

A Multiobjective Optimization Approach for COLREGs-compliant Path Planning of Autonomous Surface Vehicles Verified on Networked Bridge Simulators

Liang Hu, Wasif Naeem, Eshan Rajabally, Graham Watson, Terry Mills, Zakirul Bhuiyan, Craig Raeburn, Ivor Salter and Claire Pekcan

Abstract—This paper presents a multiobjective optimisation approach for path planning of autonomous surface vehicles (ASVs). A unique feature of the technique is the unification of the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) with good seamanship’s practice alongside hierarchical (rather than simultaneous) inclusion of objectives. The requirements of collision avoidance are formulated as mathematical inequalities and constraints in the optimisation framework and thus collision-free manoeuvres and COLREGs-compliant behaviours are provided in a seafarer-like way. Specific expert knowledge is also taken into account when designing the multiobjective optimisation algorithm. For example, good seamanship reveals that if allowed, an evasive manoeuvre with course changes is always preferred over one with speed changes in practical maritime navigation. As a result, a hierarchical sorting rule is designed to prioritize the objective of course/speed change preference over other objectives such as path length and path smoothness, and then incorporated into a specific evolutionary algorithm called hierarchical multiobjective particle swarm optimisation (H-MOPSO) algorithm. The H-MOPSO algorithm solves the real-time path planning problem through finding solutions of the formulated optimisation problem. The effectiveness of the proposed H-MOPSO algorithm is demonstrated through both desktop and high-fidelity networked bridge simulations.

Index Terms—Autonomous surface vehicles, collision avoidance, path planning, multi-objective optimisation, COLREGs

I. INTRODUCTION

In recent years, the study and development of autonomous surface vehicles (ASVs) has become an active area of research [1], [2]. One of the basic requirements for ASVs is that they should navigate (by themselves) safely and avoid collisions with other ships/obstacles or with land features. There has been much recent discussion on the future of unmanned ships

and the requirement for collision avoidance is clearly elevated for the potential of using ASVs in the maritime transportation industry [3]. Furthermore, to operate harmoniously with other ships (either manned or unmanned), an ASV should behave in a manner similar to that of other ships in the vicinity. Since all manned craft are required to adhere to the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) defined by the International Maritime Organisation (IMO) [4], it is imperative to impose COLREGs-compliant behaviour as an integral element of any ASV navigational system.

The rules in COLREGs have been written for manned vessels, i.e. for human consumption and are thus not trivial to programme or automate. Due to their subjective nature, COLREGs are subject to various interpretations causing uncertainty, which in the worst case can lead to collisions. Minimising collisions between the vessels requires consistent understanding of COLREGs that is executable by ASVs with little or no human intervention. Motivated by such a navigational requirement, our aim is to develop a local path-planner that is capable of generating collision-free and COLREGs-compliant paths automatically and in real time.

In practice, depending on mission requirements, a number of objectives/preferences such as safety, efficiency, short-distance and comfort, may need to be considered when planning a path for ASVs [5]–[7]. For instance, it is desired that an ASV for mine counter measure operations does not deviate far from pre-specified areas, and an ASV used as cruise ship should manoeuvre smoothly without abrupt course/speed changes. Nonetheless, it should be noted that besides all the mentioned objectives, collision avoidance should always be considered as a hard constraint that should be satisfied during path planning. In addition, from a good seamanship viewpoint, an evasive manoeuvre with course changes alone is always preferred over one involving speed changes. The simple reason is that course changes are much more visible to a naked eye than speed changes. How to address the above-mentioned multiple objectives, and more importantly to take into account the priorities among all objectives and good seamanship in the path planner constitute the second motivation of our research in this paper.

A variety of path planning techniques with consideration of COLREGs have been developed in recent years, such as

This work was supported by the Innovate UK project “Robust Machine Executable Collision Regulations at Sea” under the grant TSB 102308.

L. Hu is with School of Computer Science and Informatics, De Montfort University, Leicester, UK. (e-mail: liang.hu@dmu.ac.uk)

W. Naeem is with School of Electronics, Electrical Engineering and Computer Science, Queen’s University Belfast, Belfast, UK (e-mail: w.naeem@qub.ac.uk)

E. Rajabally and G. Watson are with Rolls Royce Future Technology Group, Derby, UK (e-mail: {Eshan.Rajabally, Graham.Watson}@rolls-royce.com)

T. Mills, Z. Bhuiyan, C. Raeburn, I. Salter C. Pekcan are with Warsash Maritime Academy, Southampton Solent University, Southampton, UK (e-mail: {terry.mills, zakirul.bhuiyan, craig.raeburn, ivor.salter, claire.pekcan}@solent.ac.uk)

artificial potential fields [8], [9], evolutionary algorithms [10], fuzzy logic [11]–[14], interval programming [15], velocity obstacle method [16], model predictive control [17] and heuristic A^* method [18], to name a few. However, most, if not all of the existing techniques do not scale well to multiple target ships and multiple/conflicting COLREGs rules. Exceptions are fuzzy rule-based methods as in [13], [14], however, a single objective is considered when using those techniques. In addition, fuzzy rule-based methods exhibit failures due to inferred contradictory decisions. Recently, [14] presents solutions to overcome such problems in fuzzy-based methods, but it remains an open and challenging problem to optimize parameters in the fuzzy membership functions.

On the other hand, a number of multiobjective evolutionary algorithms have been developed [5]–[7], [19] to tackle multiple objectives simultaneously in path planning. These results are developed either as general robotic motion planning algorithms [5], [7], or applied in other kinds of autonomous vehicles including aerial and ground vehicles [6], [19]. One direct challenge of adapting existing multiobjective optimisation approaches for ASVs' path planning lies in how to incorporate COLREGs into the optimisation framework. More importantly, note that all the objectives are considered simultaneously and equally in all the above research work. However, as their overwhelming priority, mariners adopt paths that are safe. The practitioners' choice implies that the traditional "equal objective" approach may not be suitable for autonomous collision avoidance since it will not be considered as seaman-like. Instead a new hierarchical non-dominated sorting (Hi-NDS) rule is introduced in this paper, which prioritizes the course/speed change objective over other objectives.

In this paper, the path planning problem is formulated as a multiobjective optimisation problem (MOP) where a solution of the MOP represents a feasible path. In the proposed multiobjective optimisation framework, COLREGs rules and practical constraints such as collision avoidance and speed limits are interpreted as mathematical inequalities of decision variables, and viewed as constraints in the decision space that can be explicitly incorporated within a generic multiobjective evolutionary optimization algorithm. Additionally, objective functions are formulated as an integral part of the multiobjective optimisation framework. Particularly, the safety objective is deliberately designed such that it is in accordance with the commonly used closest point of approach (CPA) based risk assessment criterion in maritime navigation.

In order to solve the above problem, a well known evolutionary algorithm called multiobjective particle swarm optimisation (MOPSO) [20] is chosen due to its merits of fast convergence and rather straightforward implementation. Our proposed Hi-NDS rule, which seeks a collision-free with course-change-only preference, is incorporated into the standard MOPSO algorithm. The proposed variant of MOPSO algorithm called hierarchical MOPSO or H-MOPSO, is able to find collision-free and COLREGs-compliant paths by optimising multiple objective functions.

The contribution of this paper is threefold: 1) A multiobjective optimisation framework is developed for path planning, which is flexible and scalable to accommodate multiple target

ships and objective functions; 2) A novel and unified representation in the form of mathematical inequalities is proposed for COLREGs rules selection and other ASV constraints; 3) A modified MOPSO algorithm is proposed such that the hierarchical priorities among different objectives are taken into consideration explicitly during path planning. The proposed algorithm has been tested on high-fidelity networked bridge simulator that simulate various ship-encounter scenarios under different environmental conditions. Nonetheless, the research in this paper is only a first step of ASV development. To make fully autonomous surface vehicles ready for real-world applications, more dedicated simulation/test methods such as naturalistic experimental method [21]–[23], should be used.

