Fuzzy logic and unmanned Surface Vehicles

Implementing collission avoidance in Python

Emil AURA

Masters's thesis (30 ECTS)
Supervisors: Prof. Johan LILIUS
Adj. Prof. Marina WALDÉN
Department of Information Technologies
Faculty of Science and Engineering
Åbo Akademi University
March 26, 2018

ÅBO AKADEMI UNIVERSITY

Abstract

Faculty of Science and Engineering
Department of Information Technologies

Master of Science (Technology)

Fuzzy logic and unmanned Surface Vehicles
Implementing collission avoidance in Python

by Emil AURA

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too. . .

Keywords: COLREGS, Unmanned Surface Vessels, Fuzzy Logic

Acknowledgements

The acknowledgments and the people to thank go here, don't forget to include your project advisor. . .

Contents

List of Figures

List of Tables

Todo list

List of Abbreviations

COLREGs	Convention on the International Regulations for Preventing Collisions at Sea, 1972
FMF	**F**uzzy **M**embership **F**unction
OPRA	**O**riented **P**oint **R**easoning **A**lgebra
SA	**S**ituational **A**wareness
FIS	**F**uzzy **I**nference **S**ystem

1 Introduction

[1] [2] [3]

1.1 Problem statement

1.2 Thesis structure

2 Unmanned Surface Vehicles

Unmanned Surface Vehicles (USV) or Autonomous Surface Crafts (ASC) are vehicles operating the seas without a crew on-board. USVs encompass both fully autonomous vehicles, from now on referred to as ASCs, and semi-autonomous vehicles. Development of USVs has been ongoing for the last two decades [4]. However, the majority of the USVs developed are of the semi-autonomous type [5], [6], meaning that they depend on human intervention to some extent usually by a supervisor located on shore. Although semi-autonomous USVs greatly increase the safety of the operating personnel [5], they do not completely remove the need for human interaction. Supervision of several semi-autonomous vehicles can be admittedly be handled by a single person, which significantly decreases the number of person-hours needed to accomplish a specific mission [4]. The person-hours needed for surveillance could, however, be removed completely by a ASC. It is, therefore, of great interest to overcome the challenges associated with ASCs.

2.1 Usage

Yuh, Marani, and Blidberg [7] mention that roughly two-thirds of the earth's surface is covered by water, with an average depth of the oceans being 3688 m [8]. Thereby, adding up to a vast amount of explorable areas of which 95 % is yet to be seen by human eyes [9]. Utilization of autonomous vehicles could notably facilitate the exploration of these, yet unknown, areas. Although ASCs are situated on the surface of the ocean they can greatly increase the efficiency of Unmanned Underwater Vehicles (UUV) by acting as a gateway between UUVs and services, such as GPS, not easily available in underwater environments [5].

Liu, Zhang, Yu, *et al.* [5] have, furthermore, compiled a list of potential applications of USVs, along with previous research on the topics. The list is divided into five major categories: scientific research, environmental missions, ocean resource exploration, military use, and other applications.

2.2   Advantages

More advantages

USVs have, apart from the obvious decrease in man hours, a few advantages over traditional manned vessels. The absence of humans on board means that facilities and resources such as canteens, manned bridges, showers etc. are no longer needed. The weight saved increases the manoeuvrability and deployability and can also be used to increase the payload the vessel is able to carry. USVs can, furthermore, conduct longer and more hazardous voyages than manned vessels since the personnel is located safely on land. This has the additional benefit of decreasing the operational costs [5].

2.3   Challenges

Even though USVs bring many advantages to the maritime industry, there are still quite a few challenges that need to be solved. The following section presents some of these challenges and their relation to the scope of this thesis.

USVs should be able to operate in a highly variable environment, all over the world, in terms of climate, traffic density and communication channels available. Algorithms and systems developed should, therefore, ideally be usable in all situations that might arise in this environment. This introduces challenges both hardware and software wise [5]. The limited scope of this thesis narrows the research to only the software related challenges with a focus on path planning, especially collision avoidance.

All vessels operating on water are bound to follow the international maritime "Rules of the Road" called the Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs) explained in section 3. This is also true for USVs, since they should be able to operate in situations that comprise both other USVs and manned vessels that follow the

COLREGs rules. Furthermore, they should be able to act in a safe manner in situations where other vessels, for some reason, are not complying with the COLREGs rules. Hence, it is crucial for the USVs to follow COLREGs to ensure safe operations.

COLREGs were originally introduced in 1972, to reduce the amount of collisions at sea. They are written to be interpreted by humans. Situations where multiple rules apply and contradict each other might therefore arise. These situations are usually solved with the use of good seamanship and human deduction. These aspects pose challenges when translating the COLREGs rules into computer understandable code [10].

3 COLREG

The Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs) were adopted 1972 and entered into force 1977 in it's original form. COLREGs consist of 38 rules, grouped into five different sections as is designed by the International Maritime Organization (IMO). The original objective of COLREGs was to ensure traffic separation between maritime vessels in an increasingly populated environment back in 1972 and it is, therefore, sometimes referred to as the Rules of the Road for maritime vessels. COLREGs have since then been in use and is still mandatory to adhere to on international waters. However, a great deal has changed since the COLREGs initially entered into force and COLREGs has, therefore, received amendments several times [11].

National regulations might differ from the COLREGs to some extent. However, COLREGs mention that 'Nothing in these Rules shall interfere with the operation of special rules made by an appropriate authority for roadsteads, harbours, rivers, lakes or inland waterways connected with the high seas and navigable by seagoing vessels. Such special rules shall conform as closely as possible to these Rules.' [12]

The following sections will present the different parts of the COLREGs regulations, with emphasis on the parts related to manoeuvring a USV in international waters. Information in this subsection is taken from the official COLREGs regulations [12] if not otherwise specified. Part D and E will not be discussed further, since they are not of interest for the scope of this thesis. Moreover, rules that are not of interest for the scope of this thesis will not be mentioned.

3.1   Part A General

Part A consist of three rules and address general conditions when and to which vessels COLREGs apply. This includes all vessels navigating on high seas and all waters connected therewith. Additionally, it specifies that special rules regarding roadsteads, harbours, rivers, lakes or inland waterways connected with the high seas and navigable by seagoing vessels shall conform as closely as possible to these Rules (*Rule1*). Furthermore words, such as vessel, power-driven vessel, sailing vessel, length, breadth, and other words used in the regulations are defined (*Rule3*). Finally, it is stated that the COLREGs rules do not in any way free the vessel, owner, master or crew from responsibility to follow the rules and act according to the ordinary practice of seamen (*Rule2*).

3.2   Part B Steering and Sailing

Part B is split up into three different sections. The first section consists of rules 4-10 and apply in any condition of visibility. Section II (rules 11-18) apply to vessels in sight of each other, while the last section consist of just one rule that specifies reduced visibility operations. It states that vessels should proceed at speeds appropriate to the circumstances and use radar to determine collision risks. Speed should be reduced to a minimum if a vessel can hear another vessels fog horn apparently forward of her beam (*Rule19*).

words or numerals

Section I addresses how vessels shall operate in order to ensure proper situational awareness with emphasis on collision risk detection. It is mandatory for all vessels to at all times maintain proper look-out by sight and hearing and all other available means to ensure the best possible situational awareness (*Rule5*) and use all means possible to determine if a risk of collision exists. This includes the use of radar equipment although caution should be exercised not to trust insufficient data. A collision is considered to be imminent when two vessel maintain constant compass bearings to each other over a prolonged time. However, a constant bearing is only a sufficient condition for a collision risk, not a necessary one. Close range, large vessels or a tow might pose a collision risk without a constant bearing.

Moreover, it is stated that any doubt whether a risk of collision exists shall be treated as if a risk exists (*Rule7*). Vessels shall, furthermore, maintain a speed so that they can take proper and effective action to avoid a collision and stop within a distance appropriate to the prevailing circumstances and conditions (*Rule 6*). Any actions made in order to to avoid collisions shall, additionally, be taken according to the rules and as far as possible be conducted in ample time and large enough so that they are easily observable by other vessels. Small alterations should, in other words, be avoided. Course alterations might also be accompanied by a lowering of speed, if considered necessary to ensure that the vessels pass each other at safe distance (*Rule 8*).

