A Simple Reactive Obstacle Avoidance Algorithm and Its Application in Singapore Harbor

Tirthankar Bandyophadyay, Lynn Sarcione, Franz S. Hover

Abstract Autonomous surface craft (ASC) are increasingly attractive as a means for performing harbor operations including monitoring and inspection. However, due to the presence of many fixed and moving structures such as pilings, moorings, and vessels, harbor environments are extremely dynamic and cluttered. In order to move autonomously in such conditions ASC's must be capable of detecting stationary and moving objects and plan their paths accordingly. We propose a simple and scalable online navigation scheme, wherein the relative motion of surrounding obstacles is estimated by the ASC, and the motion plan is modified accordingly at each time step. Since the approach is model-free and its decisions are made at a high frequency, the system is able to deal with highly dynamic scenarios. We deployed ASC's in the Selat Pauh region of Singapore Harbor to test the technique using a short-range 2-D laser sensor; detection in the rough waters we encountered was quite poor. Nonetheless, the ASC's were able to avoid both stationary as well as mobile obstacles, the motions of which were unknown *a priori*. The successful demonstration of obstacle avoidance in the field validates our fast online approach.

1 Introduction

The need for monitoring and securing harbor environments has grown in recent years, as a result of increased attention to pollution from runoff or other sources, natural processes such as sediment transport, water properties, and algal blooms, as well as security against threats. Harbors, with high density of goods, vessels, and people, are heavily utilized but fragile infrastructures. Among the world's harbors, Singapore Harbor is recognized as one of the largest in terms of total tonnage shipped [10], with several hundreds of large ships present at any given time. At the same time, the city of Singapore is intimately linked with the harbor. Any development on land directly affects the harbor. In many ways, Singapore represents the most important and difficult worldwide harbor environment for monitoring.

Autonomous systems are now at the level of maturity that they can be brought to bear on the overall needs of harbor observation. Autonomous surface craft (ASC)

Tirthankar Bandyophadyay
Singapore-MIT SMART Program, Singapore, e-mail: tirtha@smart.mit.edu

Lynn Sarcione
Massachusetts Institute of Technology, Cambridge, MA USA e-mail: sarcione@mit.edu

Franz S. Hover
Massachusetts Institute of Technology, Cambridge, MA USA e-mail: hover@mit.edu

such as robotic kayaks are particularly well-suited due to their low unit cost and high loading capacity; such ASC's can be used in extremely shallow waters, where an autonomous underwater vehicle would be impractical physically and acoustic navigation would be difficult.

Several difficulties dominate autonomous agents in harbor environments. First, harbors have numerous structures and vessels both small and large which must be detected. The smaller vessels may not use the Automatic Identification System, or AIS, to broadcast the ship's data and are prone to unexpected maneuvers. Larger vessels, while presumably easier to detect at large distances, cannot realistically take actions to avoid hitting an ASC. Numerous underwater or near-surface obstacles such as shipwrecks are common in harbors. Above-water structures and vessels can also endanger communications between the vehicles and other parts of the system. Secondly, harbors can experience strong tides and tidal currents, which are often complicated by variable bathymetry. Currents can be predicted and made available to operators, but sometimes significant deviations occur, perhaps in the form of large eddies. Autonomous systems have to be able to develop optimized paths and adaptive actions that are robust against such disturbances. In this paper, we describe a series of tests that utilized autonomous kayaks in Singapore Harbor during January 2009, with a focus on the obstacle avoidance problem.

Prior Work on Obstacle Avoidance : Local reactive obstacle avoidance techniques [1, 2, 3] have been quite popular due to their simplicity and fast computation. Some works [2, 3], utilize the natural robot-centric polar frame to choose the best direction to move. While these algorithms plan in position space, others [11, 4] map the obstacles in the velocity space and choose suitable control parameters to satisfy kinematic constraints. Velocity obstacles (VO) [6], incorporate the dynamics of obstacle motion into the velocity space. A common way to handle obstacle motion in known environments is to augment the configuration space by a time axis [5, 7]. When exact motion of the obstacle are unknown, predictive techniques [8, 9] are used to identify the motion parameters.