The rest of the paper is organised as follows. In Section II we briefly review some preliminary knowledge of risk assessment and COLREGs rule selections that are relevant to path planning, and prove a lemma that will be used in subsequent sections. Section III presents how the path planning problem is formulated as a MOP. Section IV describes the proposed path planning algorithm for ASV's navigation. Several results from desktop simulation and ships bridge simulator are presented in Section V. Section VI summarises the research in the paper with some concluding remarks.

II. PRELIMINARIES OF MARITIME NAVIGATION

A complete autonomous maritime navigation system consists of a global off-line (sometimes referred to as the mission planner) and a local online path planning modules, as depicted in the flowchart of Fig. 1. Given the destination waypoint, the global path planner generates the desirable path off-line, and presents it as a sequence of waypoints. The online path planning module will only be activated if any obstacles posing a collision risk are detected. When a risk of collision is confirmed, a collision-free and COLREGs-compliant local path (from current location to the next waypoint) will be generated which is the main focus in this paper. In the following, we briefly introduce two functional modules that are relevant to path planning: risk assessment and COLREGs rules selection, and then present the specifications that should be considered when re-planning a path.

A. Risk Assessment

This module is only activated if a target/obstacle is detected and a detection range can thus be set to reflect sensor capability. To assess a risk of collision, the widely-used closest point of approach (CPA) method has been adopted for evaluating whether there is a collision risk in the near future [16], [18], [24]. Briefly speaking, this method compares the time to closest point of approach (TCPA) and the distance to closest point of approach (DCPA) with prescribed parameters t_{\min} and d_{\min} , respectively, where t_{\min} and d_{\min} are related to the vessel type and the environment where she is being operated.

A risk of collision is deemed to exist and labelled as risk = 1 (see Fig. 1) if

$$0 \leq \text{TCPA} \leq t_{\min} \quad \text{and} \quad \text{DCPA} \leq d_{\min}. \quad (1)$$

Otherwise, risk = 0 indicating no risk.

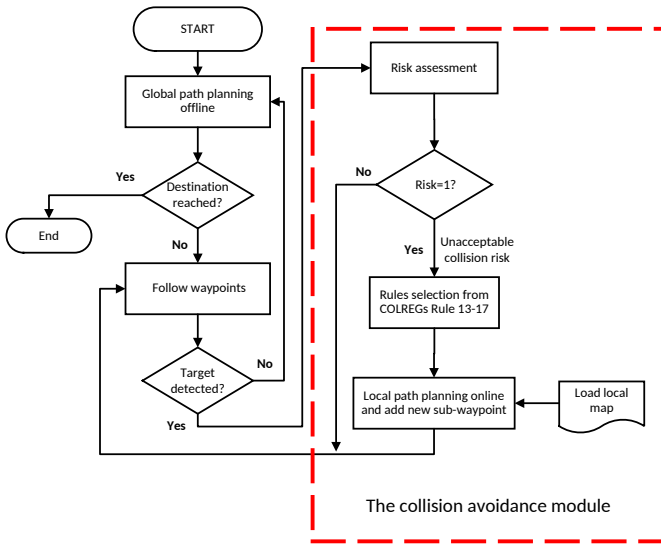


Fig. 1. A flowchart of autonomous maritime navigation

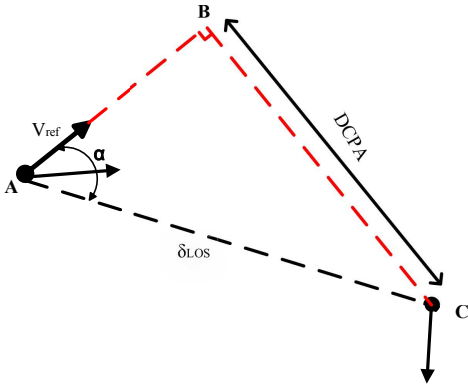


Fig. 2. Closest point of approach (CPA) calculation, adapted from [18]

A property of DCPA and TCPA is presented in the following lemma, which will be used to develop the proposed path planning algorithm with a low computational complexity.

Lemma 1: Given the line-of-sight distance δ_{LOS} , the relative velocity V_{ref} between two vessels at time instant $t = 0$, the angle α between the line-of-sight and the relative heading, as shown in Fig. 2, if both vessels maintain constant course and speed of their own during the time interval $[0, T]$, then the following properties of DCPA and TCPA during time interval $[0, T]$ hold: 1). DCPA(t) is time-invariant; 2). TCPA(t) is a continuous function of time t , and $\min TCPA(t) = \frac{\delta_{LOS} \cos \alpha}{V_{ref}} - T$ if $T \leq \frac{\delta_{LOS} \cos \alpha}{V_{ref}}$; Otherwise $\min |TCPA(t)| = 0$.

The proof of the above lemma is provided in Appendix A.

B. COLREGs rules selection

Once a collision risk deemed to exist, the next stage is to determine which type of encounter, i.e., “head-on”, “crossing” or “overtaking” and which type of action, i.e., “give-way” or “stand-on” should be applied. By choosing an appropriate COLREGs rule for the specific encounter scenario, the decision making of the vessel is ensured to be compliant with the rules all the time. A brief summary of the relevant COLREGs rules [4] is provided in Appendix B. Given the relative bearing

and heading of the target vessel to the ownship, it is then possible to uniquely determine a specific COLREGs rule that may apply [25]. In particular, according to the rules, in a “head-on” situation, both vessels are required to manoeuvre starboard; in a “crossing” situation, the give-way vessel should manoeuvre starboard while the stand-on vessel should keep the course and speed. Moreover good seaman practice indicates that in an “overtaking” encounter, the overtaking vessel can manoeuvre either side but preference is given to the starboard side if possible, and the vessel being overtaken should keep the course and speed.

Once a risk is confirmed and appropriate COLREGs rule is selected, the next step is to plan an evasive path by generating new sub-waypoints. Specifications for generating the sub-waypoints are provided in the following section. Though the local planner “looks ahead” into the future, it is constrained by the limited detection range around the ownship. It is therefore possible to construct a case where the vessel is trapped because of short-sight. However, such a situation is unlikely to occur in practice for local path planning, where there are typically only a few vehicles that need to be avoided simultaneously. If one needs to generate a path in a maze-like environment while avoiding moving hazards, it can be combined with a global path planner that takes into account the obstacles at a longer range.

C. Specifications in ASV’s path planning

An evasive trajectory can be planned either by generating one or a sequence of sub-waypoints. In our research a single evasive waypoint is preferred as it is computationally efficient and hence is well suited for real-time applications. Fig. 3 illustrates the basic idea of the proposed local path planning process where a simple 1-1 head-on encounter situation is depicted. The shaded area represents the safe working area that the ASV is constrained to travel within, while the non-shaded area represents the no-go area that is prohibited to travel due to various reasons such as presence of shallow water or landmass. At the beginning, the ASV at location A follows the nominal path \overline{AC} towards waypoint C with speed s and heading angle θ . When the ASV detects a risk of collision with a target ship, then the local path planner generates a new sub-waypoint B. Subsequently, the ASV alters its heading angle from θ to θ' and traverses the new path \overline{AB} until arriving at sub-waypoint B, finally returning to the original waypoint C. Note that the other alternative sub-waypoint B' is unacceptable since sections of the paths $\overline{AB'}$ and $\overline{B'C}$ fall outside the restricted area.

For path planning, an ideal path is usually expected to be 1) safe, keeping clear of any other targets/obstacles; 2) smooth, avoiding abrupt changes in the course; and 3) shortest, when possible. In addition, from the mariners’ perspective, course changing manoeuvres are preferred over speed changes when avoiding a collision. This lends the ASV’s path planning naturally to a MOP. Besides the above-mentioned objectives, the following constraints should also be considered provided there is sufficient sea-room:

- course alteration should neither be less than a minimum

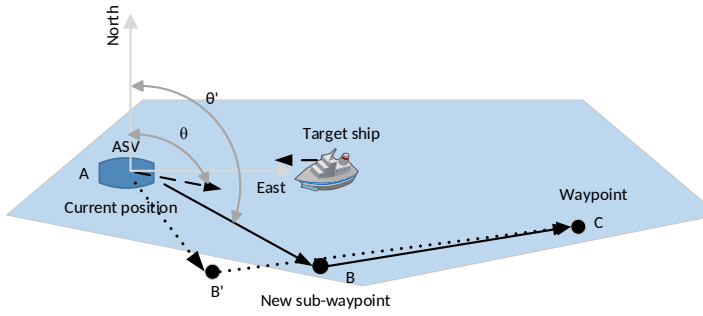


Fig. 3. The illustration of local path planning

acceptable threshold $\Delta\theta_{\min}$ nor bigger than a maximum value $\Delta\theta_{\max}$;

- the demanded speed should not be bigger than the maximum speed v_{\max} the vessel can reach;
- manoeuvres to starboard are favoured over to port if both sides are allowed e.g. in overtaking encounters.

