underestimate the costs in situations where the original paths are planned with less distance than their approximations. Since the sets of trajectories which are collision free, were collision free before the approximation as sets of paths in the path planner, no introduction of a collision according to the collision-definition of the path planner is expected.

4.7.4 TIME DELAY

The system is anticipated to be used in real time, which presents a problem. Every full negotiation over r rounds, to achieve a certain quality needs

$$s_{\text{final}} = \underbrace{r \cdot s_{\text{bc}}}_{\text{broadcasting each round}} + \overbrace{r \cdot n \cdot (s_{\text{path}} + s_{\text{penal}^{**}})}^{\text{trajectory search in } n \text{ sets}} + s_{\text{sequential}} \tag{4.7}$$

to reach a solution and a ship with a linear speed v_{Ship} can not plan its trajectory from its initial position because by the time a solution is negotiated and agreed the ship will have moved by a distance of $d_{\text{start}} = v_{\text{ship}} \cdot s_{\text{final}}$. The way in which this is handled is to plan from a future point in time and space in front of the ship which has some drawbacks. First the available time and space is further constrained and the time s_{final}, or at least an upper limit has to be known. In this work the needed time for the algorithm was determined on average in a simulation and a point chosen however, the distance has to be verified in a real setting. To find a good basis for a prediction for s_{final} is yet an open topic and the worst-case approximation in section 4.7.2 a compromise. Further parameters which are expected to influence the time are the number of ships in a situation as well as their speeds and distances, the capabilities of the crew, the performance of the hardware and the bandwidth of the communication channel. Furthermore this is an issue in which the slowest participant will determine

---

[**] Time to finish the cost calculations.

the speed of the negotiation. One possible solution would be to exclude the slowest participant from a number of rounds which has to be investigated and is not within the scope of this work.

4.8 QUALITY MEASUREMENT OF COLLISION AVOIDANCE

The quality of the approach is an abstract term which can be measured according to different metrics. It contains factors as for example the simple length of all trajectories, which indicates the needed space in the area, as well as more complex measures as the cost distribution among all ships, used to judge the fairness of the solution. We must differ between the quality of the final set of trajectories which is the outcome of a negotiation process and the course of this quality, a function over the number of negotiation rounds. This quality function can be used to make statements about the convergence of the process and the feasibility of the approach. In addition there are feasibility-constraints on the quality function, i.e. to converge to a satisfying degree within an appropriate time-frame. If the process can not produce near optimal solutions before a collision may happen, reaching near optimal solution with this approach is not considered feasible. The main measurement for the quality of sets of trajectories in this work is the augmented cost function, described in section 4.4.7. The course of the global augmented cost function as well as the several single vehicle and penalty cost functions for each ship will be discussed as part of the interpretation of the results in chapter 6. In software engineering, an approach to judge the quality of a software solution is the "Goal-Question-Metric" procedure. It is used in this work to determine which metrics can be used best to compare different parameter sets, whether the requirements are met and to judge the quality of the prototypical implementation of the process. The following questions are also stated to plan with which metrics the fulfilment of the requirements from chapter 2 will be traced.

4.8.1 GOAL

The stated goals in table 4.2 are a trivial repetition which however lead to numerous questions and metrics.

Object	Ships with trajectories in a simulation
Objective	Avoid Collisions, Minimise resource demand, Maximise human comfort, Minimise risk
Focus	Safety, Scalability, Performance
Perspective	Captain on a Bridge, Scientist, Engineer

Table 4.2: Goals of maritime collision avoidance

4.8.2 QUESTIONS

From these goals, a number of questions can be deducted which will clarify if a solution has been found and to what degree on a measurable metric. Questions about the goals are divided into two kinds of questions: Questions about the final outcome of the process, which are the actual trajectories and information about their course, length, number of turns etc. and questions about the process, which regard the performance of the trajectory-finding.

	Outcome	Requirements
1.	Are planned trajectories too close?	R_7, R_6, R_8
2.	How many turns are needed?	R_4, R_13
3.	How close is the closest point of approach?	R_7, R_6, R_8
4.	How many changes in speed are needed?	R_5, R_1
5.	How equal/fair are all trajectories?	R_15, R_16
6.	Are all collision regulations observed?	R_3
	Process	**Requirements**
7.	When will the procedure not work?	R_20, R_7
8.	How long does it take to generate collision free trajectories?	R_2, R_12
9.	How long does it take to achieve the best solution?	R_12, R_14
10.	How does the process scale with the number of ships?	R_10
11.	How fast does the algorithm produce a final collision free outcome?	R_2, R_12
12.	How close in front of the ship can a trajectory start?	R_19
13.	Does it reach the optimal solution?	R_12, R_14
14.	How is the convergence behaviour of the trajectories?	R_12

Table 4.3: Questions Regarding the Process and the Outcome

4.8.3 METRICS

The following metrics are derived to answer the questions. Conforming with the goal of GQM, the minimum number of metrics needed is selected even though, through the logging process, it is possible to evaluate a much wider range of performance indicators. The majority of experiments will be measured by these metrics:

Metric	Referred Question
All trajectories of all ships at all negotiation rounds.	1-14
Computing time of the algorithm	7-12
Initial Position, heading and speed of all ships.	1-6
Performance measure of the trajectories	1-3,5,7-11,13-14

Table 4.4: Derived Metrics

5

ManTra

In this chapter the developed software prototype for maritime trajectory negotiation is described. First the cooperation with the HTWG-Konstanz and the Mason-Framework is introduced, in which the system is embedded. An overview over the class-design is given and the interaction with the external path planner is described. In the end the internal state-succession of an agent is shown.

5.1 ManTra System Design

To evaluate the suggested procedures in chapter 4 and to provide a starting point and proof of concept for further research a software system was implemented which shows promising results in a simulation. The Maritime n-Ship Trajectory Negotiation System for Collision Avoidance (ManTra) Framework was created: In figure 5.1 all software components and their associations are illustrated. The presentation of packages, classes and associations is limited to important core functionalities to preserve a comprehensive view.

5.2 Cooperation with HTWG Konstanz

In order to fulfil the various requirements which concern the dynamic model of the ship and the implementation of the COLREGs, a path-planner is used which was developed in parallel to this work. The negotiation component communicates via the IMC-Framework using an agreed interface specification to the path planner component implemented by Michael Blaich at the HTWG Konstanz (Blaich et al., 2012). A specialised A* algorithm, which is explained in detail section 4.3.1, is used to search for an optimal single trajectory, given the presented environment information. The negotiation component developed provides a system to negotiate all single trajectories towards a near-optimal solution. Other trajectories can be included explicitly or via position, heading and speed information of other ships after which the path-planner calculates probable trajectories to be used as obstacles in the configuration space. The path planner is a *C++* implementation which is started on the same, or a different system and a *UDP* connection in between the negotiation component and the path planner must be possible and configured in the respective configuration files.

5.3 Mason Framework

The basis for the implementation is provided by *Mason*, a Java Multi-Agent Framework by the George Mason University, mainly developed by Sean Luke (Luke et al., 2005). It supplies a scheduler for several agents, utility classes for map representation and a graphical user interface as well as a more elaborate random number implementation. A vivid community contributes with classes for geodetic calculations and representation. From the multi agent framework only the use of a single agent class is used to communicate with every other agent implementation on another ship via a communication protocol.

5.4 CLASSES AND INTERACTIONS

All classes and associations are shown in figure 5.1: The ManTra - class is used to add the *Ship*-agent which is located based on externally provided information about the position and heading of the own ship. It starts all other components and is an implementation of the main SimState-class from the Mason framework. The Agent-class holds the most important part of the functionality, in combination with the TrajectoryNegotiator-Facade and the TrajectoryCostCalculator. The states of the agent as well as the conditions for their succession are shown and described in figure 5.2. The TrajectoryCostCalculator holds the methods to evaluate a given TrajectorySet using the penalty functions implemented which base the resulting cost on the vehicle cost, given by the PathPlannerService, and the constraint satisfaction. The TrajectoryNegotiator encapsulates the interaction with the different components for path planning and trajectory negotiation. It uses the PathPlannerService to query new own trajectories for the own ship which relay that query to the external path planner. Furthermore it compiles full sets of sets of trajectories for each round when all other ships send sets of trajectories. In the current implementation, the negotiation component and the path planner facade use IMC-messages to communicate with all other ships and the path planner backend. Therefore, the same IMCMessageHandler is used. The IMC based communication is implemented according to the Observer-Pattern where the IMCReceiver calls methods of the IMCMessageHandler once they are received. The IMCReceiver receives dedicated IMCMessages, which are based on generated Java bindings and include among others an AStarResponseMessage and a TrajectorySetMessage. The IMCSender is used for broadcasting trajectory sets and sends messages to the path planner.

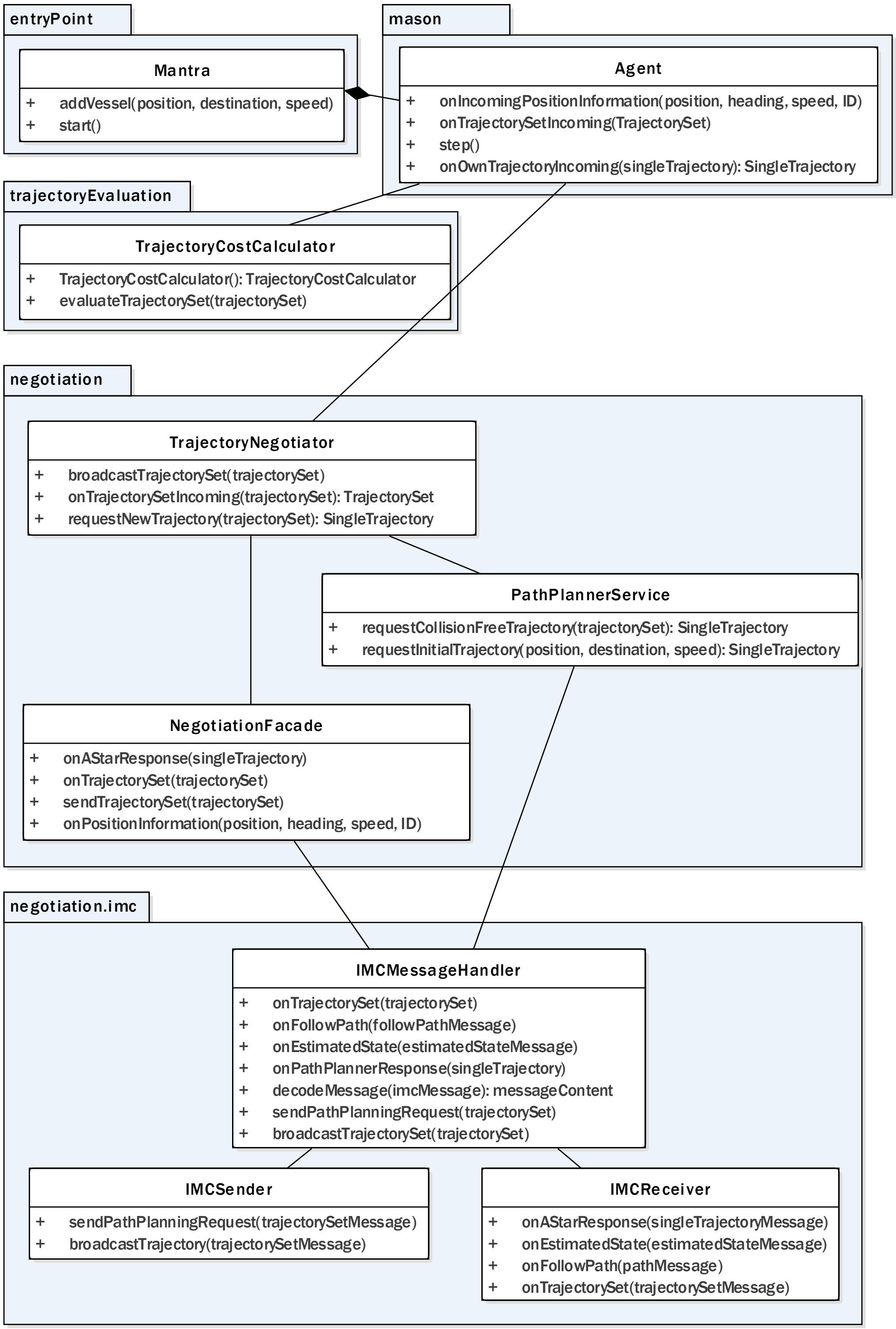


Figure 5.1: Class Diagram of ManTra

In the beginning, the agent receives all ship positions, headings and speeds. It chooses a destination and queries the PATH PLANNER for the shortest trajectory to be used as the desired trajectory, without considering other agents. Once it is planned the TRAJECTORYNEGOTIATOR broadcasts it to all other agents on all other ships while all other desired trajectories are received. Once they are received, the PREROUND (*the name in the implementationt for the sequential round*) is started, where all ships plan their trajectory in a defined order. After the last agent planned its trajectory and a collision free initial solution is achieved, the parallel planning continues, referred to as PLANNING FULL SOLUTION. During that state, full sets of trajectories are received, the own trajectory is optimised in each one, the best one according to the performance measure is chosen and broadcasted. After the solutions converge to within an ϵ- environment the final solution is chosen.

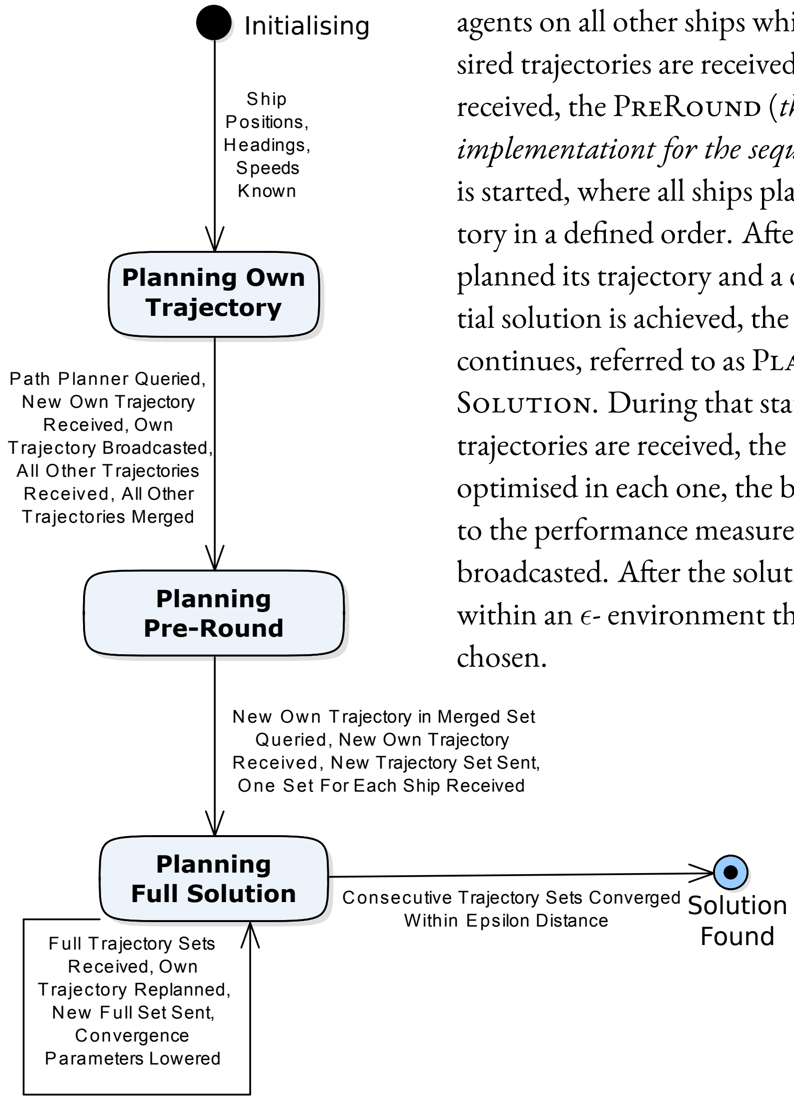


Figure 5.2: States of the Ships' Agent

6

Evaluation of Collision Free Trajectories

The system created in the previous chapter is used to evaluate the feasibility and quality of the trajectory negotiation process. The most suitable metrics, as derived in section 4.8 for the evaluation are traced during several simulated experiments. The questions in section 4.8.2 motivate the set-up where 2,3 and 5 ships are in difficult situations to judge among other factors how the procedure scale. Questions about the number of turns or the smallest distance between two ships on any trajectory can be answered by observing how ships plan in an increasingly crowded environment. Especially the convergence behaviour of the various cost functions is of interest to answer the formulated questions about the time needed to reach a first feasible and final, near-optimal solution. Questions about the fulfilment of the COLREGs will be interpreted according to final set of trajectories to which the procedure converges. The experiments are described in the following, sorted by the type of the collision situation, starting with a detailed set-up of each experiment followed by the results. The interpretation of the results is done for each experiment as well as for all experiments at the end of this chapter, where in section 6.5 the interpretation of each question can be found along with a discussion about the fulfilment of all requirements.