Section II defines three different scenarios, involving two vessels, and specifies the action to take for both vessel. The scenarios are a head-on, a crossing and an overtake situation. A vessel overtaking another vessel shall keep out of the way of the stand on vessel. A vessel is considered to overtake another vessel when approaching it from a direction more than 22.5 °abaft the stand on vessels beam, i.e with a relative bearing between 112,5 °and 247,5 °from the stand on vessel (*Rule 13*). A head-on situation, on the other hand, is when a vessel sees another vessel ahead or nearly ahead, i.e. it can see the masthead light in line or both sidelights. The give way vessel shall, in this case, alter its course to starboard (*Rule 14*). Finally it is stated that vessel coming from starboard, i.e right, has the right of way in a crossing situation. The give way vessel shall if possible avoid crossing ahead of the other vessel and, therefore, alter its course to starboard (*Rule 15*). All actions by the give away vessels shall be taken as early as possible (*Rule 16*), while the stand on vessels shall keep its course and speed, provided that the give way vessel acts according to the regulations. However, the stand on vessel might take action to best avoid a collision if it finds that the actions by the give way vessels are insufficient to prevent a collision (*Rule 17*). Responsibilities between vessels of different types are also addressed. For instance, a power-driven vessel shall give way for vessels engaged in fishing, vessels not under command, vessels restricted in her ability to move, and finally fishing vessels (*Rule 18*).

3.3 Part C Lights and Shapes
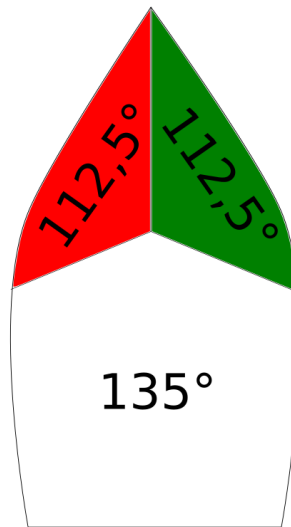
FIGURE 3.1: Mandatory navigation lights

Part C specifies the lights and shapes a vessel shall exhibit, and when they should be used. These are therefore not of interest for the scope of this thesis, with the exception of *rule 21* and *rule 22*, which specify the arcs that specific lights should cover as well as the distance the should be visible from. Figure 3.1 shows the lights of interest for this thesis.

Masthead lights cover a 225 °arc from 247,5 °to 112,5 °and should be visible from at least 2-6 miles, depending on the length of the vessel.

Sidelights are green on the starboard side and red on port. They cover a 112,5 arc from 0 relative bearing to 112,5 and 247,5 respectively. They should be visible from at least 1-3 miles, depending on the length of the vessel.

4 Alternative COLREG compliant algorithms

USVs have been in development since at least 1993 when MIT started its Sea Grant College Program, Autonomous Surface Craft (ASCs) [4]. Many different approaches and algorithms have since then been tried out. This section will briefly look into some of the COLREGs compliant approaches, before finally presenting the approach which this thesis is based on.

Larson, Bruch, Halterman, *et al.* [13] uses a two-level two-dimensional obstacle map to create a near-field reactive control technique. One layer for a near field model and the other for a far field model. COLREGs rules are incorporated into the solution by utilizing a rule-based approach in the path planner. The solution is made with computational efficiency in mind and the provided trajectory is, therefore, not optimal. It is also stated that the sensors and processing algorithm need more work. However, the authors have plane for future improvements and the algorithm is regularly tested in San Diego Bay.

Another approach, developed by Benjamin, Leonard, Curcio, *et al.* [10] and Benjamin and Curcio [14] utilizes multi-objective optimization and interval programming, within a behaviour based architecture. The solution has been tested with kayak-based autonomous surface crafts and initial tests show promising results, but the solution cannot yet be called COLREGS compliant.

Naeem, Irwin, and Yang [15] propose a path-planer that uses line of sight guidance between way-points. Obstacles are avoided by introducing a starboard heading bias to the line of sight heading until clear of conflict. The paper compares the trajectory, produced by the proposed on-the-fly algorithm, to that of a modified off-line DPSS strategy. Multiple simulations

show that both algorithms yield COLREGs compliant results. However, a simulation with multiple dynamic targets was not conducted.

Spatio temporal reasoning have also been considered as an way to translate the COLREG rules into computer understandable rules. Kreutzmann, Wolter, Dylla, *et al.* [16] and Wolter, Dylla, and Kreutzmann [17] suggest the use of Oriented Point Reasoning Algebra (OPRA) with m granularity to represent vessels position and relative moving direction. A qualitative spatio-temporal reasoning toolbox is used to convert the geometric scenario consisting of the vessels Cartesian coordinates into a qualitative scene description based of OPRA relations. The same toolbox can then generate all possible scenarios that are spatially or temporally possible. Model checking can then be conducted on the observations to check whether the behaviour of the involved vessels is COREGS compliant.

Lee, Kwon, and Joh [18] combine fuzzy logic with modified virtual force fields to create a COLREGs compliant algorithm. The fuzzy logic rule-set used consists of about two hundred rules, which might be computational challenging. The authors do, however, mention that the rules-set might be streamlined. Fuzzy logic is also used Perera, Carvalho, and Soares [19] which, furthermore uses bayesian networks to solve cases with multiple target vessels. The presented simulations consist of one vessel utilizing the algorithm to avoid three dummy vessels that are keeping their course and speed. The rule-set is in this paper also quite extensive.

Both fuzzy logic and spatio temporal reasoning with model checking was considered to be used as the foundation for this thesis. However, fuzzy logic was finally chosen due to the fact that the qualitative representation of a scenario in spatio temporal reasoning is limited the definitions of the calculus used, in this case OPRA.

5 Fuzzy set theory and logic

The concept of fuzzy sets was first mentioned by Zadeh [20] in 1964 as a way of dealing with sets of objects, where an objects membership to a set can be represented by a value in the real interval [0, 1]. Compared to classic set theory where membership is restricted to the two values of one and zero. Fuzzy logic enables one to define sets such as "old men" or "vessels on reciprocal course" [20], which facilitates the process of converting human abstract reasoning into computer understandable logic.
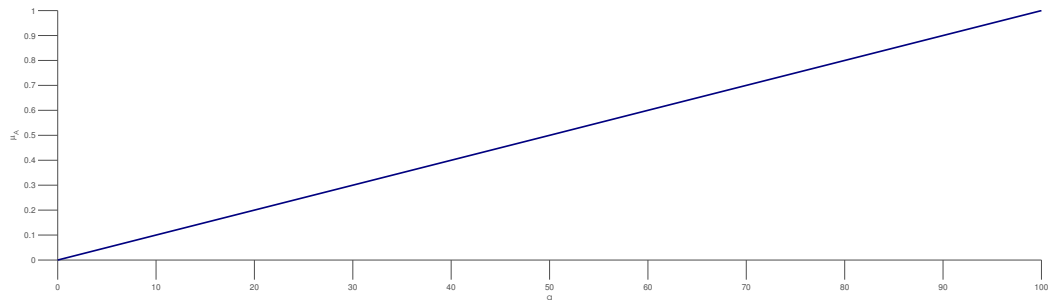
5.1 Fuzzy sets

The fuzzy set "all old men" can be defined in the following way. Let the *universe set* **X** be the set of ages [0, 100]. The set of old men can then be written as $\underset{\sim}{A} = \{a \in \mathbf{X} | a \text{ is old}\}$ [21]

However, the set requirement 'is old' is still undefined and therefore vague. Defining the threshold of being old at an age of 60 does divide the universal set **X** into subsets of 'old' and 'not old', but the division does not distinguish between humans aged 1 and 59. Fuzzy set theory introduces the concept of membership functions, to solve this problem. A Fuzzy Membership Function (FMF) describes the grade of membership of a in $\underset{\sim}{A}$. This function is often written as $\mu_A(a)$. Figure 5.1 shows a very simple linear FMF for the example $\underset{\sim}{A} = \{'\text{is old}'\}$, where for instance $\mu_A(10) = 0.1$. Similar FMFs can be defined for young and middle-aged which results in the graph seen in figure 5.2 [22].

