Radar-based Maritime Collision Avoidance using Dynamic Window

Bjørn-Olav H. Eriksen **Erik F. Wilthil** **Andreas L. Flåten** **Edmund F. Brekke** **Morten Breivik**

Centre for Autonomous Marine Operations and Systems
Department of Engineering Cybernetics
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
{bjorn-olav.h.eriksen, morten.breivik}@ieee.org
{erik.wilthil, andreas.flaten, edmund.brekke}@ntnu.no

Abstract—**Collision avoidance systems are a key ingredient in developing autonomous surface vehicles (ASVs). Such systems require real-time information about the environment, which can be obtained from transponder-based systems or exteroceptive sensors located on the ASV. In this paper, we present a closed-loop collision avoidance (COLAV) system using a maritime radar for detecting target ships, implemented on a 26 foot high-speed ASV. The system was validated in full-scale experiments in Trondheimsfjorden, Norway, in May 2017. The probabilistic data association filter (PDAF) is used for tracking target vessels. The output from the PDAF is processed through a least-squares retrodiction procedure in order to provide the COLAV system with sufficiently accurate course estimates. A tracking interface provides estimates of target states to the COLAV system, which is based on the dynamic window (DW) algorithm. DW is a reactive COLAV algorithm originally designed for ground vehicles, and we therefore make a number of modifications to adapt it for use with high-speed ASVs. The closed-loop experiments demonstrated successful COLAV with this system, but also disclosed several challenges arising from both the DW algorithm and the tracking system, motivating for further work.**

TABLE OF CONTENTS

1. INTRODUCTION

The cost efficiency and safety of marine operations such as seabed surveying, surveillance, passenger and goods transport have potential for improvement by moving in the direction of more automatic and autonomous operations. An enabling technology for this to happen is autonomous collision avoidance (COLAV). For autonomous surface vehicles (ASVs), COLAV is a complex task involving challenges with perception, planning and regulations.

COLAV algorithms may in general be divided into reactive and deliberate approaches. Reactive algorithms consider a limited amount of information, often just currently available sensor data, making the algorithms computationally cheap at the cost of possibly producing sub-optimal behavior. The dynamic window (DW) algorithm [13], [10] and the velocity obstacle (VO) algorithm [16] are examples of reactive COLAV algorithms. Deliberate algorithms usually use greater amounts of information and plan in a longer temporal setting, making the algorithms more optimal in a global sense at the cost of increased computational requirements. To be able to both react to sudden changes in the environment and performing meaningful long-term maneuvers, reactive and deliberate algorithms can be combined in a hybrid architecture [18].

ASVs are in general required to follow the International Regulations for Preventing Collisions at Sea (COLREGS), which act as “rules of the road” for avoiding collisions at sea [4]. Reactive COLAV is intended as a “last line of defense” in close-quarter situations, where deliberate algorithms fail in resolving the situation. In such situations, it may be necessary to violate COLREGS to avoid collision. In fact, COLREGS require a vessel to violate the rules if collision cannot be avoided without violating the rules. Hence, we do not consider COLREGS in these experiments.

For perception, a transponder-based communication system may be used. An example of such a system is the automatic identification system (AIS), which is used to transmit the position and velocity of a vessel to other vessels. Passenger ships and vessels with gross tonnage over 300 are obliged to carry AIS transponders. AIS is being used for navigation at sea, and can obviously provide useful information about other vessels’ whereabouts and intentions to a COLAV system. However, since AIS depends on satellite navigation and data input from the user, it may contain inaccurate or invalid data [14]. The information is also restricted to vessels equipped with AIS transponders. Although lower-cost transponders for smaller vessels (AIS class B) exist, many vessels such as leisure craft and kayaks are not equipped with AIS. This also extends to other objects in water such as navigational aids and debris. Transmitted information may also be faulty, either by accidental errors such as transmission failures or through intentional actions or negligence by the crew [14].

Other means of perception include exteroceptive sensors such

978-1-5386-2014-4/18/$31.00 ©2018 IEEE

as radars, lidars and cameras. The main advantage of these sensors are that they do not depend on other ships to transmit information, but instead transmit a signal and wait for a return, or passively capture information from the environment. This makes detection of kayaks and other small obstacles possible, but the measurements may be affected by clutter. The sensors may also have limited range, which means that objects can be very close before they are detected. This is usually not an issue for radars. There are several reports of use of exteroceptive sensors for maritime COLAV. The complementary properties of different sensors are thoroughly discussed in [6], with experimental validation. In [20], a maritime broadband radar intended for smaller recreational vessels is used for tracking in the joint probabilistic data association (JPDA) and interacting multiple model (IMM) framework.