6.1 Maritime Simulation Experiments

The evaluated scenarios are selected to cover certain standard situations on sea. All experiments were conducted on an Intel i5-2520M Quadcore with 2.5 GHz and 4GB RAM. The *Path Planner*and Maritime n-Ship Trajectory Negotiation System for Collision Avoidance (ManTra) performed on the same machine and used the defined *IMC*-Interface for communication. Even though live experiments were planned and the whole framework is developed for a decentralised use, prolonged technical difficulties with core functionalities of the test-bed at Lake Konstanz made live tests impossible. Hence it was decided to base the evaluation on a simulation. As simulation environment the Boat Operating System (BOS) was used, which was developed by Michael Blaich and provided during the cooperation with the HTWG-Konstanz. It is based on the Uniform Navigational Environment Documentation (DUNE) which was created as part of an overarching framework from the *Underwater Systems and Technology Laboratory* from the university of Porto, which also entails the IMC. It enables to configure scenarios in which several vessels can be simulated, traced in a GUI and especially the trajectories, generated by the path planner, can be viewed.

6.1.1 STANDARD SITUATIONS

Based on century old experience on typical encounters on sea, situations were defined in the COLREGs as base for the rules for behaviour. In the following the negotiation approach will be evaluated in the most prevalent situations on sea. In Cockcroft and Lameijer (2003), Rule 14 defines the situation *"...When two power-driven vessel are meeting on reciprocal or nearly reciprocal courses ..."* as a *Head-On* Situation. The required behaviour to avoid a collision is for both ships to alter their course to starboard. Rule 15 defines a crossing situation and urges to *"When two power-driven vessels are crossing so as to involve risk of collision, the vessel which has the other on her own star-board side shall keep out of the way and shall, if the circumstances of the case admit, avoid crossing ahead of the other vessel."*. A crossing situation is defined in the following for three and five ships and a Head-On situation for two.

6.1.2 MEASURED PERFORMANCE INDICATORS

The questions about the quality of the approach can be answered by tracing the metrics, defined in in section 4.8.3 as absolute values as well as their convergence properties over 30 negotiation rounds, which is on average the maximum number needed to reach a convergent behaviour. Furthermore, because of the assumed limited bandwidth more than 30 rounds are considered to costly. The round-counter in all figures starts at 3 because round 1 and 2 are counted are the rounds of the initial exchange of desired trajectories and the pre-round to reach a first solution. The following cost functions will be traced:

INDIVIDUAL SHIP VEHICLE COST The individual ship vehicle costs are a measure which show as a function how costly the path planner regards its found trajectories. They are one of the most important performance measure metrics of a trajectory and will be in the diagrams often abbreviated to *vehicle costs*. Due to the used path-planner, the ship vehicle cost function is based on the length of the trajectories, the number of needed turns and the probability of evading all current and future obstacles. Tracing the function over all negotiation rounds shows if the costs converge to a stable solution and how fast they converge. In situations where convergence is observed on two or more sub-sequences, which seem divergent in the beginning, this shows the existence of certain candidate funnels for optimal solutions in the state space. The difference in their value of the cost function in between ships shows how independent ships converge towards a global solution where trajectories are increasingly equal in their quality. Since in the following, the lengths of the desired trajectories are all the same and all other penalising factors are also equal, the convergence towards the same value on all ships is one of the best indicators of a decentralised optimisation towards a common fair solution. Apart from the ship vehicle cost function other results are also plotted against their global mean to interpret the global convergence. To this end the length of a trajectory is also measured independently from the path planner and used in the evaluation even though it is not on its own a factor in the final augmented cost function.

Constraint Penalty Cost Another performance measure metric are the constraint penalty costs. Ships which follow planned trajectories should stay at all times out of a safety radius around any other ship, as described in section 4.4.2. The course of the penalty cost function is traced as it gives insight into constraint violation over the course of the process. The absolute value is usually very high since the value γ was chosen to be 2 in all experiments, leading to a quadratical rise in costs if any two ships got closer than the chosen radius R. At the same time the function may seem chaotic since its value is only calculated once the safe distance is too small and is 0 at all other times. The convergence of this cost function has several implications. Larger values towards the end of the process show, in combination with the mean distance from the CPA, trajectories which are planned narrower, usually leading to lesser individual ship cost. Larger values towards the beginning which decrease suggest the process finds more solutions with acceptable distances.

Minimum Distance from CPA In order to interpret information from all other costs, the mean distance from the closest point of approach from the nearest trajectory to the own trajectory is traced, since it gives information on the smallest distance two ships have at any point in time. Its final value, if it converges, shows if the desired distance can be kept. The minimum distance is calculated based on the first metric which measures the course of all trajectories over all rounds.

Further indicators The execution time for the process is also measured as required in the second metric, although it is considered only a guideline. The system was designed and implemented in Java with requirements in mind which help a researcher to investigate the convergence process and reimplement parts often to test alternative approaches aided by an object-oriented approach. A fixed implementation as an embedded system on dedicated hardware could speed up the process as well as distributed execution on more than one computer. Other factors which may slow the execution were not investigated as i.e. a smaller bandwidth.

6.2 5-SHIP CROSSING

In figure 6.1a five ships are in a complicated crossing situation where every decision of every ship limits strongly the space of possible alternative trajectories of all other ships. The ships are trying to reach their destination on straight line trajectories and the solution at round 7 is shown, where almost all ships deviate already, to some degree even further than is necessary in the final solution after approximately 10-12 rounds.

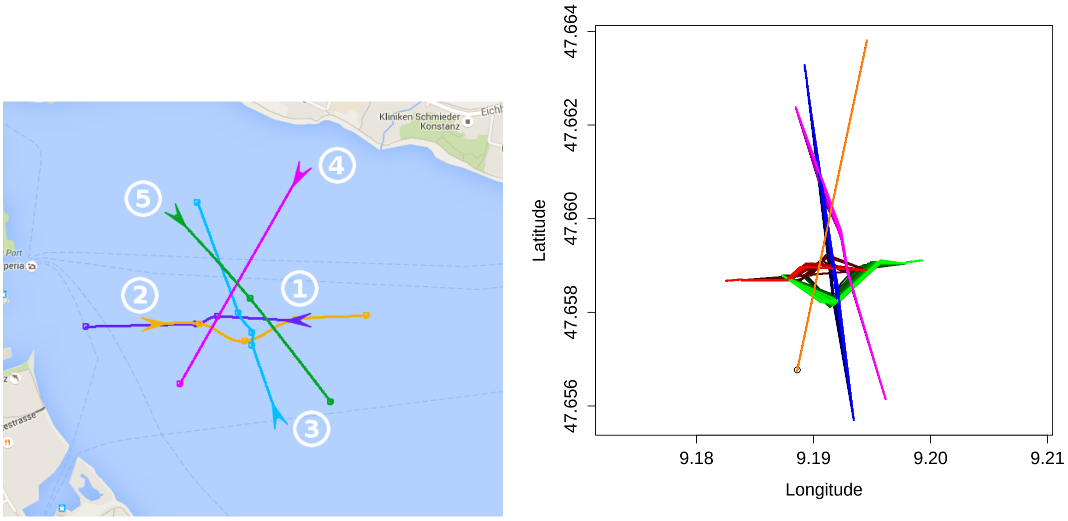
6.2.1 OFFSET

In the 5-Ship Scenario the ships were at the locations shown in table 6.1. From the information given it is possible to calculate the minimum average trajectory length of the desired trajectory because every length of the initial trajectories is chosen to be 900 meters.

	Start Positions		Destinations	
Ship No.	Longitude	Latitude	Meter N/S	Meter W/E
1	47.65890	9.19451	-25.20	-895.64
2	47.65879	9.18730	35.20	895.17
3	47.6557	9.19343	800.0	-300.0
4	47.6638	9.19455	-700.0	-400.0
5	47.66237	9.18848	-600.0	500.0

Table 6.1: Start Locations and Destinations of Five Ships

In the situation described the ships reach an agreement for feasible collision free trajectories within the first 5-10 rounds. It can be seen in figure 6.1b that the actual considered search-space is already quite small. Feasible alternative courses of trajectories are limited to distinct sets. Before the individual cost functions are discussed, the 5-ship scenario is used in section 6.2.2 to illustrate the two defining parts of the cost function.

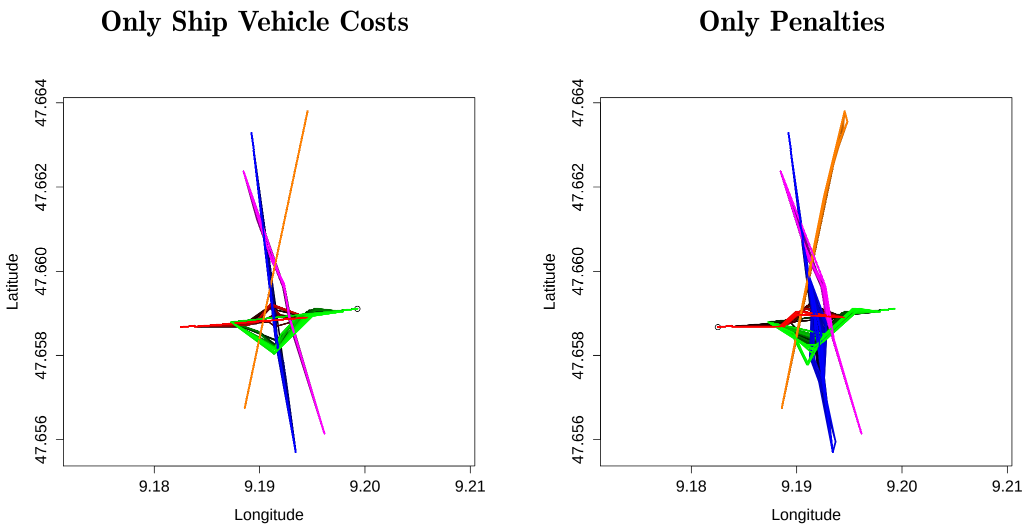


(a) 5-Ship Scenario. The destinations are chosen to provoke a collision in the middle of the area. However, the ships 5 and 4 do not necessarily cross the same point as 1,2 and 3.

(b) Using the traced trajectories from all sets, the convergence behaviour in the situation is illustrated. The coloured lines represent all suggested trajectories during the negotiation process. The first found fade out towards black, while the most current suggestions are coloured with full saturation and brightness. One colour stands for the trajectories of each ship. In this case the ship with the orange trajectory did not have to change its course at all.

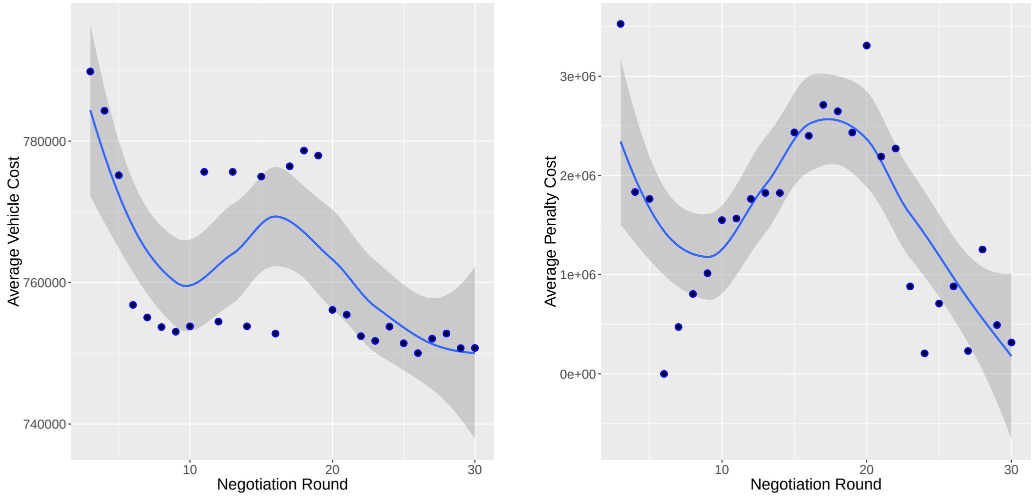
6.2.2 SHIP VEHICLE COST FUNCTION AGAINST CONSTRAINT PENALTY

Special preliminary test runs in the five-ship experiment were conducted, to investigate the influence of the two main factors from the augmented cost function, which structure the search for an optimal solution. In the first run only the ship's vehicle cost function is used in the augmented cost function and in the second only the penalty cost function contributes. In every other experiment the parameter β increases the influence of the ship's vehicle cost function towards the penalty cost function over the course of all rounds. In figure 6.2 the two different optimisation processes are shown side by side. On the left side the penalty costs are kept at 0 which leads to a search where the suggested trajectories change often in between distinct smaller areas, depicted in figure 6.2a. It is remarkable that even though the process converges only after 20-30 rounds, near optimal solutions are found as early as 5-10 rounds. On the right side the ship vehicle cost are kept at 0 which leads to a wider search in the available space. The length of the trajectories in both cases converges slowly in a similar way however at a slightly different offset as can be seen in figures 6.2e and 6.2f. Note that the absolute numerical values from the average vehicle costs are multiplied by β in the final cost function and therefore their offset is on its own not comparable to the penalty costs. Also, since one of each cost function is kept at 0 it is not possible to compare the same costs in both experiments therefore each time only one cost function is shown.