A fuzzy set can then be written as an array of tuples consisting of the objects and its associated membership value: $\underset{\sim}{A} = \left\{ (x, \mu_A(x) | x \in X) \right\}$ [23], which gives
$\underset{\sim}{A} = \{(0,1), (10,0.9), (20,0.8), \dots, (90,0.1), (100,0)\}$ for the example $\underset{\sim}{A}=\{'\text{is young}'\}$ in figure 5.1 [22].

FIGURE 5.1: Fuzzy membership function for $A = \{\text{'Is old'}\}$
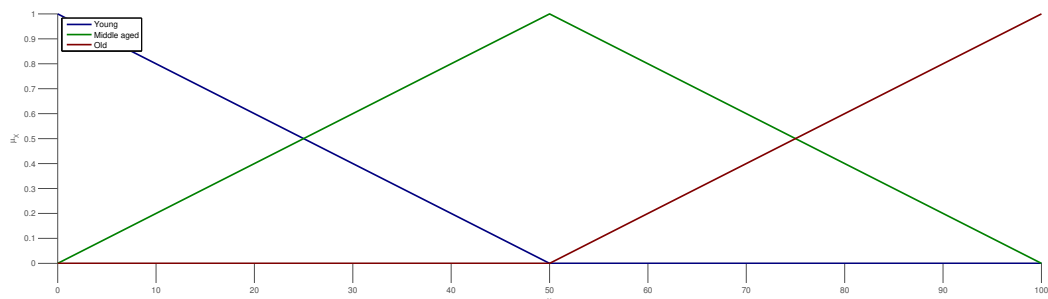
FIGURE 5.2: Fuzzy membership function for $A = \{\text{'Is young'}\}, B = \{\text{'Is middle-aged'}\}$ and $C = \{\text{'Is old'}\}$

5.2 Fuzzy logic

Fuzzy sets and set theory can, in the same manner as classical boolean sets, be used to define logical expressions. Fuzzy logic allows for half-truths, i.e. truth values in the interval [0,1]. Whereas boolean logic is restricted to truth values of one and zero. The truth value T of a fuzzy preposition P is, therefore a mapping from [0,1] to the universe T, as can be seen in 5.1 [22].

$$T : u \in U \rightarrow (0,1) \tag{5.1}$$

The truth value of proposition P is therefore given by the membership grade $\mu_A(x)$ of x in A

Many of the same operators and connectives used in classical logic, does also apply to fuzzy logic. This thesis will only present the rules necessary

for the scope of the thesis, which are the following [22]:

Negation

$$T(\overline{\underset{\sim}{P}}) = 1 - T(\underset{\sim}{P}))$$ (5.2)

Disjunction

$$\underset{\sim}{P} \vee \underset{\sim}{Q} : x \text{ is } \underset{\sim}{A} \text{ or } \underset{\sim}{B} \quad T(\underset{\sim}{P} \vee \underset{\sim}{Q}) = max(T(\underset{\sim}{P}), T(\underset{\sim}{Q}))$$ (5.3)

Conjunction

$$\underset{\sim}{P} \wedge \underset{\sim}{Q} : x \text{ is } \underset{\sim}{A} \text{ and } \underset{\sim}{B} \quad T(\underset{\sim}{P} \wedge \underset{\sim}{Q}) = min(T(\underset{\sim}{P}), T(\underset{\sim}{Q}))$$ (5.4)

Implication

$$\underset{\sim}{P} \to \underset{\sim}{Q} : x \text{ is } \underset{\sim}{A}, \text{ then } x \text{ is } \underset{\sim}{B}$$ (5.5)

$$T(\underset{\sim}{P} \to \underset{\sim}{Q}) = T(\overline{\underset{\sim}{P}} \vee \underset{\sim}{Q}) = max(T(\overline{\underset{\sim}{P}}), T(\underset{\sim}{Q}))$$

Implication can also be written in rule based form

$$\underset{\sim}{P} \to \underset{\sim}{Q} \text{ is, IF } x \text{ is } \underset{\sim}{A}, \text{ THEN } t \text{ is } \underset{\sim}{B}$$

$$\Leftrightarrow$$

$$R = (\underset{\sim}{A} \times \underset{\sim}{B}) \cup (\overline{\underset{\sim}{A}} \times \underset{\sim}{Y})$$

With the membership function:

$$\mu_{\underset{\sim}{R}}(x, y) = max[\mu_{\underset{\sim}{A}}(x) \wedge \mu_{\underset{\sim}{B}}(y), (1 - \mu_{\underset{\sim}{A}}(x))]$$ (5.6)

Equation 5.6 is the fuzzy equivalent to the material implication $\neg A \vee B$ used in traditional boolean logic[24], where $\neg A = 1 - \mu_A(x)$ and $B = \mu_{\underset{\sim}{A}}(x) \wedge \mu_B(y)$. However, fuzzy logic has multiple implication operators [22]. The work in this thesis will use Mamdani's implication, see equation 5.7, since it is the most commonly used one.

$$\mu_{\underset{\sim}{R}}(x, y) = min[\mu_{\underset{\sim}{A}}(x), \mu_{\underset{\sim}{B}}(x))]$$ (5.7)

5.3   Fuzzy (rule-based) systems

Fuzzy logic can be used to model complex systems described in natural language, originally written to be interpreted by humans. Such knowledge can often be written as rules in the following form [22].

$$\text{IF premise (antecedent), THEN conclusion (consequent)} \tag{5.8}$$

Combining multiple rules enables one to describe complex systems in a relatively simple structure. Rules can, furthermore, contain multiple antecedents and consequents. However, this raises the question of how multiple antecedents, as shown in rule 5.9, can be decomposed into a single antecedent and the rules aggregated into a single consequent [22].

$$\text{IF } x \text{ is } \underset{\sim}{A_1} \text{ and } \underset{\sim}{A_2} \ldots \text{ and } \underset{\sim}{A_L} \text{ THEN } y \text{ is } \underset{\sim}{B_s} \tag{5.9}$$

Conjunctive antecedents can be rewritten as a new fuzzy set

$$\underset{\sim}{A_s} = \underset{\sim}{A_1} \cap \underset{\sim}{A_2} \cap \cdots \cap \underset{\sim}{A_L}$$

with the membership function

$$\mu_{\underset{\sim}{A_S}}(x) = min \left[\mu_{\underset{\sim}{A_1}}(x), \mu_{\underset{\sim}{A_2}}(x), \ldots, \mu_{\underset{\sim}{A_3}}(x) \right]$$

Rule 5.9 can then be rewritten as

$$\text{IF } \underset{\sim}{A_S} \text{ THEN } \underset{\sim}{B_S}$$

Disjunctive antecedents can similarly be written as

$$\underset{\sim}{A_s} = \underset{\sim}{A_1} \cup \underset{\sim}{A_2} \cup \cdots \cup \underset{\sim}{A_L}$$

$$\mu_{\underset{\sim}{A_S}}(x) = max \left[\mu_{\underset{\sim}{A_1}}(x), \mu_{\underset{\sim}{A_2}}(x), \ldots, \mu_{\underset{\sim}{A_3}}(x) \right]$$

The same principle can be applied to find the overall consequent when multiple rules apply. Conjunctive rules where both consequents must be applied can be found by the fuzzy intersection of all the rule consequents. Whereas disjunctive rules use the fuzzy union of the rule consequents.

5.4 A fuzzy logic model for COLREGs

Antecedents consequents and rules are needed in order to make a fuzzy system for a COLREGs compliant auto pilot. The model used in this thesis is created by Perera, Carvalho, and Soares [19] and has four antecedents, two consequents and an extensive rule set of nearly 200 rules. The antecedents used are: relative bearing to the target vessel, relative course of the target vessel, distance between the vessels, and the ratio between the vessels speed. The main vessels relative bearings are divided into ten sets. These resulting sectors represent the different sets in the relative course universe [0, 360]. The relative courses of the target vessel are ,likewise, divided into eight sectors, which represent sets in the relative course universe [0, 360]. The distance universe consist of three sets, representing radii from the main vessel, called R_A, R_B and R_{VD}. R_{VD} is the vessel domain into which other vessel shall not enter. R_A and R_B depicts the area around the main vessel in which other vessels are detected. R_A when the main vessel is the give way vessel and R_B when it is the stand on vessel. A graphical representation of the bearing, range and course sectors can be seen in figure 5.3. The final antecedent, speed ratio $= \frac{V_{target}}{Vmain}$, consist of three sets: $< 1, \approx 1$ and > 1. Visualizations of the antecedent FMFs, as well as their distribution in their corresponding universes, can be seen in figure 5.4.

