Motion Attribute-based Clustering and Collision Avoidance of Multiple In-water Obstacles by Autonomous Surface Vehicle

Mingi Jeong and Alberto Quattrini Li

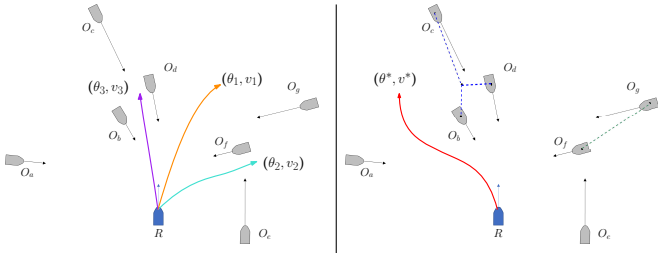
***Abstract*— Navigation and obstacle avoidance in aquatic environments for autonomous surface vehicles (ASVs) in high-traffic maritime scenarios is still an open challenge, as the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) is not defined for multi-encounter situations. Current state-of-the-art methods resolve single-to-single encounters with sequential actions and assume that other obstacles follow COLREGs. Our work proposes a novel real-time non-myopic obstacle avoidance method, allowing an ASV that has only partial knowledge of the surroundings within the sensor radius to navigate in high-traffic maritime scenarios. Specifically, we achieve a holistic view of the feasible ASV action space able to avoid deadlock scenarios, by proposing (1) a clustering method based on motion attributes of other obstacles, (2) a geometric framework for identifying the feasible action space, and (3) a multi-objective optimization to determine the best action. Theoretical analysis and extensive realistic experiments in simulation considering real-world traffic scenarios demonstrate that our proposed real-time obstacle avoidance method is able to achieve safer trajectories than other state-of-the-art methods and that is robust to uncertainty present in the current information available to the ASV.**

I. Introduction

This paper addresses the problem of obstacle avoidance and navigation of Autonomous Surface Vehicles (ASVs) in highly-congested waters – see Fig. 1. Augmenting the autonomy of ASVs can enable and automate many high-impact societal applications, including shipping and monitoring [1]. One of the current main challenges limiting the widespread use of ASVs is navigation safety [2]. Differently from car driving, waterways are not clearly marked. In addition, while there are some traffic rules – Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) [3] – governing how to handle single obstacle encounters, COLREGs do not explicitly cover scenarios of multiple obstacle encounters [4]. The unstructured environment and lack of regulatory framework in aquatic scenarios create a challenge especially in high-traffic waters, such as the Ningbo and Shanghai waterways, Singapore and Malacca Straits, and the Dover Straits where hundreds of vessels navigate on a daily basis [5].

Current state-of-the-art methods for ASVs operate following COLREGs on several *single-to-single encounters with sequential actions* (e.g., [6], [7]) and *reciprocal cooperative actions by the obstacles* (e.g., [8], [9]). These sequential and myopic methods may produce conflicting actions in real-world scenarios. For example, when the own controlled ASV is obliged to turn to the right as a '*give-way*' vehicle with respect to *head-on* obstacle(s), but simultaneously keeping the speed and heading as a '*stand-on*' vehicle with respect to *overtaking* obstacle(s). Moreover, other vessels might not follow the rules and the same reciprocal collision avoidance algorithm as the own controlled ASV.

The authors are with Department of Computer Science, Dartmouth College, Hanover, NH USA {mingi.jeong.gr, alberto.quattrini.li}@dartmouth.edu



(a) State-of-the-art rule-compliant (b) Proposed method

Fig. 1. Collision avoidance behavior for controlled ASV, R, using state-of-the art vs. proposed method under congested traffic with multiple obstacles O: **(a)** a desired action (θ_1, v_1) complies with a left-to-left maneuver for the head-on O_b, O_c, O_d, while the bow-crossing maneuver conflicts with the crossing O_f, O_g; (θ_2, v_2) complies with a stern-crossing maneuver for the crossing O_f, O_g, while the change of action conflicts with stand-on status for overtaking O_e; (θ_3, v_3) partially complies with a left-to-left maneuver for the head-on O_b, O_c except O_d, while it leads to entering between the obstacles. **(b)** (θ^*, v^*) achieves a safe and adaptive maneuver with only one bow-crossing for O_a, while clusters (O_b, O_c, O_d) and (O_f, O_g) are identified based on similar motion attributes with respect to R such that R will not pass areas (dotted lines) between obstacles in a cluster.

To address the above mentioned limitations, this paper proposes a novel motion attribute-based clustering and non-myopic collision avoidance based on multi-objective optimization for complex in-water obstacle scenario(s). The proposed method identifies **three near future motion attributes** of encountered obstacles with respect to the controlled ASV – *temporal* (Time to the Closest Point of Approach (TCPA)), *spatial* (Closest Point of Approach (CPA)), and *angular* (relative bearing) similarity – and clusters group(s) of obstacles based on the similarity. The clustering allows an increased safety: the ASV can holistically consider the obstacles, by prohibiting the entrance within the cluster. From the predicted cluster(s), our method calculates an evasive action by **geometric analysis**, which finds feasible action space boundaries, and **multi-objective optimization**. The objective function considers change(s) of *velocity*, *heading*, and *expected safety level*. Note clusters are updated or added, if a motion attribute of obstacle(s) with respect to the controlled ASV changes or new obstacles are sensed within the range. After calibrating parameters of a simulated ASV from field experiments with a real ASV, we validate the proposed approach with extensive simulations under realistic diverse traffic scenarios. We vary the number of

 DOI: 10.1109/IROS47612.2022.9981226

obstacles and their dynamic properties and compare other state-of-the art methods. The proposed method shows 1.2–1.7 times higher success rate performance (0.927 on average) than other methods, while achieving real-time computation (61 ± 13 ms) in congested traffic with 30 obstacles.

II. Related Work

The robotics literature is rich on collision avoidance methods for mobile robots. One classification of such methods is according to whether they consider global vs. local domain [10] – i.e., full vs. partial knowledge. Here, we discuss methods that have been proposed specifically for maritime autonomous navigation, which are based on those available for other types of robots.

As the majority of the encountered obstacles in aquatic environments are unknown or dynamic – detected by the ASV on-board sensors – many studies (e.g., [7], [11]) have addressed the collision avoidance problem based on the local domain. Local-based methods include Artificial Potential Field (APF) [6], [12], Velocity Obstacle (VO) [7], [13] and its variant [8], Dynamic Window [14], model-predictive [15], evolutionary algorithm [16], learning-based model [17]. Other examples include fuzzy adaptive control model [18] and Differential Evolution algorithm [19].

