

2017-10-10

Obstacle Avoidance Approaches for Autonomous Navigation of Unmanned Surface Vehicles

Polvara, R

<http://hdl.handle.net/10026.1/10053>

10.1017/S0373463317000753

Journal of Navigation

Cambridge University Press (CUP)

All content in PEARL is protected by copyright law. Author manuscripts are made available in accordance with publisher policies. Please cite only the published version using the details provided on the item record or document. In the absence of an open licence (e.g. Creative Commons), permissions for further reuse of content should be sought from the publisher or author.

"This is the author's accepted manuscript. The final published version of this work (the version of record) is published by *Journal of Navigation*, Cambridge University Press, pp. 1–16. doi: DOI: 10.1017/S0373463317000753. This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher."

Obstacle Avoidance Approaches for Autonomous Navigation of Unmanned Surface Vehicles

Riccardo Polvara¹, Sanjay Sharma¹, Robert Sutton¹, Jian Wan¹
and Andrew Manning^{1,2,3}

¹(Autonomous Marine Systems Research Group, Faculty of Science and Engineering,
University of Plymouth, Plymouth, Devon, PL4 8AA, UK)

²(HR Wallingford, Howbery Park, Wallingford, Oxfordshire, OX10 8BA, UK)

³(Geography & Geology, School of Environmental Sciences, Faculty of Science &
Engineering, University of Hull, Hull, HU6 7RX, UK)

(E-mail: riccardo.polvara@plymouth.ac.uk)

The adoption of a robust collision avoidance module is required to realise fully autonomous Unmanned Surface Vehicles (USVs). In this work, collision detection and path planning methods for USVs are presented. Attention is focused on the difference between local and global path planners, describing the most common techniques derived from classical graph search theory. In addition, a dedicated section is reserved to intelligent methods, such as artificial neural networks and evolutionary algorithms. Born as optimisation methods, they can learn a close-to-optimal solution without requiring large computation effort under certain constraints. Finally, the deficiencies of the existing methods are highlighted and discussed. It has been concluded that almost all the existing methods do not address sea or weather conditions, or do not involve the dynamics of the vessel while defining the path. Therefore, this research area is still far from being considered fully explored.

KEYWORDS

1. Unmanned Surface Vehicle. 2. Collision Avoidance. 3. Path Planning.

Submitted: 21 March 2016. Accepted: XX August 2017.

1. INTRODUCTION. Since their first development, the military community showed interest in Unmanned Surface Vehicles (USVs) for their deployment in force protection and surveillance scenarios (Bertram, 2008). Other examples are given by the *Spartan* produced by the US Space and Naval Warfare System Center (**SSC**) in San Diego, the *Delfim* developed by the Portuguese Dynamical Systems and Ocean Robotics (**DSOR**) Laboratory (Alves et al., 2006), and the *Springer* developed by Plymouth University (Naeem and Sutton, 2009). Most of the vessels are dual-purpose vehicles, i.e. they can both be controlled remotely and operated autonomously.

USVs usually adopt catamaran and kayak shapes, for their roll robustness or ease of manufacture, and they are generally provided with a rudder-propeller system for propulsion

and steering. Their kinematics are modelled ignoring pitch and roll, and considering an Earth-fixed frame and a body-fixed one (Bibuli et al., 2009). The first is used to express the position and the orientation $[x \ y \ z]^T$, while the second for surge u_r and sway velocities v_r relative to the water, yaw rate r and forces and moments. Linear and angular velocities are defined by the following equations, expressed in the Earth-fixed frame:

$$\begin{cases} \dot{x} = u_r \cos\omega - v_r \sin\omega + \dot{x}_c \\ \dot{y} = u_r \sin\omega + v_r \cos\omega + \dot{y}_c \\ \dot{\omega} = r \end{cases} \quad (1)$$

where $[\dot{x}_c \ \dot{y}_c]^T$ are the sea currents that are considered constant. Although in general, the marine environment introduces a number of serious disturbances which affect the control of an USV (Bandyopadhyay et al., 2010). Examples are sea currents, tides, and sea-bed conformation. When transiting close to the coasts, the presence of tides dictates where a ship can go and how much time it will take to reach its destination. Despite the importance of weather conditions while traversing, there are only a small number of papers addressing this issue.

The existence of an obstacle detection and avoidance module requires the combining the sensing and decision making components, as shown in Figure 1 to navigate autonomously (Statheros et al., 2008; Tam et al., 2009; Hasegawa and Kouzuki, 1987; Hasegawa; 2009). The path planning problem has a long history in robotics, especially for Unmanned Ground Vehicles (UGVs) (Fahimi, 2008). A Path Planner (PP) is generally composed of a global path planner and a local one. The first (*deliberative*) must find a safe path connecting the actual pose of the robot and its destination. Whereas the local (*reactive*) must react to immediate collisions, moving the vehicle on a different route. Referring to unmanned boats, obstacles are represented by other vessels, divers, buoys, rocks, harbours, islands, and coasts.



Figure 1. Architecture of an autonomous system.

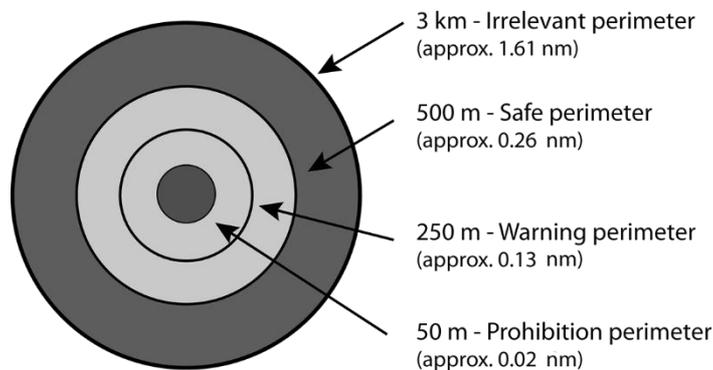


Figure 2. The proximity of the USV (Almeida et al, 2009).

The sensing module covers a key role towards full autonomy since it must identify the surrounding obstacles. When static, finding a safe path is easy and any graph search algorithm

can be used. However, detecting collisions in time is still an open problem because of sensor limitations (e.g., false positive/negative due to noisy data). Moreover, the conformation of the environment can occlude the sensors field of view, preventing a moving obstacle's detection. Therefore, autonomous navigation is a more complex task than connecting two points; multiple situations must be considered, especially under uncertainty. Hence, the scope of this paper is to offer an overview concerning the latest methods regarding collision avoidance for USVs and how to solve the problems described above.

This review is divided in the following way: Section 2 illustrates how to perceive the surrounding environment; in Section 3 the necessity of having a robust path planner is discussed, focusing on the difference between global and local path planners, and introducing intelligent methods. Finally, in Section 4 concluding remarks are given.