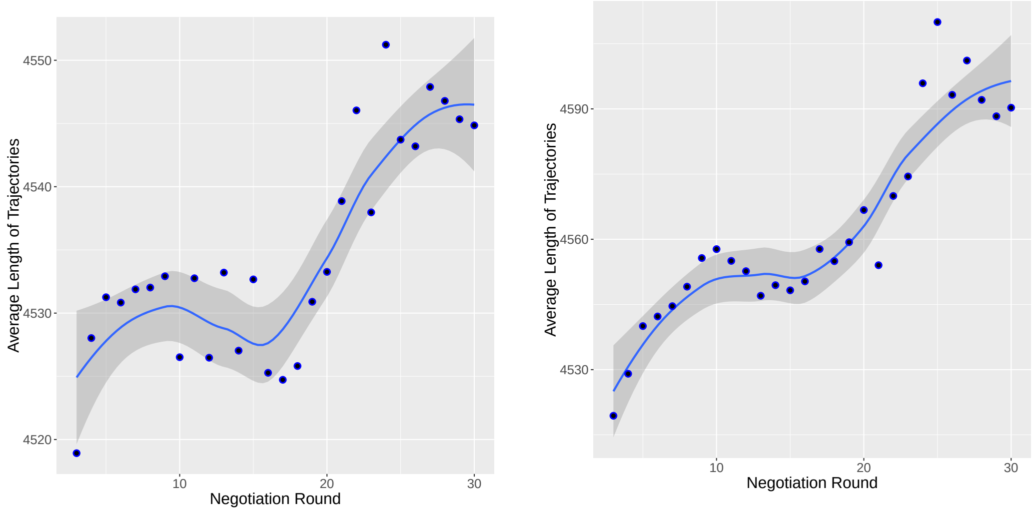
(a) Search close to candidate solutions

(b) Broader search in the state space



(c) Distinct candidate solutions

(d) Continuous search and improvement



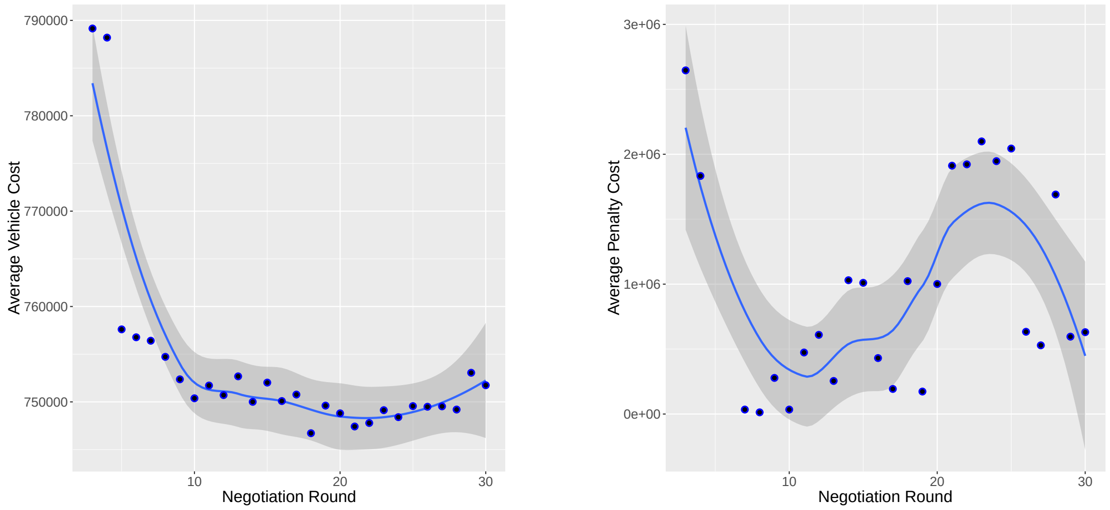
(e) Average length slowly converging

(f) Offset changed but comparable to figure 6.2e

Figure 6.2: The two possible extremes of the cost function

6.2.3 INDIVIDUAL SHIP COST

In the normal 5-ship experiment with the cost functions transitioning in the designed way, the ship vehicle cost function converges, as can be seen in figure 6.3a. In the beginning, the average ship vehicle cost fall quickly within 5 rounds by approx. 4.5 % relative to the absolute maximum and then converge towards 95% of the first exchanged solution.

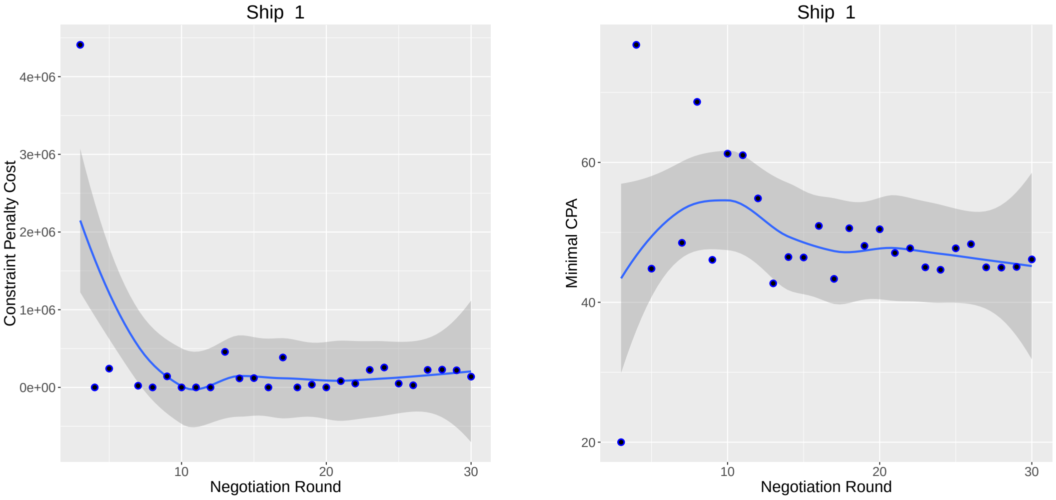


(a) Average Individual Ship Cost **(b)** Average Penalty Cost

Figure 6.3: Average cost functions over all ships in the situation

6.2.4 CONSTRAINT PENALTY COST

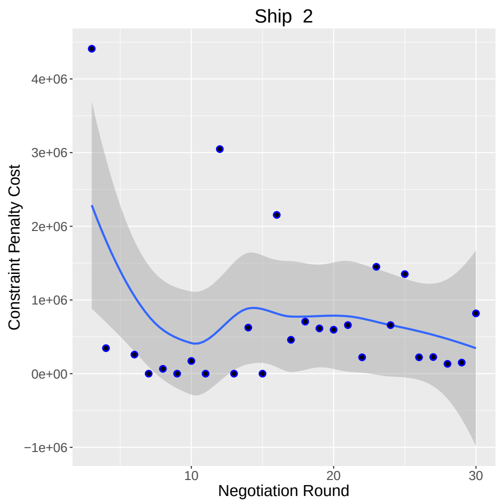
The Penalty Cost Function is non-zero if any target ship on its trajectory violates the chosen safety radius of 50 meters around the own ship. The average constraint violation as seen in 6.3b refers to the penalty cost incurred by all ships. It shows that in the beginning constraints are violated but this is not the case for any ship in round 7, 8 and 10 because the penalty costs are always positive and 0 on average for all ships in those rounds. Later in the process, as the trajectories are planned tighter, two effects contribute to the rise and later convergence which can be seen separately in figure 6.4a and 6.4b. In figure 6.4a, the penalty cost function from ship 1 converges after the first round towards values in between 0 and 500 000 which in combination with a high β value of 10 000 000 contributes only marginally towards the result of the augmented cost function. This implies only minor violations of safety constraints which over the course of 30 negotiation rounds converge towards 0. This can also be seen in figure 6.4b by the absolute value and convergence of the minimal distance to the CPA which for all trajectories converges close to the desired distance of 50 meter. In contrast, trajectory planning for ship 3 shows a less convergent behaviour of the constraint penalty costs, as can be seen in figure 6.6



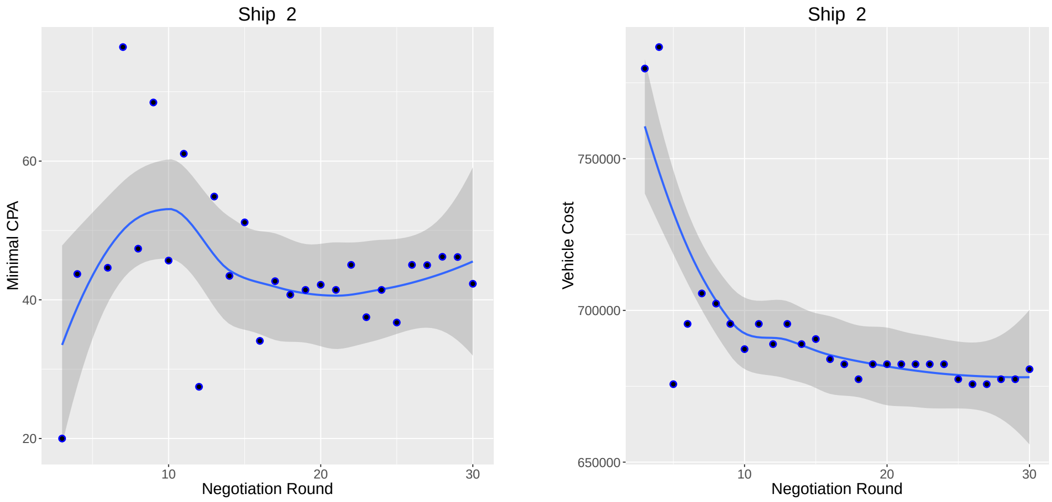
(a) Constraint penalty costs decline and converge **(b)** Course of the minimal distance of the own trajectory to any other trajectory.

Figure 6.4: Cost Effects of Safety-Radius Violation for Ship 1

The path-planner generates trajectories for ship 2 in the beginning which avoid a collision, but infringe on the safety area of 50 meters. In figure 6.5a the penalty costs converge towards 0 which shows a steady improvement. It can be seen in figure 6.5b that after 15 rounds all generated trajectories are at least 30 meters apart. The ship vehicle cost improve steady as seen in figure 6.5c which, in combination with the average results in figure 6.3, leads to the final conclusion that safe and near optimal trajectories can be achieved in as early as 10 rounds, since exactly in round 10 all three indicators show satisfying results.



(a) Constraint penalty costs start high but converge towards 0

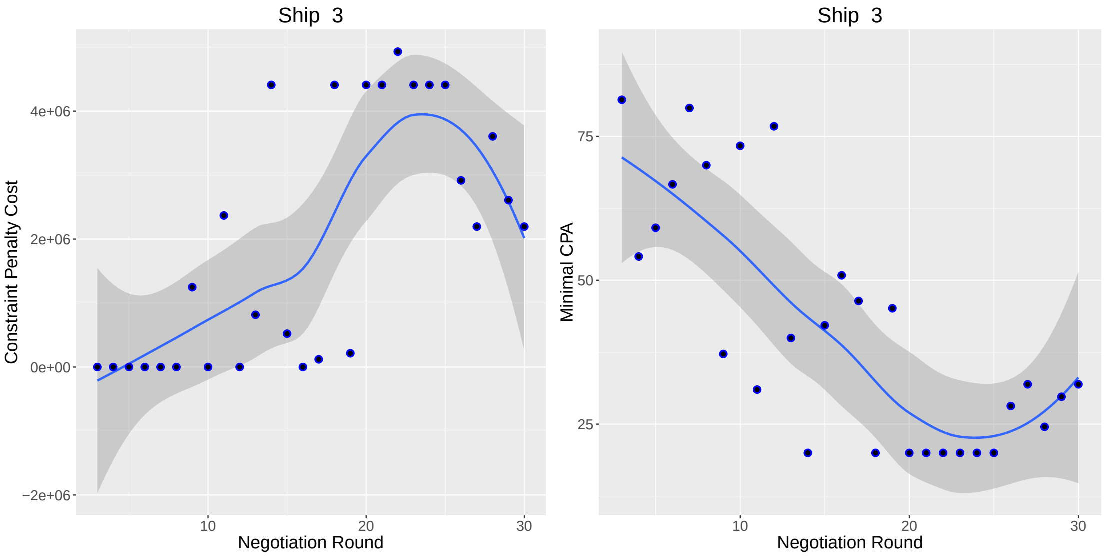


(b) Course of the minimal distance to the CPA of the own trajectory to any other trajectory.

(c) Ship Cost for Ship 2 converging rapidly.

Figure 6.5: Cost Effects of safety-radius violation for ship 2

Ship 3 is in the beginning not affected by planned trajectories of other ships as these do not cross its path too close. However, this changes during the optimisation and negotiation which forces ship 3 to accept less direct trajectories than its initial proposal. The costs start at 0 indicating no constraint violation at all but rise sharply as more own trajectories are generated which are too close to other trajectories. The desired safety radius of 50 meters can not be achieved in 30 negotiation rounds, however the results shown in figure 6.6b suggest more rounds would improve on the minimal distance of approximately 18 meters.



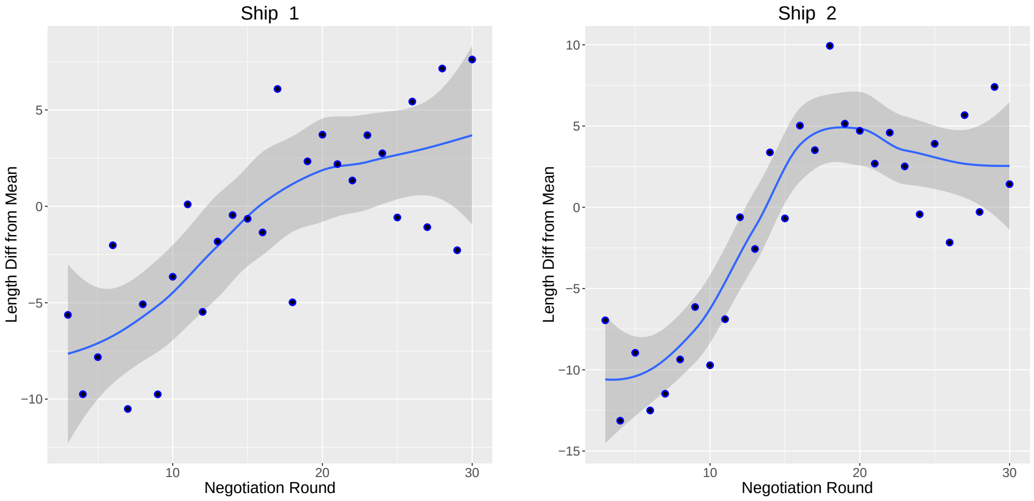
(a) Constraint penalty costs start low but increase towards the end of the process

(b) Course of the minimal distance of the own trajectory to any other trajectory.

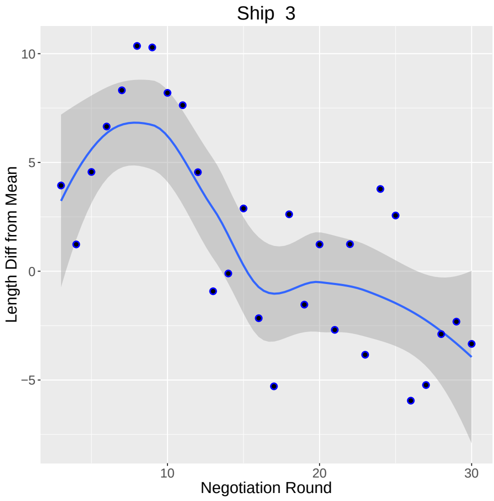
Figure 6.6: Cost Effects of Safety-Radius Violation for Ship 3

6.2.5 Cost Distribution

Of all ships in the situation the costs are measured and the distance to mean values is shown in figure 6.7. It is used to trace if the experiment converges towards fair costs, which would show as a convergence towards 0 for all mutual length differences in the data. The results show a behaviour where shorter trajectories for some ships are necessarily detours for others in the complex situation. It must be noted that the final values must not necessarily converge to 0 since the length is not the only contributing factor to the cost function.