Finally a rules set is needed to connect the antecedents with the consequents. The rule set used for this thesis is based on the rules, seen in table A.1 developed by Perera, Carvalho, and Soares [19]. A few rules have furthermore been added to handle overtake scenarios. These can be seen in table A.2

An analysis and optimization of the values used to specify the FMFs and the rules in the rule set are unfortunately outside of the scope for this thesis. The values are therefore used as specified in previous research [19], with

<div style="color:orange; border:1px solid orange;">Move section to Impl?</div>

<div style="color:orange; border:1px solid orange;">Explain tables?</div>

the exception of a few added rules to ensure collisions avoidance when one vessel is overtaking another.
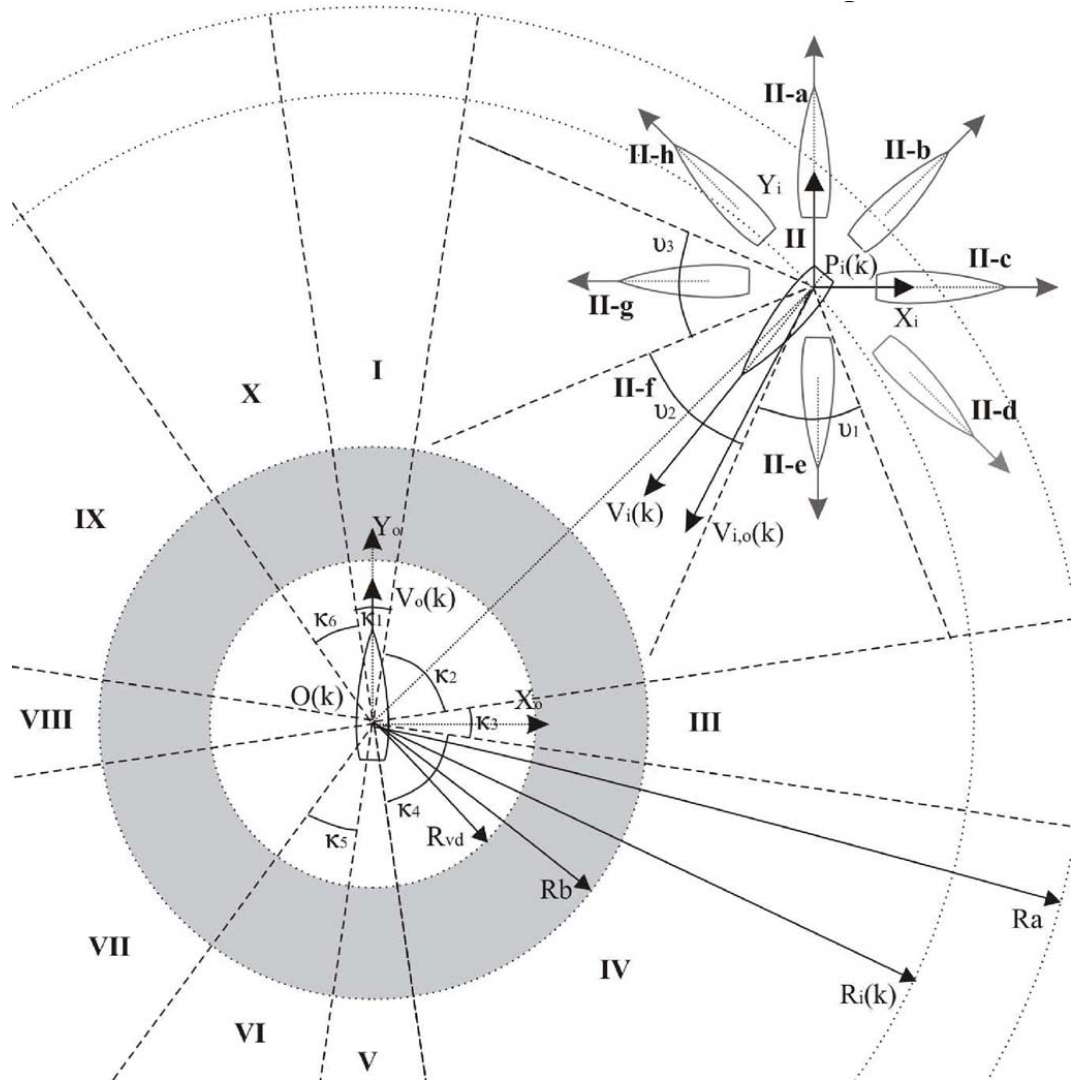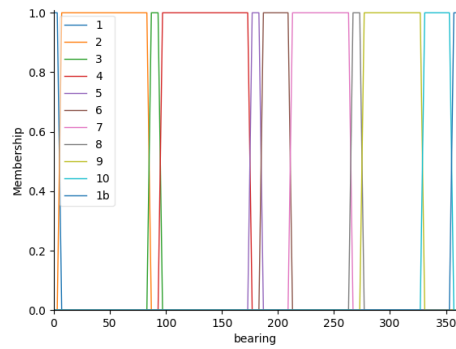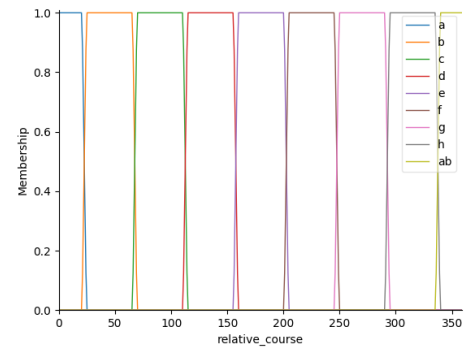

FIGURE 5.3: Mathematical representation of the inputs to a FIS for a two vessel collision situation[19]

(A) Relative bearing of the target vessel from the main vessel

(B) Relative course of the target vessel to the main vessel
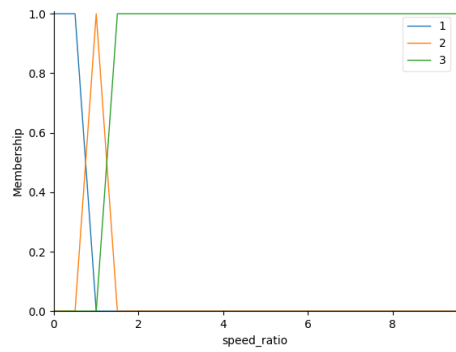
(C) Speed ratio target vessel/main vessel
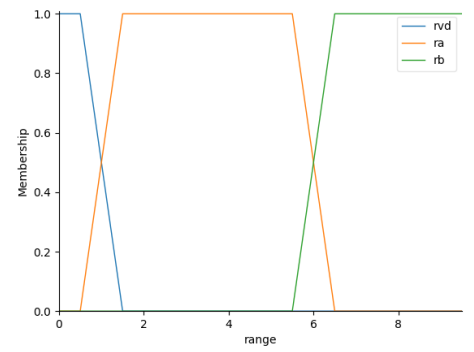
(D) Distance between the main vessel and the target vessel

FIGURE 5.4: Antecedent FMFs

(A) Course change
of the main vessel

(B) Speed change
of the main vessel
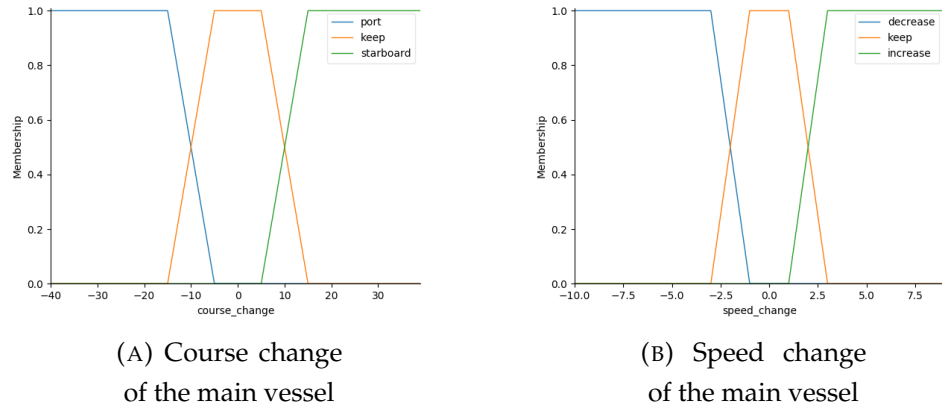
FIGURE 5.5: Consequent FMFs

5.5 Mamdani's fuzzy inference method

This thesis will, as mentioned in section 5.2, use the Mamdani implication operator in its Fuzzy Inference System (FIS). Mamdani's method was chosen since it is the most common one in practice literature [22].