2. COLLISION DETECTION. To avoid obstacles, an accurate representation of the environment is required. To obtain it, data from sensors (often integrated with electronic charts) are combined to give a Two-Dimensional (2D)/Three-Dimensional (3D) model.

With regards to technology advancements, the methods for collision identification have dramatically changed in the last fifty years (Kemp et al., 2012) and sensor advantages and limitations are summarised in Table 1.

In Almeida, et al. (2009) a radar classifies targets in terms of a collision threat. A set of perimeters (Figure 2) is defined around the USV: irrelevant (3 km), safe (500 m), warning (250 m) and prohibition perimeter (50 m). Based on the Closest Point of Approach (CPA), the estimated shortest distance between the detected object and the USV, targets are classified as *no threat*, *low threat*, *potential threat* and *dangerous*. Low-cost radars are also used in Schuster et al. (2014) to compensate for the lack of an automatic identification system. Despite the fact that the radar has been extensively used for far-field obstacle detection, its accuracy is low for short distances. It is then difficult to detect changes in course in time. Moreover, high waves and water reflectivity still represent a major challenge.

Other approaches use vision methods for recording obstacles near the vessel. For example, Wang et al. (2011; 2012) use two parallel cameras. In the first monocular phase, the horizon is extracted using the Random Sample Consensus (RANSAC) (Fischler and Bolles, 1981) method (Figure 3(a)), a binary mask is built (Achanta et al., 2009) and detected salient features are considered of potential interest (Figure 3(c)). Surface obstacles are distinguished (Harris and Stephens, 1988; Bouguet, 2000) (Figure 3(d)) and interesting ones are tracked in consecutive frames to validate them as potential obstacles (Figure 4). The stereo correspondence was then realised applying an epipolar constraint to reduce the search space. This approach showed good results up to 100 m from the USV using a low-resolution image. Monocular greyscale images are used in Azzabi et al. (2014). The Sobel operator and the Hough transform are applied to extract the edges (Maini and Aggarwal, 2009; Hough, 1962), then the horizon is identified and moving objects are detected using optical flow estimation. Finally, the distance to obstacles is estimated using geometric relationships (Figure 5). Using greyscale image, the authors reduce the computational effort required by other algorithms (e.g., Ettinger et al., 2003).

Other systems use Infra-Red (IR) cameras due to their ability to discretize temperatures. The main advantage of IR cameras is the possibility to overcome the impact of various light conditions (darkness or fog), allowing operations in day and night to take place with little deterioration in performance. In Borghraef et al. (2010), floating mines were identified in the Persian Gulf. Two background subtraction algorithms, the visual background extractor (ViBe) (Barnich and Droogenbroeck, 2011) and the behavioural subtraction (Jodoin et al., 2008), were implemented. Without specifying any details about the target object, this approach

detects every anomaly in respect to the background. IR data are used also in Park and Jeong (2012), where a multi-step algorithm extracts the shape of a floating object. Vertical and horizontal edges are initially extracted to find candidate objects. Then, the background is removed and a logical AND operation with the original frame is performed to highlight the object region. The limit shown by the previous approaches relies on detecting small obstacles at long distances but, unfortunately, IR applications are still very limited. For this reason, it has not been proved that marine situations such as sea fogs, wave occlusions, variations of lighting and weather, and object reflections do not affect IR camera's performances like common video cameras (Liu et al., 2016).

From the consolidated use with an UGV, Lebbad and Nataraj (2015) adopt a LIDAR (Laser Direction and Ranging) because a laser entering the water dissipates itself, segmenting any object present there. Sensor data are combined with that of a camera and the object area is then identified. The authors discovered that the vessel's motion can distort the LIDAR image and some of the targets may appear tilted. A similar approach was proposed in Halterman and Bruch (2010) where sensor data were analysed with regards to different parameters (i.e., scanning rate and density, coverage in azimuth, elevation and range etc.). It was noticed that most of the obstacles appear clearly in the returned data while the sea surface does not provide any return. On the other hand, problems are posed by the presence of salt spray, winds, waves, currents and tides that produce image blurring and vibrations (Gal and Zeitouni, 2013).

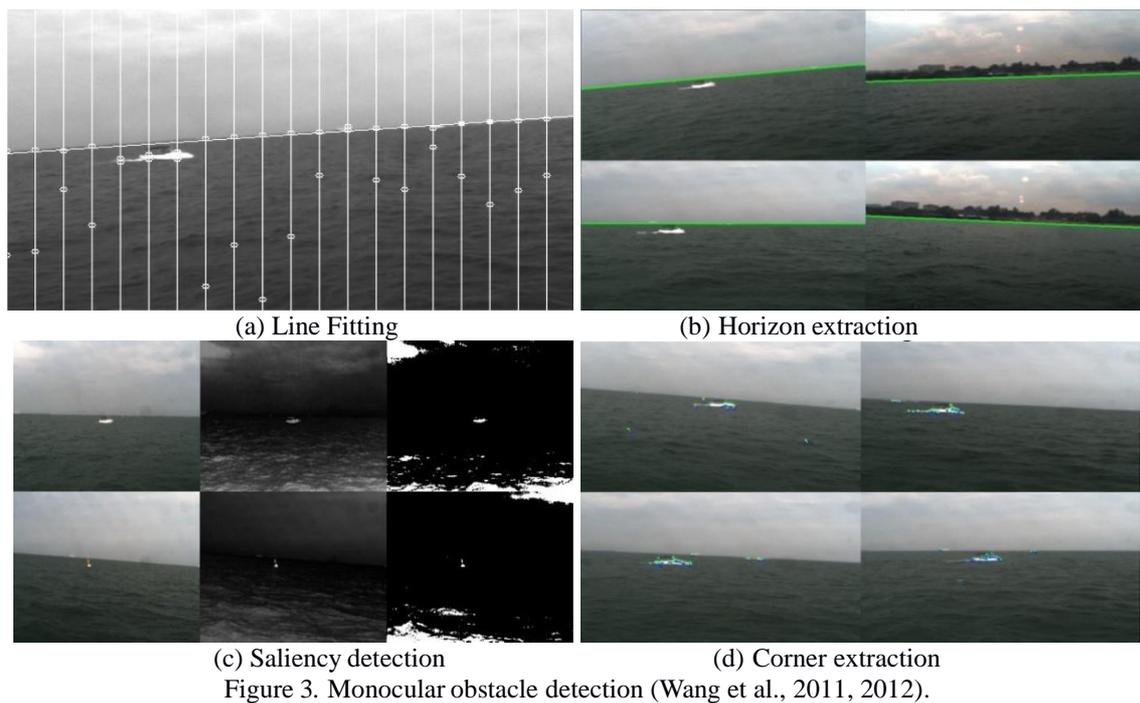




Figure 4. The obstacle is tracked among successive frames (Wang et al., 2011, 2012).