Our approach is similar in principle to the VO approach except that it is formulated in position space. By planning in a relative frame, we avoid modeling the kayak and the obstacle motion individually in the rough sea. A simple linear prediction based on the immediate history is used to determine the relative obstacle velocity. This keeps the computation load light as the algorithm runs at a high frequency and helps in bounding the uncertainty errors at each step.

2 Working in the Singapore Harbor Environment

Equipment : The ASC's utilized in Singapore Harbor are each equipped with a GPS receiver, a compass, and wireless communications gear in the base configuration. To support the aquisition of evironmental data a number of other sensors were added. A *Blueview* blazed array imaging sonar was used to image corals and shipwrecks. A *Velodyne* 3-D scanning laser imaged above-water structures including an oil platform. Also, an *RDI* doppler velocimeter measured the ASC's speed over ground. Each ASC has a full-thrust mission duration of about three hours.

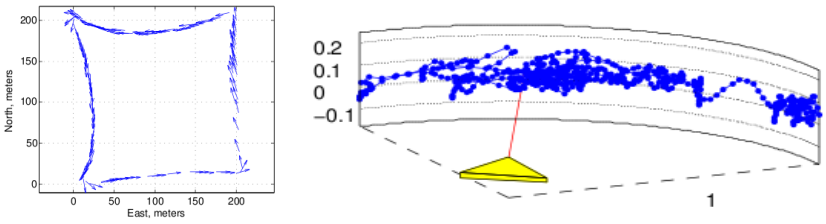
(*a*) (*b*)

Fig. 1. (a) Selat Pauh operational area. (b) A range sensor on the ASC is used for data acquisition in real-time preparation for traffic avoidance

For obstacle avoidance, a single *SICK* 2-D laser scanner was utilized. The range is 250m and the resolution is on the order of centimeters. We were able to use the full 10Hz scan rate of the sensor in our algorithm. The obstacle avoidance operation uses the onboard GPS and compass for waypoint navigation, and the 2-D laser for obstacle detection. In the remainder of this section, we describe several operational issues relevant to the Singapore Harbor environment: the effect of strong currents on navigation and the effect of waves on obstacle detection.

Effect of Ocean Currents on Navigation : It can be observed that ocean currents greatly affect navigation. The currents we encountered in Singapore Harbor reached 1.6m/s, whereas the maximum speed through water of the kayaks was about 2.4m/s on a fully charged battery. As seen in Figure 2, the closeness of these two values means that a simple waypoint-following controller can be unsatisfactory depending on the goal of the mission. Here the kayak was given four waypoints, effectively defining a square box to be traversed. The controller, developed for operation in low-current conditions, gives the kayak one of the desired waypoints to travel towards. Once the kayak reaches that waypoint, with some error tolerance, the next point of the square becomes the desired waypoint. In the test shown, significant currents to the southwest have deformed several of the vehicle paths up to fifteen percent of the leg length. If the goal of the mission was to navigate a straight line for data sampling, this current effect would be unsatisfactory, and a controller with true cross-track error regulation would be needed. The results from this run also serve to motivate path and mission planning overall, because the current influences the amount of time and energy needed to complete each leg.

Effect of Waves on Object Detection Rate : Although utilizing the *SICK* 2-D laser has many benefits, it limitations are demonstrated when ocean waves are present. The unit is fixed on the vehicle, at a height of about 35cm above the waterline. Figure 2*b* shows the projected motion of a single laser beam fixed to the vehicle, and with the vehicle heading ranging from $[0-90]^\circ$. The beam spends a fraction of time below the surface of the water, leading to no return. Other points are well above the water surface, perhaps yielding a return from the superstructure of a vessel instead of the desired hull. Figure 3*c* shows the overall performance of our cluster-based detection, as a function of range. In relatively calm waters, good hit rates can be found at about half the specified range of the sensor, but in waves



(*a*) (*b*)

Fig. 2. (a) The GPS log data for an ASC being asked to navigate the perimeter of a square under waypoint control. The distortion of the paths are due to currents. (b) Projections of a single laser beam onto a constant-radius sheet showing the effects of large roll and pitch motions of the craft, at about 1Hz.

virtually no hits are obtained outside 20m. The observations are based only on the data that was obtained during the runs, as we did not do a systematic study of the sensor clustering characteristics as a control experiment. We note that mounting the unit on a gimbal could take out some of the roll and pitch effects seen.