The following subsections will explain Mamdani's method by going through an example based on the model presented in section 5.4. The example uses a scenario where two vessels are located in a 2 dimensional 10*10 NM Cartesian coordinate system.

Vessel A starts at coordinate (0; -4,5) with heading 0, speed 5 kts and rate of turn 2 degrees per second.

Vessel A starts at coordinate (-4,5; -4,5) with heading 203, speed 10 kts and rate of turn 2 degrees per second. The initial state of the scenario is depicted in figure 5.6
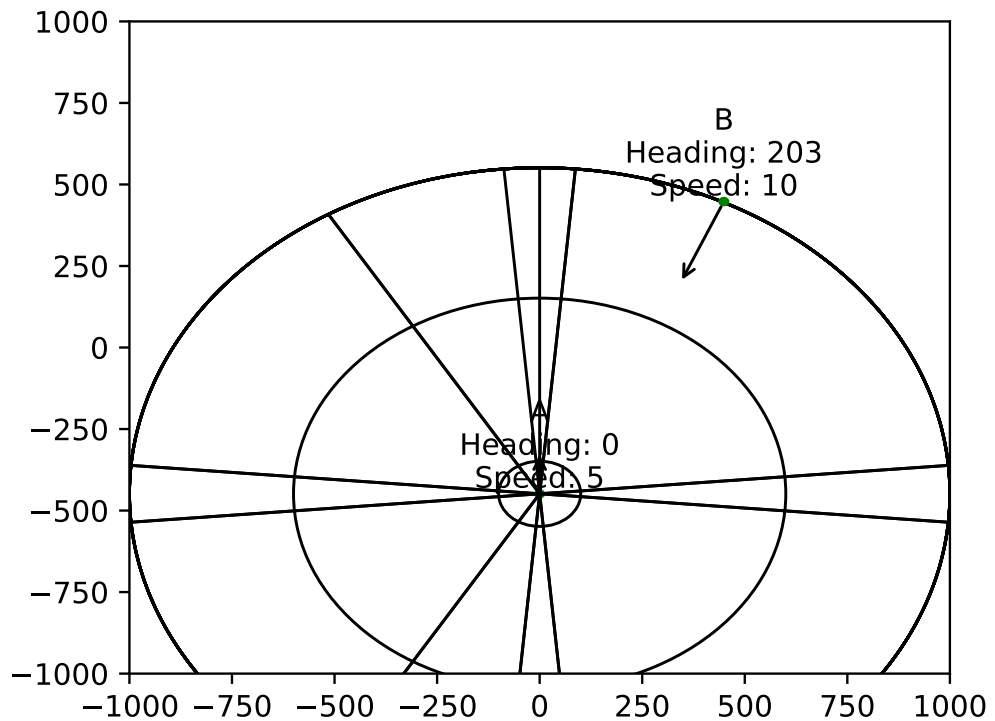
FIGURE 5.6: Initial state of the example scenario

A FIS can be said to consist of six steps [25]:

1. Rule determination

2. Input fuzzification

3. Combination of antecedents

4. Obtain consequences

5. Aggregate consequences

6. Defuzzification

This example will use a subset of the rules used in the real system. The rules used are:

Rule 1 IF target in sector II AND relative course is in sector f AND speed ratio
is greater than 1 AND range is $R_v d$ OR R_b OR R_a
THEN change course to starboard AND reduce speed.

Rule 2 IF target in sector II AND relative course is in sector e AND speed ratio
is greater than 1 AND range is $R_v d$ OR R_b OR R_a
THEN change course to starboard AND reduce speed.

Next the crisp values, gained when applying the model presented in
section 5.4 on the scenario in figure 5.6, are fuzzified. The target vessel is
located in sector II with a relative bearing of 26,7°, which results in a mem-
bership value of 1 in sector 2 as shown by the red line in figure 5.7a

The same principle is applied to the three other inputs: relativist course,
distance, and speed ratio. Distance and speed ratio will also, in this case,
yield membership values of 1, for the fuzzy sets distance R_a and speed ra-
tio>1 . However, the relative course of 203 is located in the fuzzy area be-
tween sector 2-e and 2-f which results in membership values of 0,4 and 0,6
respectively, as shown in figure 5.7b.

(A) Bearing
antecedent

(B) Relative course
antecedent
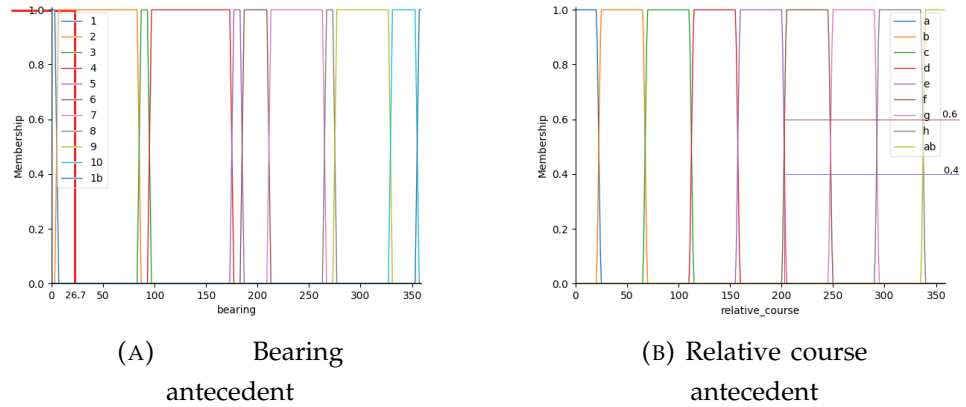
FIGURE 5.7: Fuzzified antecedent FMFs for the example sce-
nario

The membership values gained for the individual antecedents can then
be combined using the AND and OR operators specified in the rule set. This

results in the following calculations.

$$Rule1 : min(1; 0,6; 1; max(0; 0; 1)) = 0,6 \tag{5.10}$$

$$Rule2 : min(1; 0,4; 1; max(0; 0; 1)) = 0,4 \tag{5.11}$$

Mamdani's inference method is then used on each rule to obtain the consequent value. This is done by taking the minimum of the antecedent value from the previous step and the consequent. The result is a clipped consequent FMF, as seen in figures 5.8 and 5.9

(A) Course change
of the main vessel

(B) Speed change
of the main vessel
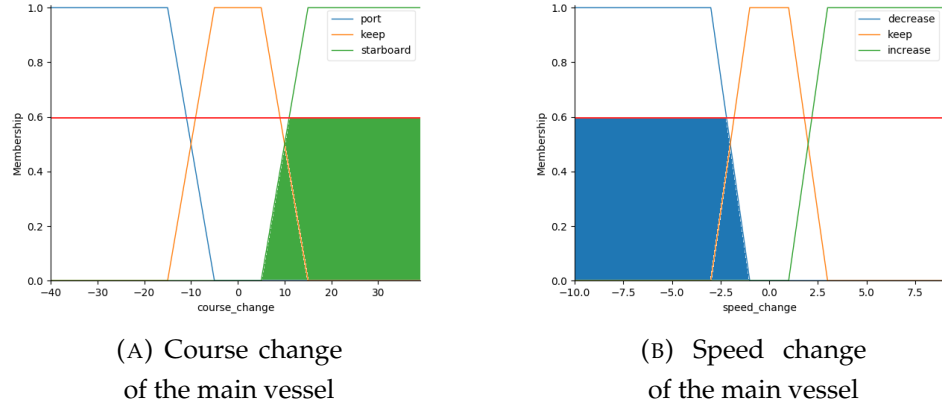
FIGURE 5.8: Consequent FMFs for rule 1 after applying Mamdani's inference method