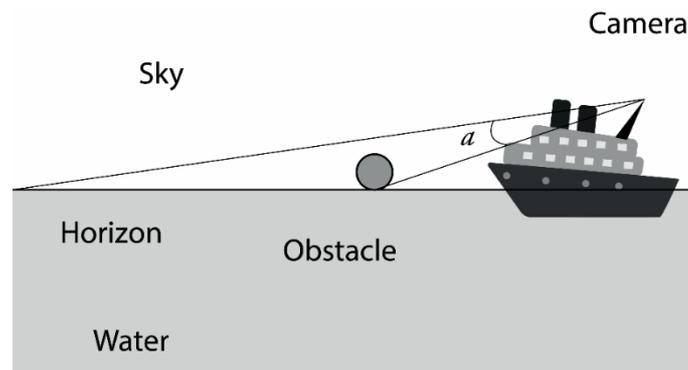


Figure 5. Estimation of the USV-obstacle distance (Azzabi et al., 2014).

3. PATH PLANNER. After perception, a representation of the world is realised. This can be done in a 2D or 3D space but the first option is often preferred because it is less computationally demanding (even though the depth dimension can also be useful for navigation purposes). As stated in Section 1, the Path Planner (PP) is usually subdivided: the global PP identifies a free path from the actual pose of the robot to a destination, while the local one tries to avoid close moving obstacles.

In the following subsections, the most recent techniques used in marine robotics are described. Before that, certain navigation rules are introduced.

3.1. *The International regulations for avoiding collisions at sea.* While underway, vessels must obey rules designed to avoid any kind of collision. Despite being manned or not, the rules must be obeyed by any kind of vessel operating at sea (Wilson et al., 2003; Lee et al., 2004; Kemp 2002; Belcher 2002). Differing, confused behaviour can lead to collisions against other marine craft. The **COLREGS (Commandant, 1999)** an abbreviation for Collision Regulations, are divided into three parts. Part A defines vessel and authority responsibilities, Part B regulates the conduct of vessels in an encounter, and Part C establishes communication protocols. Rules contained in Part B are defined as *Steering and Sailing Rules*, and the more important are reported in Table 2.

The rapid increase in the number of USVs posed interesting questions about their responsibilities while underway (Allen, 2012). This is still an open problem, the US Navigation

Table 1. Advantages and limitation of various sensors within USVs (Liu et al., 2016).

Sensors	Advantages	Limitations
Radar	1) Long detecting range; 2) Good velocity estimates; 3) Provides nearly all-weather and broad-area imagery;	1) Skewed data from fast turning manoeuvres depending on the context; 2) Limited small and dynamic target detection capability;

	4) High depth resolution and accuracy.	3) Susceptible to high waves and water reflectivity.
LIDAR	1) Good at near range obstacle detection; 2) High depth resolution and accuracy.	1) Angular resolution both vertically and horizontally; 2) Sensitive to environment and USV motion.
Sonar	1) No visual restrictions; 2) High depth resolution and accuracy.	1) Limited detecting range in each scan; 2) Impressionable to the noise from near-surface
Visual sensor	1) High lateral and temporal resolution; 2) Simplicity and low weight in practical application.	1) Low depth resolution and accuracy; 2) Challenge to real-time implementation; 3) Susceptible to light and weather condition.
Infrared sensor	1) Applicable for dark conditions; 2) Low power consumption.	1) Indoor or evening use only; 2) Impressionable to interference and distance.

Safety Advisory Council seems inclined to recognise their status as ‘vessels’ and, therefore, their obligations in an encounter.

3.2. *Global path planner.* A global path planner must continuously adapt the existing path to new long-range obstacles. Larson et al. (2007a, 2007b) discretise the environment in a bi-dimensional grid. Stationary obstacles are provided by a chart server, while moving ones by the radar. The A* algorithm (Hart et al., 1968) was chosen as search technique and a proximity cost added to prevent the USV going close to obstacles. To avoid the moving ones, safe velocity ranges are determined using the Velocity Obstacles (VO) method (Figure 6): a velocity space v - θ grid (where v identifies the vehicle’s speed and θ its heading angle) is built where the obstacle’s size is increased and the robot treated as a point. To avoid any collision, the robot’s velocity must lie outside the VO. If a collision cannot be avoided, a Projected Obstacle Area (POA) (Figure 7), the future occupied area, is created for each obstacle and a new safe route is found. The authors also implemented the COLREGS, in particular rules 13, 14 and 15.

Casalino et al. (2009) suggest an approach based on the Visibility Graph (VG) concept (Figure 9): for each couple of inter-visible points, a straight line connecting them and not passing into an obstacle was drawn. After transforming obstacles into polygons, the VG is built and Dijkstra’s Algorithm (Dijkstra, 1959) was applied to find a safe path.

A different solution has been proposed by Xie et al. (2014) who modified the Artificial Potential Field (APF) (Khatib, 1985) method (Figure 10) in a way that, in the presence of an obstacle, attractions decrease as a linear function and repulsions as a higher-order function. Chen et al. (2013) propose to adapt the scanning angle of a sonar to the distance from the obstacle using fuzzy logic to make the strategy timely and effective.

Table 2. Selected Rules from the International regulations for avoiding collisions at sea (COLREGS).

Number	Situation	Action
13	Overtaking	Any vessel overtaking any other shall keep out of the way of the vessel being overtaken.
14	Head on	When two power-driven vessels are meeting on a reciprocal course to involve risk of collision, each shall alter her course to starboard so that each shall pass on the port side of the other.
15	Crossing	When two power-driven vessels are crossing to invoke risk of collision, the vessel which has the other on her own starboard side shall keep out of the way and shall, if the circumstances of the case admit, avoid crossing ahead of the other vessel.
16	Give way	Every vessel which is directed to keep clear of another vessel shall, so far as possible, take early and substantial action to keep well clear.

Where one of two vessels is to keep out of the way, the other shall keep her course and speed. The latter vessel may, however, act to avoid collision by her manoeuvre alone, as soon as it becomes apparent to her that the vessel required to keep out of the way is not taking appropriate action in compliance with these rules.

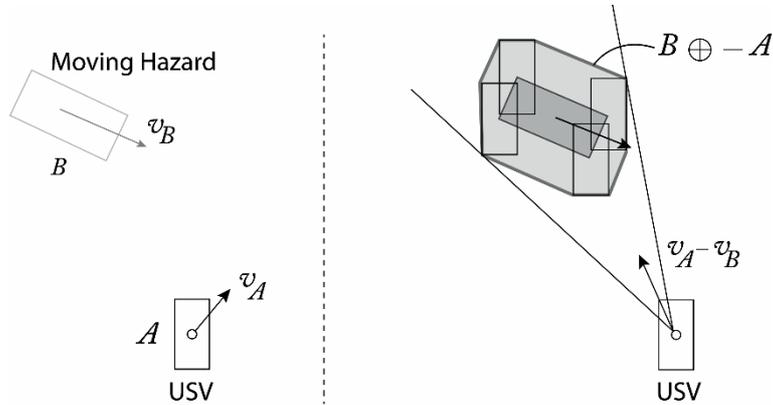


Figure 6. To avoid the collision, the relative velocity $\mathbf{v}_A - \mathbf{v}_B$ must be outside the cone defined by the USV centre and the expanded obstacle (Larson et al., 2007a, 2007b).