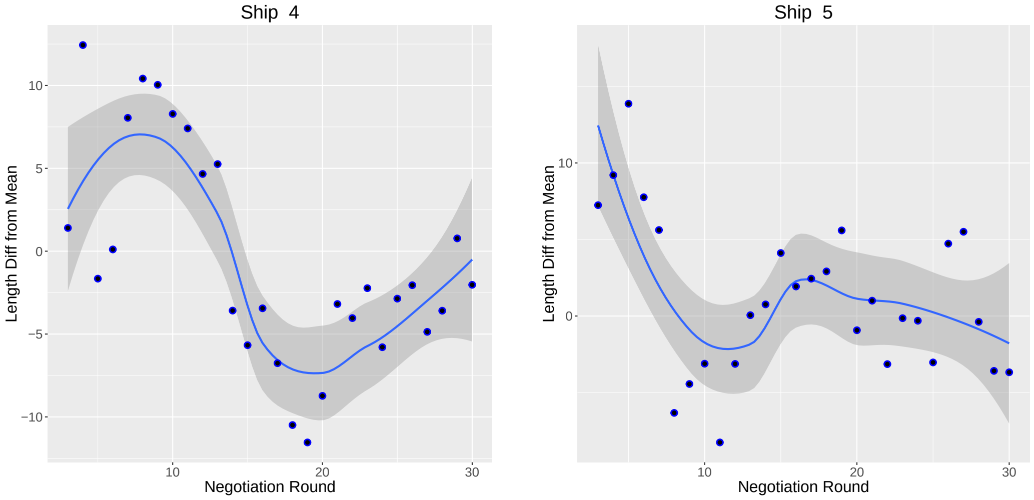


(a) Length-distance to Mean Length **(b)** Length-distance to Mean Length



(c) Length-distance to Mean Length

As can be seen in figures 6.7a to 6.7e the difference of the individual trajectory length of any ship, compared to the mean length varies around ±10 meters or approx. 11% compared to the initial trajectory. Convergence towards 0 can be observed over 30 rounds. An interesting point is that after 10 to 15 rounds, all trajectory lengths are very close to 0 which shows a fair solution, regarding the trajectory length, is reached, however a more efficient solution was found at round 30.



(d) Length-distance to Mean Length **(e)** Length-distance to Mean Length

Figure 6.7: Course of Mean Costs and Deviation from Global Mean

6.3 3-Ship Crossing

A simpler simulation scenario with three ships was evaluated to investigate the convergence behaviour further and answer the research questions in section 4.8.2. The following scenario is a Head-On situation, which is disturbed by a third ship on a crossing course. The two ships are therefore not free in their choice of evasive trajectories and the influence on the third ship, when trajectories are planned in its path, can be investigated.

6.3.1 Offset

In the 3-Ship Scenario the ships were at the locations, and start with their short term destinations as shown in table 6.2. In a crossing scenario the three ships start with speed and direction chosen to lead to a collision, as shown in figure 6.8a.

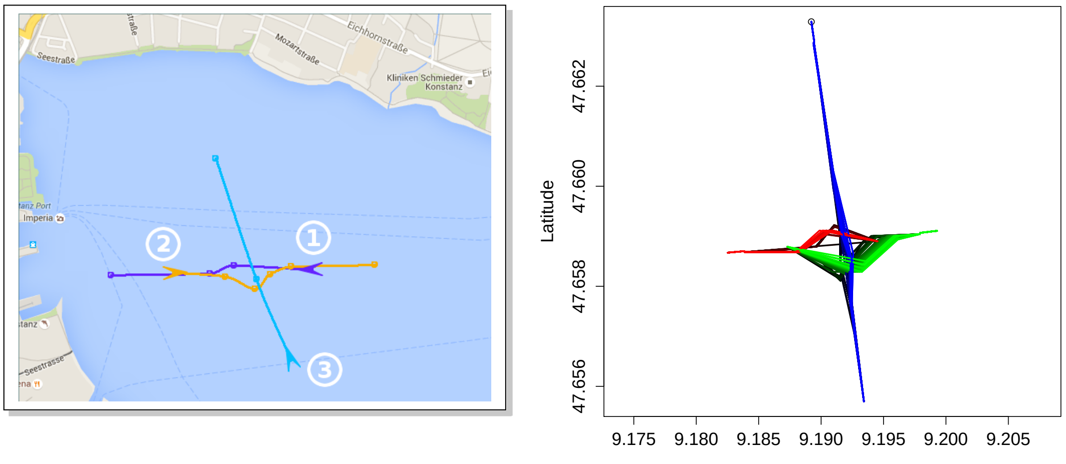
	Start Positions		Destinations	
Ship No.	Longitude	Latitude	Meter N/S	Meter W/E
1	47.65890	9.19451	-25.20	-895.64
2	47.65879	9.18730	35.20	895.17
3	47.6557	9.19343	800.0	-300.0

Table 6.2: Start locations and destination of the three ships

Ship 1 and 2 try to find a trajectory from the east to the west side of the area and vice versa. According to the COLREG the appropriate behaviour is to cross with each port side to one another. This was successful in all scenarios.

6.3.2 Individual Ship and Constraint Penalty Cost

The vehicle cost decline fast and converge after 20 rounds towards 90.63 % of the maximum cost. This shows an improvement over the first found solution of 9.37 %. The average penalty cost, as shown in figure 6.9, converges slowly over all 30 rounds towards 0 which indicates that no infringement of safety ranges is present in the final solutions. Convergence

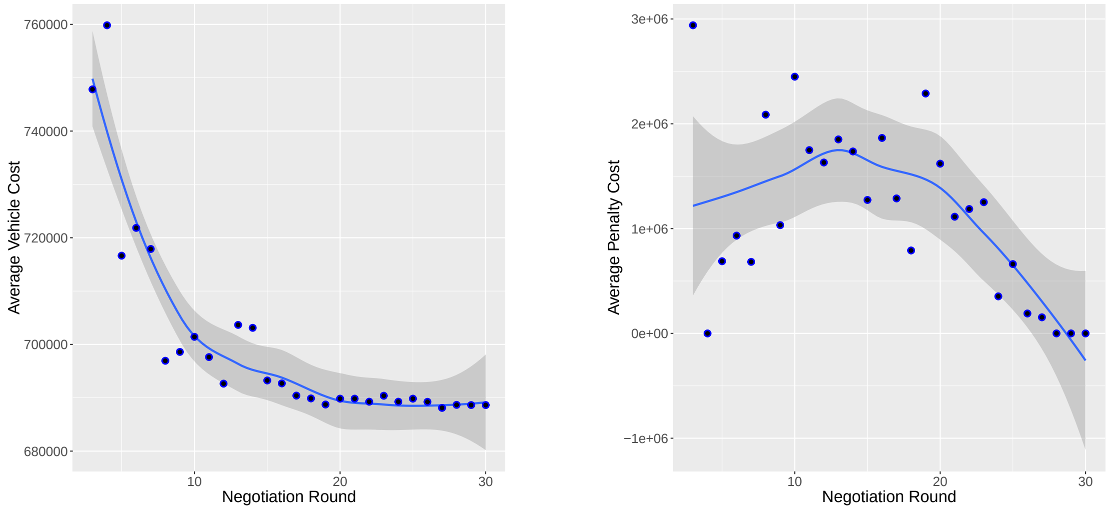


(a) The situation on Lake Konstanz. A trajectory search of 3 ships

(b) Every coloured line shows all generated and favoured trajectories which were broadcasted.

Figure 6.8: Generated trajectories in a 3-Ship Situation in the Boat Operating System (BOS) - Framework from the HTWG-Konstanz

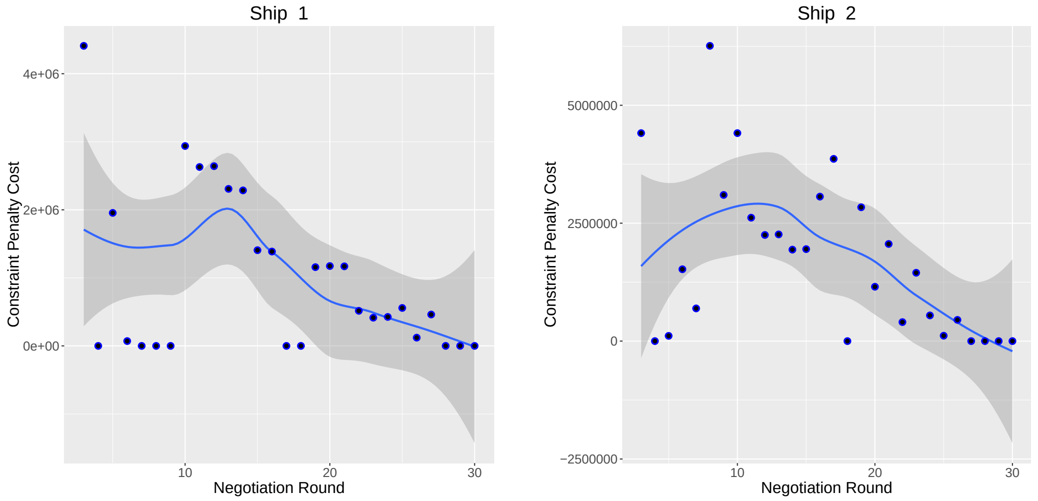
of the penalty cost vary between ships, as is shown in detail in figure 6.10. Length differences as for the 5-ship scenario are measured and the distance to the mean is shown in figure 6.11.



(a) Average Individual Ship Cost

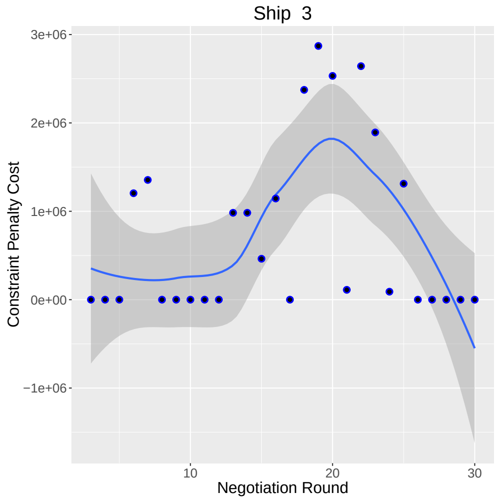
(b) Average Penalty Cost

Figure 6.9: Average Cost Functions of 3 ships ships



(a) Penalty cost converging steady

(b) Penalty cost of the opposite ship



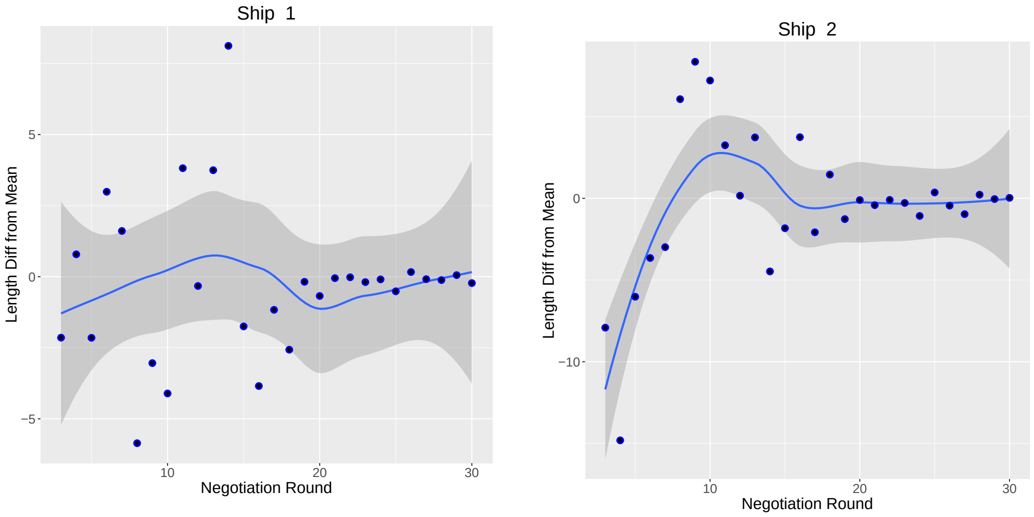
(c) The crossing ship's cost avoids a penalty most of the time

The three constraint penalty cost functions for three ships show that the two ships in a head-on situation converge against a situation in which no constraint penalty costs are incurred. At distinct negotiation rounds their solution is planned in the trajectory of ship 3 whose penalty cost rise as a consequence. In the end a solution is found for all three ships which does not infringe on any safety radius. This can be seen also in figures 6.10d and 6.10e where the minimum distance to the CPA converges towards the chosen safety radius.

(d) Minimum Distance to CPA Converging Towards 50 meters

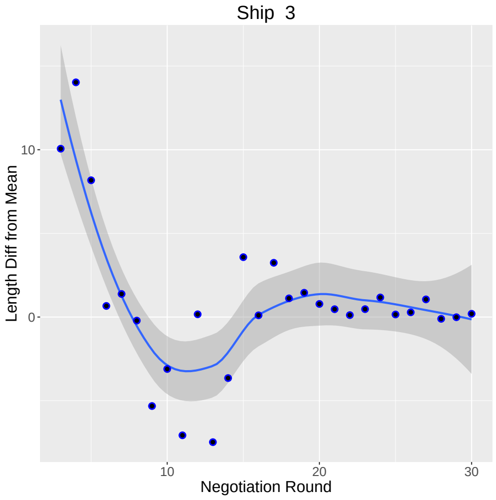
(e) Converging Distance to CPA

Figure 6.10: Penalty Costs Of 3 Ships



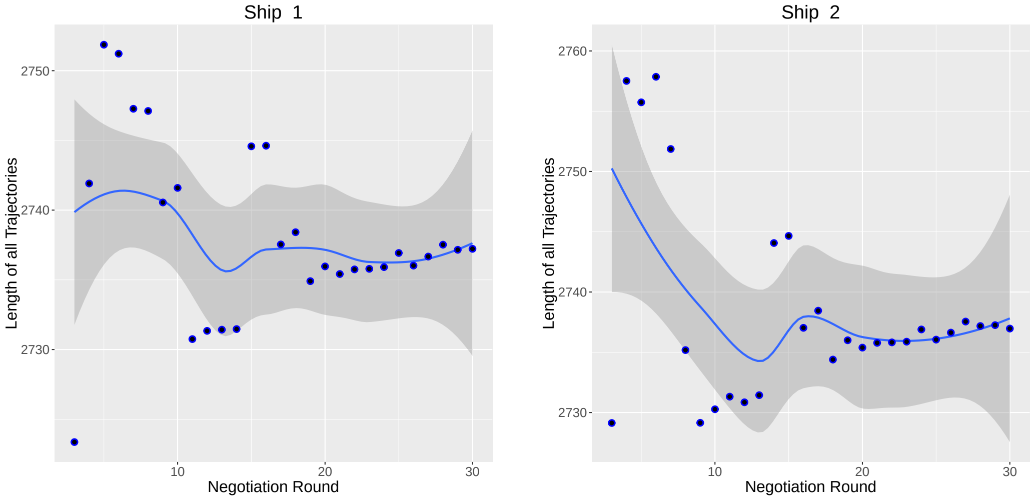
(a) Mean Absolute Costs of a Set of Trajectories

(b) Distance of Mean Quality



(c) Distance of Mean Quality

In figures 6.11a to 6.11c the converging difference to the mean trajectory length is shown. The difference indicates that after round 18 a final solution is reached where the length of each trajectory is approximately the same for each ship. Figures 6.11d and 6.11e show the length of all resulting trajectories for ship 1 and 2 which converge at 101.4 % ± 0.1 % of the length of all desired trajectories of 2700 meter. Shown are the added lengths of all trajectories for sets which are suggested by two different ships because they differ depending on whether they are accepted in the negotiation or not.