These methods take decisions by considering one obstacle at the time or aggregating the information of all obstacles together. As a result, deadlocks or collisions in heavy traffic scenarios can happen as the number of obstacles increase [20], as shown also in the experimental section.

Furthermore, most of the existing methods are in compliance with COLREGs [21], which, as mentioned above, do not specify multi-obstacle encounters. This compliance introduces a critical problem for these methods based on multiple single-to-single encounter relationships: '*role conflicts*', e.g., '*give-way*' obligation while '*stand-on*' privilege. To address this *role conflict* problem, Cho *et al.* [9] proposed a symmetric role-classification criterion and probabilistic VO. Kim *et al.* [22] presented a Distributed Stochastic Search Algorithm for complex ship encounters. Zhao and Roh [17] proposed a deep reinforcement learning for multi-ship collision avoidance. These methods assume that obstacles behave (1) in a *distributed*, *reciprocal*, and *cooperative* manner as their proposed algorithms, and (2) fully *rule-compliant* achieved by constraining the action space. For example, Cho *et al.* [9] assumed a starboard maneuver only for every evasive action. This fully compliant behaviors are not always present in real-world marine traffic, e.g., departure from the rule under a special case like multiple encounters with impending dangers ([23], [24], and Rule 2 in [3]).

Some work focused on maritime traffic management instead of real-time collision avoidance. Zhen *et al.* [25] and Chen *et al.* [26] proposed a density-based clustering of multiple ships, which allows for identification of a geographical collision hot spot. Their proposed approaches are applied to shore traffic monitoring operators, since the clustering is not based on the first person perspective (i.e., the local observation from a controlled vehicle) that requires an on-board evasive action.

This paper's main insight, differing from previous work, is that it is fundamental to holistically consider multiple obstacles, by identifying clusters that should not be crossed. In addition, we assume neither a *reciprocal* algorithm nor *distributed, cooperative* action(s) by obstacle(s) under multiple encounter scenarios. Instead, we consider that our controlled ASV is a *Give-way* vehicle – a vehicle that takes an evasive action – according to definition [3] whereas obstacle(s) are *Stand-on* vehicle(s) – a vehicle that maintains its action. This assumption on challenging cases for the controlled vehicle is aligned with defensive, proactive actions in many real-world cases where human-driven vehicles do not take a proper action despite a *Give-way* vehicle status [27], [28]. Finally, we prioritize on safety rather than COLREGs rule following, as no explicit regulation is available for multi-obstacle encounters. This prioritization complies with the main principle of COLREGs, i.e., safety.

III. Approach

The proposed method evaluates complex marine traffic situations and determines optimized action(s) – speed and heading – to avoid obstacles that are located within an ASV's *sensible range* defined as $\mathcal{S} \subset \mathbb{R}^2$ and get to the goal. We assume that any obstacle within $\mathcal{S}$ can be detected via a broadcast message format – which is a realistic assumption in the maritime domain thanks to the presence of Automatic Identification System (AIS). In the experimental section, we relax the assumption by considering potential noise present in the type-A AIS with 1 Hz – e.g., delay time when obstacle(s) enter $\mathcal{S}$ due to processing of obstacle information collected from received messages [29] as in real-world scenarios.

Fig. 2 shows the overview of the pipeline, with each module described in the following subsections.

A. Relative Perspective Framework

For computational efficiency, instead of using the global reference frame, we utilize a relative perspective considering two local reference frames: the ASV's reference frame $\{R\}$ and an obstacle's reference frame $\{O\}$, similarly to our previous work [27]; here, we include additional metrics that

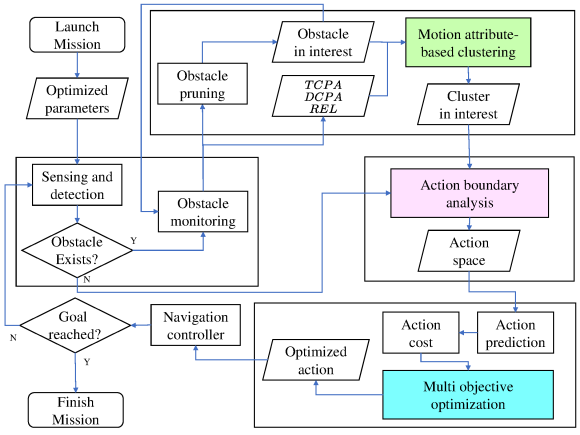


Fig. 2. Obstacle avoidance pipeline overview.

assess collision risks important in maritime navigation [21] and will be used for clustering.

From the relative perspective of $\{O\}$, CPA is defined as the closest point of approach until ASV fully passes the obstacle. **DCPA** is defined as distance between two vehicles when an ASV passes CPA and is calculated by

$$DCPA(O) = ||\mathbf{x}_R - \mathbf{x}_O|| \sin(\phi) \tag{1}$$

where $\mathbf{x}_R$, $\mathbf{x}_O \in \mathbb{R}^2$ is the current pose of R, O in $\{W\}$, respectively, $||\mathbf{x}_R - \mathbf{x}_O||$ is the Euclidean distance, and ϕ the angle between the line of sight vector and the relative motion vector of R with respect to O. **TCPA** is defined as time to reach CPA and is calculated by

$$TCPA(O) = \frac{||\mathbf{x}_{cpa} - \mathbf{x}_R||}{||\dot{\mathbf{x}}_R - \dot{\mathbf{x}}_O||} \tag{2}$$

where $\mathbf{x}_{cpa} \in \mathbb{R}^2$ is the CPA position on $\{W\}$, and $\dot{\mathbf{x}}_R - \dot{\mathbf{x}}_O$ is the relative motion vector of R with respect to O.

On the other hand, the **relative bearing** of O with respect to R is defined as $\psi_{O|R}$ on $\{R\}$. As per maritime convention, $\psi_{O|R}$ represents a compass bearing of O in clockwise direction measured from R's heading.

B. Motion Attributes-based Clustering

The goal of the clustering of static and dynamic obstacles with similar **near future motion attributes** with respect to a controlled ASV is to holistically find an action that can avoid the whole cluster of obstacles in a safer way than avoiding one obstacle at the time.

We define a **cluster** as a group of static and dynamic obstacles with temporal (s_t), spatial (s_d), and angular similarity (s_a), such that a controlled ASV should not enter an obstacle's domain as well as narrow areas between obstacles until they are clear. More formally:

Definition 1: (Cluster of obstacles) If $|TCPA(O_i) - TCPA(O_j)| \leq s_t \wedge |DCPA(O_i) - DCPA(O_j)| \leq s_d \wedge |\psi_{O_i|R} - \psi_{O_j|R}| \leq s_b$, a pair $\{O_i, O_j\}$ becomes a cluster C_k. For $O_m \neq O_i, O_j$, if O_m satisfies the clustering condition with O_i or O_j, the cluster C_k is enlarged to $C_k = \{O_i, O_j, O_m\}$.