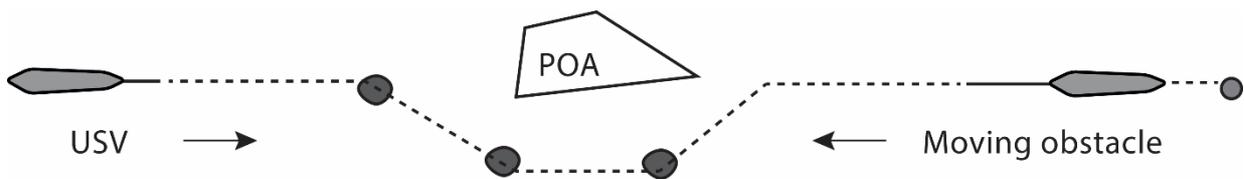


Figure 7. The projected obstacle area of a moving obstacle (Larson et al., 2007a, 2007b).

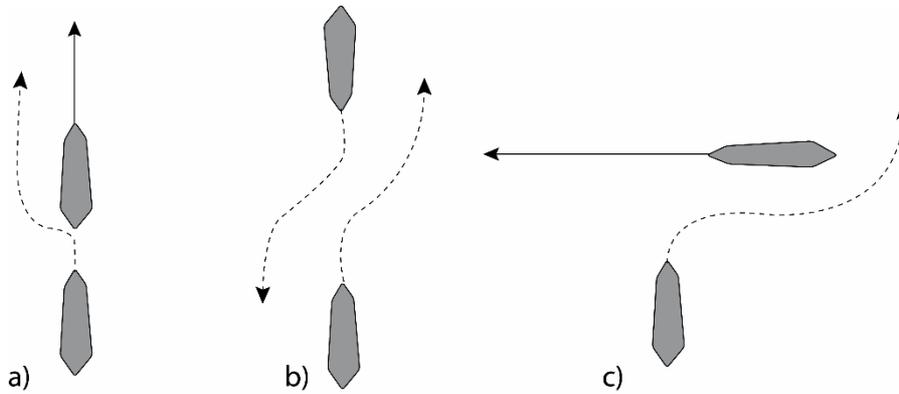


Figure 8. Manoeuvres defined for addressing overtaking (a), meeting (b), and crossing (c) obstacle (Larson et al., 2007a, 2007b).

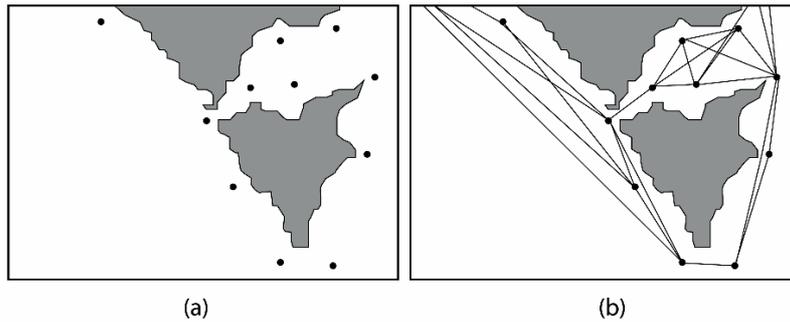


Figure 9. A visibility graph (Casalino et al., 2009).

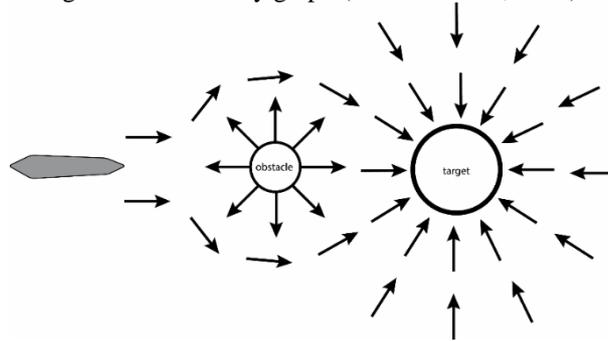


Figure 10. The APF model

Four scan range levels are defined: if no obstacle is detected in a cycle, the scan range will jump to the upper level, otherwise, it will move down to the nearest level bigger than the distance existing between the sonar mounted on the vessel and the identified obstacle. At this point, the type of the obstacle is determined and the slope is calculated. The only limitation is given by the low precision of the Inertial Measurement Unit (IMU) used, which compromised the reaction accuracy.

Chang et al. (2003) extended the maze routing algorithm developed by Lee (1961) to general λ -geometry for $\lambda \geq 4$. A route is defined as a data structure composed of the vessel path, the source cell, the destination cell and the speed. The vessel path is a list of all the cells traversed, characterised by their coordinates, the arrival time, and a link to the next one. Unfortunately, the shortest path on a raster grid is not always the optimal one since it could contain too many course alterations. Moreover, a change in speed often corresponds to a longer passage time. This problem has been addressed by Szlapczynski (2006), which found the shortest path with bend penalisation. The data structure of the previous work is modified to account for the cost of all course alterations.

3.3. *Local path planner.* One example of a local path planner is given by Kuwata et al. (2014), where the COLREGS and VO are considered. Initially, the CPA for every obstacle was calculated and the most suitable rule is applied. A cost for every velocity v_j and heading angle θ_j admissible was generated and the (v_j, θ_j) pair with the minimum cost sent to the controller. Similarly, Leng et al. (2013) integrated VO with mixed linear integer programming after linearizing the USV's properties, sensors data and uncertainty about the environment. The collision risk is checked by calculating the CPA and its distance from the vessel.

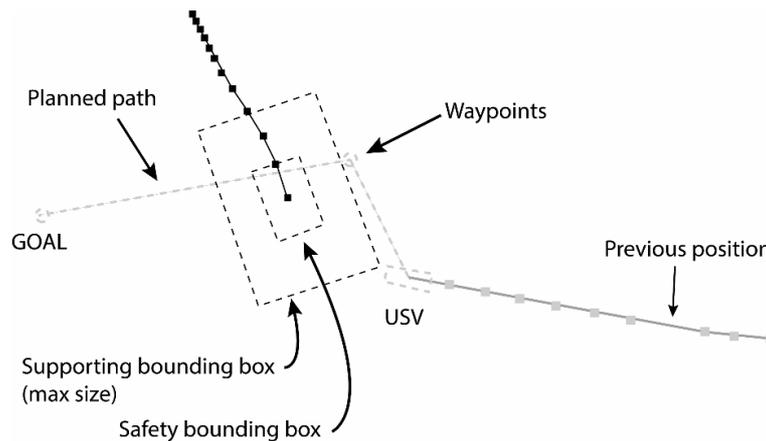


Figure 11. The bounding boxes of a moving obstacle (Simetti et al., 2014).