3 Online Navigation Approach

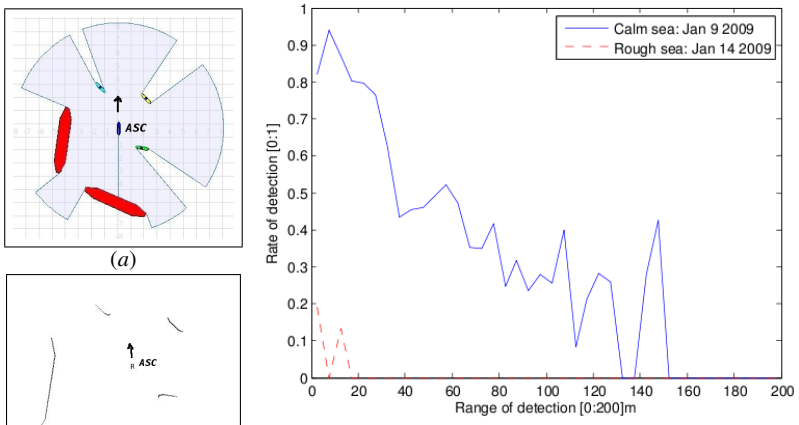
The purpose of this work is to devise a motion strategy that enables safe navigation along a desired direction for an ASC using only local 2-D range readings in the presence of unknown ocean currents and surface waves from nearby boats. At each step the ASC estimates the relative obstacle position and motion and subsequently chooses a direction that avoids collision. The approach is to follow the sense-plan-act paradigm at each step at a high frequency.

The basic model of the ASC is that of a point with controllable direction and a maximum powered velocity. A major difference between our application and terrestrial robotics is the ability of the uncertain environmental factors, such as wind and currents, to move the ASC in any arbitrary direction. We model the velocity vector of the vehicle V_{asc} by a simple superposition of the velocity arising out of environmental factors, V_{drift}, and the velocity due to the ASC's own propulsion, V_{thrust}:

$$V_{asc} = V_{drift} + V_{thrust} \tag{1}$$

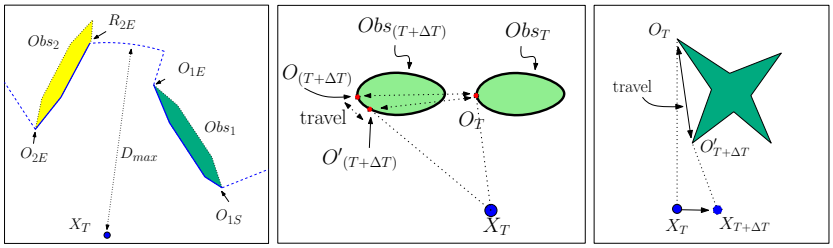
We represent the world in terms of clusters, which are determined from the scan data by a simple thresholding based range segmentation. As detected from the laser data range and angle data, any one obstacle is considered to be a single cluster, so that it has a starting point and an end point. Each such terminal point can either be an *occlusion point* that occludes the sensor's line of sight visibility, or a *range point* which is the limit of the cluster visible due to the sensor's range limit (Figure 4*a*). The range points are an artifact of the sensor limitations and do not reflect information about the obstacle.

In general, the occlusion points represent the shape characteristics of the silhouette and not of the actual object. Due to this, the motion of the occlusion points



(*a*) (*b*) (*c*)

Fig. 3. (*a*) The typical schematic environment with the ASC facing north and surrounded by boats of varying size. (*b*) The local information available to the ASC using the 2-D laser. The ASC must plan its path while avoiding dynamic obstacles. (*c*) The fraction of boat detections per second vs. the distance. The higher curve is for a relatively calm day, while the lower curve is for a choppy day. In the second case, detection beyond 20m was impossible, and even below the 20m mark, it was less than 20%.