Definition 2: (Evasive action for cluster) For a cluster C_k, expected track $\mathcal{T}$ of R should satisfy $\nexists \mathcal{T} \cap \mathcal{C}(O_i) \wedge \nexists \mathcal{T} \cap \mathcal{L}(O_i, O_j) \quad \forall\, O_i, O_j \in C_k$ where $\mathcal{C}(O_i)$ is a collision boundary of O_i defined in Section III-C, and $\mathcal{L}(O_i, O_j) \in \mathbb{R}^2$ is a line segment as a linkage between $\mathbf{x}_{O_i}$ and $\mathbf{x}_{O_j}$ until R clears C_k.

The threshold parameters (s_t, s_d, s_a) are determined by maneuvering characteristics aligned with an operational goal [27], e.g., tactical diameter, abort distance, advance/transfer – see Section IV-A for the values.

1) temporal similarity s_t: determined by **TCPA**, represents 'How much temporal room does an ASV have prior to CPA by obstacle(s) in a cluster?'. For example, as shown in Fig. 3 and Table I, obstacle O_b and O_c are grouped because of similar TCPAs with respect to the controlled vehicle R. Intuitively, two obstacles – one moving at fast speed (O_c), the other one at slow speed (O_b) – can be grouped based on similar passing time with respect to R, despite being far located to each other at the current time. On the other hand, two obstacles (e.g., O_d, O_f) would not necessarily belong to a cluster with respect to R, because they have large TCPA difference albeit located in proximity at the current time.

2) spatial similarity s_d: determined by **DCPA**, represents 'How much safety room does an ASV have as a distance to the closest point of approach?'. If the ASV expects to pass obstacles located in close proximity with similar DCPA, the obstacles are to be clustered at the current time. More specifically, as shown in Fig. 3, while obstacle O_a, O_b, O_c meet s_t criteria, O_b and O_c are grouped whereas obstacle O_a and O_b are not, due to the big difference of DCPAs.

3) angular similarity s_b: determined by **relative bearing**, represents 'How closely are the obstacles distributed as relative view angles from the ASV?'. The relative bearing is an important factor to identify any risk of collision ([21] and Rule 6 in [3]) with an obstacle that is is on a *collision course* to the controlled ASV without a noticeable change of relative bearing. For example, as shown in Fig. 3, if obstacle O_b and O_d (meeting both s_t, s_d criteria) are monitored with similar relative bearings ($|\psi_{O_b|R} - \psi_{O_d|R}| \leq s_a$), the ASV can consider the directional distribution of the cluster as a collision indicator. On the other hand, while obstacles O_f and O_g are not on a collision course due to the shift of $\psi_{O_f|R}, \psi_{O_g|R}$ across time, a cluster generated by O_f, O_g still remains valid by meeting s_a criteria.

Our proposed method keeps track of the obstacles and updates cluster(s) from t_i to t_{i+1}, if there is a change in the marine traffic. Such a change includes encounter situations (e.g., head-on to crossing), motions of obstacles and ASV, and obstacle appearance or disappearance in $\mathcal{S}$. The updates are performed using a *moving average window* to obtain motion-attribute information – TCPA, DCPA, relative bearing – of each detected obstacle. If the ASV detects an obstacle at a certain time (t_d) within $\mathcal{S}$ and receives minimum required number of data (e.g., n_{tcpa} for TCPA) from AIS broadcast messages, the ASV continues to process and update data by just shifting the average window at t_{d+1}. That minimum required number is set according to a *delay* during obstacle detection and tracking in real-world maritime navigation. The moving window also has a *smoothing effect*

TABLE I

NUMERICAL DETAIL OF EACH VEHICLE MOTION UNDER CONGESTED TRAFFIC IN FIG. 1 SCENARIO.

ID	Start [m, m]	Heading [°]	Speed [m/s]
R	(0, -100)	0	2.5
O_a	(-20, 20)	175	1.0
O_b	(-5, 40)	180	1.0
O_c	(-40, 90)	170	3.0
O_d	(90, -10)	250	2.0
O_e	(80, -20)	250	2.0
O_f	(-120, -20)	100	1.0
O_g	(50, -150)	0	3.5

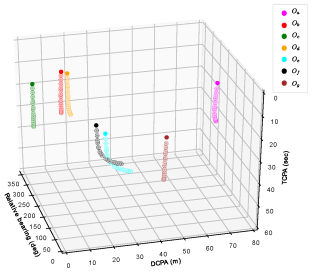


Fig. 3. Motion attributes – TCPA, DCPA, relative bearing – by multiple obstacles under congested traffic in Fig. 1 scenario. The three attributes are monitored with respect to the controlled ASV (R) while it is assumed that obstacles do not take an evasive action.

on noises caused by disturbing factor(s), e.g., environmental disturbances, robot motion oscillation, while focusing on the recent streamed data.

Our ASV will keep track of any obstacle O_i whose TCPA satisfies ASV's *lookahead* monitoring time, i.e., $0 \leq TCPA(O_i) \leq T_{monitor^+}$ where $T_{monitor^+}$ is a forward looking threshold, a positive TCPA of an obstacle decreases as it approaches. This pruning process contributes to not only computational efficiency by reducing obstacles in interest [30], but also maneuver robustness by preventing a potential dead-lock scenario similar to Guard Zone on a marine RADAR [31]. After obstacle pruning, we perform a hierarchical clustering to find group(s) of obstacle(s) meeting all s_t, s_d, s_a thresholds in accordance with *Definition* 1. Note that the proposed algorithm performs a new clustering process in case of the following events: (1) when a new obstacle O_j appears with $0 \leq TCPA(O_j) \leq T_{monitor^+}$ inside $\mathcal{S}$; (2) when a monitored obstacle O_j either goes out of $\mathcal{S}$ or gets cleared, i.e., $TCPA(O_j) \leq T_{monitor^-}$ inside $\mathcal{S}$ where $T_{monitor^-}$ is backward looking threshold to check clearance, as an already passed obstacle has a negative TCPA; and (3) when periodic monitoring catches a significant motion change of an obstacle, i.e., θ or v. Such process can adaptively update cluster(s) formed by obstacle(s) in dynamic marine traffic scenarios.

C. Geometric Analysis of Action Boundary

To find evasive action(s) for cluster(s), we use a virtual domain concept – so called *ship domain* – of an obstacle [27] as shown in Fig. 4 (a). The boundary is divided into two areas: *collision boundary* $\mathcal{C}$ where a controlled ASV shall not enter as otherwise it would be considered collision even if ASV might pass without a physical contact; and *risky boundary* $\mathcal{R}$ where the ASV can enter, but need to navigate with caution while maintaining a safety level. We include uncertainty coefficient to determine the size of the domains – see the details in [27] where we introduced this geometric boundary for a single obstacle; here we report the main elements to understand the extension to clusters of multiple obstacles.