Many target tracking models parametrize the target states as north and east position and velocity, while COLAV methods often use the total speed and course of the target. This may lead to fluctuations in these values, and the course angle in particular can fluctuate more than it should.

There are several solutions that can be applied to obtain better course estimates. One could argue that course and speed, instead of the linear world frame-parameterized velocity vector, should be used in the state vector. This would lead to a nonlinear process model. It is not clear that such a model would give improved performance. More refined models such as the Best-Norton model [2] require additional tuning to work well. Current research by the authors is exploring the utility of particle filtering for heading estimation [11]. Another approach would be to use multiple models within an IMM framework as in [20]. However, IMM is known to have limited effect when the so-called maneuvering index [15] is low. This can often be the case in maritime COLAV, when ships move relatively slowly compared to the measurement noise. In these cases, it may be feasible to smooth the target estimates using a form of retrodiction [5].

In this article, we report on a full-scale experiment where the DW algorithm was used in conjunction with a probabilistic data association filter (PDAF) radar-tracking filter to perform autonomous COLAV. A number of modifications to the DW algorithm in order to adapt the algorithm for use with high-speed ASVs are presented. We highlight a number of challenges arising from the results, motivating for further work.

The rest of the paper is structured as follows: Section 2 describes the platform used for the experiments. Section 3 describes the tracking system. Section 4 describes the DW algorithm, and the modifications applied to it. Section 5 describes the interface between the tracking and COLAV systems, while Section 6 show the results. Finally, concluding remarks and possibilities for further work are presented in Section 7.

2. Experimental platform

The vessel used in the experiments described in this paper is the dual-use Maritime Robotics Telemetron ASV, shown in Fig. 1. The ASV was equipped with a high grade navigation system from Kongsberg Seatex, supplemented by real-time GNSS corrections (CPOS) from the Norwegian mapping authority [22]. Conservative performance estimates for the navigation system are given in Table 1. Given the performance of the Seapath navigation system, navigation uncertainty is of minor impact to the target tracking system, and will be neglected. This is supported by simulation studies in [23] and [3] which indicated that navigation uncertainty would have to be significantly higher to have noticeable impact. The vessel tracking and control system is implemented in the robot operating system (ROS) [19], and the vessel actuators are interfaced via a proprietary interface developed by Maritime Robotics. A summary of the vessel parameters is given in Table 1.

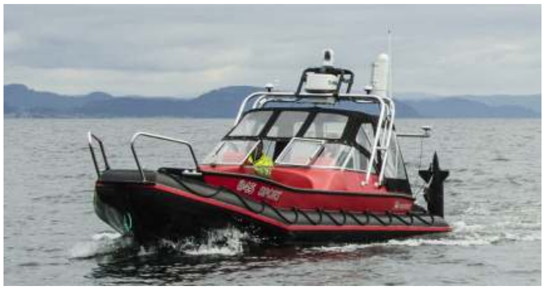

Figure 1. The Telemetron ASV, a dual-use vessel for both manned and unmanned operations. Courtesy of Maritime Robotics.

Table 1. Telemetron ASV specifications.

Component	**Description**
Vessel hull	Polarcirkel Sport 845
Length	8.45 m
Width	2.71 m
Weight	1675 kg
Propulsion system	Yamaha 225 HP outboard engine
Motor control	Electro-mechanical actuation of throttle valve
Rudder control	Hydraulic actuation of outboard engine angle with proportional-derivate (PD) feedback control on engine angle
Navigation system	Kongsberg Seatex Seapath 330+
Heading/roll/pitch accuracy	$0.1°$ RMS
Position accuracy	0.1 m RMS
Radar	Simrad Broadband 4G$^{\text{TM}}$ Radar
Processing platform	Intel® i7 3.4 GHz CPU, running Ubuntu 16.04 Linux

The Telemetron vessel operates at speeds of up to 18 m/s, which makes modeling and control of the vessel difficult since it operates in both the displacement, semi-displacement and planing regions [12]. In [8], a control-oriented model of the Telemetron ASV was developed, together with experimental validation of vessel controllers based on the vessel model. The model is defined in 2DOF using the speed over ground (SOG) and rate of turn (ROT) as states, denoted as U $[\mathrm{m/s}]$ and r $[\mathrm{rad/s}]$, respectively. The DW algorithm specifies a desired velocity which the vessel should follow. We therefore employ a velocity controller for SOG and ROT combining a proportional integral (PI) feedback controller with model-based feedforward of a desired velocity and acceleration shown to have good performance for the Telemetron ASV [8].