(A) Course change
of the main vessel

(B) Speed change
of the main vessel
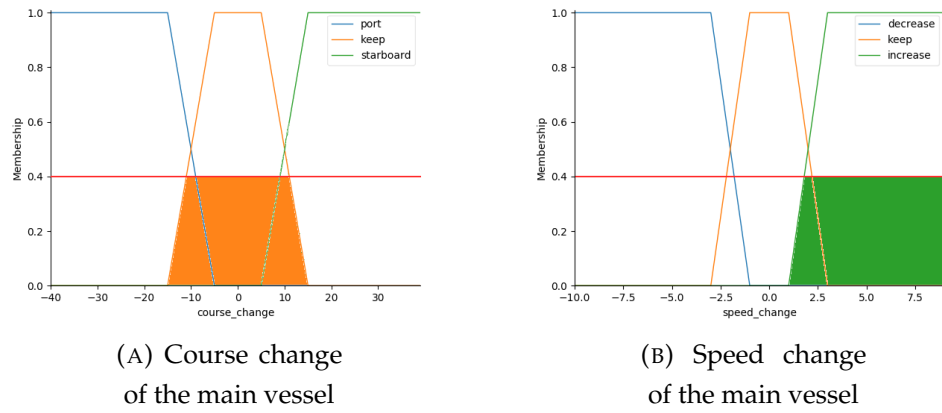
FIGURE 5.9: Consequent FMFs for rule 2 after applying Mamdani's inference method

The clipped consequents, one for each rule, must then be combined. This is achieved by taking the maximum of each consequent, to create a combined FMF. The result is presented in figure 5.10

(A) Course change
of the main vessel

(B) Speed change
of the main vessel
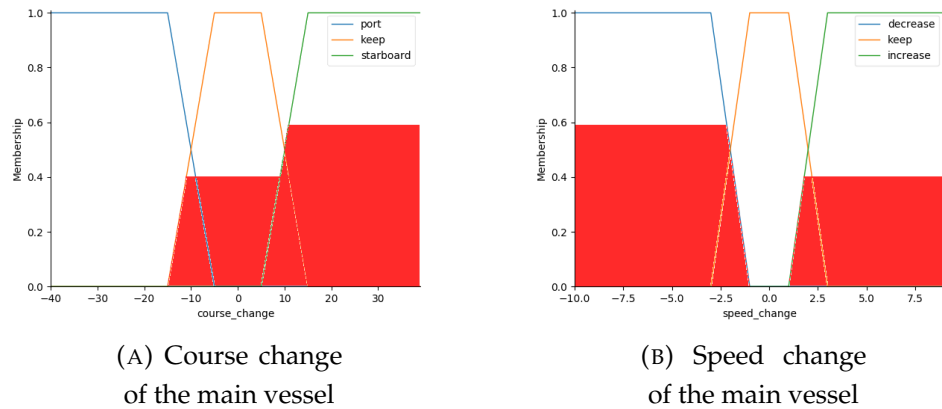
FIGURE 5.10: Combined Consequent FMFs for the example scenario

Finally the fuzzy sets gained from the previous step need to be converted into a crisp numerical value that can be used by the auto pilot. This process is called defuzzification and is in this case achieved by calculating the centroid value of FMFs in figure 5.10. The results are a +15.5°course change and a -1,7 knot speed change.

6 Implementation

The objective of this thesis was to implement and evaluate a collision avoid-ance algorithm for USVs. Several related approaches where analysed be-fore the fuzzy logic approach presented by Perera, Carvalho, and Soares [19], was chosen to be implemented. The solution presented in the origi-nal paper is implemented in MATLAB, while the solution presented here is implemented in Python. Python was chosen due to the writers previous knowledge of the language as well as the availability of fuzzy logic python libraries. The library used in this implementation is called SciKit-Fuzzy [26].

6.1 The fuzzy inference system

The fuzzy logic framework SciKit-Fuzzy is used to implement the model described in section 5.4. The framework provides a simple API which en-ables one to define Antecedents, Consequents, their corresponding FMFs as well as the rules that unite them.

First antecedents are initialized as shown in listing 1. This example cre-

```python
bear = ctrl.Antecedent(np.arange(0, 360, 1), 'bearing')
```

LISTING 1: Antecedent initialization

ates the antecedent *bearing* with a universe spanning from 0 to 360 with steps of 1. FMFs are then added to the antecedent. Listing 2 shows the FMF for sector II of the model. It starts at 5°relative bearing and ends at 85°, with a 5 °fuzziness on both sides.

The other antecedents and consequents are initialized and populated with FMFs in the same manner, according to the model described in sec-tion 5.4.

```
bear['2'] = fuzz.trapmf(bear.universe, generate_trapetzoid(5, 85, 5))
```

<div align="center">LISTING 2: FMF initialization</div>

Rules are defined in order to connect the antecedents with consequents. A rule consist of an antecedent and a consequent as follows:

```
Rule(antecedent, consequent)
```

Where the parameter *antecedent* can be a combination of antecedents using the & and | operators. An example based on Rule 1 in section 5.5 can be seen in listing 3.

```
Rule(
    'II' &
    'f' &
    '>1' &
    ('Rvd'|'Ra'|'Rb'),
    'starboard','decrease')
```

<div align="center">LISTING 3: Rule initialization</div>

These rules are then passed to the ControlSystem that each vessel utilizes to calculate proper course and speed corrections.

6.2 Architecture

This section will present the high level structure of the implementation with the help of the class diagram seen in figure 6.1. Each class and their interactions will be briefly presented.
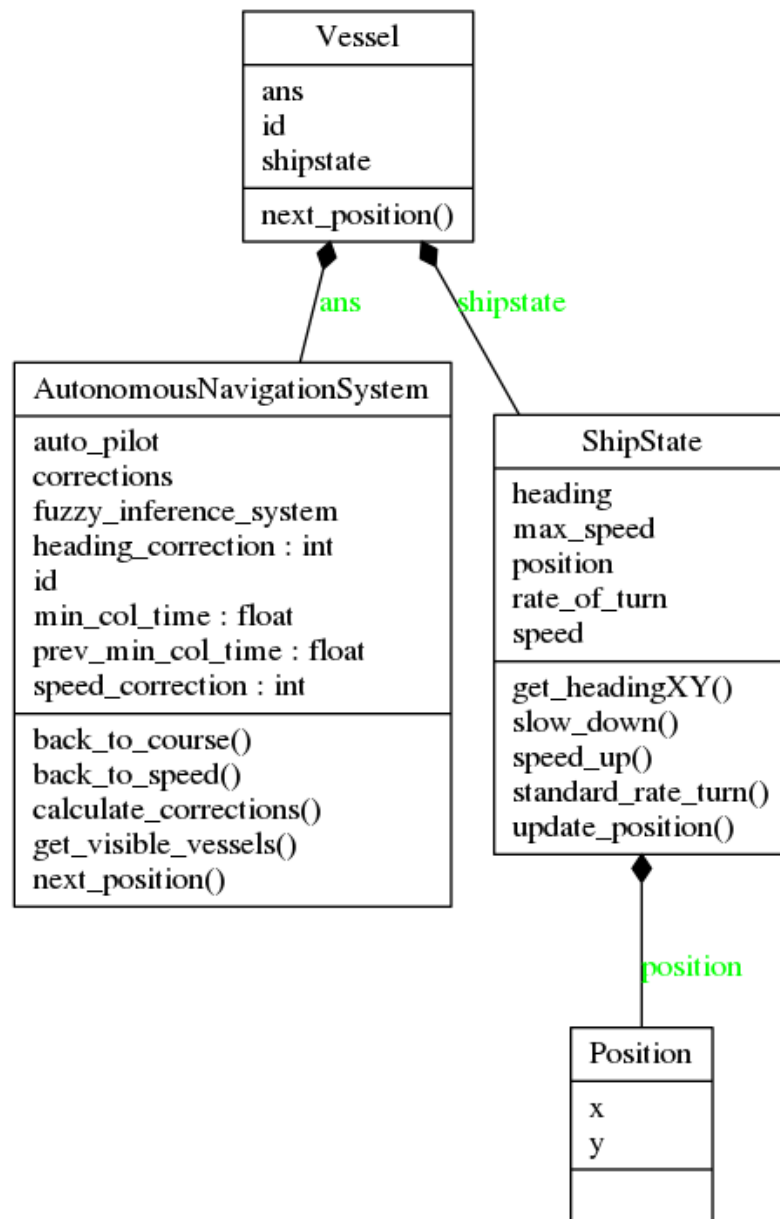
FIGURE 6.1: Class diagram

6.2.1 Classes

Each simulation scenario consists of at least one vessel. These vessels are represented by the *Vessel* class. The current state of the vessel is represented by the *ShipState* class which also contains the *Position* class. Furthermore, each vessel need a navigation system, which is represented by

the *AutonomousNavigationSystem* class. The classes will be more thoroughly presented in the following subsections.