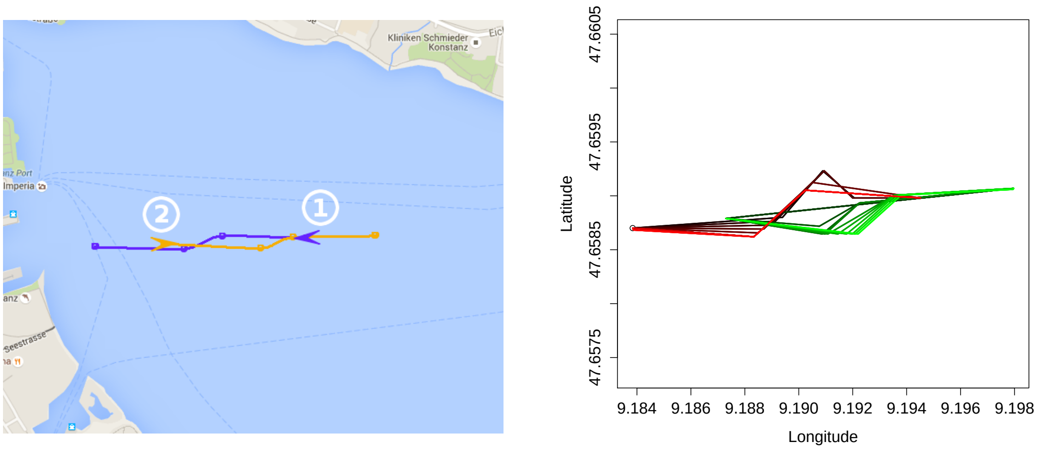
(d) Length of all trajectories converge towards 2737 m

(e) Convergence trend is the same for all three ships.

Figure 6.11: Convergence towards the global average trajectory length.

6.4 2-SHIP HEAD-ON

In figure 6.12 two ships are in a Head-On situation where their desired trajectories lie exactly opposite to each other. A two-ship situation is on the one hand a good way to see the continuous improvement by repeated queries to the path planner, on the other hand in this special situation all ships receive only one set of trajectories to choose from. It is therefore not possible to order the solution-space according to the vehicle costs or penalty costs with the negotiation component. The results are shown for the sake of completeness and to illustrate the behaviour of the path planner, when repeatedly presented with sets of planned trajectories. There is no significant change in the behaviour after round 15, therefore the experiments stop at this round.



(a) COLREG Compliant Evasion

(b) The course of the found trajectories over all negotiation rounds

Figure 6.12: 2-Ship Head-One Scenario

6.4.1 OFFSET

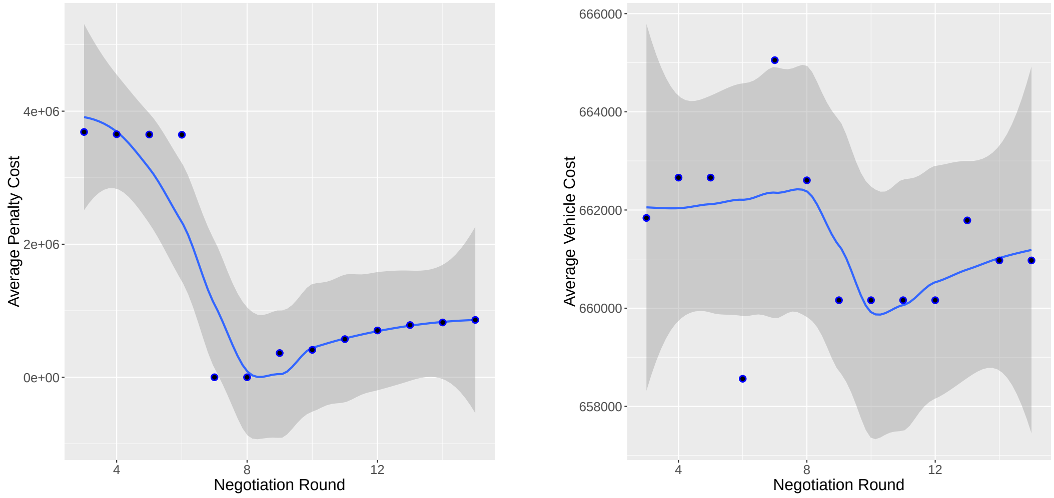
In the 2-Ship Scenario the ships were at the following locations:

	Start Positions		Destinations	
Ship No.	Longitude	Latitude	Meter N/S	Meter W/E
1	47.65898	9.19451	35.20	-895.64
2	47.65879	9.18730	35.20	895.17

Table 6.3: Start locations and destinations of two ships

6.4.2 Cost Functions

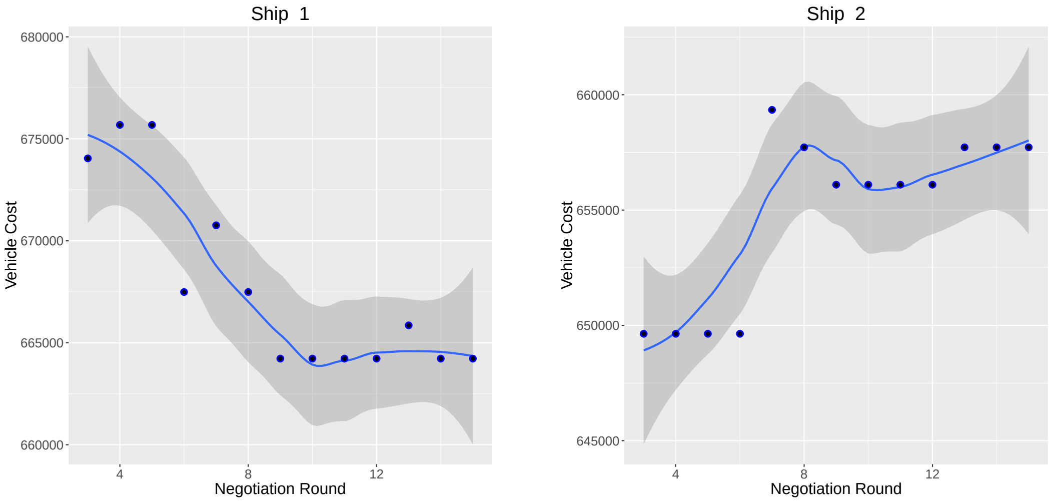
The ship vehicle cost funtion and the penalty cost function are both close to their final value after 15 rounds. After 8 rounds both trajectories are planned with enough distance to avoid any penalty costs as can be seen in figure 6.13a, later in the process however penalty costs rise marginally as the trajectories are planned closer to each other. The vehicle costs converge on average over two ships a bit chaotic. The reason can be seen in figure 6.14, where both vehicle cost functions of both ships converge from different directions. Ship 2 finds a very favourable trajectory first but has to deviate from it while ship 1, which first plans a bigger detour as response to the initial desired straight line trajectory, improves its trajectories.



(a) Penalty cost function rising later in the process

(b) Vehicle cost function converging a bit chaotic

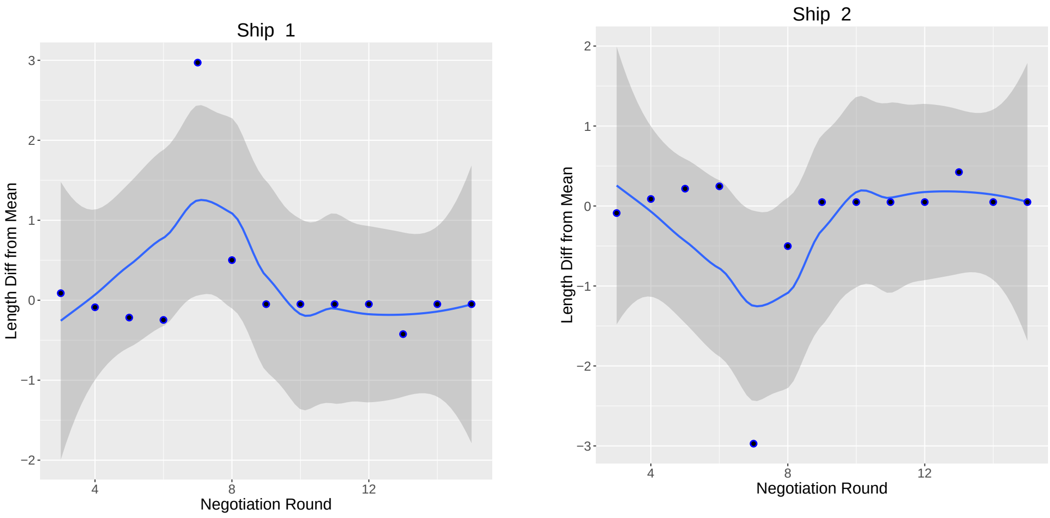
Figure 6.13: Average penalty and ship vehicle cost functions



(a) Vehicle costs decreasing as trajectory-quality improves

(b) Vehicle costs rising as trajectory-quality degrades

Figure 6.14: Ship Vehicle Cost Function Comparison



(a) Length is almost always equal

(b) Deviation in round 7 shows linked planning

Figure 6.15: Difference in length of the trajectories

6.5 OVERALL RESULTS AND DISCUSSION

The experiments showed results which where as expected in chapter 4. Interdependent constraint satisfaction could be observed when trajectories were planned near to other trajectories and convergence of all cost functions could be shown. The emergence of sub-sequences in the cost functions illustrate a search near several candidate funnels of sub-optimal solutions and the optimal solution in the solution space. The stated questions can be answered as follows:

Questions	Simulation Result
Are planned trajectories too close? How close is the closest point of approach?	A desired final distance of 50 meter is chosen for the ships simulated in the experiments. This distance should be kept at all times, though is not a hard limit. The questions can be answered by observing the minimal distance to their CPA of two trajectories, which converges towards a minimum of $\approx 35 \pm 1$ meter with ship 3 in the five ship scenario, figure 6.6. It can be observed that during the negotiation an intermediate solution plans two trajectories as close next to each other as $\approx 20 \pm 1$ meters In conclusion it could be observed that in no scenario the final value after 15-30 rounds was smaller than 40 meters, apart from the 5-ship scenario where this could be improved by choosing a higher desired distance
How many turns are needed?	The average number of all waypoints for all ships is considered the number of turns of all ships, even though in reality, due to the conversion to Bézier-Splines the number of turns may be slightly smaller if two very close waypoints are reached with one large turn. The three scenarios have in their final solution sets 20, 12 and 10 waypoints for 5, 3 and 2 ships which leads to an average number of waypoints of 4, 4 and 5 respectively and this to approx. 2-3 turns because the first and the last waypoints are the start and destination of a trajectory.
How many changes in speed are needed?	All experiments were successfully conducted with constant speed therefore no changes were necessary.

Questions	Simulation Result
How equal/fair are all trajectories?	The deviation of the average ship vehicle costs and augmented costs for all ships are listed at round 30 as a measure of how fair the final trajectory sets became. Since in some experiments all final trajectories infringe the desired safety radius slightly, a huge numerical cost penalty is given which makes direct comparison to other experiments difficult where that is not the case. For that reason all deviations are shown in percent:

Relative ship vehicle cost difference:

	Ship 1	Ship 2	Ship 3	Ship 4	Ship 5
2-Ships	-0.492 %	0.492 %	–	–	–
3-Ships	6.580 %	-4.734 %	-1.845 %	–	–
5-Ships	12.400 %	10.885 %	0.835 %	-10.476 %	-13.644 %

Relative augmented cost difference:

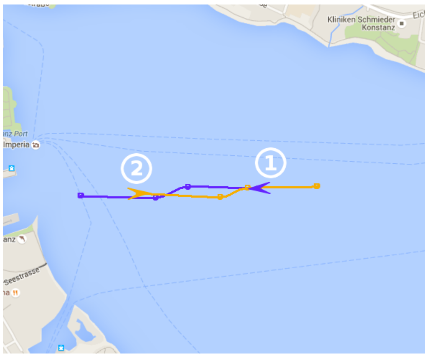
	Ship 1	Ship 2	Ship 3	Ship 4	Ship 5
2-Ships	-0.111 %	-0.111 %	–	–	–
3-Ships	-1.976 %	1.507 %	0.469 %	–	–
5-Ships	-7.548 %	-3.459 %	-0.985 %	3.572 %	8.421 %

When accounting for the penalty costs in the augmented cost function the best and the worst solution differ in the most difficult case of 5-ships by 15.969 % which is considered not optimal but satisfactory, considering that deviation from the agreed solution does not guarantee a collision free trajectory. In the 3-ship scenario the maximum difference between augmented costs is only 2.445 %. Overall it can be observed that differences in the quality of the trajectories are significantly smaller in the 2-ship scenario and rise moderately for three and five ships. Due to the unbounded ship vehicle cost function, when an arbitrary path planner would be used, the optimisation procedure can not theoretically be shown to converge to a Nash Bargaining Solution because the proof depends on a strictly decreasing penalty function (Waslander, 2004). Nevertheless, convergence towards a solution with equal costs, with the final deviations just stated can be observed.

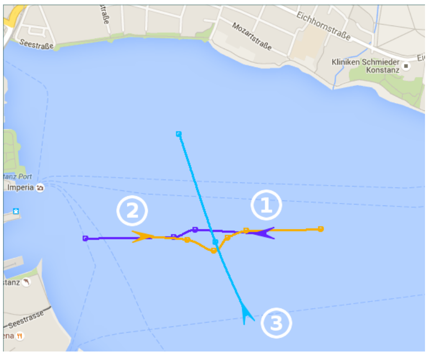
Questions	Simulation Result
Are all collision regulations observed?	RULE 5, 6 & 7 - LOOK-OUT, SAFE SPEED AND RISK OF COLLISION The system can not replace a visual look out, but enhance the situation awareness due to the exchange of intentions and aid as a system for observation of detected objects. At the end of the process when a feasible set of trajectories to be followed with a chosen speed is found, this speed is safe enough to pursue the trajectories since they were planned based on the physical model of the ship. Additional factors as fog or bad weather are not accounted for.

RULE 8 - ACTION TO AVOID COLLISION

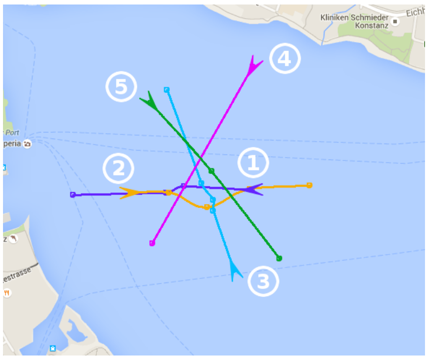
The three situations are revisited to argue the fulfilment of rule 8. The ships cross at the port side of each other, their action is taken as early as possible and with an apparent course. Passing at a safe distance is possible.



Ship 1 and 2 cross again in a correct way and ship 2 crosses behind of ship 3 while ship 3 crosses behind of ship 1. Courses are apparent and ship 3 does not change its course at all.



Even though the situation is getting more difficult ship 1 and 2 cross as required and 1,2 and 3 pass behind each other. From figure 6.5b, the minimum distance at the CPAs, it can be seen that ship 2 does not come closer to any ship than 40 meters thus passing at a safe distance with all other ships.



Questions	Simulation Result
When will the procedure not work? How does the process scale with the number of ships?	The procedure worked in all scenarios in a satisfying time. However, it remains unknown what may happen for groups of ships, larger than five. The main factor which is used to compare all the scenarios is the average ship vehicle cost because it is the cost which is assigned by the path planner to the trajectories and therefore considered as an external performance measure which uses information more relevant to the individual ship. It can be observed that in all three scenarios the ship vehicle cost function converges after 10 rounds even though other measures as the trajectory length or the penalty costs converge only after 20 rounds in the five ship scenario. It seems to be the case that the approach scales well, however any limit or behaviour beyond five ships can not be anticipated with certainty and further experiments are needed.
How long does it take to generate collision free trajectories?	After the sequential round each ship has a collision free trajectory even though the first trajectory exchange may not produce efficient or fair trajectories yet. The pre-round scales almost linear with the number of ships since for each additional ship one information exchange and one path planning process is added in an environment where for each of the n-ships there are $n - 1$ trajectories already present as obstacles. In the most difficult situation for 5 ships the path planner could be queried and the communication could be performed in under one second which leads to a final time of below five seconds for all ships to reach an initial conclusion. This time is however heavily dependent on the bandwidth of the communication medium.