Table 2. Tracking module parameters used in the experiments.

Parameter	Value	Comment
T_r	2.4 s	Radar revolution time
q	$0.5\ \mathrm{m/s^2}$	Process noise standard deviation
r	6.86 m	Measurement noise standard deviation
N_s	5	Retrodiction window length

3. TARGET TRACKING: THEORY AND IMPLEMENTATION

The core algorithm of the target tracking system is the probabilistic data association filter (PDAF), which is a single target tracking method which calculates the association probabilities for each measurement in the validation region of the target of interest [1]. It is not a multitarget method, but a parallel bank of PDAFs will be able to handle targets that are sufficiently temporally and/or spatially spaced. This covers the majority of the situations encountered in commercial ship traffic, and is thus chosen for its simplicity. The PDAF requires point measurements with known measurement noise covariance. The radar listed in Table 1 provides an array of spokes consisting of resolution cells with intensity information. The radar data must be processed in a detection, projection and clustering pipeline before it can be used in the PDAF. This pipeline is described in detail in [24]. The target motion and measurement model is the nearly constant velocity (NCV) model [17] with position measurements, given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}(T_r)\boldsymbol{x}_k + \boldsymbol{v}_k, \;\; p(\boldsymbol{v}_k) = \mathcal{N}(\boldsymbol{v}_k; 0, \boldsymbol{Q}(T_r)), \quad (1)$$

$$\boldsymbol{z}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{w}_k, \quad p(\boldsymbol{w}_k) = \mathcal{N}(\boldsymbol{w}_k; 0, r^2\boldsymbol{I}), \quad (2)$$

where $\boldsymbol{x} = [p_N \quad v_N \quad p_E \quad v_E]^T$ is the state vector, containing position and velocity in north and east directions and $\boldsymbol{v}$ is the process noise. The variable $\boldsymbol{z}$ is the position measurement, while $\boldsymbol{w}$ is the measurement noise with covariance $r^2\boldsymbol{I}$ where $\boldsymbol{I}$ is the identity matrix. The matrix $\boldsymbol{H}$ extracts the position elements of the state vector, while the matrices $\boldsymbol{F}(T_r)$ and $\boldsymbol{Q}(T_r)$ are given as

$$\boldsymbol{F}(T_r) = \begin{bmatrix} 1 & T_r & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_r \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

$$\boldsymbol{Q}(T_r) = q^2 \begin{bmatrix} T_r^4/4 & T_r^3/2 & 0 & 0 \\ T_r^3/2 & T_r^2 & 0 & 0 \\ 0 & 0 & T_r^4/4 & T_r^3/2 \\ 0 & 0 & T_r^3/2 & T_r^2 \end{bmatrix}. \quad (4)$$

In addition to the radar sampling time T_r, which is given by the radar revolution time, the NCV model is parametrized by the covariance of the white noise acceleration q^2. This simple model reduces the risk of overfitting and tailoring the model to a specific target. A suitable value for the white noise acceleration standard deviation q was determinted to be $0.5\ \mathrm{m/s^2}$ through an analysis of covariance consistency, by considering AIS data from several vessels, including the Telemetron ASV [24]. This is a fairly large process noise value, which ensures good resilience against track-loss but also inevitably leads to some fluctuations in the course estimates, as shown in Fig. 6. The course angle is the angle of the velocity vector, measured clockwise from straight north, and is calculated as $\arctan_2(v_E, v_N)$ where $\arctan_2$ is the four-quadrant arctangent function.

Improved target course estimation

Several methods for improved course estimation was discussed in section 1. In this case, the target maneuvering index is found to be 0.42 from the values in Table 2, which falls below the limit of where the IMM estimator is preferred over the regular Kalman filter estimator [15]. Since the assumed target dynamics are moderately slow, the speed and course estimates will be improved using a retrodiction procedure.