In Larson et al. (2007a, 2007b) the Morphin algorithm (Simmons and Henriksen, 1996) is applied drawing multiple arcs in front of the vehicle over the local occupancy map. Thus, all the safe paths are considered, covering with at least one arc, every free cell of the map. A weight is assigned to each arc depending on its vicinity to obstacles. The votes are then scaled from 0 to 1 and combined with those coming from other navigation behaviours.

Casalino et al. (2009) describe a method based on the Bounding Box (BB) concept, defined as the rectangular area the vessel should avoid. The algorithm integrates the BB vertices with the vehicle position S and the goal G into a graph. The solution is any path from S to G , identified using A^* . Since kinematic constraints are neglected, the work suffers from sub-optimality. In successive work (Simetti et al., 2014), a safety BB is added and all the computations are now performed against it; after entering the safety box, the USV must leave it without intercepting the main diagonals and ensuring in this way the collision BB is not crossed. To address uncertainty, a supporting BB is adopted (Figure 11), whose dimensions may vary from coinciding with the safety one's to be identical to the maximum bounding box's. Thus, the path's robustness to changes in speed and heading of the obstacle is increased.

An approach based on lane-constrained trajectory is proposed in Tan et al. (2010). Here, while avoiding obstacles, the vessel must meet several objectives such as maintaining a minimum distance from each obstacle, respecting the COLREGs, keeping as close as possible to the intended path and complete the trajectory in the shortest time possible. Initially, the platform's motion is forward simulated for a fixed number of time steps using simple models of a manoeuvre tracker and the vessel. Then, objectives are ordered and an elimination multi-stage process takes place. Regrettably, the algorithm may sometimes be unable to find a solution since only a sample of possible manoeuvres is generated.

Blaich et al. (2015) modified the A^* algorithm to allow velocity variations and different turning circles. The cost function is adapted adding a penalty representing the amount of path skipped during the evasion manoeuvre. In this way, the USV is led back to the original path after the deviation. Velocity and time are added to the search space to guarantee the feasibility of the trajectory.

Tang et al. (2012) developed a new method called "Obstacle Avoidance Algorithm Based on a Heading Window" (OAABHW). To obtain the best navigation angle θ_{out} , the heading yaw, is considered as an optimisation objective, while the heading window and the set of unfeasible heading angles as constraints. From θ_{out} and the current heading angle θ_{USV} , it is possible to obtain the avoidance rotational velocity ω_{out} . Zhang et al. (2014) illustrated a new adaptive collision avoidance algorithm based on **Sarsa (State-Action-Reward-State-Action)** on-policy reinforcement learning (Sutton and Barto, 1998) to face changing trajectory due to external conditions. It is composed of two modules: the Local Obstacle Avoidance Module

(LOAM) and an Adaptive Learning Module (ALM). LOAM focuses on the avoidance manoeuvre, ignoring external disturbance factors, dealt with instead by ALM while searching for a course compensation angle. LOAM calculates the guidance angle with OAABHW, which is the forwarded as input to the ALM together with sea winds and currents.

3.4. *Intelligent path planner.* Differing from the deterministic methods proposed before, some research groups developed intelligent algorithms based on Fuzzy Logic (FL), Evolutionary Algorithms (EAs) and Artificial Neural Networks (ANNs). These optimisation approaches offer multiple advantages such as lower computation effort and the ability to learn the near-optimal solution. On the other hand, they can fail by getting trapped in local minima or fail to find a solution at all. To overcome these problems, they are often combined in 'hybrid systems' (Campbell et al., 2012).

Hwang (2002) used a knowledge base and an inference fuzzy engine. Adopting a circular ship domain, the author solves a multi-ship collision one by one, suggesting the manoeuvres as a visual aid or sends them to an autopilot. Liu et al. (2006) coupled FL with an ANN in a three module algorithm. The first module classifies the collision risk given CPA, course, and distance of the incoming vessel and provides an avoidance manoeuvre. The second calculates the membership function of the speed ratio. The last one generates magnitude and action time. Kao et al. (2007) addressed port navigation safety using FL to calculate a Guarding Ring (GR) and a Danger Index (DI). The radius of the GR is established starting from the length of the ship, its speed, and the sea state. When two GRs start to overlap, the collision alert system is activated. Perera et al. (2012; 2015) and Lee and Kim (2004) provide an example where FL has been combined with a Bayesian network and an APF, respectively.

EAs are heuristic approaches developed to find a close-to-optimal solution in large search spaces. They follow the Darwinian principle that only the best elements of every generation survive a process of variation and selection (Glick and Kohn, 1996). Smierzchalski (1999) and Smierzchalski and Michalewicz (2000) use genetic mutation operations to modify a ship's speed. COLREGS and the time variable are incorporated during the generation of a new population of trajectories and the solution is searched based on cost function combining space, time and smoothness. In this model, based on Ito et al. (1999), each chromosome has a different number of genes representing the coordinates of the turning points, the ship's speed, and the interconnections among genes of the same chromosome. A more realistic approach is explained by Zeng (2003), where a single gene is characterised by some noise parameters, such as wind, wave and sea current. The length of each chromosome depends on the navigation (Figure 12). The same approach is also used by Tsou (2010). The selection of the best route is made using the roulette selection method, after ordering all of them per a linear fitness function. In Szlapczynski (2013; 2015), evolutionary algorithms were used to find a set of safe trajectories for all ships involved in a collision. Six types of violations were identified and corresponding penalties introduced. Ship tracks, modified in course and speed, are therefore evaluated with a fitness function inversely proportional to penalties.

In contrast, in Lazarowska (2015) the path planning problem is solved as an Ant Colony Optimisation (ACO) problem. ACO is inspired by the indirect communication mechanism observed among ants.

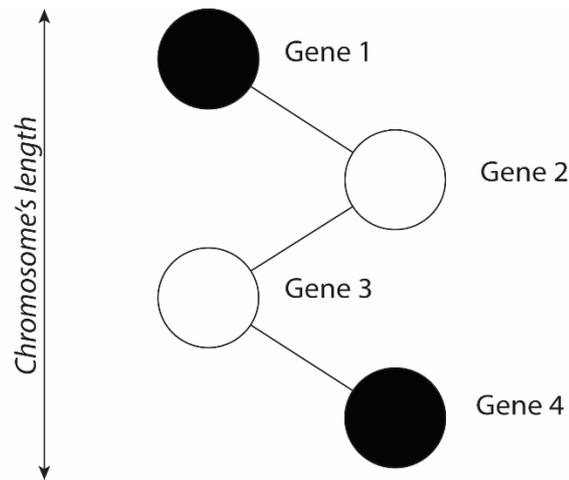


Figure 12. The route as a chromosome.