The above constraints are explicitly imposed by either COLREGS rules or good seamanship or by ship dynamics.

On the other hand, the COLREGS rules selected by the rule selector also impose constraints on changes in the ASV's course. For example, "starboard manoeuvring" implies that the course alteration is positive according to the local north-east coordinate. The difficulty with COLREGS is that there are no quantitative constraints provided in the rules and mariners may interpret the rules in a different manner which has resulted in near misses and in the worst case caused collisions. Automating such rules poses significant challenges especially in autonomous vessels. To overcome such a difficulty, an optimisation framework is proposed in the next section, in which COLREGS rules and constraints are represented as mathematical inequalities.

III. PATH PLANNING FORMULATED AS A MULTIOBJECTIVE OPTIMIZATION PROBLEM

A general multi-objective optimisation framework includes three elements: decision variables, objective functions and constraints. In the following, we present how to formulate the local path planning problem as a MOP.

A. Decision variables

Define the decision variable by the vector x

$$x := [\theta', \quad s', \quad T]^T, \quad (2)$$

where θ' denotes the new heading angle of ASV for collision avoidance (i.e. the heading of vector \overrightarrow{AB} in Fig. 3), s' denotes the demanded speed of the ASV from current position to the new sub-waypoint B whilst T represents the time the ASV takes to reach the new sub-waypoint B , respectively.

B. Objective functions

Four objectives/preferences used in path planning are formulated in this subsection. By minimising the objective functions, the search of the algorithm is directed towards optimal paths.

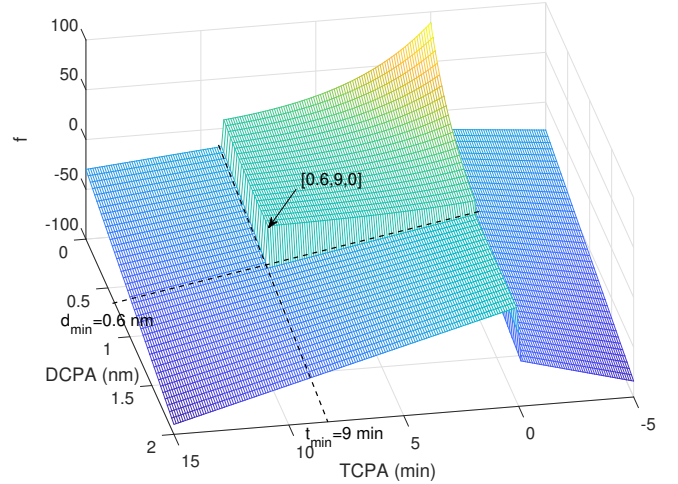


Fig. 4. The safety objective function

1. **Margin of safety:** this objective function employs the ASV's safety parameters (TCPA and DCPA) with respect to a target vessel to estimate the risk of collision. For an open water environment, the following function is chosen to describe the safety objective:

$$f(x) = \max_{1 \leq i \leq n} f_i(x), \quad (3)$$

where $f_i(x)$ defined in (4) is adapted from [26], [27], d_{\min} and t_{\min} are the pre-specified thresholds, and $DCPA_i(x)$ and $TCPA_i(x)$ are obtained from the pre-process of risk assessment, $i = 1, 2, \dots, n$ represents the i^{th} target vessel posing a collision risk, and a is a constant scaling parameter.

From (4), it is found that 1) if the safety objective $f \leq 0$, then there is no risk of collision; 2) otherwise there is a risk of collision and 3) the smaller the safety objective f , the safer the candidate path is. In robotic path planning literature, the safety objective is typically a single variable function of the clearance distance [26]. However, since both DCPA and TCPA play equally important roles in maritime navigation, in this paper $f_i(x)$ is constructed as a function of two variables DCPA and TCPA, as plotted in Fig. 4 for a specific case of $t_{\min} = 9$ min and $d_{\min} = 0.6$ nm. The constructed objective function suitably reflects the safety requirement in maritime transportation owing to its following properties:

- For a given TCPA, the function f monotonically decreases with increasing DCPA, i.e., $DCPA \nearrow, f \searrow$;
- For a given DCPA, when $TCPA \geq 0$, the function f monotonically decreases with increasing TCPA, i.e., $TCPA \nearrow, f \searrow$; when $TCPA \leq 0$, the function f monotonically increases with increasing TCPA, i.e., $TCPA \nearrow, f \nearrow$;
- The function $f = 0$ when $TCPA = t_{\min}$ and $DCPA = d_{\min}$

The above properties of the objective function f reflect the safety requirement in path planning, i.e. a path with larger DCPA and TCPA (TCPA is absolute here as it can be negative) is always safer than another path with smaller values. The proposed function f is thus suitable for the maritime path planning problem.

Next we consider the objective function for a restricted water environment where the surrounding features such as landmass and shallow water may be present. In this research, a restricted water area where the vessel is allowed to travel is represented by polygons and provided in a map beforehand.

Referring to Fig. 3, the shaded zone represents the area that the ASV is allowed to stay within. Obviously, the new path \overrightarrow{AB} is feasible while $\overrightarrow{AB'}$ is not. Denote the restricted area and its boundary by \mathcal{S} and $\partial\mathcal{S}$, respectively, and define the set $\mathcal{J} = (\overrightarrow{AB} \cap \partial\mathcal{S}) \cup (\overrightarrow{BC} \cap \partial\mathcal{S})$, then \mathcal{J} is the set of all intersection points between the path and the boundary of the restricted water area. The safety objective function in (3) is modified as follows:

$$f(x) = \max_{1 \leq i \leq n} f_i(x) + e^{a|\mathcal{J}|} - 1 \quad (5)$$

where $|\mathcal{J}|$ is the number of elements in set \mathcal{J} . The newly added term $e^{a|\mathcal{J}|} - 1$ acts as a penalty function on the safety objective. It is clear from (5) that the more the number of intersections of potential paths with the boundary of the restricted water, the bigger the value of the safety objective is and the less likely the candidate path will be chosen. Otherwise, $|\mathcal{J}| = 0$ and $f(x)$ in (5) reduces to the form in (3).

Remark 2: Since it is required that the ASV's journey is collision-free throughout, the safety objective function at every point of the trajectory (\overrightarrow{AB} and \overrightarrow{BC}) should be taken into account during the path planning stage. However, this would incur high computational overload. Instead, we only evaluate the maximum value of function $f(x)$ over time interval $(0, +\infty)$. Lemma 1 reveals that the absolute minimum value of $\text{TCPA}(t)$ is uniquely determined depending on the value of T whilst $\text{DCPA}(t)$ is constant when $t \in (0, T)$. The maximum value of function $f(x)$ is then obtained by substituting the two values of the minimum absolute value of $\text{TCPA}(t)$ and $\text{DCPA}(t)$ into (3) or (5).

2. **Path smoothness:** this objective function minimises or prevents any abrupt changes to the modified path. Referring to Fig. 3, the objective is to minimise the sum of angle changes in the successive manoeuvres from current position A to the waypoint C . Mathematically, it is equal to the sum of the angles $\angle A$, $\angle C$ and $\angle(\pi - B)$ where $\angle A$, $\angle B$ and $\angle C$ are the angles at vertices A , B , C of the triangle $\triangle ABC$ respectively. From geometry, $\angle(\pi - B) = \angle A + \angle C$, therefore $\angle(\pi - B)$ is used to quantify the path smoothness and it follows from basic

trigonometry:

$$g(x) = \pi - (\theta' - \theta) - \arctan \frac{s'T \sin(\theta' - \theta)}{l_{AB} - s'T \cos(\theta' - \theta)} \quad (6)$$

where l_{AB} is the distance from the current position A to the newly generated sub-waypoint B .