(*a*) (*b*) (*c*)

Fig. 4. (*a*) Both the end points O_{1S} and O_{1E} of the cluster for Obs_1 are occlusion points and can be used as reliable features in a short time duration. R_{2E} is a range point and does not provide any distinctive information about the obstacle; (*b*) Occlusion point travel due to curvature. $O_{(T+\Delta T)}$ is the actual point, while $O'_{(T+\Delta T)}$ is the detected point. (*c*) Occlusion point travel due to visual discontinuity.

do not exactly represent the motion of the object, *i.e.* the rotational motion of the object can change the shape of the silhouette and give the occlusion points some velocity. However, the occlusion points can act as distinctive features of the obstacle, however, under the following conditions:

Low obstacle rotation rate : In open water, the translational velocity of many moving objects is quite high compared to their rotational speed. Due to this, the velocity of the occlusion point closely approximates the linear velocity of the

moving objects:

$$V_{occ/asc} = V_{obs/asc} + \omega_{obs} \times r_{occ/obs} \approx V_{obs/asc}$$

Here $V_{occ/asc}$ is the relative velocity of the occlusion point with respect to the ASC, $V_{obs/asc}$ is the actual velocity of the obstacle with respect to the ASC, ω_{obs} is the rotation rate of the boat, and $r_{occ/obs}$ is the radius vector from a reference frame on the obstacle to the occlusion point.

Small radius of curvature : The occlusion points may travel along the physical object surface due to its curvature. The distance error in occlusion point, however, is usually much smaller than the actual travel if the radius of curvature is small compared to the distance from the sensor. In Figure 4*b* let the obstacle move as shown relative to the sensor. If the radius of curvature is small, the actual distance discrepancy is small, and $\overline{O_{T+\Delta T}O_T} \approx \overline{O'_{T+\Delta T}O_T}$

Limited sharp edges : Depending on the inherent shape of the obstacle, the occlusion points may jump a large distance. Figure 4*c* shows such a case. This sudden jump gives an erroneous measure of the motion of the obstacle. Running the detection at high frequency and maintaining a short motion history helps the algorithm recover from such an error which is unavoidable unless the obstacle is fully modeled.

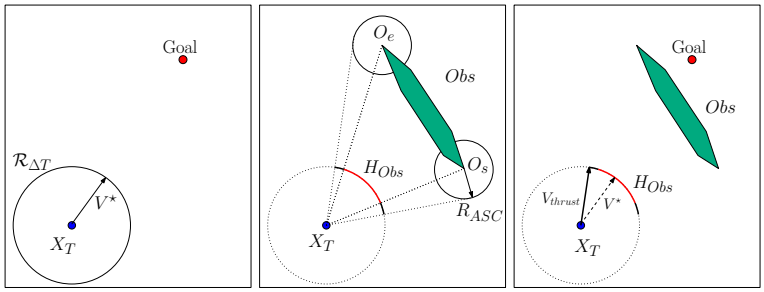
Following these assumptions and ignoring the obstacle rotation, the obstacle motion is estimated simply as the average motion of the two occlusion points: $V_{obs} = (V_{occ,s} + V_{occ,e})/2$.

3.1 Navigation Algorithm

In general, the ASC has an underlying objective such as waypoint navigation which generates a desired heading. Our algorithm modifies the heading command in light of nearby moving obstacles for collision avoidance. The higher level planning that generates the desired heading command is responsible for avoiding local minima, as the local reactive approach fails to address it.

We plan in the position space rather than velocity space due to the unreliability in velocity measurements of the ASC as well as the obstacles. Since the motion of the ASC and the environment are not modeled, we extrapolate the current velocity measurements in a simple linear model for a short duration ΔT. The position space in the planning horizon ΔT becomes the reachable set $R_{\Delta T}$ which is the set of all positions that the ASC can reach in time ΔT using this linear model. Using the simplified motion model in the previous section, we can determine directions that will cause collision with nearby obstacles. Each obstacle corresponds to one or two continuous sets of directions, termed as forbidden headings, that should be avoided. We denote the forbidden heading for a given obstacle *Obs*, by H_{Obs}.

Stationary case : In Figure 5, $R_{\Delta T}$ shows the reachable region. The optimal velocity V^* vector towards the goal position is shown in Figure 5*a*. The obstacle is represented by the occlusion points, O_e and O_s. Let the ASC have a bounding radius of R_{ASC}. To accommodate the size of the ASC, we extend the cluster by this measure.