<table>
<tr><th>Questions</th><th>Simulation Result</th></tr>
<tr><td>How long does it take to achieve the best solution?
How fast does the algorithm produce a final/feasible outcome?
How is the convergence behaviour of the trajectories?</td><td>In all experiments the constraint penalty cost and the ship vehicle cost converge after 30 rounds, even though in some settings the very same quality could be observed at 10 rounds. Analogous to the calculation of the worst-case predictable fail time of the algorithm in section 4.7.2 the typical runtimes which can be expected are shown here. If the path planner needs a runtime of s_{path} to plan one trajectory, this leads to a minimum runtime s_{min} of the approach of

$$s_{\text{min}} = n \cdot (n-1) \cdot r \cdot s_{\text{path}} \tag{6.1}$$

because n agents plan a new trajectory in $n-1$ trajectory sets in r rounds. In the following typical times of the approach after 10 and 30 rounds are listed along with their minimum path planning time after 30 rounds ($r = 30$), based on a typical observed s_{path} of 50 ms, 75 ms and 100 ms for 2,3 and 5 ships resp. In addition, in the last column the actual measured time of the first feasible solution s_{feasible} is shown.:

<table>
<tr><th></th><th>s_{min}</th><th>$s_{\text{avg-10}}$</th><th>$s_{\text{avg-30}}$</th><th>s_{feasible}</th></tr>
<tr><td>2 ships</td><td>9 s</td><td>7 s</td><td>20 s</td><td>0.9 s</td></tr>
<tr><td>3 ships</td><td>13.5 s</td><td>24 s</td><td>69 s</td><td>1.7 s</td></tr>
<tr><td>5 ships</td><td>60 s</td><td>50 s</td><td>160 s</td><td>2.2 s</td></tr>
</table>
</td></tr>
</table>

These times are based on experiments on the described architecture in section 6.1. The negotiation component needs in between two and four times the runtime of the path planner, however both times could be further improved by decentralised execution. The first feasible solution is reached at all times in less than 3 seconds after all ship positions are known, which is considered satisfactory because, even though this solution is not expected to be fair or optimal, it supplies a first solution fast enough and its improvement can be stopped at any time to pursue the found trajectories in a matter of emergency. It can be seen that the ship vehicle cost function usually converges logarithmic in between 5-15 rounds towards a minimum value. In the special case of 5-ships, where only the ship vehicle cost function is part of the augmented cost function as shown in figure 6.2c, the function switches in between two sub-sequences of values, but converges in both. This is an expression of an alternating search near two distinct candidate solutions. The constraint penalty cost function converges in a more erratic way since it is only non-zero in certain cases and is dependent on the position of all nearest trajectories. Since in the beginning all trajectories change their course over a large area, it converges slowly in the beginning and towards the end. When trajectories are planned tighter, i.e. in the 5-ship scenario, it ascents temporarily as a result.

Questions	Simulation Result
How close in front of the ship can a trajectory start?	As described in section 4.7.4, the time needed for the execution of the algorithm determines how far ahead in the direction of the ship the trajectory could be set to start. If the first feasible solution after 3 seconds would be implemented, planning for a ship with a speed of 16 knots could start as early as $25.5\,m = 8.5\,\frac{m}{s} \cdot 3\,s$. The worst case of the 5-ship scenario with the current, not optimised implementation would need a distance of $1360\,m = 8.5\,\frac{m}{s} \cdot 160\,s$ if all 30 rounds should be planned. and still $425\,m$ for 10 rounds. This might not be impossible and it confines the available state space further, an effect which was not simulated in the experiments. A solution could be to accept and start following the feasible solution and try to plan an optimised solution from a new position. This was not investigated in this work and further research is necessary.

Questions	Simulation Result
Does the trajectory set reach the optimal solution?	The original criteria for converging to a solution is that the solution converges into an ϵ-environment of an optimal solution. In the original algorithm by Waslander this is ensured by checking if two successive solutions x^{i-1} and x^{i} differ in their course for an agent j less than a defined ϵ: $\|\|x^{i,j} - x^{i-1,j}\|\| < \epsilon$. His procedure is thereby based on two assumptions, one that the process converges towards the optimal solution and second that is does so with a certain stepsize in between solutions which strictly decreases and is only less than ϵ near the optimal solution. Due to the use of the path-planner which produces a different convergence behaviour which can not be compared it will be determined if the ship vehicle costs of successive solutions stay within an ϵ-cost-environment. Using this modified definition, convergence could be observed in every experiment usually around 10 rounds or, as the case in figure 6.9a around 20 rounds. This states only general convergence and reaching the optimal solution could only be judge in combination with knowledge about the minimal costs which the path planner assigns to its trajectories. An estimation is however possible since one of the biggest contributions to the cost of the path planner is the length of the trajectory which is i.e. in the 3-ship experiment set to 900 m per trajectory which would lead to a combined length of 2700 m. It can be observed in the results, i.e. for ship 2 in figure 6.11e that the length converges below 2740 m which signifies prolonged trajectories of around 40m or 1.5% in the final solution.

Table 6.12: Questions regarding the process and the outcome

6.6 FULFILMENT OF THE REQUIREMENTS

From the answers to the questions and the detailed evaluation it is possible to judge the degree to which the requirements could be fulfilled.

R_1 (**Safe Speed**) AND R_2 (**Ample Time**) can both be fulfilled in the experiments even though after the first initial solution a near-optimal solution in a 5-ship scenario needs a minimum of 50 seconds which would be too long for situations with very fast ships in close-quarter.

R_3 (**Action to Avoid Collision**) was fulfilled in the observed cases with an appropriate distance and collision avoidance by course alteration only, however the interpretation of *small angles* in rule 8 of the COLREGs is an ambiguous term, not backed by concrete angles so the fulfilment of R_4 (**Apparent Course**) is uncertain, even though R_5 (**Course Alteration Only**) and R_6 (**Appropriate Distance**) are fulfilled.

R_7 (**Collision-Free Confidence**) AND R_8 (**No collision**) for the final solutions could be shown in the limits of the distance parameters σ and R, even though in some rounds the distance could be improved as can be seen in figure 6.6b where the distance of the closest points of approach in the final solution is under the desired value of 50 m.

R_9 (**Common Intention Knowledge**) is achieved after the very first round where ships exchange their intentions, even though they still contain conflicts and during the process a solution is found for n-ships, fulfilling R_10 (**N-Ship**).

R_11 (**Passive Ships**) were not part of the evaluated scenarios because including their behaviour would have to be examined in the same way in all 2,3 and 5 ship scenarios again which was left out of the scope of this work, however in general the path planner is capable of planning around passive ships, which are detected with the described constant speed or constant speed and velocity model and a fast initial solution could be achieved in earlier work, as seen in figure 6.16, where yet the COLREG compliant behaviour was more difficult.

R_12 (**Solution Convergence**) was reached in all scenarios and due to the path planner all trajectories were feasible, fulfilling R_13 (**Feasibility**).

R_14 (**Highest possible quality**) could be achieved with the stated limitations while R_15 (**Fairness of Solution**) was always achieved. Unfortunately it is not possible to be certain that R_16 (**No regrets**) is achieved because it would mean to theoretically proof that every other solution contains at least one collision or has a higher cost, even though the results suggest it when considering the cost distributions and convergence. To be able to show this, the approach could be changed and reimplemented in a deterministic way, i.e. as a mixed integer linear program to calculate the theoretically possible optimal Nash Bargaining Solution, which would increase the computational complexity and is therefore expected to be slower than the current approach. If this requirement can be fulfilled with the used path planner and the special penalty cost function is therefore still an open topic.

R_17 (**Minimal Communication**) is considered to be fulfilled because in a 5-ship scenario as examined, on average 4-5 waypoints are needed leading to a number of 600-750 waypoints to be sent in 30 rounds. Considering double-precision and three fields for each waypoint this leads to a worst case data usage of 144kbit for the waypoints over the course of 30 rounds which would be transmitted with an off the shelf VHF radio with 19.2 kps in under 8 seconds.

R_18 (**Decentralised Method**) The method is decentralised in its design fulfilling the requirement. A thorough evaluation of R_19 (**Implementation Delay**) in a more confined state space has to remain an open topic even though it is implemented as an option in the negotiation component. Due to the predictable fail time, it is decidable if the algorithm can come to a first collision free solution at all in the worst-case, thus fulfillling R_20 (**Decidable**). Using extensive logging to produce files with all values of the cost functions and xml files which show the content of each trajectory set of each round in every experiment, R_21 (**Traceability**) is considered to be fulfilled.

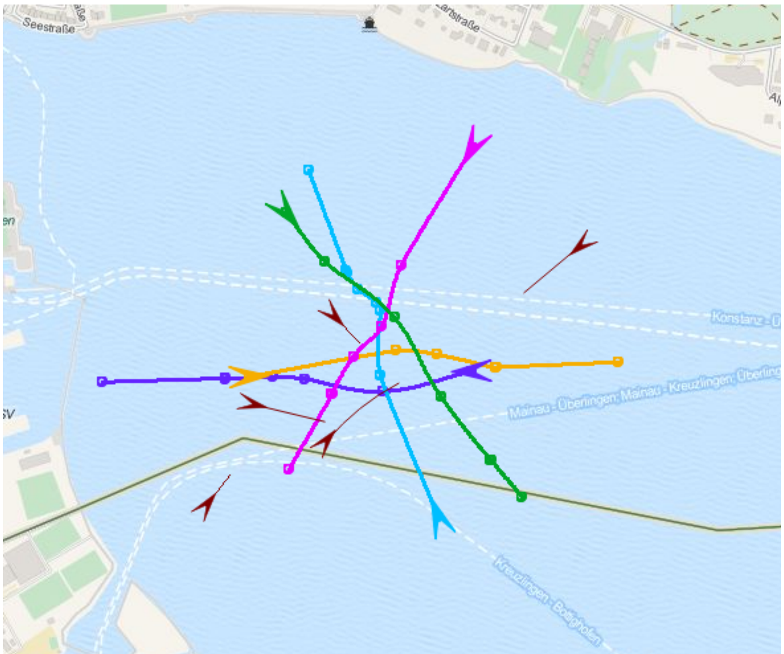


Figure 6.16: 5 active planning ships with 5 passive uncooperative ships

7

Conclusions

The problem of negotiating a number of trajectories for n-ship collision avoidance can be solved with a very good quality for two to three ships and with some limitations for five and more ships. It was possible to implement a Java-agent which interacts with a path-planner to achieve a fast initial collision free trajectory which is then optimised in a satisfactory time over further collision free sets. The procedure can be observed to converge fast towards a near-optimal solution which maximises the utility for each ship and minimises the cost-differences in between all ships. This is achieved with trajectories which minimise the number of turns and conform to the COLREGs in all observed cases. Speed changes were not necessary and the process scales almost linear when planning the first collision free set while the number of ships have a strong impact on the performance of the optimisation with three or more ships. When five ships are in a collision situation, which is chosen in a worst-case manner where all ships try to follow a course through almost the same point in their centre, the system finds after 2-3 seconds the first collision free set and the first near-optimum after 10 rounds/$\approx$ 50 seconds. The process could be accelerated by implementing the negotiation component in a real-time programming language, while the proof-of-concept was successfully performed using Java.

The use of the developed system in the maritime domain could change the way in which collision avoidance is performed today, support the transition phase when autonomous and manned ships navigate the sea while at the same time and help to discuss possible alternatives to current procedures.

7.1 Support of Intention Based Negotiation

As mentioned in section 3.3, a prototype ECDIS application with the ability to send and receive routes was developed by the Danish Maritime Authority and the Maritime Human Factors Group from Chalmers University of Technology (Porathe et al., 2012a). In studies in a simulator and during a Search-and-Rescue mission, the authors identified a number of implications and open issues when intended routes were to be exchanged among ships. The suggested approach in this work is a step towards successful integration of route exchange in maritime routines. Route negotiations over VHF can lead to misunderstandings about the intended behaviour. Manual route negotiations over explicitly stated routes in an ECDIS application can remove that ambiguity but at the expense of spending crucial time on entering and negotiating the routes manually. Porathe et al. (2012b) showed a manual route negotiation between two crews, which were able to exchange their intentions and thus come to a solution which might not be compliant with the COLREGs but is a common procedure in a specific situation. The manual negotiation may however be too time consuming when more than two ships are involved and best practises for manual, on-screen negotiation would have to be developed over a transition phase. With the automated negotiation presented in this work, only long term destinations would have to be entered and the crew of each ship in a situation could accept or reject a final found solution. In close-quarter situations when the system would need more time to come to the best solution, but it is unclear if this time is available, a feasible backup solution could be generated which is a snapshot of a specific round during the negotiation, to be implemented when a critical time is reached. This could give mariners time to come to a decision about the solution and state consent on implementing it on all participating ships. This would leave the executive decisions to the captain while reducing workload, which was another concern raised. At the same time nearby VTS operators can receive solutions and observe the intended behaviour, giving them the opportunity to bridge the technological gap to other ships without the system.

7.2 PREDICTION OF THE PHYSICAL MODEL

Since the external path planner uses the predicted trajectories of detected ships in its configuration space as obstacles, a very practical approach was examined where other ships are included using a linear constant velocity or a constant velocity and turn rate model. This is a simplification but makes the approach computationally fast enough and leaves the position, heading and speed of other ships as the only necessary information. It would be also possible to use other trajectories if they are known or predicted by another external source. A different trajectory planning approach than the one investigated in this work could plan for all other ships and not just for the own ship based on a predicted physical model of detected ships, i.e. via technologies as introduced in section 3.6. The negotiation is designed in the suggested way to keep communication to a minimum by negotiating only trajectories and not explicitly environment information or physical models, however, if predicted this could make it possible to plan trajectories locally for all other ships and thereby enhance the quality and convergence speed of the approach. In that case solutions would have to converge towards a solution which is not only collision free and optimal but also feasible because a predicted model might be only valid with a certain accuracy which would have to improve over the course of the negotiation.

7.3 STANDARDISATION OF COLLISION AVOIDENCE PROCEDURES

Due to the need to find rules which can be learned and used by human navigators in unforeseen situations, the COLREGs contain some ambigous definitions of situations and procedures. Some terms as *"an angle which differs appreciably form* 90°*"* (Cockcroft and Lameijer, 2003), may lead to a differing understanding. If the proposed system would be used on a large scale, additions the COLREGs to be used by automated and autonomous systems could be passed by the legislative authorities which could include strict ideal terms for safe distances or angles to adhere to. The behaviour in situations could be more nuanced to layout the ideal behaviour in i.e. a crossing situation not only dependent on the relative side of a ship but also on factors as the distance, the intention or the environment more than it is possible today. This could enhance further standardisation of the way in which collision avoidance is performed. In addition, new or refined rules could be in the future included in the path planner and the crew would with every trajectory suggestion in an assistance system be made instantly aware of changed procedures in every encountered situation.

7.4 CONTINOUS TRAJECTORY SURVEILLANCE

If the system would be used continuously even in situations where no collision is to be expected, ships could monitor not only intentions of a target ship but the system could negotiate and display the best trajectories during standard operations. Since the exchanged trajectories contain information about the dynamic model of the ship, unfamiliar members of the crew on a new ship could observe which manoeuvres are possible and most efficient. This could safe resources and help avoid getting into close-quarter situations due to false judgement. VTS operators could receive a continuous stream of updates of the intentions of ships in their area as well as the available emergency manoeuvres. This could even aid designing traffic separation schemes when areas can be detected which give ships very little option to evade a collision if there ever were a close-quarter situation. Especially near the exit of a harbour or near landmasses, where visibility is limited, the system could in addition to AIS and VTS display the already coordinated intention of ships which may first be hidden from sight but ultimately crossing in an unforeseen way as in the scenario of Porathe et al. (2012a).