A graph considering all the constraints and the ship position is built. **At this point, every ant constructs its path from the starting vertex until the ending one:** at each step, the ant chooses its next position based on the value of the pheromone trail. After all the ants complete the task, the pheromone trail is updated penalising those vertices with a constant value and increasing the others. The algorithm terminates when a maximum number of iterations is reached or after a maximum computational time. The path selected is shown to be the shortest.

4. CONCLUDING REMARKS. The review presented in this paper began comparing different sensors employed for acquiring information about the environment. Classic sensors used in the marine environment such as sonar and radar, as well as those mounted on a ground vehicle, i.e. cameras and LIDARs, have been evaluated to establish how good they are in identifying static and moving obstacles. Studies involving them are reported and their respective advantages and drawbacks have been shown. It is concluded that the radar, despite being the sensor most used for preventing a collision at sea in the past, suffers from a lack of accuracy while trying to identify objects at close distance. Vision-based (mono and stereo) approaches can achieve better results than radar despite maritime phenomena like lightning changes and waves still posing a big challenge in the field. Infra-red cameras can offer a partial solution to the previous problems but only a few works have employed them thus far. Many efforts have been made to adopt the use of LIDAR from unmanned ground vehicles but, despite calibration problems now being solved, the same issues for visual sensors still hold. Therefore, it is possible to conclude that the obstacle detection problem remains open and a definitive robust solution is still missing.

Multiple path planners have been reported, classifying them as global or local ones. A stand-alone section has been dedicated to the so-called 'intelligent' methods, based on soft computing techniques. ANNs have been applied to this problem due to their nonlinear mapping, learning ability and parallel processing. Fuzzy logic, instead, can simulate the human thinking represented by knowledge-based conditional rules. All the methods described must face the difficulty of addressing complex encounter scenarios that would require human-like experience to choose the best action. Considering that most of the time they work as a black-box (e.g., ANN), the convergence to the optimal solution is not guaranteed. In addition, these techniques are computationally time-consuming. Therefore, they should be discarded while considering real-time navigation but they can be adopted in different tasks where they have already proved to be efficient (e.g., convolutional neural networks for identifying possible obstacles in front of the vehicle). A major problem with most of solutions reported is that they have been tested only in simulation while proving their validity in the real world still has to be done. Moreover,

their reliability is limited by the results that can be incomplete due to a small subset of scenarios tested, or small for a proper probabilistic analysis (Niu et al., 2016). It is therefore plausible to affirm that there is no guarantee in avoiding obstacles in all conditions with the methods above. In addition, three serious shortfalls have been identified:

- while defining the path, most of the methods reported do not consider the vessel's dynamics;
- others ignore the COLREGS despite this being a mandatory requirement while underway in open waters;
- almost all of them do not take in account weather and sea conditions.

Taking these points in order, most of the path planners previously described produce unexecutable paths. Treating the USV as a point and ignoring its dynamics (e.g., the minimum turning radius of the vehicle), generate way-point paths in which two points are generally connected by a straight line. This results in a trajectory characterised by high turning rates with the possibility of damage the vessel's actuators. To avoid this problem, the solution is to incorporate the vessel's dynamics (in terms of velocity, drag and heading angle) within a cost function in the formulation of the path to making it suitable for the USV's turning abilities.

The second main point to address is the implementation of the COLREGS. This is an important requirement for all types of USVs since they must show an intelligent behaviour for understanding and executing the standard rules for marine navigation. There is no a universally agreed solution to incorporate them and most of the existing works either disregard them or adopt different safety domains to emulate them. Only in few cases, they are included in the solution space but the optimality is not guaranteed. Therefore, to simplify the decision-making process in an encounter, it has been decided that an autonomous vessel always has to give away if a doubt arises while meeting a manned vehicle.

Finally, future work should address environmental disturbances and uncertainties. USV path planning is dependent on the combined effect of wind, waves and currents, where wind and wave effects can be neglected for larger vessels (Lee et al., 2015). High speed currents can deviate the vehicle from the planned path instead. The incorporation of such environmental effects within the planning process, while improving the overall understanding of the surrounding world increases the computational complexity for the optimal path planners. In the same way, varying environmental conditions (e.g. fog, lighting, rain, wave occlusions, sophisticated background) as variations in the view angle and range, still represent an issue for real-time vision-based perception.

REFERENCES

- Achanta, R., Hemami, S., Estrada, F. and Susstrunk, S. (2009). Frequency-tuned Salient Region Detection. *IEEE Conference on Computer Vision and Pattern Recognition*. CVPR 2009, 1597–1604.
- Allen, C. H. (2012). The Seabots are Coming Here: Should they be Treated as ‘Vessels’? *The Journal of Navigation*, **65**, 749–752.
- Almeida, C., Franco, T., Ferreira, H., Martins, A., Santos, R., Almeida, J. M. and Silva, E. (2009). Radar Based Collision Detection Developments on USV ROAZ II. *Oceans 09*, Bremen, Germany, 1–6.
- Alves, J., Oliveira, P., Oliveira, R., Pascoal, A., Rufino, M., Sebastiao, L. and Silvestre, C. (2006). Vehicle and Mission Control of the DELFIM Autonomous Surface Craft. *14th Mediterranean Conference on Control and Automation*, 1–6.
- Azzabi, T., Amor, S. B., and Nejim, S. (2014). Obstacle Detection for Unmanned Surface Vehicle. *International Conference on Electrical Sciences and Technologies in Maghreb 2013*, **5**, 1–7.