3. **Shortest path:** this objective function minimises the deviation from the global path thus bringing the ASV back to the pre-defined waypoint as soon as feasible.

$$h(x) = s'T + l_{BC}, \quad (7)$$

where l_{BC} is the distance from the newly-generated sub-waypoint B to the next waypoint C .

4. **Course/Speed preference:** this objective function reflects the good seamanship's practice of course changes over speed changes. If this fails to find a collision-free path, then a speed change option with or without a course change is considered. Since speed change is visually less obvious than course change, a discrete variable with only three values: half speed, double speed (maximum speed instead if double speed is not applicable) and zero speed, is selected. The half speed option is typically preferred over double speed as it permits additional time for decision making whereas zero speed is only considered as the last resort. Mathematically, the course/speed preference objective function to be minimised can be defined as follows:

$$z(x) = \begin{cases} 0, & \text{if } \theta' \neq \theta, s' = s \text{ (course change only),} \\ 1, & \text{if } s' = s/2 \text{ (half speed),} \\ 2, & \text{if } s' = 2s \text{ (double speed),} \\ 3, & \text{if } s' = 0 \text{ (stop),} \end{cases} \quad (8)$$

From the foregoing, this research considers and formulates four different objectives, however, the proposed multiobjective framework has been designed to be flexible, therefore additional objectives could be added such as the energy efficiency objective employed in [28], [29]. Note that the aforementioned objectives are not weighted equally rather the course/speed preference $z(x)$ is prioritised over other three objectives. To understand this, a collision-free path with only course change is preferred by mariners over another collision-free path with both speed and course changes, even though the latter solution performs relatively better in other three objectives than the former one.

C. Constraints

- **Hard constraint of collision avoidance:** since elimination of a collision risk is a necessity, a hard constraint needs to be defined and met when planning a path.

$$\text{risk}(x, t) = 0, \quad t \in (t_i, t_{i+1}), i = 0, 1, \dots, \quad (9)$$

where t_i denotes the time instant that the ASV detects its i th collision risk with other vessels.

$$f_i(x) = \begin{cases} e^{a(d_{\min} - \text{DCPA}_i(x) + t_{\min} - \text{TCPA}_i(x))} - 1, & \text{if } \text{DCPA}_i(x) \leq d_{\min}, 0 \leq \text{TCPA}_i(x) \leq t_{\min}, \\ -\text{DCPA}_i(x) \times (\text{sign}(\text{TCPA}_i(x))(\text{TCPA}_i(x) - 0.5t_{\min}) - 0.5t_{\min}), & \text{otherwise} \end{cases} \quad (4)$$

- Limits on course alteration:

$$\Delta\theta_{\min} \leq |\theta' - \theta| \leq \Delta\theta_{\max}; \quad (10)$$

- Speed constraint:

$$s' \leq s_{\max}; \quad (11)$$

- Length of manoeuvres:

$$\underline{T} \leq T \leq \bar{T}; \quad (12)$$

Once a manoeuvre is initiated, the ASV continues at least the minimum duration of time \underline{T} , making ASV's decision obvious and predictable to other users of the sea-space. Additionally, the ASV should not continue indefinitely on the new path ensuring minimum possible deviation from the nominated trajectory. This is specified by the variable \bar{T} , defining the maximum allowable time constraint.

- Preference of starboard manoeuvre:

$$\theta' - \theta \geq 0^0 \quad (13)$$

When the ASV is crossing and giving-way to other vessels, this constraint must be satisfied as manoeuvring to the starboard side is required according to COLREGs rules. For overtaking scenarios, both the starboard or port side manoeuvres are allowed but the former is preferable. In this case, the following strategy is adopted to bias the starboard manoeuvre in the local path planning sub-process: first the inequality constraint (13) is imposed and if a solution is found, a sub-waypoint on the starboard side is generated. However, if no feasible solution was found, then the inequality constraint (13) is relaxed and a port side manoeuvre is allowed.

In summary, the inequalities in (9)-(13) serve as constraints on the local path planning problem. Particularly, (10)-(13) are formulated explicitly in the form of upper and lower bounds of decision variables, which define the decision spaces. The local path planning problem is represented as the following constrained multi-objective optimisation problem:

$$\mathcal{P} : \begin{cases} \min & F(x) \\ \text{subject to:} & (9) - (13) \end{cases} \quad (14)$$

where the objective function $F(x) = [f(x) \ g(x) \ h(x) \ z(x)]^T$, and $f(x)$, $g(x)$, $h(x)$, $z(x)$ are defined in (3)-(8), respectively, whilst the decision variable x is given by (2).

IV. PATH PLANNING USING H-MOPSO ALGORITHM

A. The hierarchical non-dominated sorting rule

In this subsection, the MOP problem is first briefly introduced followed by a detailed account of the proposed hierarchical non-dominated sorting (Hi-NDS) rule in the context of path planning.

For MOPs, there typically exist no solutions that minimise or maximise all the objectives simultaneously due to the inherent partial order relation when comparing different objective vectors. For a single objective function, only one of the three possible orders ($<$, $>$ or $=$) exists when comparing two values. However, for a multi-objective function in the form of vectors,

two different values of objective functions cannot be compared using the aforementioned three orders. For example, consider three different values of a 2-dimensional objective function $F(x) = [0, \ 1]$, $F(y) = [1, \ 0]$ and $F(z) = [2, \ 2]$, then none of the three orders ($<$, $>$ or $=$) are valid when comparing $F(x)$ and $F(y)$. Instead, a new non-dominated relation, as outlined below is adopted in MOPs for comparing such values of objective functions.

Consider a n -dimensional objective function $F(x) = (F_1(x), \dots, F_n(x))$ of a MOP whose aim is to minimise all n objectives. $F(x)$ is said to be dominated by $F(y)$ if $F(y)$ is as small as $F(x)$ in all objectives and smaller in at least one objective than $F(x)$, i.e., $\forall i \in \{1, \dots, n\}, F_i(y) \leq F_i(x)$, and $\exists j \in \{1, \dots, n\}, F_j(y) < F_j(x)$. Moreover, $F(x)$ and $F(y)$ are said to be non-dominated by each other if for at least one objective, $F_i(x)$ is smaller whilst $F_j(y)$ is smaller for at least one of the other objectives. The Pareto front (\mathcal{P}) is defined as the set of elements of the objective space that are not dominated by any other elements in the objective space. In the above example, $F(x)$ and $F(y)$ are non-dominated, $F(z)$ is dominated by both $F(x)$ and $F(y)$, so the Pareto front $\mathcal{P} = \{F(x), F(y)\}$.

One approach commonly used in solving MOPs is to treat all objectives equally. In this way, all objectives are considered simultaneously and a set of trade-offs is obtained. However, for surface vessels, mariners would always prefer course changes over speed changes if both actions can lead to a collision-free path. In such a situation, the traditional ‘‘equal-objective’’ approach may not be suitable for path planning of ASVs since it will not be considered as seaman-like. This suggests that the course/speed preference (Objective $z(x)$) takes priority over the remaining objectives. In addition, the paths which satisfies the hard constraint of collision avoidance in (9) are always preferred over the ones which do not, irrespective of their performance in any of the objective functions. If the problem is synthesised as a single-objective optimization problem with weighted objectives, it is very difficult to take into account quantitatively such preferences (heading change over speed change) through assigning weighting factors to different objectives.

Based on the above discussion, a new Hi-NDS rule has been implemented where the course/speed preference is considered as the *main* objective. All remaining objectives are viewed as equal *secondary* objectives to be optimised. More specifically, the following Hi-NDS rule is used: 1) prefer the paths that satisfy the hard constraint of collision avoidance; 2) prefer the paths with smaller values in course/speed preference and 3) prefer the paths with better overall fitness (determine by all objectives) if the paths have the same course/speed preference value.