7.5 Seemless Integration of Autonomous Ships

The system has the potential to be part of an autonomous system which could in the future help to realise fully autonomous ships. Some of the difficulties with the introduction of any autonomous system is the integration into procedures with other human decision makers. The negotiation can be used at the same time as part of an autopilot without any crew and as an assistance system, to be used on a manned ship, which are both in the same situation. This would not only make the behaviour of the autonomous system predictable and understandable for a human crew with a traditional training but also make the human behaviour accessible to an autonomous system within a certain margin of error.

7.6 Steps Towards a Widespread Use

In order to build a holistic collision avoidance system, which can be implemented in a modern ship's bridge, a number of further problems have to be addressed. This work was conducted under a few simplifications and under certain assumptions, which may not hold for every situation.

7.6.1 Collision Detection

An important assumption is that a collision situation is detected before the procedure is started. Even though it can be used in a scenario where it guides ships in continuous use, the procedure will always assume that a collision was detected by an external component and it has to avoid that collision. The development of a collision prediction component was not the focus of this thesis, even though the path planner developed in Blaich et al. (2012) has the ability to predict behaviour based on its described models, this may not be accurate for each situation. In a continuous use the system would use its described σ and R distances to find collision free trajectories, though an ideal point where mariners would have to be warned in such a scenario was not examined. A sensible choice would be however, if the worst-case time to come to an initial solution in addition to the time needed for alternative measures i.e. coming to a complete stop is going to be exceeded.

7.6.2 Special Circumstances

The application of the procedure in certain areas, with different legislation or under the guidance of a VTS might pose requirements which are not identified and regarded in this work. In a situation where different ships may or may not enter a specific geographic area because of their draught or near an Emission Control Area this may be modelled in the procedure as a simple obstacle in the configuration space however this obstacle would be present for some ships and not for others. The negotiation would benefit from a common knowledge about similar environmental factors and since they are usually agreed and fixed in maps it would be possible to include them on every active ship, without communication. This would also apply to Traffic Separation Schemes, which add further requirements and could be handled analogous to the approach of Szlapczynski (2013) with an adjusted path generation dependent on the positions and destinations of all ships.

7.6.3 Heterogenos Participants

The possible range of ships to which this procedure is applicable is right now focused on ships propelled by fuel or electricity and therefore manoeuvrable at all times, even though this manoeuvrability might differ vastly between slow tankers and fast recreational ships. The possibility that a ship negotiates a trajectory but is then unable to follow it i.e. due to changing winds or currents, is not investigated. Trajectories, which have to have a different progression due to different directions of the wind i.e. for sailing ships, are not investigated.

7.6.4 Emergent Behaviour

It might be possible to reduce the number of needed rounds or the content of the communication further by implementing certain additional procedures which are the same on every ship, to reach an emergent, implicit behaviour, which can be relied upon. The negotiation is performed under the assumption that ships do not know every information which may only be available locally on every ship. Those information include the perspective on the environment, the physical model of the ship and the intention of the crew. If in the future those information could be predicted and gathered via improved prediction methods and information sharing schemes as the maritime cloud, (Weintrit, 2014), this could make approaches possible where no communication or negotiation would be necessary. If a system as the deterministic path planner is used and ships can rely that it is used with the same information on every ship no explicit negotiation is necessary and a common behaviour can still be executed. This is analogue to introducing new collision avoidance regulations but would include regulations which are detailed with regards to angles, distances and speed changes in a way that only a computer based system could process the environment information and produce and execute the only possible deterministically defined behaviour. However, this would change the approach to on-line planning which is due to its reactive behaviour hard to predict. Safety properties may be difficult to guarantee and a simulation of the behaviour would be necessary in each situation to see the result of such an approach before it would be implemented by an autonomous system.

7.6.5 Limiting the State Space Further

Fulfilling the demands of the COLREGs can not only be a challenge but also a simplification for a collision avoidance system because they limit the solution space. If in a head-on situation only one way of passing each other is conforming to the regulations, other ways do not have to be planned, evaluated and then discarded. Blaich et al. (2012) already limits the state space by its T-shaped operator which efficiently limits the search to physically feasible trajectories. Every single COLREG is in a similar way a chance to limit the search further and introducing *Stand-On* and *Give-Way* ships would leave the burden of planning to the Give-Way ship with a very good prediction about the behaviour of the Stand-On ship. This reasoning can also be applied to large inert ships which may according to the COLREGs have to change their course but would need several hundred of meters to even initiate a manoeuvre. The solution might not be fair in the terms defined in this work but speed up the process if Give-Way and large ships above a certain size relative to all others would not have to change their course at all.

Further artificial rules may be used to discretise the state space more purposely even though this may produce suboptimal solutions when compared to a continuous solution. If only certain step sizes between incoming and outgoing edges of a waypoint or distances of waypoints would be allowed, this would speed up the path generation process and could be done in the same way without communication on each ship if the step size would be globally set and fixed for all ships.

A

Appendix

A.1 Results 5 Ship Experiment

All data of a 5-ship experiment are given in table A.1, however without some improvements which accelerated the procedure in later experiments. The course of the values is also shown graphically in figure A.1:

Table A.1: Values Over 30 Rounds

TimeStamp	Negotiation Round	Number of Waypoints	Penalty Cost	Vehicle Cost	J* Cost	Augmented Cost	Length of all Trajectories	Minimal Closest Point of Approach	Beta	Epsilon
16.432	3.0	13.0	4410000.000000029	715716.0	-12.124233324913693	-1.168323332491369E8	4418.411942195109	19.999999999999833	1.0E7	1.0
21.423	4.0	16.0	25477.199717442854	717409.0	-12.115003957866973	-1.1870156158737889E8	4344.612823692928	48.37751763156511	9800000.0	1.0
27.6	5.0	16.0	223090.0261425371	672400.0	-12.335344981234313	-1.1819622179370686E8	4343.270155429412	45.02972275081216	9600000.0	1.0
33.803	6.0	16.0	60759.63588160862	675684.0	-12.320811051000332	-1.1575486424352151E8	4341.832886301809	47.47109965577387	9400000.0	1.0
39.712	7.0	16.0	0.0	675684.0	-12.320811051000332	-1.1335146166920306E8	4344.8334695336835	50.39983486475673	9200000.0	1.0
46.868	8.0	16.0	1012462.3557558048	669124.0	-12.349636048840292	-1.1013426208380683E8	4341.617245700233	38.64955518339497	9000000.0	1.0
52.4	9.0	20.0	997552.6225520859	664225.0	-12.370633239412086	-1.0786401988427427E8	4298.7888587617435	38.74563766862783	8800000.0	1.0
58.756	10.0	17.0	794463.9572896142	672400.0	-12.335344981234313	-1.0528950288132547E8	4300.808532193095	40.10826502738517	8600000.0	1.0
64.065	11.0	16.0	1101502.147101436	667489.0	-12.356692814718336	-1.0269471749653259E8	4342.045791361701	38.08510569254245	8400000.0	1.0
70.802	12.0	16.0	0.0	667489.0	-12.356692814718336	-1.0132488108069035E8	4287.002959779099	149.16568724574304	8200000.0	1.0
76.315	13.0	17.0	0.0	664225.0	-12.370633239412086	-9.896506591529669E7	4288.434678438938	74.9828802777679	8000000.0	1.0
82.026	14.0	16.0	0.0	664225.0	-12.370633239412086	-9.649093926741427E7	4286.707505114263	143.57102404311192	7800000.0	1.0
87.993	15.0	16.0	0.0	660969.0	-12.384348529615378	-9.412104882507688E7	4286.642389261355	143.2364490692523	7600000.0	1.0
94.757	16.0	16.0	0.0	660969.0	-12.384348529615378	-9.16441791191538E7	4286.932157994293	143.23645214332072	7400000.0	1.0
100.609	17.0	16.0	0.0	660969.0	-12.384348529615378	-8.916730941323072E7	4287.042899225851	143.23645527649433	7200000.0	1.0
106.149	18.0	16.0	0.0	662596.0	-12.377518610215422	-8.664263027150795E7	4285.448886419889	143.57097232341997	7000000.0	1.0
112.651	19.0	16.0	0.0	662596.0	-12.377518610215422	-8.416712654946487E7	4285.559619218741	143.570978589011	6800000.0	1.0
119.229	20.0	16.0	0.0	662596.0	-12.377518610215422	-8.169162282742178E7	4284.709025598265	143.889921825684	6600000.0	1.0
125.232	21.0	16.0	0.0	662596.0	-12.377518610215422	-7.92161191053787E7	4283.748251957878	143.88986669325786	6400000.0	1.0
130.951	22.0	16.0	0.0	662596.0	-12.377518610215422	-7.674061538333562E7	4283.859036401347	143.88986669325786	6200000.0	1.0
137.356	23.0	16.0	0.0	662596.0	-12.377518610215422	-7.426511166129252E7	4283.480842494266	143.9280171645008	6000000.0	1.0
144.109	24.0	17.0	0.0	662596.0	-12.377518610215422	-7.178960793924944E7	4284.402521207468	125.38398062556935	5800000.0	1.0
150.152	25.0	16.0	0.0	662596.0	-12.377518610215422	-6.931410421720636E7	4283.679445956191	143.75899665142748	5600000.0	1.0
156.219	26.0	18.0	0.0	662596.0	-12.377518610215422	-6.683860049516328E7	4285.577309478909	144.1943156098575	5400000.0	1.0
162.864	27.0	17.0	0.0	662596.0	-12.377518610215422	-6.4363096773120195E7	4284.98429113094	125.38399504324906	5200000.0	1.0
170.8	28.0	18.0	0.0	662596.0	-12.377518610215422	-6.1887593051077105E7	4287.9386008825695	144.34139588991647	5000000.0	1.0
176.966	29.0	18.0	0.0	662596.0	-12.377518610215422	-5.941208932903402E7	4286.144246421095	144.48521463712538	4800000.0	1.0