(*a*) (*b*) (*c*)

Fig. 5. In the stationary case, the dilation of the obstacle creates forbidden zones in the heading of the ASC.

Without having to consider the whole obstacle, the Minkowski sum to represent the obstacle in the configuration space is reduced to dilating the occlusion points by R_{ASC}. The heading that the ASC must avoid in order to prevent collision is as shown in Figure 5*b* by the arc H_{Obs}. The decision of moving past O_e or O_s is made by choosing the shortest path to the goal. In general, the final choice of V_{thrust} once the H_{Obs} is established depends on the mission preferences. The *corrected* ASC heading is taken towards the corresponding endpoint of H_{Obs}. The same approach holds for multiple obstacles with the introduction of multiple forbidden regions.

Dynamic case : As discussed earlier in (Equation 1), environmental factors such as wind and current can introduce an additional velocity V_{drift} to the ASC. Also many of the obstacles in a harbor-like environment are mobile contributing to the dynamic environment seen by the craft. The velocities of these obstacles are unknown *a priori* and have to be deduced from the local range information. Let us take the case of a single obstacle moving with unknown velocity V_{obs}, while the ASC drifts with the velocity V_{drift}. As the sensing is done in the egocentric frame of the ASC, it is impossible to distinguish between these. Let V_{ext} represent the uncontrollable component of the ASC velocity towards the obstacle, *i.e.* the combined effect of the ASC drift and the obstacle motion V_{obs}, $V_{ext} = V_{drift} - V_{obs}$. Note that the ASC can only control V_{thrust}; using the onboard sensors to measure the obstacle velocity would give us $V_{ext} + V_{thrust}$. The net ASC velocity with respect to the obstacle and the reachable set $R_{\Delta T}$, is then given by:

$$\begin{aligned} V_{asc/obs} &= V_{ext} + V_{thrust} \\ X_{T+\Delta T} &= X_T + V_{ext}\Delta T + V_{thrust}\Delta T \end{aligned}$$

For a constant estimate of V_{ext} in the duration ΔT, $R_{\Delta T}$ is shown in Figure 6.

The choice of the planning horizon ΔT depends on two factors: accuracy of velocity estimation, and the distance to the nearest obstacle. If the predicted motion is considered reliable, the ASC can plan for a much longer time step with confidence. On the other hand, if the motion model is highy unpredictable or if the data is sporadic, it is advisable to plan for a shorter horizon. Given ΔT, choosing the maximum V_{thrust} is usually desirable from the point of view of the mission. However, in cases

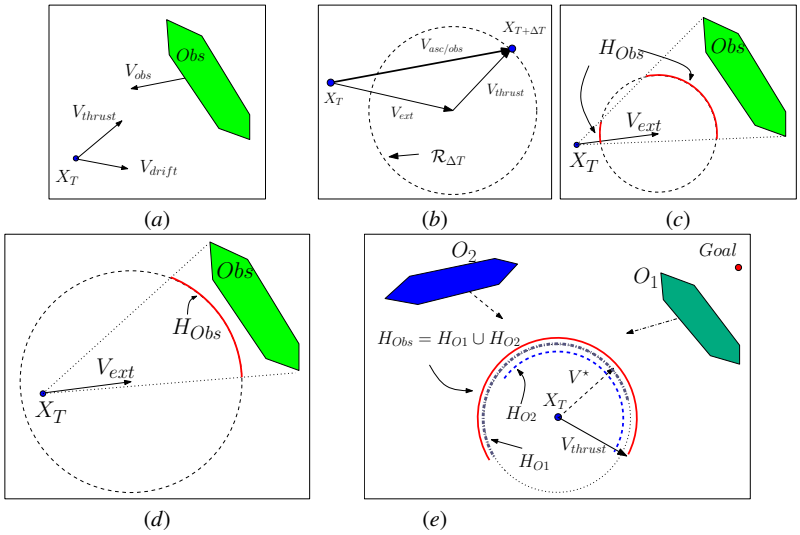


Fig. 6. (*a*&*b*) Velocity of the ASC with respect to the obstacle $V_{asc/obs}$ is the vector sum of uncontrollable velocity components affecting the obstacle and the ASC, and the controlled speed. The reachable state $R_{\Delta T}$ after ΔT is the circle shown; (*c*-*e*) The relative velocity of the ASC with respect to the obstacle is used to translate the reachable set. The forbidden heading regions are shown in red. (*c*) Scenario when $|V_{thrust}| < |V_{ext}|$; (*d*) Scenario when $|V_{thrust}| > |V_{ext}|$; (*e*) Forbidden regions for multiple moving obstacles.