The proposed Hi-NDS rules is codified in Algorithm 1 and used for comparing two candidate paths: firstly check the hard constraints and then compare their objective values in course/speed preference, and finally the relationship of non-dominance in all objective functions.

Algorithm 1 Choose the solution from candidates x and y using the Hi-NDS rule

```

if risk( $x, t$ ) = 0 & risk( $y, t$ ) > 0 then
  solution  $\leftarrow x$ 
else if risk( $y, t$ ) = 0 & risk( $x, t$ ) > 0 then
  solution  $\leftarrow y$ 
else if risk( $y, t$ ) = 0 & risk( $x, t$ ) = 0 then
  if  $z(x) < z(y)$  then
    solution  $\leftarrow x$ 
  else if  $z(x) > z(y)$  then
    solution  $\leftarrow y$ 
  else if  $z(x) = z(y)$  then
    solution  $\leftarrow a$ , where  $F(a) = \mathcal{P}\{F(x), F(y)\}$ 
  end if
end if

```

B. The Proposed H-MOPSO Algorithm

This subsection shows how the Hi-NDS rule is combined with the MOPSO to formulate the proposed H-MOPSO algorithm. It should be noted that though a variation of the standard particle swarm optimization algorithm is used to solve the MOP in this paper, the proposed Hi-NDS rule can be conveniently incorporated into other evolutionary algorithms such as genetic algorithms to solve the ASV path planning problem in our proposed multiobjective optimisation framework.

A number of evolutionary-based multiobjective optimisation algorithms have been developed demonstrating outstanding results through solving complex multiobjective benchmark test problems. Among those techniques, multiobjective particle swarm optimisation (MOPSO) algorithm has been developed by extending the basic particle swarm optimisation algorithm [30]. It optimizes a solution by iteratively searching a large space of candidate solutions. According to the results in [20], compared with other multiobjective evolutionary algorithms, MOPSO showed a highly competitive performance in that it was able to cover the full Pareto front with relatively low computational burden. Such merits of MOPSO are desirable for real-time applications and therefore is adapted to solve the given online path planning problem.

In this paper, the H-MOPSO algorithm is modified from the MOPSO algorithm by applying the Hi-NDS rule presented in the previous subsection. The flowchart of H-MOPSO algorithm is depicted in Fig. 5. The algorithm works according to the following procedure:

- 1) Initialize the population $P(0)$: the population $P(k)$ is a set of N particles, each with its own position $x_i(k)$ and velocity $v_i(k)$, $i = 1, 2, \dots, N$. Both $x_i(k)$ and $v_i(k) \in \mathbb{R}^n$, where n is the dimension of the decision variable in the particular problem addressed. For every particle in the population, its position is randomly initialized in the decision space and its velocity is initially set to 0, and the archive $A(k)$ is initialized as an empty set.
- 2) Select the local and global best: the local best $Pb_i(k)$ is the best position the i th particle achieved at time instant k whereas the global best $Gb(k)$ is the best position in the population $P(k)$ at time instant k . Each particle

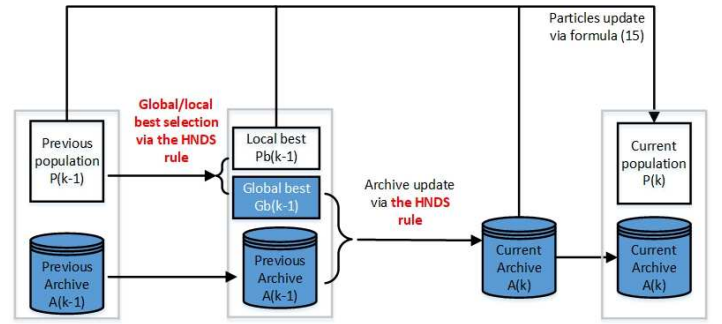


Fig. 5. The procedure of H-MOPSO algorithm, where the red parts highlight the modifications over the traditional MOPSO algorithm

compares its current objective value $F(x_i(k))$ with the best one in memory $F(Pb(k-1))$ according to the proposed Hi-NDS rule in 1, and only the non-dominated position will be kept. To choose the global best $Gb(k)$, all particles in the current population $P(k)$ are compared according to the Hi-NDS rule and all the non-dominated positions are regarded as the global best $Gb(k)$.

- 3) Archive update: the archive $A(k)$ maintains a historical record of the non-dominated particles found along the search process. The global best $Gb(k)$ found at every iteration are compared with existing positions in archive $A(k)$ according to the Hi-NDS rule, and the non-dominated positions are kept in the archive.
- 4) Particles update: the population of particles moves in the search space according to two simple mathematical formulae for the particle's position and velocity as follows:

$$\begin{cases} v_i(k) = \omega v_i(k-1) + c_1 r_1 (Pb_i(k-1) - x_i(k-1)) \\ \quad + c_2 r_2 (Gb_j(k-1) - x_i(k-1)) \\ x_i(k) = x_i(k-1) + v_i(k) \end{cases} \quad (15)$$

where $x_i(k) = [x_{i1}^T(k), \dots, x_{in}^T(k)]^T$, $x_i(k)$ is the position of the i th particle at the k th iteration, and $x_i(k) \in [x_{min}, x_{max}]$, with x_{min} and x_{max} being the lower and upper bounds for all particles' positions. $v_i(k) = [v_{i1}^T(k), \dots, v_{in}^T(k)]^T$, is the velocity of the i th particle at the k th iteration. ω is the inertia weight, c_1 and c_2 are called acceleration coefficients, namely, cognitive and social parameters, respectively. r_1 and r_2 are two uniform random number samples from $[0, 1]$. $Pb_i(k)$ is the local best position encountered by i th particle at the k th iteration, and $Gb_j(k)$ is the j th particle in the current archive $A(k)$ where the index j is selected using the roulette-wheel approach proposed in [20].

Briefly speaking, each particle's movement is influenced by its local best known position and also the global best known positions, which are updated and kept in the archive. By such an iterative approach, the particle swarm moves towards the optimal solution.

It is worth noting that the Hi-NDS rule is incorporated on two processes of the overall procedure of H-MOPSO algorithm: the local/global best selection and the archive update process. The particle compares its position $x(k-1)$ with the

current local best $Pb(k-1)$ according to the Hi-NDS rule and select the best one as the new local best. Similarly, the global best $Gb(k-1)$ is also chosen from the current population $P(k-1)$ using the Hi-NDS rule. Subsequently the global best $Gb(k-1)$ together with the current archive $A(k-1)$ is used to update the archive by the Hi-NDS rule as well.

Due to the introduced Hi-NDS rules, our proposed algorithm will require at most three times number of particle comparison operations as that in the standard MOPSO algorithm proposed in [20]. As a result, the overall worst case complexity of the H-MOPSO algorithm is the same as that of the traditional MOPSO, i.e., $O(nN^2)$ [31], where the number of objectives to be optimized is n and the size of the swarm is N , respectively.

V. SIMULATION STUDY

In this section, the performance of the proposed online path planning method is presented and discussed. The simulations considered a wide range of cases, from single to multiple obstacles, from static to moving targets, from open water to restricted water, and from COLREGs-compliant to non-compliant targets vessels. Throughout the simulations, it was assumed that the ASV was the ownship (OS) and all the ships detected in the surroundings by the ASVs were target ships (TGs). The code was executed in Matlab 2015a running on a laptop with an i7 processor and 8 GB RAM. The algorithm was set with a population of 50 particles with 40 generations for all the runs, since acceptable re-planned paths were found with such settings in all cases. The same parameters were used in the H-MOPSO algorithm. Since H-MOPSO renders a set of solutions rather than a single one, we chose the solution with smallest values in the third objective function, that is, the smoothest path was selected as the planned path. However, this could be conveniently modified to select a path based on different preferences as desired.