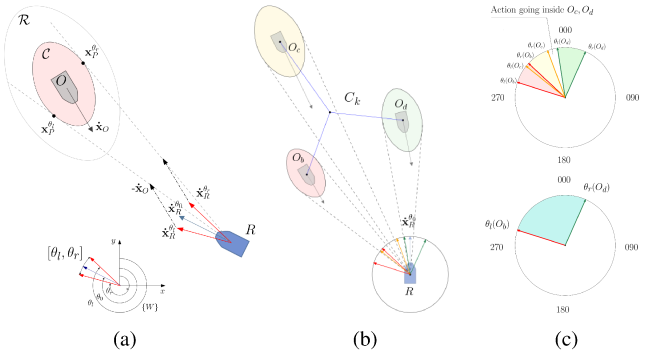


Fig. 4. Geometric representation of action boundary for obstacle(s) and cluster. **(a)** conceptual diagram of an action boundary in a single obstacle. **(b)** boundary analysis for multiple obstacles with varying ship domain sizes in the cluster C_k in Fig. 1 example. **(c)** *top*: boundary analysis result for each cluster; *bottom*: holistic analysis proposed by this study identifies determinant obstacles as O_b, O_d such that $[\theta_l(O_b), \theta_r(O_d)]$ encompasses the action boundaries of the obstacles in the cluster.

Based on this geometric boundary, our algorithm derives margins of evasive action(s), i.e., action(s) leading to tangent lines with respect to the ship domain $\{O\}$. As shown in Fig. 4 (a), R must have two tangent lines towards $\mathcal{C}$ with respect to $\{O\}$, except for a case that the ASV is on the edge of or inside $\mathcal{C}$, i.e., collision. A tangent line is a linear span of a relative motion of R with respect to O. Finding a motion vector at a specific speed towards a tangent line can be solved by a system of linear equations as shown in our previous study [27]:

$$\begin{cases} (\dot{\mathbf{x}}_R^\theta - \dot{\mathbf{x}}_O) \times (\mathbf{x}_P^\theta - \mathbf{x}_O) = \mathbf{0} \\ \left\|\dot{\mathbf{x}}_R^\theta\right\| = \left\|\dot{\mathbf{x}}_R^{\theta_0}\right\| \end{cases} \tag{3}$$

where $\mathbf{x}_P^\theta$ is a tangent point of $\mathcal{C}$ when the ASV's heading is θ, θ_0 is the current heading, and $\dot{\mathbf{x}}_R^\theta$, $\dot{\mathbf{x}}_R^{\theta_0}$ is velocity of ASV for θ, θ_0, respectively. Note that Equation (3) for two tangent lines can return vectors based on two compass heading values θ_l, θ_r on $\{W\}$ such that $\dot{\mathbf{x}}_R^{\theta_l}$, $\dot{\mathbf{x}}_R^{\theta_r}$ can lead to the left, right boundary of the obstacle domain. We define this $[\theta_l, \theta_r]$ as an **action boundary** for O. Therefore, if the controlled ASV takes an action outside of $[\theta_l, \theta_r]$, the ASV can avoid a collision. While in maritime navigation a heading change is preferred over a speed change, the proposed approach can still find motion vector(s) and relevant action boundary by prediction based on different speeds.

By extending the concept of the action boundary into a cluster level (multiple obstacles in a group), the proposed algorithm finds a motion vector to avoid collision(s) based on a holistic view – see Fig. 4 (b), (c). To determine a holistic action boundary for a cluster, the proposed algorithm first identifies determinant obstacle(s) in a cluster C_k as follows:

$$\underset{O_i, O_j \in C_k}{\arg\max}\ \Theta(\theta_l(O_i), \theta_r(O_j)) \tag{4}$$

where O_i, O_j are determinant member obstacles in a cluster C_k, and $\Theta(\cdot,\cdot)$ is clockwise angle difference on $\{W\}$ between the left and right action boundary of obstacles. Then, the action boundary of a cluster C_k is represented as $[\theta_l(C_k), \theta_r(C_k)] = [\theta_l(O_i), \theta_r(O_j)]$ such that any action outside this action boundary makes the ASV avoid the cluster C_k. Formally:

Proposition 1: For a cluster C_k and its action boundary at a specific speed v as $[\theta_l(O_i), \theta_r(O_j)]$ where $O_i, O_j \in C_k$, any action $a \notin [\theta_l(O_i), \theta_r(O_j)]$ can make ASV avoid a collision with C_k, i.e., $O_m \; \forall O_m \in C_k$.

Proof: Suppose there is an action $a \notin [\theta_l(O_i), \theta_r(O_j)]$ that makes the controlled ASV *collide* with at least one $O_m \in C_k$. In other words, there exists at least one action boundary such that $a \in [\theta_l(O_m), \theta_r(O_m)]$. According to the definition of an action boundary with Equation (4), $\bigcup_m [\theta_l(O_m), \theta_r(O_m)] \subseteq [\theta_l(O_i), \theta_r(O_j)] \; \forall O_m \in C_k$. Therefore, if $a \notin [\theta_l(O_i), \theta_r(O_j)]$, it must satisfy $a \notin [\theta_l(O_m), \theta_r(O_m)] \; \forall O_m \in C_k$. This contradicts the initial supposition. ■

The constrained action space reduces collision risk in high-traffic scenarios – see Fig. 4 (c) *top* for an action example

towards a confined space between obstacles. Note that there exists a case where Equation (4) returns $O_i = O_j$: (a) only one obstacle exists in a cluster, i.e., $||C_k|| = 1$; or (b) an action boundary formed by O_i encompasses any other combinations of action boundaries in C_k where $||C_k|| \geq 2$.

As a result, the proposed algorithm can derive the aggregated action boundary from individual clusters in interest at that time $\bigcup_k [\theta_l(C_k), \theta_r(C_k)]$. Given a possible action space $\mathcal{A}$, the aggregated boundary becomes *no-go-zone* $\mathcal{A}'$ where $\mathcal{A}' \subseteq \mathcal{A}$. An action $a \in \mathcal{A} - \mathcal{A}'$ prevents a controlled ASV from colliding with any $O_i \in C_k$ as well as entering a confined space between obstacles. Formally, this can be derived by extending *Proposition* 1: if $a \notin \bigcup_k [\theta_l(C_k), \theta_r(C_k)]$ leads to a collision with an obstacle in a certain C_p, this contradicts that a can avoid a collision with any obstacles in C_p, because $a \notin \bigcup_k [\theta_l(C_k), \theta_r(C_k)]$ means $a \notin [\theta_l(C_p), \theta_r(C_p)] = [\theta_l(O_q), \theta_r(O_r)]$ where O_q, O_r is determinant obstacle in C_p.