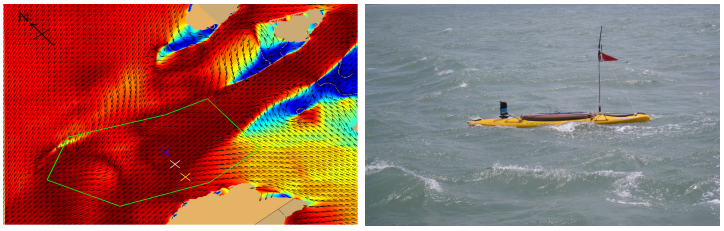
where the obstacle is too close, $R_{\Delta T}$ is further bounded by the minimum distance to the obstacle in consideration, i.e., $V_{thrust} = min(V_{thrust\_max}, dist(asc, obs)/\Delta T)$.

Estimating $\mathbf{V_{ext}}$: From our velocity definitions, we have $V_{ext} = V_{asc} - V_{obs} - V_{thrust}$ where $V_{asc} - V_{obs} = V_{asc/obs} = -\dot{r}$. Here, r is the range vector from the ASC to the obstacle, and $\dot{r}$ is the vector of the time rates of change of r's components over time. V_{thrust} is estimated from the thrust command on the vehicle or a water velocity sensor, in union with the compass heading. In the absence of these estimates, the physical thrust can be briefly turned off, forcing $V_{thrust} = 0$. Note that this use of V_{thrust} should not be confused with the circle radius in defining the forbidden angles described in previously. Here it is used to describe an actual measurement or estimate of controlled velocity.

4 Experiments at Selat-Pauh

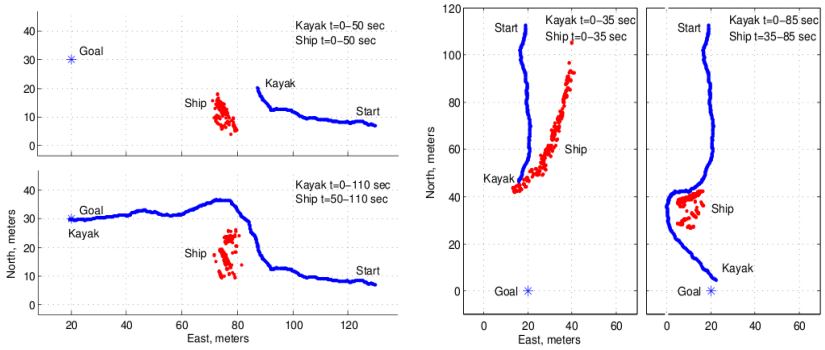
Selat Pauh was the test site utilized during the January 2009 experiments (Figure 1). Selat Pauh is located off the southern coast of Singapore where a significant amount of ship traffic is seen. In addition, the site has numerous stationary structures, such as buoys and oil rigs, and there are strong current fluctuations daily. Overall, this area is ideal for testing and observing the harbor environment.

Using the theory described, a number of obstacle avoidance tests were completed as summarized in (Figure 8). Current predictions provided for the experiment date, January 14, and a deployment photo are shown in Figure 7.



(*a*) (*b*)

Fig. 7. (*a*) Selat Pauh Jan 14 ocean current forecast. The operational area is shown by the green polygon. The current direction is shown by the arrows, and the color shows the magnitude (red being of the order 1.6m/s). *Image courtesy: N. Patrikalakis (CENSAM).* (*b*) SICK ld-1000 scanning laser being used in rough waters on Jan 14, for obstacle detection.