To ensure a feasible solution (representing a new path) found in time by the online path planner, we do require that the algorithm always terminate with a solution output in a pre-specified time period (e.g, 2 seconds in our desktop/bridge simulations and sea trials). The requirement was achieved through the following practical procedure:

- If within the pre-specified time, the proposed H-MOPSO algorithm completes running the specified number of generations (40 in our research), then a solution (approximating the global optimum) is generated automatically by the algorithm;
- If within the pre-specified time, the specified number of generations are not complete, however a non-empty set of candidate solutions (particles) are still generated in the last iteration, then a near-optimal solution will be selected from the set of candidate solutions according to the Hi-NDS rule. The solution will still satisfy collision avoidance constraints, though it might not be an optimal one due to incomplete number of iterations;
- If a collision-free path is not found even after the specified time, then the contingency is to stop the vessel immediately by demanding a speed of zero ensuring that the zero speed does not lead to a collision with

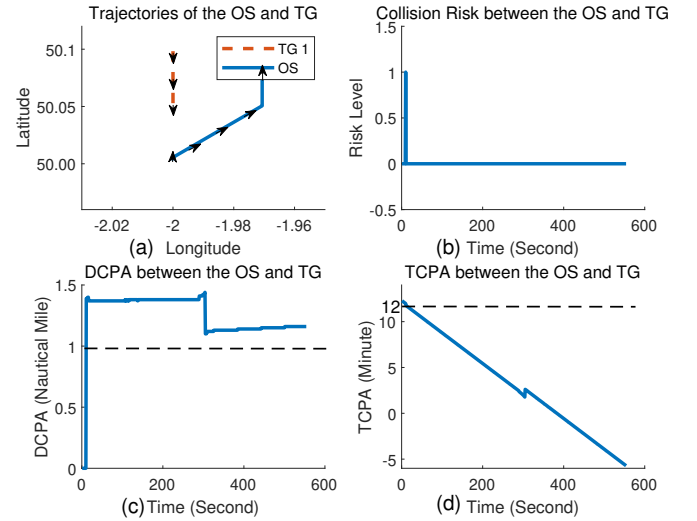


Fig. 6. Single vessel head-on simulation

an oncoming vessel. Theoretically this is possible, for example, when the traffic density in the vessel's neighbouring area is too high to find a safe path with a large DCPA/TCPA thresholds. A practical solution might be to reduce the DCPA/TCPA thresholds but it is out of the scope of this paper. Nonetheless, thanks to the appropriate DCPA/TCPA parameters and scenarios designed by experienced maritime practitioners in our research team, we have not encountered this situation during our on-water and bridge simulation trials, where typically less than 10 vehicles needed to be considered simultaneously for local path planning to avoid a collision.

The simulations were tested both in the stand-alone desktop Matlab-based environment as well as on the high-fidelity networked bridge simulators available at the Warsash Maritime Academy (WMA) in Southampton Solent University. Compared with desktop simulations, the bridge simulators are capable of running scenarios where interaction between manned ships and autonomous ships is required and can simulate different environmental conditions such as sea state and degraded sensors with intermittent signal and/or sensor uncertainty. In this research, the bridge simulator platform provided the capability of utilising former seafarers to design representative trials as well as simulate and assess the ASV's performance as they would assess humans. In addition, the bridge simulator recorded data and generated visual feeds for post analysis. It also allowed for configuration in terms of different vessel types as well as geographical setting to allow trials at any corner of the world. For more details of tests using bridge simulators, please refer to [32].

In the stand-alone desktop simulations, a local relative coordinate system was used where the OS was assumed to start at the origin $(0, 0)$.

A. Desktop simulation

Single vessel encounter: Simulation results involving a single vessel head-on scenario are depicted in Fig. 6 (a)-(d) where the thresholds for DCPA and TCPA parameters were

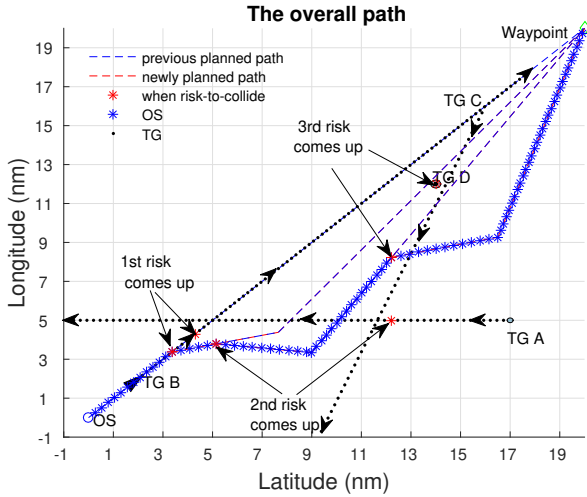


Fig. 7. Path re-planning under multiple-vessel encounter

set as 1 nm and 12 min respectively. At the beginning, as the OS and TG were approaching on reciprocal courses (6(a)), the DCPA was almost zero (6(c)) however the TCPA was over the pre-specified threshold (6(d)). When the TCPA falls below 12 min, a collision risk arose as shown in Fig. 6(b). At that moment, even though the TG had not altered the course (6(a)) as required in the head-on scenario (possibly due to smaller DCPA/TCPA thresholds set by the TG), the OS removed the risk of collision by changing course immediately to starboard (6(a)) passing the TG on her port side. The evasive path was chosen such that the DCPA of the OS was always equal to or bigger than the threshold at all times (6(c)). Thus the OS behaviour fully complied with COLREGs rules 14 and 16 (see Appendix B).

Multiple-vessel encounter: One of the main advantages of the proposed multiobjective optimization framework is that it can deal with multiple-vessel encounters simultaneously. As illustrated in Fig. 7, the ASV travelling from the bottom-left to the top-right was required to negotiate 3 moving targets A, B, C and 1 static obstacle D in turn. The OS thus re-planned the path three times. Firstly, TG B presented an “overtaking” scenario, then the collision was avoided by re-planning a local path performing a COLREGs-compliant manoeuvre to the starboard with course change only; Secondly, when the OS travelled along the newly updated path, TG A presented a “crossing” scenario. According to COLREGs, the ASV became a give-way vessel to TG A and manoeuvred to starboard whilst maintaining her speed. Finally, the static TG D was in the way of the ASV whilst TG C could be a potential risk if the OS manoeuvred only slightly. Taking into account of both TGs C and D for path re-planning, the ASV made an *obvious* course alteration to starboard preserving her speed and thus avoided collision from both TGs C and D simultaneously.

For a better understanding of the proposed H-MOPSO strategy, consider the particular moment that the ASV had the first risk of collision with TG B in the scenario illustrated in Fig. 7. After detecting the risk of collision and selecting the corresponding COLREGs rules (rule 13 & 16), the path

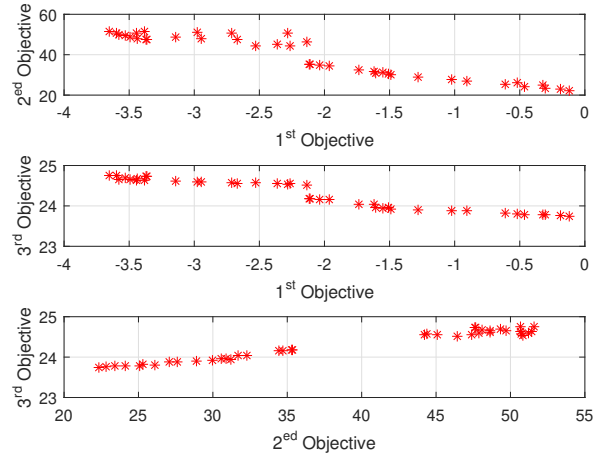


Fig. 8. Pareto front between different objectives

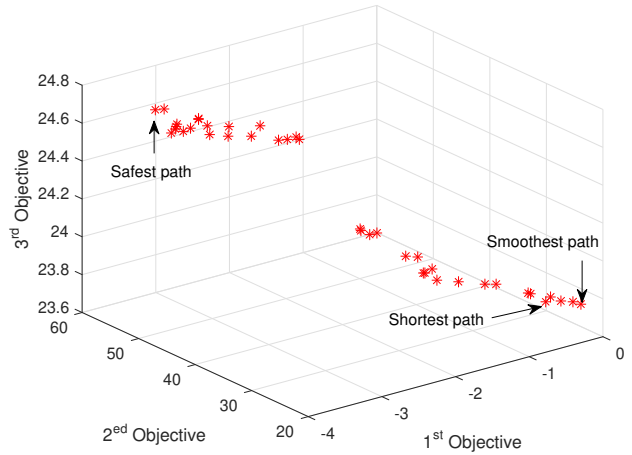


Fig. 9. Three dimensional plot of Pareto front

planner was activated. A set of feasible solutions were found as a Pareto front using our proposed algorithm. For better illustration, we plotted the Pareto front of three objectives by ignoring the objective of course/speed change since no speed change was made in this scenario. Fig. 8 shows Pareto front for different objectives and Fig. 9 depicts the three-dimensional plot of the Pareto front. It demonstrates that the three objectives are contradictory with each other and there are no paths that minimize all three objectives simultaneously.