D. Multi-Objective Optimization for Obstacle Avoidance

We define a multi-objective optimization to find the best action in the action space $\mathcal{A}$ of a controlled ASV. The action space $\mathcal{A}$ is a discrete grid space determined by a combination of heading and speed θ, v. We use $[0, 360)$ with 1° increment for θ and ratio $[0, 1]$ of the max target speed with 0.25 increment for v. Possible actions are evaluated from the feasible action space $\mathcal{A} - \mathcal{A}'$. Such actions are evaluated as a weighted sum of four objectives **(a)** heading change from a waypoint; **(b)** heading change from a local target heading; **(c)** speed change; and **(d)** safety level:

$$J(\theta, v) = w_f f(\theta) + w_{f_2} f_2(\theta) + w_g g(v) + w_h h(\theta, v) \quad (5)$$

where (θ, v) is an action $a \in \mathcal{A} - \mathcal{A}'$ to be evaluated; $f(\cdot)$, $f_2(\cdot)$ are a heading change cost for required heading towards a destination or local target heading while avoiding obstacles, respectively; $g(\cdot)$ is a speed change cost; $h(\cdot, \cdot)$, is a safety level cost; and w_f, w_{f_2}, w_g, w_h are related weights.

More specifically, f cost function represents 'how much an evaluated heading will be offset from a direction towards a waypoint and can be calculated as follows:

$$f(\theta) = \frac{|\theta_{wp} - \theta|}{\Delta_{max}(\theta)} \quad (6)$$

where θ_{wp} is a true bearing towards the current waypoint, and $\Delta_{max}(\theta)$ is the possible maximum heading change in $\mathcal{A}$ for prediction. f cost can determine the extent of how strictly the ASV follows a path towards the current waypoint. f_2 cost function represents 'how much an evaluated heading will be offset from a local target heading' and calculated in the same way as f except θ_{tgt} instead of θ_{wp} where θ_{tgt} is the current target heading while avoiding obstacles. To overcome limitations of the trivial hysteresis method [7] as noted in [15] and prevent an ASV from *chattering*, we introduced f_2 cost. Intuitively, f_2 has a conflicting role against f, but stabilize the motion such that the robot maintains a passage direction (e.g., left to left) unless an imminent risk arises (e.g., newly detected obstacle(s) blocks the passage). g cost function represents 'how much an evaluated speed is offset from a target speed' and can be calculated as follows:

$$g(v) = \frac{|v_{target} - v|}{\Delta_{max}(v)} \quad (7)$$

where v_{target} is a target speed to the waypoint, and $\Delta_{max}(v)$ is the possible maximum speed change in $\mathcal{A}$ from the current speed. g cost can determine the extent of how strictly the ASV keeps its speed compared to the target speed. Last, h cost function represents 'how much safety level is expected when the ASV passes an obstacle' and can be calculated as:

$$h(\theta, v) = \begin{cases} 1 & \text{if} \quad (\tau_{\theta,v} \leq \check{\tau}), \\ 0 & \text{else if} \, (\tau_{\theta,v} \geq \hat{\tau}), \\ \frac{|\hat{\tau} - \tau_{\theta,v}|}{\hat{\tau} - \check{\tau}} & \text{else} \end{cases} \quad (8)$$

where τ is a safety level metric (DCPA in this study), $\hat{\tau}, \check{\tau}$ are the upper, lower bound of the safety level, $\tau_{\theta,v}$ is a DCPA value when a specific action (θ, v) is taken. h cost determines the safety extent of collision avoidance by ensuring a safe distance from the obstacle. Note that the h cost is normalized based on the upper and lower bound of the safety level as done in other work [11].

Finally, after combining individual costs and weights defined in a feasible action space, the optimized action θ^*, v^* within the feasible action space $\mathcal{A} - \mathcal{A}'$ is :

$$(\theta^*, v^*) = \underset{\theta, v \in \mathcal{A} - \mathcal{A}'}{\arg\min} \; J(\theta, v) \quad (9)$$

In general, the weight parameters can be set to account for different traffic situation and end-user goal. How to set all the parameters is discussed in Section IV-A. Note that an optimal action may differ depending on a combination of the weights, although, fixing the weights, the best action is found on Pareto optimal front [32] while ensuring collision avoidance as discussed in Section III-C.

IV. Results and Evaluation

We conducted extensive simulation tests for validation: (1) Monte Carlo simulations on a 2D simulator including noises – Stage [33] – to optimize the parameters and quantitatively demonstrate the performance of the proposed algorithm compared to state-of-the-art methods; (2) realistic 3D simulations on Gazebo [34] which included plugins for disturbances in addition to 2D simulations on actual historical data of a collision accident in congested traffic by testing generality on different robotic platforms with varying size and speed.

A. Calibration Test

We performed the Monte Carlo simulations considering our custom-made ASV, *Catabot* with length $2.5\,\mathrm{m}$, beam $1.4\,\mathrm{m}$, sensible range $100\,\mathrm{m}$, the max linear, angular speed $2.5\,\mathrm{m/s}$, $45\,^\circ/\mathrm{s}$ found by performing a real experiment shown in [27]. By finding the motion characteristics such as tactical diameter and advance, we chose $s_t = 10\,\mathrm{sec}, s_d = 15\,\mathrm{m}, s_a = 15^\circ$. For example, ASV is not allowed to enter between two obstacles forming a cluster whose bearing offset

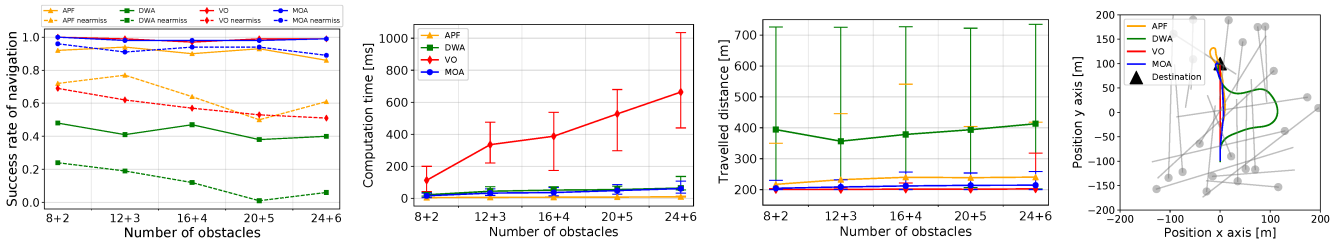


Fig. 5. Comparison of each method for collision avoidance of multiple obstacles under congested traffic. **(a)** Success rate of navigation. *Solid:* success rate without consideration of near miss area entrance, *Dotted:* success rate with near miss area entrance regarded as *collision*; **(b)** Computational time; **(c)** Travelled distance of ASV when the navigation task was successful; and **(d)** Trajectories in an example scenario with 30 obstacles (*gray*).