(*a*) (*b*)

Fig. 8. (*a*) Avoiding a stationary boat (*b*) Avoiding a moving obstacle

The weather conditions were challenging and tended towards strong winds and a choppy sea. Detection quality was on a par with that shown in the lower curve of Figure 3*c*, and this explains why the ASC only took avoidance action at a short range. However, as shown, the online approach was successful even under these dismal circumstances. The algorithm ran at 10Hz, on a Mini-ITX with 1GB RAM.

In Figure 8 the GPS position logs (the blue points) of the ASC are broken into two time sets for clarity. The red points are the laser hits from the ASC plotted in the global frame. A simple waypoint based controller was used to navigate from the Start to Goal with known GPS locations. Since the percentage of detection is so low, 20%, we averaged the laser data over a moving window of 1sec before applying the obstacle detection algorithm, improving the detection rate significantly.

In Figure 8*a* the boat was kept stationary. We see that initially the ASC follows the V^* direction to go straight towards the goal before it detects an obstacle at a distance of about 20m. The ASC motion is then modified to go around the boat and as soon the the obstacle is safely cleared, it executes a new V^* direction. In the second run Figure 8*b*, the boat actively obstructs the path of the ASC, moving from top right corner of the plot to right in front of the kayak, from the side. The ASC detects the obstacle and modifies its motion accordingly. Such close range dynamic obstacle avoidance requires fast online algorithms like the one proposed.

5 Conclusion

In large-scale autonomous vehicle testing in Singapore Harbor, we have found that strong currents and heavy traffic are serious robustness concerns. Autonomous vehicles need to have more available speed and substantially increased energy storage in order to perform meaningful missions in these waters. Path planning for known current and robust control to reject unknown currents are also critical. We have made specific progress in obstacle avoidance which, as described here, is appropriate for day-to-day use to avoid fixed and slowly-moving obstacles. The main features of our algorithm are that it is neither probabilistic nor model-based, and that it is posed in position space; as a result, it scales seamlessly to situations with many objects, and with very low computational cost. In turn, our simple approach requires good confidence in the range data and obstacle detection, and for this we have successfully employed a clustering algorithm. The avoidance behavior is demonstrated for detection rates under 20%.

In future work, we plan to test avoidance of faster moving obstacles and to use vessel motions reported by AIS. The algorithm can be extended to formations, and including range information in the forbidden regions could lead to addtional trajectories that may be useful.

Acknowledgements The research described in this project was funded in whole or in part by the Singapore National Research Foundation (NRF) through the Singapore-MIT Alliance for Research and Technology (SMART) Center for Environmental Sensing and Monitoring (CENSAM).

References

1. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. The Int. J. of Robotics Research, 5(1):90-98, 1986.
2. J. Minguez and L. Montano. Nearness diagram navigation (nd): A new real time collision avoidance approach for holonomic and no holonomic mobile robots. In Proc of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Takamatsu, Japan, November 2000.
3. L. Ulrich and J. Borenstein. VFH*: Local obstacle avoidance with look-ahead verification. In IEEE Int. Conf. on Robotics and Automation, 2505-2511, April 2000.
4. D. Fox, W. Burgard, and S. Thrun. Controlling synchro-drive robots with the dynamic window approach to collision avoidance. In Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1996.
5. D. Hsu, R. Kindel, J.C. Latombe, and S. Rock. Randomized kinodynamic motion planning with moving obstacles. Int. J. Robotics Research, 21(3):233-255, 2002.
6. P. Fiorini and Z. Shiller. Motion planning in dynamic environments using velocity obstacles. Int. J. of Robotics Research, 17(7):760-772, July 1998.
7. J. Latombe, Robot Motion Planning. Boston, MA: Kluwer Academic Publishers, 1991.
8. C.C. Chang and K.-T. Song, Dynamic motion planning based on real-time obstacle prediction, Proceedings on IEEE Int. Conf. on Robotics and Automation (1996), 2402-2407.
9. A. F. Foka and P. E. Trahanias, Predictive Autonomous Robot Navigation IROS, IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2002
10. J.R.M. Gordon, P. M. Lee, and H.C. Lucas, Jr.. A resource-based view of competitive advantage at the Port of Singapore. J. Strategic Information Systems, 14(1):69-86, 2005.
11. R. Simmons, "The Curvature-Velocity Method for Local Obstacle Avoidance," Int. Conf. on Robotics and Automation, April, 1996.