Vessel

Vessel is the main class and the class with which the simulation script interacts. It contains, apart from the previously mentioned *AutonomousNavigationSystem* and *ShipState* an id and a method to calculate it's position in the next time frame. A new vessel object in created with the following call:

```
Vessel(id, heading, position_x, position_y, speed, max_speed,
    rate_of_turn, fuzzy_inference_system, auto_pilot)
```

This constructor call specifies the ID for the vessel as well as its initial position, speed and heading. Furthermore it defines the vessels maximum speed and rate of turn. The two final parameters specify the fuzzy inference system to use and whether the vessel shall use the navigation system. Setting the final boolean to false creates a rogue vessel that will just keep its initial speed and heading, thereby not complying to the COLREG rules. The *Vessel* class has one just one method, which calculates the vessels position in the next time frame after applying possible corrections to heading ans speed.

Position

The position class simply holds the vessels current coordinates in the Cartesian coordinate system used.

ShipState

ShipState holds information about the state of the vessel in the current time frame. This includes the vessels current position, heading and speed. The simulation does not distinguish between course and heading since no drift is simulated. Furthermore, limits such as maximum allowed speed and standard rate of turn is specified in this class. Finally, the class holds methods to change the ships heading by the specified standard rate of turn or speed by 1 kt, for the next time frame.

```
1  rel_course = observed_vessel.shipstate.heading - shipstate.heading
2  if rel_course < 0:
3      rel_course = 360 + rel_course
```

LISTING 4: Relative course calculation

AutonomousNavigationSystem

The AutonomousNavigationSystem class from now on referred to as ANS is what separates an autonomous vessel from an ordinary vessel. The ANS combines the information from the ShipState class with situational awareness information provided by a separate Situational Awareness (SA) module, in order to calculate needed corrections to speed and course.

The SA is in this simulation case represented by a service that holds all information regarding the current scenario. A real system would have a SA module that reads and processes information from different sensors, such as LiDAR, cameras, on board the vessel. Information needed about a target vessel is its heading, speed, and position. These are used to calculate the four different inputs to the FIS system. Listing 5 shows the method used to calculate the relative bearing fed into the FIS system. Compass bearing is first calculated from the two provided coordinate pairs after which the result is converted into a relative bearing. Relative course is then calculated as the observed vessels heading - the own vessels heading, as shown in listing 4. Distance can be obtained by using the Pythagorean theorem on the differences between the vessels in the X and Y axis. Finally speed ratio is defined as $\frac{\text{Speed of the observed vessel}}{\text{Speed of the own vessel}}$.

These inputs are fed into the FIS system, for each target vessels, and the recommended corrections are stored in an array. The recommendations might however contradict each other and a way to prioritize the recommendations is therefore needed.

Furthermore the expected time until collision needs to be calculated for each target vessel to prioritize the recommended corrections. Knowing the distance between the vessels, their relative velocity is needed to calculate the time. Equation 6.1 shows the relative velocity calculation, based on the law of Cosines. V stands for velocity, θ for heading, the subscript m for own

vessel and *t* for target vessel.

$$V_r = \sqrt{V_m^2 + V_t^2 - 2V_m V_t cos(\theta_m - \theta_t)} \tag{6.1}$$

6.2.2 Simulation

The goal of the implementation is to simulate a real word situation with respect to time, speed, acceleration, and rate of turn, while neglecting environmental factors such as weather. The simulation is therefore limited to two dimensions in a Cartesian coordinate system. The interval between time frames in the simulation is set to correspond to 1 second in the real world. This is handled by a for loop, that calls the vessels *next_position* method.

7 Scenarios

Use scenarios from http://advanced.ecolregs.com/index.php?option=com_k2&view=item&layout=item&id=109&Itemid=359&lang=en

8 Results

9 Discussion

10 Conclusion

A Fuzzy Inference system rule set

TABLE A.1: Rule set for a COLREGs compliant fuzzy inference system [19]

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
I	d	<1		<1	
		≈ 1		≈ 1	
		>1		>1	
	e	<1	SB	<1	SB
		≈ 1	SB	≈ 1	SB
		>1	SB	>1	SB
	f	<1		<1	
		≈ 1		≈ 1	
		>1		>1	
II	e	<1		<1	
		≈ 1	Acc	≈ 1	Acc
		>1	Acc	>1	Acc
	f	<1		<1	
		≈ 1	SB, Dec	≈ 1	SB, Dec
		>1	SB, Dec	>1	SB, Dec
	g	<1		<1	
		≈ 1	SB	≈ 1	SB
		>1	SB	>1	SB
III	f	<1		<1	
		≈ 1	Acc	≈ 1	Acc
		>1	Acc	>1	Acc

SB= Starboard	P= Port	Empty decision = No change
Acc = Accelerate		Dec = Decelerate

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
	g	<1		<1	
		≈ 1	Dec	≈ 1	Dec
		>1	Dec	>1	Dec
	h	<1		<1	
		≈ 1	Dec	≈ 1	Dec
		>1	Dec	>1	Dec
IV	g	<1		<1	
		≈ 1	Acc	≈ 1	Acc
		>1	Acc	>1	Acc
	h	<1		<1	
		≈ 1	P, Dec	≈ 1	P, Dec
		>1	P, Dec	>1	P, Dec
	a	<1		<1	
		≈ 1	P, Dec	≈ 1	P, Dec
		>1	P, Dec	>1	P, Dec
V	h	<1		<1	
		≈ 1		≈ 1	
		>1		>1	
	a	<1		<1	
		≈ 1	P	≈ 1	P
		>1	P	>1	P
	b	<1		<1	
		≈ 1		≈ 1	
		>1		>1	
VI	a	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
	b	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	

SB= Starboard	P= Port	Empty decision = No change
Acc = Accelerate		Dec = Decelerate

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
	c	<1		<1	
		≈ 1	Acc	≈ 1	
		>1	Acc	>1	
VII	a	<1		<1	
		≈ 1	SB	≈ 1	
		>1	SB	>1	
	b	<1		<1	
		≈ 1	SB, Dec	≈ 1	
		>1	SB, Dec	>1	
	c	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
VIII	b	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
	c	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
	d	<1		<1	
		≈ 1	Acc	≈ 1	
		>1	Acc	>1	
IX	c	<1		<1	
		≈ 1	P	≈ 1	
		>1	P	>1	
	d	<1		<1	
		≈ 1	P, Dec	≈ 1	
		>1	P, Dec	>1	
	e	<1		<1	
		≈ 1	Acc	≈ 1	
		>1	Acc	>1	

SB= Starboard	P= Port	Empty decision = No change
Acc = Accelerate		Dec = Decelerate

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
X	c	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
	d	<1		<1	
		≈ 1	Dec	≈ 1	
		>1	Dec	>1	
	e	<1		<1	
		≈ 1	Acc	≈ 1	
		>1	Acc	>1	