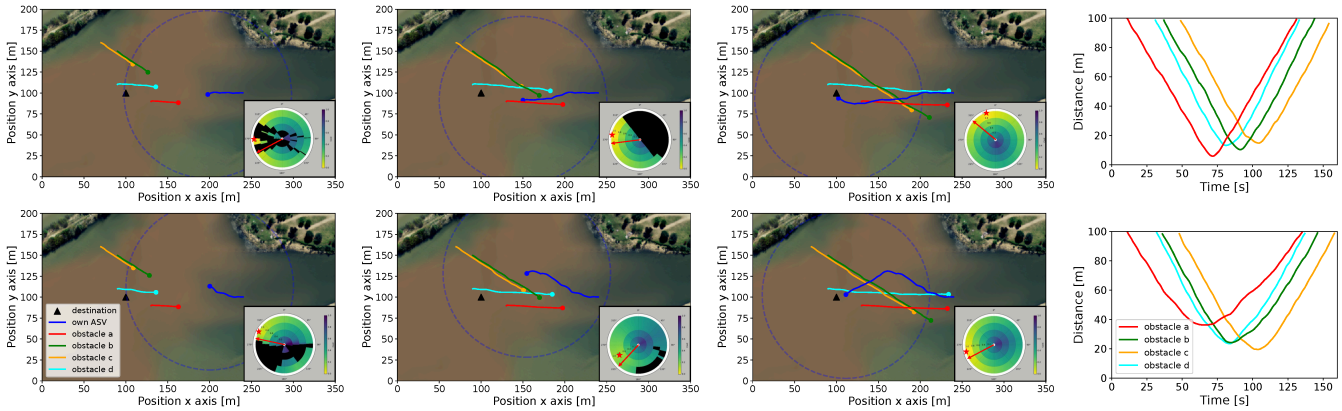
is less than s_a. We then optimized the weight parameters proposed in Section III-D by evaluating the behaviors of the controlled ASV. We used different combinations of weights (w_f, w_{f_2}, w_g, w_h) in 50 scenarios in $200\,\mathrm{m} \times 200\,\mathrm{m}$ area. For each scenario, we built a randomized environment that has 30 obstacles with varying size, speed, start and goal position, and encounter situations: head-on, overtaking, and crossing. The ranges of the weight values were set to $w_f = [0.25, 0.4], w_{f_2} = [0.05, 0.1], w_g = [0.3, 0.55], w_h = [0.1, 0.3]$, respectively, with total sum as 1. Note that each combination of the weights was set to the following inequalities for typical behaviors of vehicles in maritime navigation: $w_f \leq w_g$, $w_h \leq w_f$, and $w_{f_2} < w_f$. Intuitively, $w_f \leq w_g$ denotes that a vehicle prefers a change of its heading to its speed for collision avoidance. $w_h \leq w_f$ denotes that a vehicle can adjust its heading to enter a space between obstacles belonging to different clusters, if the safety level determined by w_h permits. Note that according to Definitions 1 and 2, a desired action output ensures the ASV does not enter an obstacle's collision boundary $\mathcal{C}$ in a same cluster regardless of the safety level. Last, $w_{f_2} < w_f$ denotes that a vehicle heads to its goal, while two weights act in a compensatory manner to prevent chattering behavior. Based on these tests, we chose $w_f = 0.34, w_{f_2} = 0.05, w_g = 0.49, w_h = 0.12$ for the best performance observed across the total scenarios. Note that the performance was not heavily affected by varying w_f, w_g, w_h within ± 0.15 while keeping the aforementioned inequality principles.

B. Comparative Analysis of Performance

We compared behaviors by our proposed method (MOA) and by the state-of-the-art methods. We chose as state-of-the art methods those that have shown success in mobile robotics: Velocity Obstacle (VO) [7], APF (Artificial Potential Field) [12], and DWA (Dynamic Window Approach) [35]. Note that we did not include rule cost parameters which can be applied to only a single encounter (e.g., COLREG cost in [7]), since our research focuses on multiple encounters, and role conflicts. During calibration of the proposed method, we also calibrated the state-of-the art methods' principal parameters (e.g., attractive, repulsive force coefficient in APF). We then tested each method by incrementing the number of obstacles – $10, 15, 20, 25, 30$ obstacle cases with random 100 scenarios per method, i.e., total $2,000$ simulations with start, goal position as $[0, -100], [0, 100]$. We adopted the same randomized scheme used in Section IV-A for Monte Carlo simulations. We compared the performances based on the following quantitative metrics: *success rate*, *computational time*, and *travelled distance*.

First, the proposed MOA method shows significantly better navigation *success rate* than the other methods as shown in Fig. 5 (a). We also considered cases where the ASV entered $\mathcal{C}$ of any obstacles as 'collision' despite non-physical contact, according to the definition of the ship domain. MOA's performance is not heavily affected as 0.927 on average, while outperforming other methods (1.2–1.7 times higher). On the other hand, VO's performance drastically decreases, as we observed (1) VO's non-holistic action decision on single-to-single encounters makes the ASV enter a small space between obstacles (see Fig. 4) as shown in Fig. 5 (d); and (2) a chattering behavior of the ASV, confused by many detected obstacles, leads to late, inefficient evasive actions. DWA showed the lowest success rate due to the algorithm originally designed for static environments. We analyzed the small number of collisions/near misses by the proposed method. We found the cause of entering $\mathcal{C}$ under congested traffic as: (1) there is latency in the sensing capability via the AIS broadcast message. While we modelled fast type-A AIS with $1\,\mathrm{Hz}$, a relative robot motion by the controlled ASV and the obstacle makes meters of blind period ; (2) there are dynamic constraints in turning or reducing speed. Despite motion characteristics adopted in the proposed approach, the prediction of the motion at a current time stamp may not necessarily correspond exactly to real motion. In other words, an optimized action might not be always reached at the next time stamp, e.g., opposite direction of the heading to the current heading in the action space.