B. Networked bridge simulations

Open vs Restricted water: Bridge simulators provide a unique way of high-fidelity testing of the algorithm in a safe yet effective “real world” environment without sea trials. For this work, the collision avoidance system was interfaced directly with one of the networked bridge simulators made available by the project partners WMA. A representative crossing give-way encounter is shown Fig. 10 where the OS should give-way to another vessel on her starboard side. A performance comparison was also made of the same scenario in open and restricted workspace. Figs. 10(a)-(b) depict two

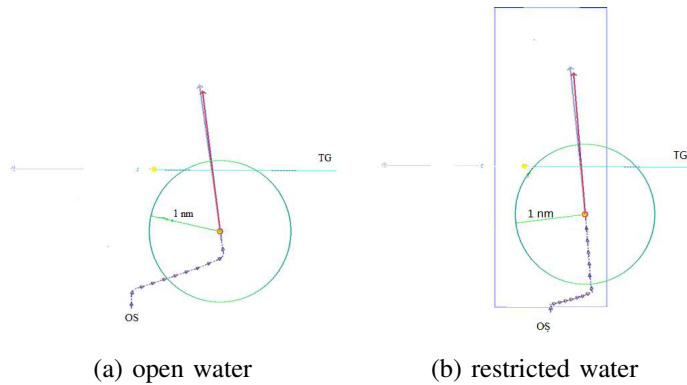


Fig. 10. The same ship encounter scenario in open and restricted water

similar scenarios: one in the open water sea-space while the other one in restricted sea space. The rectangle in Fig. 10(b) represents the boundary of restricted water. In open water case only the course change was demanded, whereas due to the narrow sea space in restricted water, the ASV was unable to remove the collision risk completely by course change only. Hence, the ASV reduced to half speed and altered course to starboard returning to nominal speed after reaching the sub-waypoint. By comparing the simulation results, it is evident that if both course change and speed change solutions are available, then the former is preferred (Fig. 10(a) for open water). However, if a course change alone does not eliminate the risk of collision, then a speed change will be considered (Fig. 10(b) for restricted water).

Real-world Scenario: In previous simulations, the TGs were assumed to maintain a constant heading. A representative overtaking scenario shown in Fig. 11 was conducted on the WMA's bridge simulator: the TG was simulated to change heading which made the collision avoidance more difficult to achieve, and the bridge simulation was run in a restricted water area in the San Francisco Bay. As shown in Fig. 11, the polygonal area surrounded by the boundary (the solid lines) was artificially chosen as the test area and the OS was constrained within it. For this narrow channel with a maximum width of about 1 nm and with parameters chosen as $d_{min} = 0.2 \text{ nm}$ and $t_{min} = 6 \text{ min}$, any manoeuvre maintaining DCPA/TCPA bigger than the thresholds is quite challenging. Figs. 12(a)-(c) depict snapshots showing the original offline and the updated paths at different time instants. In Fig. 12(a), the ASV and the TG were initially following the same heading with no collision risk. In Fig. 12(b), the ASV detected a collision risk and selected an "overtaking" manoeuvre, then the collision was avoided by re-planning a local path performing a COLREGs-compliant manoeuvre to the starboard with no speed changes. From COLREGs, the TG in this encounter was the stand-on vessel and should keep its course with caution. However, the TG made an unexpected alteration to starboard causing a new risk of collision with the OS. Detecting the urgent risk caused by the TG's improper action, the ASV re-planned an alternate evasive path based on Rule 2(b) (see Appendix B) in the process, as shown in Fig. 12(c).



Fig. 11. The restricted area where the OS can go in the San Francisco Bay, courtesy of Google Map 2017.

VI. CONCLUSION

This paper has presented a new rule-based reactive collision avoidance method for ASVs which generates safe and COLREGs-compliant paths using a multiobjective optimisation approach. The COLREGs rules are expressed as inequality constraints, which is rather straightforward to incorporate in the presented optimisation framework. The Hi-NDS rule is proposed which manages priorities for different objectives in path planning such that a collision free path is always guaranteed with preference given to course alteration over speed changes. The effectiveness of the proposed algorithm has been validated through extensive simulations of various ship encounters at sea in both desktop and bridge simulator environments.

It should be pointed out that the safety objective function presented in (3) is not unique but how to choose the optimal one is still an open problem to be addressed in the future. Another research direction is to find the most suitable optimization algorithm through comparative simulations of different existing multi-objective optimization algorithms.

ACKNOWLEDGEMENT

The authors would like to thank the project partners, Rolls Royce, Warsash Maritime Academy, Lloyd's Register and Atlas Elektronik UK for providing useful input and support throughout the course of this research.

REFERENCES

- [1] S. Campbell, W. Naeem, and G. W. Irwin, "A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres," *Annual Reviews in Control*, vol. 36, no. 2, pp. 267–283, 2012.
- [2] 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.
- [3] "Ship intelligence - transforming future marine operations," <http://www.rolls-royce.com/products-and-services/marine/ship-intelligence.aspx>, 2016, [White paper produced by Rolls-Royce plc.].
- [4] A. N. Cockcroft and J. N. F. Lameijer, *Guide to the collision avoidance rules*. Butterworth-Heinemann, 2003.
- [5] F. Ahmed and K. Deb, "Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms," *Soft Computing*, vol. 17, no. 7, pp. 1283–1299, 2013.

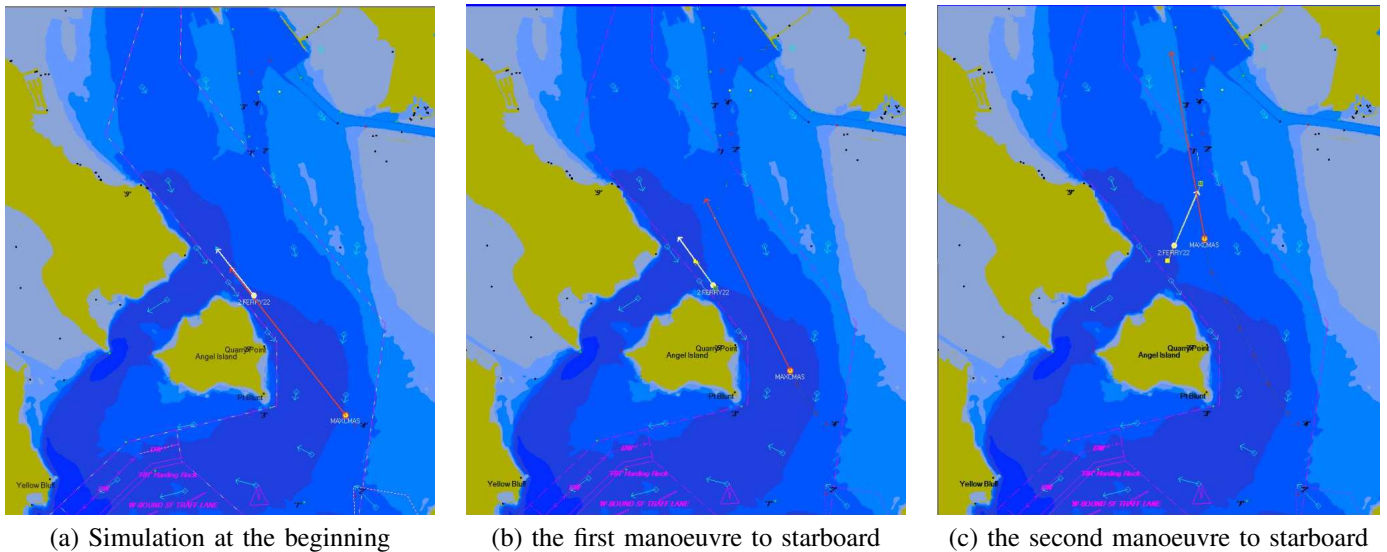


Fig. 12. Snapshot of bridge simulation in restricted water of the San Francisco Bay, where MAXCMAS is the ownship. The red and white arrows represent the relative magnitude of the instantaneous velocity of the OS and the TG, respectively