SB= Starboard   P= Port   Empty decision = No change

Acc = Accelerate   Dec = Decelerate

TABLE A.2: Rules regarding COLREG rule 15 added to the rule set developed by Perera, Carvalho, and Soares [19]

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
I	a	<1	SB	<1	SB
		≈ 1		≈ 1	
		>1		>1	
	h	<1	SB	<1	SB
		≈ 1		≈ 1	
		>1		>1	
	b	<1	SB	<1	SB
		≈ 1		≈ 1	
		>1		>1	
II	a	<1	P	<1	P
		≈ 1		≈ 1	
		>1		>1	
X	a	<1	SB	<1	SB

SB= Starboard   P= Port   Empty decision = No change

Acc = Accelerate   Dec = Decelerate

Bearing	Course	Range($R_{VD} - R_A$)		Range($R_A - R_B$)	
		Speed ratio	Decision	Speed ratio	Decision
		≈ 1		≈ 1	
		>1		>1	

SB= Starboard	P= Port	Empty decision = No change
Acc = Accelerate		Dec = Decelerate

B Python Code

```python
def theta_to_nav(theta):
    heading = theta + 90
    heading = heading - theta * 2
    if heading > 360:
        heading = heading - 360
    elif heading < 0:
        heading = heading + 360
    return heading

def compass_to_relative(compass, heading):
    x = compass - heading
    if x < 0:
        return x + 360
    else:
        return x

def calculate_initial_compass_bearing(pointA, pointB):
    if (type(pointA) != tuple) or (type(pointB) != tuple):
        raise TypeError("Only tuples are supported as arguments")
    if pointB[0] > pointA[0]:
        x_diff = abs(pointA[0] - pointB[0])
    else:
        x_diff = -abs(pointA[0] - pointB[0])

    if pointB[1] > pointA[1]:
        y_diff = abs(pointA[1] - pointB[1])
    else:
        y_diff = -abs(pointA[1] - pointB[1])

    initial_bearing = math.atan2(y_diff, x_diff)

    initial_bearing = math.degrees(initial_bearing)
    initial_bearing = theta_to_nav(initial_bearing)
    compass_bearing = (initial_bearing + 360) % 360
    return compass_bearing

def cartesian_coords_to_compass(main_vessel, observed_vessel):
    main_observed = calculate_initial_compass_bearing(
        (main_vessel.position.x
        main_vessel.position.y),
        (observed_vessel.position.x,
        observed_vessel.position.y))

    return main_observed

def cartesian_coords_to_relative(main_vessel, observed_vessel):
    main_observed= cartesian_coords_to_compass(main_vessel,
     observed_vessel)

    main_observed = compass_to_relative(main_observed,
     main_vessel.heading)

    return main_observed
```

LISTING 5: Methods used to calculate the relative bearing from a ship to another from their heading and Cartesian coordinates

Bibliography

[1]   European Marine Casualty Information Platform, "Annual Overview of Marine Casualties and Incidents 2016", 2016.

[2]   B. Luin and T. Brcko, "Framework for Simulation of Collision Avoidance Strategies", in *Proc. of the 24th International Symposium on Electronics in Transport (ISEP 2016)*, 2016.

[3]   E Jokioinen, "Remote and autonomous ships-the next steps", London: AAWA, Tech. Rep., 2016.

[4]   J. E. Manley, "Unmanned surface vehicles, 15 years of development", in *OCEANS 2008*, IEEE, 2008, pp. 1–4.

[5]   Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges", *Annual Reviews in Control*, vol. 41, pp. 71–93, 2016.

[6]   J. Park, M. Kang, T. Kim, S. Kwon, J. Han, J. Wang, S. Yoon, B. Yoo, S. Hong, Y. Shim, *et al.*, "Development of an Unmanned Surface Vehicle System for the 2014 Maritime RobotX Challenge", *Journal of Field Robotics*, vol. 34, no. 4, pp. 644–665, 2017.

[7]   J. Yuh, G. Marani, and D. R. Blidberg, "Applications of marine robotic vehicles", *Intelligent Service Robotics*, vol. 4, no. 4, p. 221, 2011, ISSN: 1861-2784. DOI: 10.1007/s11370-011-0096-5. [Online]. Available: https://doi.org/10.1007/s11370-011-0096-5.

[8]   The National Ocean Service, *How deep is the ocean?*, https://oceanservice.noaa.gov/facts/oceandepth.html, [Online; accessed 25.09.2017], U.S. Department of Commerce.

[9]   The National Ocean Service, *How much of the ocean have we explored?*, https://oceanservice.noaa.gov/facts/exploration.html, [Online; accessed 25.09.2017], U.S. Department of Commerce.

[10] M. R. Benjamin, J. J. Leonard, J. A. Curcio, and P. M. Newman, "A method for protocol-based collision avoidance between autonomous marine surface craft", *Journal of Field Robotics*, vol. 23, no. 5, pp. 333–346, 2006.

[11] International Maritime Organization, *Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs)*, `http://www.imo.org/en/About/conventions/listofconventions/pages/colreg.aspx`, [Online; accessed 07.11.2017].

[12] International Maritime Organization, *Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs) 30/1977*, `http://www.jag.navy.mil/distrib/instructions/COLREG-1972.pdf`, [Online; accessed 08.11.2017], 1977.

[13] J. Larson, M. Bruch, R. Halterman, J. Rogers, and R. Webster, "Advances in autonomous obstacle avoidance for unmanned surface vehicles", SPACE AND NAVAL WARFARE SYSTEMS CENTER SAN DIEGO CA, Tech. Rep., 2007.

[14] M. R. Benjamin and J. A. Curcio, "COLREGS-based navigation of autonomous marine vehicles", in *Autonomous Underwater Vehicles, 2004 IEEE/OES*, IEEE, 2004, pp. 32–39.

[15] W. Naeem, G. W. Irwin, and A. Yang, "COLREGs-based collision avoidance strategies for unmanned surface vehicles", *Mechatronics*, vol. 22, no. 6, pp. 669–678, 2012.

[16] A. Kreutzmann, D. Wolter, F. Dylla, and J. H. Lee, "Towards safe navigation by formalizing navigation rules", *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, vol. 7, no. 2, pp. 161–168, 2013.

[17] D. Wolter, F. Dylla, and A. Kreutzmann, "Rule-compliant navigation with qualitative spatial reasoning", in *Robotic Sailing*, Springer, 2011, pp. 141–155.

[18] S.-M. Lee, K.-Y. Kwon, and J. Joh, "A fuzzy logic for autonomous navigation of marine vehicles satisfying COLREG guidelines", *International Journal of Control, Automation, and Systems*, vol. 2, no. 2, pp. 171–181, 2004.

[19] L. Perera, J. Carvalho, and C. G. Soares, "Intelligent ocean navigation and fuzzy-bayesian decision/action formulation", *IEEE Journal of Oceanic Engineering*, vol. 37, no. 2, pp. 204–219, 2012.

[20] L. A. Zadeh, "Fuzzy sets", in *Fuzzy Sets, Fuzzy Logic, And Fuzzy Systems: Selected Papers by Lotfi A Zadeh*, World Scientific, 1996, pp. 394–432.

[21] G. Chen and T. T. Pham, *Introduction to fuzzy sets, fuzzy logic, and fuzzy control systems*. CRC press, 2000.

[22] T. J. Ross, *Fuzzy logic with engineering applications*. John Wiley & Sons, 2009.

[23] H.-J. Zimmermann, "Fuzzy set theory", *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 3, pp. 317–332, 2010.

[24] M. Ying, "Implication operators in fuzzy logic", *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 1, pp. 88–91, 2002.

[25] R. B. Knapp, *Fuzzy inference systems*, http://www.cs.princeton.edu/courses/archive/fall07/cos436/HIDDEN/Knapp/fuzzy.htm, [Online; accessed 12.02.2018].

[26] J. Warner, J. Sexauer, scikit fuzzy, twmeggs, A. M. S., A. Unnikrishnan, G. Castelão, F. Batista, T. G. Badger, and H. Mishra, *JDWarner/scikit-fuzzy: Scikit-Fuzzy 0.3.1, month = oct, year = 2017, doi = 10.5281/zenodo.1002946, url = https://doi.org/10.5281/zenodo.1002946.*