The experimental computation time by MOA shows real-time capability of collision avoidance ($61\,\mathrm{ms}$ on average in 30 obstacles), negligibly affected by the increasing number of obstacles – see Fig. 5 (b). Here, we measured *computational time* taken by the planner to return a desired action output, considering that the fast action update is a key to successful collision avoidance. APF shows the fastest performance based on simple force calculations, while we observed problems such as a local minima and oscillation of actions caused by multiple obstacles' forces acting on the ASV. VO showed the largest linear increases with the largest varying ranges. This observation is aligned with the complexity $\mathcal{O}(|\theta||v||n|)$ where $|\theta|, |v|$ are the size of an



(a) time elapsed: 50 s (b) time elapsed: 100 s (c) time elapsed: 150 s (d)

Fig. 6. Comparison of trajectories and action spaces by the-state-of-the-art VO (*top*) vs the proposed MOA (*bottom*) method for collision avoidance of multiple obstacles in Porto Alegre, Brazil with locally varying water/wind currents. The controlled ASV (*Blue*) with length 4 m, max speed 1 m/s departed from the river mouth $(250, 100)$ to the open waters $(100, 100)$, while the obstacles merged with arrival traffic. In the action space, *red* star and *red* line represents the best action and current action, respectively. **(a)** VO attempted to go between obstacle a, d. MOA proactively took a right-side action while obstacle a, b, d were clustered. Note that obstacle d was under tracking process as it just entered the sensible range 100 m; **(b)** VO passed the obstacles, particularly in very close proximity to obstacle a, b. MOA safely passed obstacles. Note that clusters were adaptively identified, e.g., obstacle (a, b, d) to obstacle (a, d) and (b, c) based on changes of motion attributes; **(c)** VO arrived at the destination. MOA cleared obstacles while still proceeding to the destination; and **(d)** Distance comparison shows MOA avoid obstacles under congested traffic in a safer manner.

action space in terms of heading, speed change, respectively, and $|n|$ is the number of obstacles [9]. We noted that we were able to achieve real-time performance for our method because of the following algorithmic optimizations: (1) TCPA-based pruning of obstacles in priority; (2) parallel processing of detection, monitoring, and clustering module despite the clustering as $O(|n|^2)$ (see Fig. 2); and (3) instead of $\mathcal{O}(|\theta||v||n|)$, the clustering process makes at most two obstacles (*determinant obstacles*) in a cluster considered for cost analysis such that the complexity is $\mathcal{O}(|\theta||v||k|)$ where $|k|$ is the number of clusters. Note that $|k| \leq |n|$ theoretically, but $|k| < |n|$ in real-world traffic scenarios.

Last, Fig. 5 (c), (d) shows MOA has reasonable travelled distances, while avoiding multiple obstacles in congested traffic. In the most challenging situations with 30 obstacles, MOA still showed stable performance, while VO without TCPA-priority process experienced cases with high variations of distance due to *deadlock* situations. DWA showed the longest distance travelled while avoiding obstacles, because the method is originally designed for static environments.

C. Real-world like Environment and Real Accident Case

To further test the robustness of our proposed method under real-world conditions, we validated the proposed method on a different robotic platform and in a 3D simulator with environmental disturbances [36]. As shown in Fig. 6, the experimental area covers Dilúvio river mouth (30°03′ S, 51°14′ E) in Porto Alegre, Brazil. The environmental disturbances include buoyancy, time- and spatial- varying 3D waves and water/wind currents modelled by HEC RAS[1] and openFoam[2], affecting the 6-DOF of the ASV and the obstacles. Note that we could test up to total 5 vehicles (own+obstacles) due to the computationally-demanding simulation which dropped significantly the real-time factor of the 3D simulator. The traffic scenario was included according to the in and out traffic flow[3].

TABLE II
NUMERICAL DETAIL OF EACH SHIP INVOLVED IN ACCIDENT.

ID	Heading [°]	Speed [m/s]
R	220	10.0
ship a	85	8.8
ship b	40	6.2
ship c	95	8.4
ship d	60	10.0

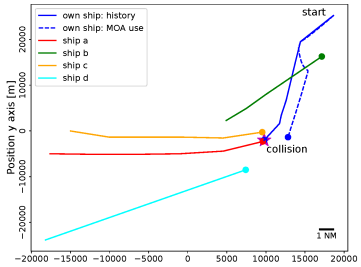


Fig. 7. Trajectories of ships during 1.5 hours prior to a collision off coast Japan. Southwest-bound ship (*solid blue*) collided with East-bound ship a while the proposed method successfully avoided all the ships (*dotted blue*) in the same scenario. Note that the involved vehicles were made anonymous.

Fig. 6 shows that MOA avoids more safely than other methods multiple obstacles. The VO-based ASV passed a small space between obstacles and thus experienced action chattering within a small portion of the feasible action space. Selecting an action within that small portion of the action space could raise potential risk particularly under locally varying real-world disturbances which caused the oscillatory motions of obstacles. Note that the proposed approach handled the noisy environment and uncertain motions of obstacles by a holistic evaluation of the action space and its cost, leaving a safety level margin to prevent a close-quarter situation as shown in Fig. 6 (d). Interestingly, despite the non-explicit implementation of rule-compliance, the MOA's best action in the given scenario turns out to be all rule-compliant (single-to-single perspective).

We additionally tested applicability of the proposed method to a large-scale vessel with high speed in a real-world accident as shown in Fig. 7. More specifically, we

[1] https://www.hec.usace.army.mil/software/hec-ras/
[2] https://www.openfoam.com/
[3] https://www.marinetraffic.com/

tested the proposed method based on historical AIS records of the collision at high-traffic Sagami Nada Bay ($34°31'$ N, $139°05'$ E), Japan on 2017 [37]. Note that the controlled ship with length $153.9\,\mathrm{m}$, linear, angular speed $10\,\mathrm{m/s}$, $20\,°/\mathrm{s}$ according to the accident report was assumed to have a sensible range of 12 nautical miles (NM). While a principal cause of the accident was *late, improper action* with respect to ship a, our approach proactively adjusted the course with respect to ship b first such that own ship could pass multiple obstacles (ship a, c as a cluster) with safe distance off.

V. Conclusion and Future Steps

Our proposed collision avoidance method in high marine traffic scenarios can achieve safer trajectories than other state-of-the-art methods by using motion attribute-based clustering, geometric framework for the feasible action space, and multi-objective optimization based on a holistic view of obstacles detected by range sensors. While the update frequency from obstacle's broadcasting message can affect the behaviors leading to near miss, we optimized the proposed algorithm by considering uncertainty of motions monitored by sensors in real-world scenarios.

We will integrate our proposed method with real multi-sensor fusion modules and on a real ASV. We plan to explicitly consider kinematic and dynamic properties of the controlled ASV to better cope with uncertainty of external disturbances. As a long-term goal, we will explore a high-level global planning such that the proposed collision avoidance from local perspective can be integrated towards full marine autonomy.