- [6] Q. McEntegart and J. Whidborne, "A multiobjective trajectory optimization method for planning environmentally efficient trajectories," in *the UKACC International Conference on Control (CONTROL)*, 2012, pp. 128–135.
- [7] T. Oral and F. Polat, "MOD* lite: An incremental path planning algorithm taking care of multiple objectives," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 245–257, 2016.
- [8] Y. Xue, B. Lee, and D. Han, "Automatic collision avoidance of ships," *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, vol. 223, no. 1, pp. 33–46, 2009.
- [9] W. Naeem, S. Campbell, and L. Hu, "A reactive colregs-compliant navigation strategy for autonomous maritime navigation," *IFAC Proceedings Volumes*, 2016.
- [10] R. Szlapczynski, "Evolutionary sets of safe ship trajectories: a new approach to collision avoidance," *Journal of Navigation*, vol. 64, no. 01, pp. 169–181, 2011.
- [11] 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.
- [12] S.-L. Kao, K.-T. Lee, K.-Y. Chang, and M.-D. Ko, "A fuzzy logic method for collision avoidance in vessel traffic service," *Journal of Navigation*, vol. 60, no. 01, pp. 17–31, 2007.
- [13] 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.
- [14] L. Perera, J. Carvalho, and C. Guedes Soares, "Solutions to the failures and limitations of mamdani fuzzy inference in ship navigation," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1539–1554, 2014.
- [15] 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.
- [16] Y. Kuwata, M. T. Wolf, D. Zarzhitsky, and T. L. Huntsberger, "Safe maritime autonomous navigation with colregs, using velocity obstacles," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 110–119, 2014.
- [17] T. A. Johansen, T. Perez, and A. Cristofaro, "Ship collision avoidance and colregs compliance using simulation-based control behavior selection with predictive hazard assessment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3407 – 3422, 2016.
- [18] S. Campbell, M. Abu-Tair, and W. Naeem, "An automatic colregs-compliant obstacle avoidance system for an unmanned surface vehicle," *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, vol. 228, no. 2, pp. 108–121, 2014.
- [19] K.-B. Lee, H. Myung, and J.-H. Kim, "Online multiobjective evolutionary approach for navigation of humanoid robots," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5586–5597, 2015.
- [20] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 256–279, 2004.
- [21] J. Balsa-Barreiro, "Application of gns and gis systems to transport infrastructures. studies focusing on naturalistic driving," *PhD thesis*, 2015.
- [22] J. Balsa-Barreiro, P. M. Valero-Mora, I. Pareja-Montoro, and M. Sánchez-García, "Quality control procedure for naturalistic driving data using geographic information systems," in *Proc. European Conference on Human Centred Design for Intelligent Transport Systems*, 2014, pp. 5–6.
- [23] —, "Proposal of geographic information systems methodology for quality control procedures of data obtained in naturalistic driving studies," *IET intelligent transport systems*, vol. 9, no. 7, pp. 673–682, 2015.
- [24] I. R. Bertaska, B. Shah, K. von Ellenrieder, P. Švec, W. Klinger, A. J. Sinisterra, M. Dhanak, and S. K. Gupta, "Experimental evaluation of automatically-generated behaviors for USV operations," *Ocean Engineering*, vol. 106, pp. 496–514, 2015.
- [25] L. Hu, W. Naeem, E. Rajabally, G. Watson, T. Mills, Z. Bhuiyan, and I. Salter, "Colregs-compliant path planning for autonomous surface vehicles: a multi-objective optimization approach," in *Proceedings of the 20th International Federation of Automatic Control (IFAC) World Congress*, Toulouse, France, 2017.
- [26] R. Smierzchalski and Z. Michalewicz, "Modeling of ship trajectory in collision situations by an evolutionary algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 3, pp. 227–241, 2000.
- [27] J. Xiao, Z. Michalewicz, L. Zhang, and K. Trojanowski, "Adaptive evolutionary planner/navigator for mobile robots," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 18–28, 1997.
- [28] T. Lee, H. Kim, H. Chung, Y. Bang, and H. Myung, "Energy efficient path planning for a marine surface vehicle considering heading angle," *Ocean Engineering*, vol. 107, pp. 118–131, 2015.
- [29] A. J. Häusler, A. Saccon, A. P. Aguiar, J. Hauser, and A. M. Pascoal, "Energy-optimal motion planning for multiple robotic vehicles with collision avoidance," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 867–883, 2016.
- [30] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. International Conference on Neural Networks*, vol. 4. Hawaii, USA: IEEE, 1995, pp. 1942–1948.
- [31] P. K. Tripathi, S. Bandyopadhyay, and S. K. Pal, "Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients," *Information Sciences*, vol. 177, no. 22, pp. 5033–5049, 2007.
- [32] J. Varas, S. Hirdaris, R. Smith, P. Scialla, W. Caharija, Z. Bhuiyan, T. Mills, W. Naeem, L. Hu, I. Renton, D. Motson, and E. Rajabally,

“MAXCMAS project: autonomous colregs compliant ship navigation,” in *the 16th Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT)*, Cardiff, UK, 2017.

APPENDIX A PROOF OF LEMMA 1

Proof: First, recall the results presented in [18]. Referring to the two vessels shown in Fig. 2, in a relative coordinate system the vessel at location C is static, and vessel at location A moves with the relative velocity vector V_{ref} . The instantaneous DCPA and TCPA are formulated as follows:

$$\begin{aligned} \text{DCPA} &= \delta_{LOS} \sin \alpha \\ \text{TCPA} &= \frac{\delta_{LOS} \cos \alpha}{V_{\text{ref}}} \end{aligned}$$

where δ_{LOS} is the line-of-sight distance between the two vessels, and α is the angle from vector V_{ref} to the line of sight between the two vessels.

Obviously, both the line-of-sight distance δ_{LOS} and the angle α change with time t , and the value of $\text{DCPA}(t)$ equals the shortest distance from point C to the line extended by relative speed vector V_{ref} , that is l_{BC} in Fig.2. It is known that l_{BC} is constant from basic principles of geometry, so $\text{DCPA}(t)$ is invariant with respect to time t .

On the other hand, $\delta_{LOS} \cos \alpha = l_{AB} - V_{\text{ref}} t$ leads to $\text{TCPA}(t) = \frac{l_{AB}}{V_{\text{ref}}} - t$, which shows that $\text{TCPA}(t)$ is a continuous function over $t \in [0, +\infty)$. If $T \leq \frac{l_{AB}}{V_{\text{ref}}}$, $\min |\text{TCPA}(t)| = \frac{l_{AB}}{V_{\text{ref}}} - T$; else if $T > \frac{l_{AB}}{V_{\text{ref}}}$, $\min |\text{TCPA}(t)| = 0$. ■

APPENDIX B COLREGS RULES

This section lists the COLREGs rules from [4] that are considered in our paper.

- **Rule 8 - Actions to avoid collision:** If there is sufficient sea-room, alteration of course alone may be most effective. Reduce speed, stop or reverse only if necessary.
- **Rule 13 - Overtaking:** Any vessel overtaking any other shall keep out of the way of the vessel being overtaken.
- **Rule 14 - Head-on situation:** Each head-on vessel shall alter her course to starboard so that each shall pass on the port side of the other.
- **Rule 15 - Crossing situation:** The vessel which has the other on her own starboard side shall keep out of the way.
- **Rule 16 - Actions by give-way vessel:** Take early and substantial action to keep well clear.
- **Rule 17 - Actions by stand-on vessel:** Keep her course and speed but may take action to avoid collision if the other vessel is not taking appropriate COLREGs-compliant action.