- Bandyopadhyay, T., Sarcione, L. and Hover, F. S. (2010). A simple reactive obstacle avoidance algorithm and its application in Singapore Harbor. In *Field and Service Robotics* (455-465). Springer Berlin Heidelberg.
- Barnich, O. and Van Droogenbroeck, M. (2011). ViBe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing*, **20**, 1709–1724.
- Belcher, P. (2002). A sociological interpretation of the COLREGS. *The Journal of Navigation*, **55**, 213–224.
- Bertram, V. (2008). Unmanned Surface Vehicles: A Survey. *Skibsteknisk Selskab*, Copenhagen, Denmark, 1–14.
- Bibuli, M., Bruzzone, G., Caccia, M. and Lapierre, L. (2009). Path-following algorithms and experiments for an unmanned surface vehicle. *Journal of Field Robotics* **26**, 669–688.
- Blaich, M., Koehler, S., Schuster, M., Reuter, J. and Tietz, T. (2015). Mission Integrated Collision Avoidance for USVs using Laser Ranger. *Oceans 2015*, Genova, Italy, 1–5.
- Borghgraef, A., Barnich, O., Lapierre, F., Van Droogenbroeck, M., Philips, W. and Acheroy, M. (2010). An Evaluation of Pixel-Based Methods for the Detection of Floating Objects on the Sea Surface. *EURASIP Journal on Advances in Signal Processing*, 1–11 2010(1), 978451
- Bouguet, J.-y. (2000). *Pyramidal Implementation of the Lucas Kanade Feature Tracker*. Intel Corporation, Microprocessor Research Labs.
- Campbell, S., Naeem, W. and Irwin, G. W. (2012). A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres. *Annual Reviews in Control*, **36**(2), 267-283.
- Casalino, G., Turetta, A. and Simetti, E. (2009). A Three-layered Architecture for Real Time Path Planning and Obstacle Avoidance for Surveillance USVs Operating in Harbour Fields. *Oceans 2009*, Bremen, Germany, 1–8.
- Chang, K.-Y., Jan, G. E., and Parberry, I. (2003). A Method for Searching Optimal Routes with Collision Avoidance on Raster Charts. *The Journal of Navigation*, **56**, 371–384.
- Chen, J., Pan, W., Guo, T., Huang, C. and Wu, H. (2013). An Obstacle Avoidance Algorithm Designed for USV Based on Single Beam Sonar and Fuzzy Control. *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2446–2451.
- Commandant, U. C. G. (1999). International regulations for prevention of collisions at sea, 1972 (72 COLREGS). *US Department of Transportation, US Coast Guard, COMMANDANT INSTRUCTION M*, 16672.
- Dijkstra, E. W. (1959). A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, **1**, 269–271.
- Ettinger, S., Nechyba, M., Ifju P. and Waszak, M. (2003). Towards Flight Autonomy: Vision-based Horizon Detection for Micro Air Vehicles. *Proceedings of the Florida Conference in Recent Advances in Robotics*, **7**, 617–640.
- Fahimi, F. (2008). *Autonomous robots: modeling, path planning, and control*, **107**, Springer Science & Business Media.
- Fischler, M. A. and Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, **24**, 381–395.
- Gal, O. and Zeitouni, E. (2013). Tracking objects using PHD filter for USV autonomous capabilities. In *Robotic sailing 2012* (3-12). Springer Berlin Heidelberg.
- Glick, T. F., and Kohn, D. (1996). *On evolution: the development of the theory of natural selection*, Hackett Publishing.
- Halterman, R. and Bruch, M. (2010). Velodyne HDL-64E lidar for unmanned surface vehicle obstacle detection. *Proc. SPIE 7692, Unmanned Systems Technology XII*, 76920D.
- Harris, C. and M. Stephens. (1988). A Combined Corner and Edge Detector. *Proceedings of the Alvey Vision Conference*, 147–151.
- Hart, P., Nilsson N. and Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, **4**, 100–107.
- Hasegawa, K. (2009). Advanced marine traffic automation and management system for congested waterways and coastal areas, *Proceedings of 8th ship control system symposium*, The Hague, Netherlands, 234–258.

- Hasegawa, K., and Kouzuki, A. (1987). Automatic collision avoidance system for ships using fuzzy control. *Journal of the Kansai Society of Naval Architects*, **205**.
- Hough, P. (1962). *Method and Means for Recognizing Complex Patterns*. U.S. Patent 3.069.654, December 18.
- Hwang, C.-N. (2002). The Integrated Design of Fuzzy Collision-Avoidance and $H-\infty$ Autopilots on Ships. *The Journal of Navigation*, **55**, 117–136.
- Ito, M., Zhnng, F. and Yoshida, N. (1999). Collision avoidance control of ship with genetic algorithm. *Proceedings of the 1999 IEEE International Conference on Control Applications*, **2**, 1791–1796.
- Jodoin, P. M., Konrad, J. and Saligrama, V. (2008). Modeling background activity for behavior subtraction. ICDSC 2008. *Second ACM/IEEE International Conference on Distributed Smart Cameras*, 1–10.
- Kao, S.-L., Lee K.-T., Chang K.-Y. and Ko, M.-D. (2007). A Fuzzy Logic Method for Collision Avoidance in Vessel Traffic Service. *The Journal of Navigation*, **60**, 17–31.
- Kemp, J. (2002). Collision Regulations - Discussion. *The Journal of Navigation*, **55**, 145– 155.
- Kemp, J., Bechley, M., Cockcroft, N., Jurdzinski, M. and Thirslund, S. (2012). Collision Avoidance at Sea in the Mid-20th Century. *The Journal of Navigation*, **65**, 191–205.
- Khatib, O. (1985). Real-time Obstacle Avoidance for Manipulators and Mobile Robots. *IEEE International Conference on Robotics and Automation*, **2**, 500–505.
- Kuwata, Y., Wolf, M. T., Zarzhitsky, D. and Huntsberger, T. L. (2014). Safe Maritime Autonomous Navigation with COLREGS, Using Velocity Obstacles. *IEEE Journal of Oceanic Engineering*, **39**, 110–119.
- Larson, J., Bruch, M., Ebken, J., Warfare, N., Diego, S. and Diego, S. (2007a). Autonomous Navigation and Obstacle avoidance for unmanned surface vehicles. *Society of Photo- Optical Instrumentation Engineers (SPIE) Conference Series*, 17–20.
- Larson, J., Bruch, M., Halterman, R., Rogers, J. and Webster, R. (2007b). Advances in Autonomous Obstacle Avoidance for Unmanned Surface Vehicles. *Techniques*, 1–15.
- Lazarowska, A. (2015). Ship's Trajectory Planning for Collision Avoidance at Sea Based on Ant Colony Optimisation. *The Journal of Navigation*, **68**, 291–307.
- Lebbad, A., and Nataraj, S. (2015). A Bayesian algorithm for vision based navigation of autonomous surface vehicles. *IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, Siem Reap, Cambodia, 59–64.
- Lee, C. Y. (1961). An Algorithm for Path Connections and Its Applications. *IRE Transactions on Electronic Computers*, **3**, 346–365.
- Lee, S.-M., Kwon, K.-Y. and Joh, J. (2004). A fuzzy logic for autonomous navigation of marine vehicles satisfying COLREG guidelines. *International Journal of Control Automation and Systems*, **2**, 171–181.
- Lee, T., Kim, H., Chung, H., Bang, Y. and H. Myung. (2015) Energy efficient path planning for a marine surface vehicle considering heading angle, *Ocean Engineering*, **107**(1), 118-131.
- Lee, Y.-I. and Kim, Y.-G. (2004). Intelligent Data Engineering and Automated Learning – IDEAL, *5th International Conference*, Exeter, UK.
- Leng, J., Liu, J. and Xu, H. (2013). Online Path Planning based on MILP for Unmanned Surface Vehicles. *Oceans 2013*, San Diego, USA, 1–7.
- Liu, Y.-H., Du, X.-M. and Yang, S.-H. (2006). *Advances in Machine Learning and Cybernetics: 4th International Conference, ICMLC 2005, Guangzhou, China, August 18-21, 2005, Revised Selected Papers. Chap. The Design of a Fuzzy-Neural Network for Ship Collision Avoidance*, edited by D. S. Yeung, Z.-Q. Liu, X.-Z. Wang, and H. Yan, 804–812. Berlin, Heidelberg: Springer Berlin Heidelberg.
- Liu, Z., Zhang, Y., Yu, X. and Yuan, C. (2016). Unmanned surface vehicles: An overview of developments and challenges. *Annual Reviews in Control*, **41**, 71-93.
- Maini, R., and Aggarwal, H. (2009). Study and Comparison of Various Image Edge Detection Techniques. *International Journal of Image Processing*, **147002**, 1–12.
- Naeem, W., and Sutton, R. (2009). An Intelligent Integrated Navigation and Control Solution for an Unmanned Surface Craft. *IET Irish Signals and Systems Conference*, Dublin, Ireland, 1–6.