Acknowledgement

We would like to thank Donki Kim and Haesang Jeong for help with historical data analysis of marine traffic. This work is supported in part by the Burke Research Initiation Award and NSF CNS-1919647, 2144624, OIA1923004.

References

[1] F. Ferreira, J. Alves, C. Leporati, A. Bertolini, and E. Bargelli, "Current regulatory issues in the usage of autonomous surface vehicles," in *Proc. OCEANS*, 2018.

[2] A. Vagale, R. Oucheikh, R. T. Bye, O. L. Osen, and T. I. Fossen, "Path planning and collision avoidance for autonomous surface vehicles i: A review," *J. Mar. Sci. Technol.*, 2021.

[3] International Maritime Organization, *Convention on the international regulations for preventing collisions at sea, 1972 (COLREGs)*, 1972.

[4] K. Woerner, M. R. Benjamin, M. Novitzky, and J. J. Leonard, "Quantifying protocol evaluation for autonomous collision avoidance," *Auton. Robot.*, vol. 43, no. 4, Apr. 2019.

[5] U. Nations, *Review of Maritime Transport 2021*. United Nations Conference on Trade and Development, 2022.

[6] H. Lyu and Y. Yin, "Colregs-constrained real-time path planning for autonomous ships using modified artificial potential fields," *J. Navigation*, vol. 72, no. 3, 2019.

[7] Y. Kuwata, M. T. Wolf, D. Zarzhitsky, and T. L. Huntsberger, "Safe maritime autonomous navigation with colregs, using velocity obstacles," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, 2014.

[8] D. K. Kufoalor, E. F. Brekke, and T. A. Johansen, "Proactive collision avoidance for asvs using a dynamic reciprocal velocity obstacles method," in *Proc. IROS*, 2018.

[9] Y. Cho, J. Han, and J. Kim, "Efficient COLREG-Compliant Collision Avoidance in Multi-Ship Encounter Situations," *IEEE Trans. Intell. Transp. Syst.*, Oct. 2020.

[10] M. Hoy, A. S. Matveev, and A. V. Savkin, "Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey," *Robotica*, vol. 33, no. 3, 2015.

[11] M. R. Benjamin, "Capturing velocity function plateaus for efficient marine vehicle collision avoidance calculations," in *Proc. OCEANS*, 2018.

[12] Y. Xue, D. Clelland, B. S. Lee, and D. Han, "Automatic simulation of ship navigation," *Ocean Eng.*, vol. 38, no. 17, 2011.

[13] P. Chen, Y. Huang, J. Mou, and P. van Gelder, "Ship collision candidate detection method: A velocity obstacle approach," *Ocean Eng.*, vol. 170, 2018.

[14] N. Wang, Y. Gao, Z. Zheng, H. Zhao, and J. Yin, "A hybrid path-planning scheme for an unmanned surface vehicle," in *Proc. ICIST*, 2018.

[15] I. B. Hagen, D. K. Kufoalor, E. F. Brekke, and T. A. Johansen, "Mpc-based collision avoidance strategy for existing marine vessel guidance systems," *Proc. ICRA*, Sep. 2018.

[16] A. Lazarowska, "Ship's trajectory planning for collision avoidance at sea based on ant colony optimisation," *J. Navigation*, vol. 68, 2015.

[17] L. Zhao and M.-I. Roh, "Colregs-compliant multiship collision avoidance based on deep reinforcement learning," *Ocean Eng.*, vol. 191, 2019.

[18] Y. He, Z. Li, J. Mou, W. Hu, L. Li, and B. Wang, "Collision-avoidance path planning for multi-ship COLREGs," *Transportation Safety and Environment*, vol. 3, no. 2, Apr. 2021.

[19] Y. Tao, K. Wei, J. Chen, D. Qi, Z. Jin, and H. Wang, "Collision avoidance path planning in multi - ship encounter situations," *J. Mar. Sci. Technol.*, 2021.

[20] "A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres," *Annual Reviews in Control*, vol. 36, no. 2, Dec. 2012.

[21] A. Vagale, R. T. Bye, R. Oucheikh, O. L. Osen, and T. I. Fossen, "Path planning and collision avoidance for autonomous surface vehicles II: a comparative study of algorithms," *J. Mar. Sci. Technol.*, 2021.

[22] D. Kim, K. Hirayama, and T. Okimoto, "Distributed stochastic search algorithm for multi-ship encounter situations," *J. Navigation*, vol. 70, 2017.

[23] X.-Y. Zhou, J.-J. Huang, F.-W. Wang, Z.-L. Wu, and Z.-J. Liu, "A study of the application barriers to the use of autonomous ships posed by the good seamanship requirement of colregs," *J. Navigation*, vol. 73, no. 3, 2020.

[24] A. T. Simon Gault Steven Hazelwood, *Marsden on collisions at sea*, 13th ed. London: Sweet & Maxwell, 2003.

[25] R. Zhen, M. Riveiro, and Y. Jin, "A novel analytic framework of real-time multi-vessel collision risk assessment for maritime traffic surveillance," *Ocean Eng.*, vol. 145, no. September, 2017.

[26] P. Chen, M. Li, and J. Mou, "A velocity obstacle-based real-time regional ship collision risk analysis method," *Journal of Marine Science and Engineering*, vol. 9, no. 4, 2021.

[27] M. Jeong and A. Quattrini Li, "Risk vector-based near miss obstacle avoidance for autonomous surface vehicles," in *Proc. IROS*, 2020.

[28] "Autonomous ship collision avoidance navigation concepts, technologies and techniques," *J. Navigation*, vol. 61, no. 1, 2008.

[29] International Maritime Organization, *International convention for the safety of life at sea, 1974 (SOLAS)*, 1972.

[30] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. ICRA*, 2008.

[31] J. K. Robson and O. O. Qj, "Overview of collision detection in the ukcs," in *AEA Technology for Health and Safety Executive*, 2006.

[32] N. Basilico and F. Amigoni, "Exploration strategies based on multi-criteria decision making for searching environments in rescue operations," *Autonomous Robots*, vol. 31, no. 4, Sep. 2011.

[33] R. T. Vaughan, "Massively multi-robot simulation in stage," *Swarm Intelligence*, vol. 2, 2008.

[34] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Proc. IROS*, vol. 3, 2004.

[35] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, Mar. 1997.

[36] M. Paravisi, D. H. Santos, V. Jorge, G. Heck, L. M. Gonçalves, and A. Amory, "Unmanned surface vehicle simulator with realistic environmental disturbances," *Sensors*, vol. 19, no. 55, 2019.

[37] NTSB, "Marine accident report: NTSB/MAR-20/02, PB2020-101007," National Transportation Safety Board, 2020.