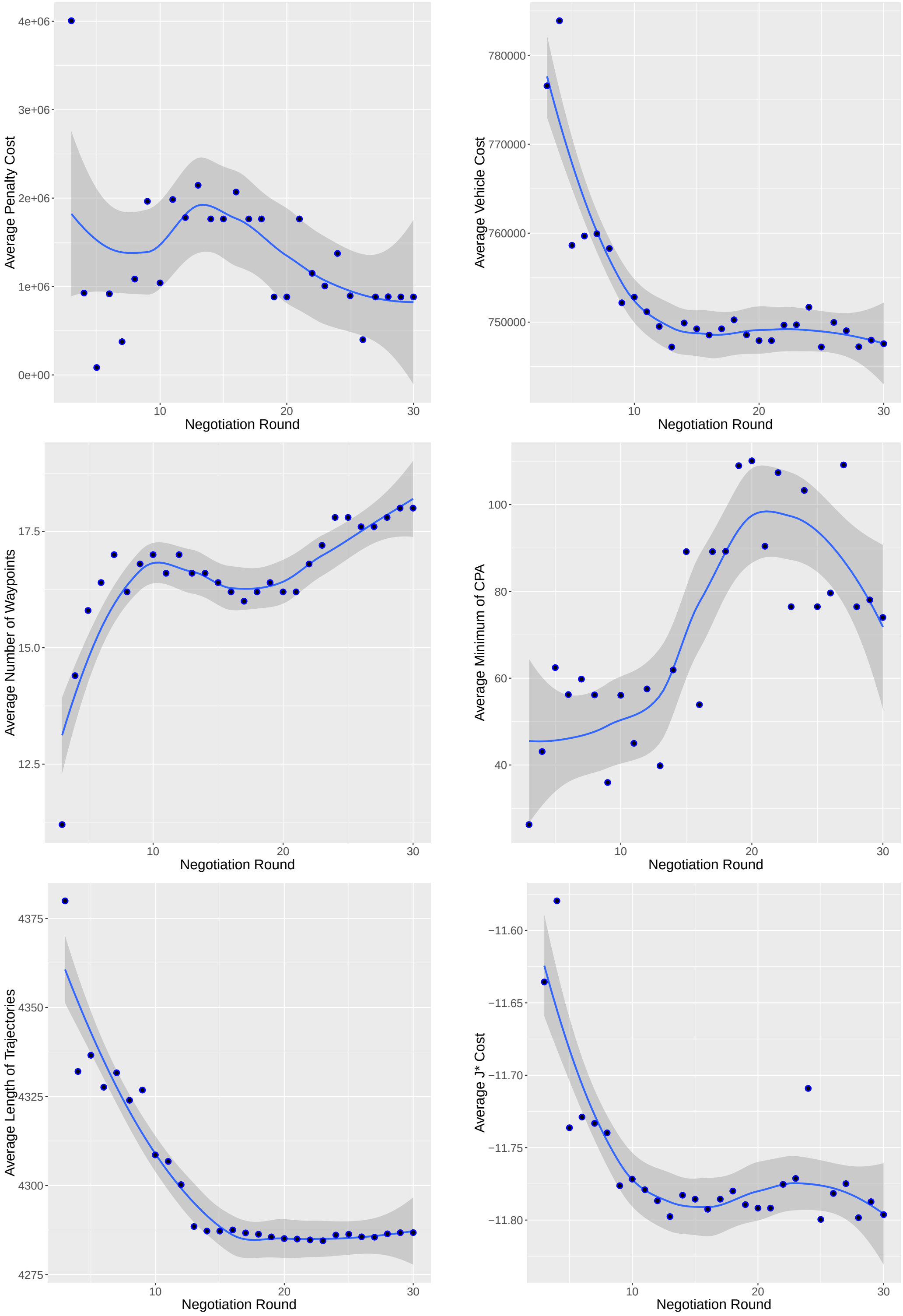


Figure A.1: Course of all Performance Measures

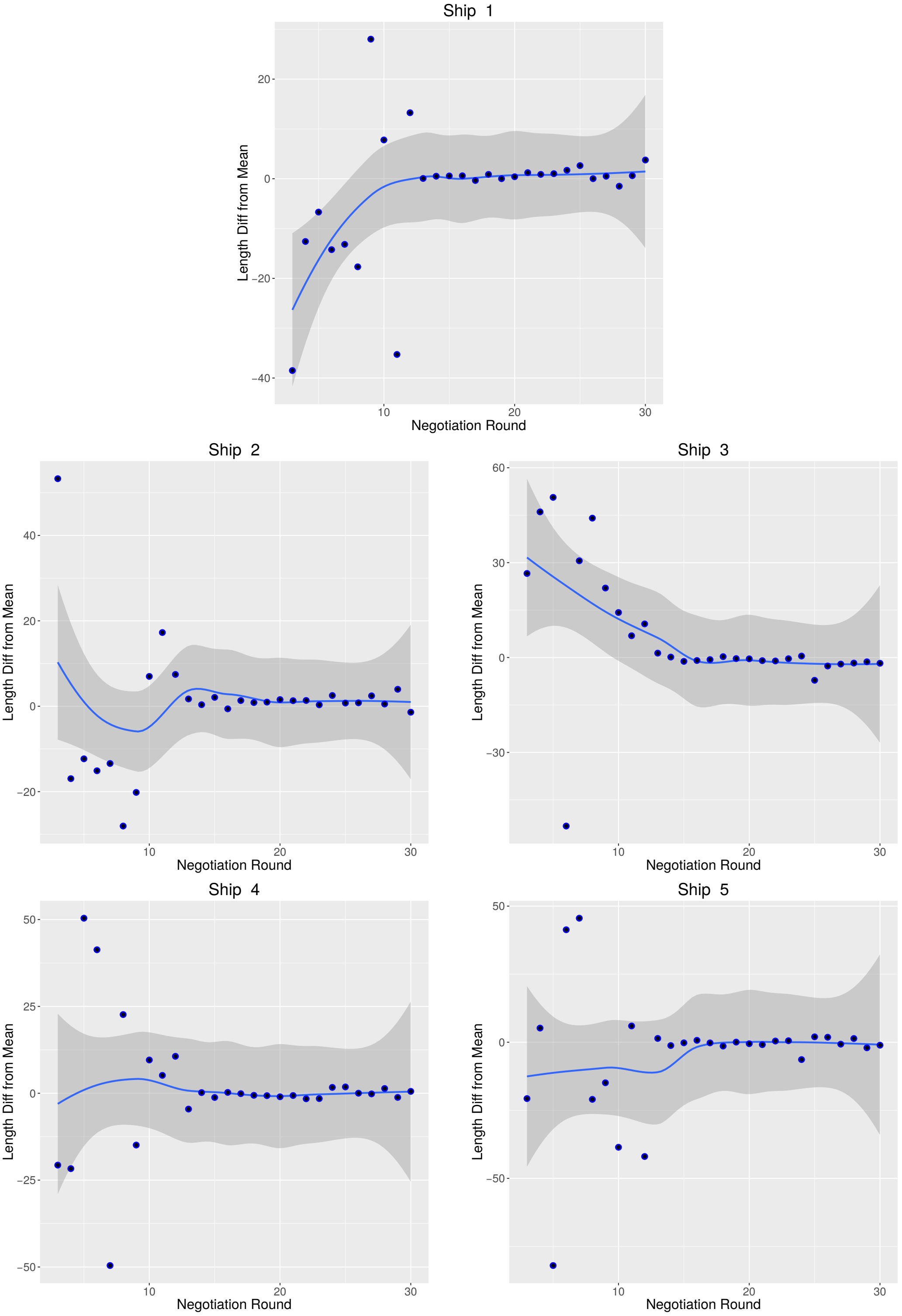


Figure A.2: Differences in the Length of all Ships

Listing of figures

Glossary

AIS Automatic Identification System. 3, 29, 31

ARPA Automatic Radar Plotting Aid. 3, 26, 28, 31

BOS Boat Operating System. 90

COLREG Conventions on the International Regulations for Preventing Collisions at Sea. 5, 10, 16, 17, 18, 26, 28, 30, 34, 36, 38, 39, 40, 41, 51, 72, 78, 90, 94

DUNE Uniform Navigational Environment Documentation. 78

EEZ German Exclusive Economic Zone. 3

HTWG Hochschule Konstanz Technik Wirtschaft und Gestaltung. 38, 72, 78

IMC Inter-Module Communication Framework. 39, 72, 73, 78

IMO International Maritime Organization. 3

ManTra Maritime n-Ship Trajectory Negotiation System for Collision Avoidance. 78

TCAS Traffic Alert and Collision Avoidance System. 29

USV Unmanned Surface Vehicle. 30

VTS Vessel Traffic Services. 3

References

Almeida, C., Franco, T., Ferreira, H., Martins, A., Santos, R., Almeida, J. M., Carvalho, J., and Silva, E. (2009). Radar based collision detection developments on USV ROAZ II. In *OCEANS 2009EUROPE*, pages 1–6. IEEE. `http://ieeexplore.ieee.org/xpl/freeabs{_}all.jsp?arnumber=5278238`.

Baldauf, M., Mehdi, R., Deeb, H., Schröder-hinrichs, J. U., Benedict, K., Krüger, C., Fischer, S., and Gluch, M. (2015). Manoeuvring areas to adapt ACAS for the maritime domain. 43(115):39–47.

Blaich, M., Rosenfelder, M., Schuster, M., Bittel, O., and Reuter, J. (2012). Fast Grid Based Collision Avoidance for Vessels using A * Search Algorithm. In *17th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 385 – 390, Miezyzdroje.

Breitsprecher, M. (2012). Methods of knowledge representation contained in COLREGS. 30(102):18–24.

Bundesamt für Seeschifffahrt und Hydrographie (2015). BSH Wind Farm Projects. `http://www.bsh.de/en/Marine{_}uses/Industry/Wind{_}farms/`.

Bundesrepublik Deutschland (1998). Seeschifffahrtsstraßen-Ordnung (SeeSchStrO) Issue 22. 10. 1998 (BGBl. I Page 3209).

Bundesversammlung Schweiz (1968). Übereinkommen über den Strassenverkehr.

Campbell, S., Naeem, W., and Irwin, G. (2012). A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres. *Annual Reviews in Control*, 36(2):267–283. `http://linkinghub.elsevier.com/retrieve/pii/S1367578812000430`.

Clarkson Research Studies Ltd (1987). *Tanker Register, 1987*. Tanker Register. International Publication Service.

Cockcroft, A. N. and Lameijer, J. N. F. (2003). *Guide to the collision avoidance rules*. Butterworth-Heinemann.

Davidson, J. and Davidson, J. (1997). United Nations Convention on the Law of the Sea Act 1996. *The International Journal of Marine and Coastal Law*, 12(3):404–412.

Davis, P. V., Dove, M. J., and Stockel, C. T. (1980). A Computer Simulation of Marine Traffic Using Domains and Arenas. *Journal of Navigation*, 33(02):215–222. `http://journals.cambridge.org/abstract{_}S0373463300035220`.

Duvenhage, B. and Nel, J. J. (2007). Inaccuracies when mixing coordinate reference frameworks in a system of systems simulation. *IEEE AFRICON Conference*, pages 1–7.

Economic Commission for Europe (2014). Report of the sixty-eighth session of the Working Party on Road Traffic Safety. (April).

Espindle, L. P., Billingsley, T., and Griffith, J. (2009). TCAS multiple threat encounter analysis. *Massachusetts Institute of Technology, Lincoln Laboratory, Project Report ATC-359*. `http://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:TCAS+Multiple+Threat+Encounter+Analysis{#}0`.

European Commission (2011). Commission Regulation (EU) No 1332/2011.

Ferguson, D., Likhachev, M., and Stentz, A. (2005). A guide to heuristic-based path planning. *Proceedings of the International Workshop on Planning under Uncertainty for Autonomous Systems, International Conference on Automated Planning and Scheduling (ICAPS)*, pages 1 – 10. `http://cs.cmu.edu/afs/cs.cmu.edu/Web/People/maxim/files/hsplanguide{_}icaps05ws.pdf`.

Goodwin, E. M. (1975). A statistical study of ship domains. *Journal of Navigation*, 3(28):328 – 344. `http://journals.cambridge.org/abstract{_}S0373463300041230`.

Heiskanen, P. (1999). Decentralized method for computing Pareto solutions in multiparty negotiations. 117:578–590.

Hetherington, C., Flin, R., and Mearns, K. (2006). Safety in shipping: the human element. *Journal of safety research*, 37(4):401–11.

Hilgert, H. (1983). Defining the Close-Quarters Situation at Sea. *Journal of Navigation*, 36(03):454. `http://www.journals.cambridge.org/abstract{_}S0373463300039801`.

Hoffmann, G., Rajnarayan, D. G., Waslander, S. L., Dostal, D., Candidate, M., Jang, J. S., and Tomlin, C. J. (2004). The Stanford Testbed of Autonomous Rotorcraft for Multi Agent Control (STARMAC). pages 1–10.

Iijima, Y., Hagiwara, H., and Kasai, H. (1991). Results of Collision Avoidance Manoeuvre Experiments Using a Knowledge-Based Autonomous Piloting System. *Journal of Navigation*, 44(02):194.

Inalhan, G. (2002). Decentralized optimization, with application to multiple aircraft coordination. *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on Decision and Control.*., 1:1147–1155.

International Civil Aviation Organization (2006). Airborne Collision Avoidance System (ACAS) Manual.

International Maritime Organisation (1997). Guidelines for Vessel Traffic Services. (A 20/Res.857, December).

International Maritime Organisation (2004). *Consolidated text of the International Convention of Safety of Life at Sea, 1974, and its Protocol of 1988: articles, annexes and certificates*. International Maritime Organization.

International Maritime Organisation (2016). Status of Conventions. `http://www.imo.org/en/About/Conventions/StatusOfConventions`.

Kuchar, J. K. and Drumm, a. C. (2007). The traffic alert and collision avoidance system. *Lincoln Laboratory Journal*, 16(2):277–296.

Kuwata, Y., Wolf, M. T., Zarzhitsky, D., and Huntsberger, T. L. (2014). Safe maritime autonomous navigation with COLREGS, using velocity obstacles. *IEEE Journal of Oceanic Engineering*, 39(1):110–119.

Lavalle, S. M. (2006). The Configuration Space. *Planning Algorithms*.

Lee, Y.-i. and Kim, Y.-g. (2004). A Collision Avoidance System for Autonomous Ship Using Fuzzy Relational Products and COLREGs. *Control*, pages 247–252.

Lozano-Pérez, T. (1983). Spatial Planning: a Configuration Space Approach. *IEEE Transactions on Computers*, C-32(2):108–120. `http://lis.csail.mit.edu/pubs/tlp/spatial-planning.pdf`.

Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., and Balan, G. (2005). MASON: A Multiagent Simulation Environment. *Simulation*, 81(7):517–527.

Martins, R., Dias, P. S., Marques, E., Pinto, J., Sousa, J. B., and Pereira, F. L. (2009). IMC: A communication protocol for networked vehicles and sensors. In *Oceans 2009-Europe*, pages 1–6. Ieee.

MathWorks (2016). MathWorks-Matlab. `http://www.mathworks.com/products/matlab/`.

Mazzarella, F., Arguedas, V. F., and Vespe, M. (2015). Knowledge-based vessel position prediction using historical AIS data. In *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6. IEEE. `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7347707`.

Mou, J. M., Tak, C. V. D., and Ligteringen, H. (2010). Study on collision avoidance in busy waterways by using AIS data. *Ocean Engineering*, 37(5-6):483–490.

Muthoo, A. (1999). *Bargaining theory with applications.*

Naeem, W., Irwin, G. W., and Yang, A. (2012). COLREGs-based collision avoidance strategies for unmanned surface vehicles. *Mechatronics*, 22(6):669–678. `http://linkinghub.elsevier.com/retrieve/pii/S0957415811001553`.

Nash, J. (1951). Non-Cooperative Games. *The Annals of Mathematics*, 54(2):286.

Perera, L. and Soares, C. G. (2010). Ocean Vessel Trajectory Estimation and Prediction Based on Extended Kalman Filter. *ADAPTIVE 2010, The Second ...*, (c):14–20.

Porathe, T. (2012). Transmitting intended and suggested routes in ship operations: cognitive off-loading by placing knowledge in the world. *A Journal of Prevention, Assessment and Rehabilitation*, 41:4873–4878.

Porathe, T., Karlsson, F., Borup, O., and Bentzen, M. (2012a). Tests of SAR services. (Deliverable EfficienSea Project):1–37.

Porathe, T., Lützhöft, M., and Praetorius, G. (2012b). What is your Intention? Communicating Routes in Electronic Nautical Charts. *Procedia - Social and Behavioral Sciences*, 48:3266–3273.

Qinyou, H., Chun, Y., Haishan, C., and Baojia, X. (2008). Planned Route Based Negotiation for Collision Avoidance Between Vessels. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, 2(4):363–368.

Qinyou, H., Qiaoer, H., and Chaojian, S. (2006). A Negotiation Framework for Automatic Collision Avoidance between Vessels. *2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 595–601.

Richards, A. and How, J. (2002). Aircraft trajectory planning with collision avoidance using mixed integer linear programming. *American Control Conference, 2002. ....* `http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=1023918`.

Smierzchalski, R. (1999). Evolutionary trajectory planning of ships in navigation traffic areas. *Journal of Marine Science and Technology*, 4(1):1–6.

Smierzchalski, R. and Michalewicz, Z. (1998). Adaptive modeling of a ship trajectory in collision situations at sea. *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360).*

Smierzchalski, R. and Michalewicz, Z. (2000). Modeling of ship trajectory in collision situations by an evolutionary algorithm. *Evolutionary Computation, IEEE Transactions on 4.3*, XX:1–18.

Statheros, T., Howells, G., and Maier, K. M. (2008). Autonomous Ship Collision Avoidance Navigation Concepts, Technologies and Techniques. *Journal of Navigation*, 61(01):129–142. `http://www.journals.cambridge.org/abstract{_}S037346330700447X`.

Sutton, J. (1986). Non-Cooperative Bargaining Theory: An Introduction. *The Review of Economic Studies*, 53(5):709–724.

Szlapczynski, R. (2006). A Unified Measure Of Collision Risk Derived From The Concept Of A Ship Domain. *Journal of Navigation*, 59(03):477–490. `http://www.journals.cambridge.org/abstract{_}S0373463306003833`.

Szlapczynski, R. (2011). Evolutionary Sets Of Safe Ship Trajectories: A New Approach To Collision Avoidance. *Journal of Navigation*, 64(01):169–181.

Szlapczynski, R. (2012). Evolutionary approach to ship ' s trajectory planning within Traffic Separation Schemes. *Polish Maritime Research*, 19(1):11–20.

Szlapczynski, R. (2013). Evolutionary Sets of Safe Ship Trajectories Within Traffic Separation Schemes. *Journal of Navigation*, 66(1):65–81.

Szlapczynski, R. and Szlapczynska, J. (2012). On evolutionary computing in multi-ship trajectory planning. *Applied Intelligence*, 37(2):155–174.

Toremar, M. (1999). *The legal position of the ship master*. Master thesis, Göteborg University.

Tvete, H. A. (2014). The Next Revolt. *DNV-GL Magazine Maritime Impact*, (2):18–23. `https://www.dnvgl.com/publications/maritime-impact-02-2014-11807`.

UNCTAD (2014). Review of Maritime Transport 2014. In *Review of Maritime Transport -Annual Report*, volume UNCTAD/RMT, page 34. `http://trid.trb.org/view.aspx?id=1238887$\delimiter"026E30F$nhttp://unctad.org/en/pages/PublicationArchive.aspx?publicationid=1734`.

Vasquez, D., Fraichard, T., and Laugier, C. (2009). Growing Hidden Markov Models: An Incremental Tool for Learning and Predicting Human and Vehicle Motion. *The International Journal of Robotics Research*, 28(11-12):1486–1506.

Volvo (2015). US urged to establish nationwide Federal guidelines for autonomous driving. `https://www.media.volvocars.com/global/en-gb/media/pressreleases/167975`.

von Neumann, J., Morgenstern, O., Kuhn, H. W., and Rubinstein, A. (1944). Theory of Games and Economic Behavior. page 776.

Waslander, S. (2004). Decentralized optimization via Nash bargaining. *Theory and Algorithms for cooperative Systems*, 4:565–585.

Waslander, S. (2007). Multi-Agent Systems Design For Aerospace Applications.

Weintrit, A. (2014). E-Navigation Revolution – Maritime Cloud Concept. In Mikulski, J., editor, *Telematics - Support for Transport*, pages 80–90. Springer, Gdynia, Polen.

Xiao, J., Michalewicz, Z., Zhang, L., and Trojanowski, K. (1997). Adaptive evolutionary planner/navigator for mobile robots. *IEEE Transactions on Evolutionary Computation*, 1(1):18–28.

This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. The body text is set in 11 point Egenolff-Berner Garamond, a revival of Claude Garamont's humanist typeface. A template that can be used to format a PhD thesis with this look and feel has been released under the permissive MIT (xii) license, and can be found online at github.com/suchow/Dissertate or from its author, Jordan Suchow, at suchow@post.harvard.edu.