- Niu, H., Lu, Y., Savvaris, A. and Tsourdos, A. (2016). Efficient Path Planning Algorithms for Unmanned Surface Vehicle. *IFAC-PapersOnLine*, **49**(23), 121-126.
- Park, K. T. and Jeong, J. (2012). Object detection in infrared image with sea clutter. *13th International Symposium on Advanced Intelligent Systems (ISIS) and 2012 Joint 6th International Conference on Soft Computing and Intelligent System (SCIS)*, 2143–2146.
- Perera, L. P., Carvalho, J. P. and Guedes Soares, C. (2012). Intelligent ocean navigation and Fuzzy-Bayesian decision/action formulation. *IEEE Journal of Oceanic Engineering*, **37**, 204–219.
- Perera, L. P., Ferrari, V., Santos, F. P., Hinojosa, M. A. and Guedes Soares, C. (2015). Experimental Evaluations on Ship Autonomous Navigation and Collision Avoidance by Intelligent Guidance. *IEEE Journal of Oceanic Engineering*, **40**, 374–387.
- Schuster, M., Blaich, M. and Reuter, J. (2014). Collision Avoidance for Vessels using a Low-Cost Radar Sensor. *19th The International Federation of Automatic Control*, Cape Town, South Africa, **2009**, 9673–9678.
- Simetti, E., Torelli, S., Casalino, G., and Turetta, A. (2014). Experimental Results on Obstacle Avoidance for High Speed Unmanned Surface Vehicles. *Oceans 2014 - St. John's, Canada*, 1–6.
- Simmons, R. and Henriksen, L. (1996). Obstacle Avoidance and Safeguarding for a Lunar Rover. *AIAA Forum on Advanced Developments in Space Robotics*.
- Smierzchalski, R. (1999). Evolutionary trajectory planning of ships in navigation traffic areas. *Journal of Marine Science and Technology*, **4**, 1–6.
- Smierzchalski, R., and Michalewicz, Z. (2000). Modelling of ship trajectory in collision situations by an evolutionary algorithm. *IEEE Transactions on Evolutionary Computation*. Citeseer.
- Statheros, T., Howells, G., and McDonald-Maier, K. (2008). Autonomous Ship Collision Avoidance Navigation Concepts, Technologies and Techniques. *The Journal of Navigation*, **61**, 129–142.
- Sutton, R. S. and Barto, A. G. (1998). *Introduction to Reinforcement Learning*. 1st. Cambridge, MA, USA: MIT Press.
- Szlupczynski, R. (2006). A New Method of Ship Routing on Raster Grids, with Turn Penalties and Collision Avoidance. *The Journal of Navigation*, **59**, 27–42.
- Szlupczynski, R. (2013). Evolutionary Sets of Safe Ship Trajectories Within Traffic Separation Schemes. *The Journal of Navigation*, **66**, 65–81.
- Szlupczynski, R. (2015). Evolutionary Planning of Safe Ship Tracks in Restricted Visibility. *The Journal of Navigation*, **68**, 39–51.
- Tam, C., Bucknall, R. and Greig, A. (2009). Review of Collision Avoidance and Path Planning Methods for Ships in Close Range Encounters. *The Journal of Navigation*, **62**, 455–476.
- Tan, A., Wee, W. C. and Tan, T. J. (2010). Criteria and Rule Based Obstacle Avoidance for USVs. *2010 International WaterSide Security Conference*, Carrara, Italy, 1–6.
- Tang, P., Zhang, R., Liu, D., Zou, Q. and Shi, C. (2012). Research on Near-field Obstacle Avoidance for Unmanned Surface Vehicle Based on Heading Window. *Proceedings of the 24th Chinese Control and Decision Conference*, Taiyuan, China, 1262–1267.
- Tsou, M.-C. (2010). Integration of a Geographic Information System and Evolutionary Computation for Automatic Routing in Coastal Navigation. *The Journal of Navigation*, **63**, 323–341.
- Wang, H., Wei, Z., Ow, C. S., Ho, K. T., Feng, B. and Huang, J. (2012). Improvement in Real-time Obstacle Detection System for USV. *12th International Conference on Control Automation Robotics Vision (ICARCV)*, Guangzhou, China, 1317–1322.
- Wang, H., Wei, Z., Wang, S., Ow, C. S., Ho, K. T. and Feng, B. (2011). A Vision-based Obstacle Detection System for Unmanned Surface Vehicle. *IEEE Conference on Robotics, Automation and Mechatronics*, Qingdao, China, 364–369.
- Wilson, P. A., Harris, C. J. and Hong, X. (2003). A Line of Sight Counteraction Navigation Algorithm for Ship Encounter Collision Avoidance. *The Journal of Navigation*, **56**, 111–121.
- Xie, S., Wu, P., Peng, Y., Luo, J., Qu, D., Li, Q. and Gu, J. (2014). The Obstacle Avoidance Planning of USV Based on Improved Artificial Potential Field. *IEEE International Conference on Information and Automation (ICIA)*, Hailar, China, 746–751.
- Zeng, X. (2003). Evolution of the safe path for ship navigation. *Applied Artificial Intelligence*, **17**, 87–104.

Zhang, R., Tang, P., Su, Y., Li, X., Yang, G. and Shi, C. (2014). An Adaptive Obstacle Avoidance Algorithm for Unmanned Surface Vehicle in Complicated Marine Environments. *IEEE/CAA Journal of Automatica Sinica*, **1**, 385–396.