Assume that the target, for the last N_s scans, have been moving in a straight line. The motion model for these steps can be written

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}(T_r)\boldsymbol{x}_k, \quad k \in [K^\dagger, K], \quad (5)$$

where K is the latest timestamp of the estimate of the target, and $K^\dagger = K - N_s + 1$ is the start of the retrodiction interval. On this form, the motion of the target for the last N_s scans are parametrized by a constant velocity and an initial position $\boldsymbol{x}_{K^\dagger}$. The estimate from the PDAF is written as

$$\hat{\boldsymbol{x}}_{k|k} = \boldsymbol{F}(t_k - t_{K^\dagger})\boldsymbol{x}_{K^\dagger} + \boldsymbol{e}_k, \;\; p(\boldsymbol{e}_k) = \mathcal{N}(\boldsymbol{e}_k; 0, \boldsymbol{P}_{k|k}) \quad (6)$$

where $\hat{\boldsymbol{x}}_{k|k}$ and $\boldsymbol{P}_{k|k}$ is the posterior estimate of the PDAF. The retrodicted estimate of $\boldsymbol{x}_{K^\dagger}$ can be calculated by a standard least-squares calculation,

$$\bar{\boldsymbol{x}}_{K^\dagger} = (\boldsymbol{F}_r^T\boldsymbol{P}_r\boldsymbol{F}_r)^{-1}\boldsymbol{F}_r^T\boldsymbol{P}_r\hat{\boldsymbol{x}}_r, \quad (7)$$

where

$$\hat{\boldsymbol{x}}_r = \begin{bmatrix} \hat{\boldsymbol{x}}_{K^\dagger|K^\dagger} \\ \hat{\boldsymbol{x}}_{K^\dagger+1|K^\dagger+1} \\ \vdots \\ \hat{\boldsymbol{x}}_{K|K} \end{bmatrix}, \boldsymbol{F}_r = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{F}(t_{K^\dagger+1} - t_{K^\dagger}) \\ \vdots \\ \boldsymbol{F}(t_K - t_{K^\dagger}) \end{bmatrix}, \quad (8)$$

$$\boldsymbol{P}_r = \mathrm{diag}(\boldsymbol{P}_{K^\dagger|K^\dagger}, \boldsymbol{P}_{K^\dagger+1|K^\dagger+1} \ldots, \boldsymbol{P}_{K|K}), \quad (9)$$

The retrodicted estimate of $\boldsymbol{x}_{K^\dagger}$ can then be used to calculate the estimate of the target at following timesteps, using (5).

We emphasize that the retrodiction is only used as a post-processing step for the output to the collision avoidance method, and not as an integral part of the tracking and prediction. Nevertheless, the estimates of the target used in the COLAV-algorithm will be affected if the target maneuvers. There are two main factors that will affect the safety of the system. The first is the time it takes to detect the maneuver in the retrodiction method, and the second is the safety regions of the ASV. The maneuver detection will depend on the retrodiction time. With the tracking system parameters shown in Table 2, this amounts to about 12 seconds. This corresponds to 60 meters with a target velocity of $5\ \mathrm{m/s}$. This means that the collision and safety regions described in Section 4 are significantly larger than the distance covered by the target during a turn. Should this not be the case, the ASV can take precautionary measures such as increasing the collision or safety regions, or decreasing its velocity.

4. Dynamic window: Theory and implementation

The dynamic window (DW) algorithm was introduced as a COLAV algorithm for indoor ground robots in 1997 [13]. The algorithm originally assumes that the vehicle is subject to constant acceleration limits, and predicts future trajectories using straight lines and circles. ASVs have nonlinear dynamics, which result in time-varying acceleration constraints. Moreover, the dynamics of an ASV is far more complex than that of an indoor ground robot, rendering the original prediction approach inaccurate. A modified DW algorithm for autonomous underwater vehicles (AUVs) moving in the horizontal plane is presented in [10], proposing a solution to these issues. This algorithm searches for feasible velocity pairs consisting of a desired surge speed u and yaw rate r, and chooses the optimal velocity pair based on an objective function. A set of feasible velocities is created by joining three search spaces, which then is sampled and used for predicting vessel trajectories with a prediction horizon T_p. The dynamic window contains velocities reachable during a time window T while respecting actuator rate saturations, and is defined as:

$$V_d = \Big\{(u,r) \in \mathbb{R} \times \mathbb{R} \Big| u \in [u^* + \dot{u}_{\min}T, u^* + \dot{u}_{\max}T] \\ \wedge r \in [r^* + \dot{r}_{\min}T, r^* + \dot{r}_{\max}T]\big\}\,, \quad (10)$$

where u^* and r^* are the current surge speed and yaw rate, and $u_{\min}, u_{\max}$ and $r_{\min}, r_{\max}$ are time-varying acceleration limits. The set of possible velocities is defined as:

$$V_s = \Big\{(u,r) \in \mathbb{R} \times \mathbb{R} \Big| g(u,r) \geq 0\Big\}\,, \quad (11)$$

where $g(u,r)$ is positive for velocities that are possible to reach with respect to actuator magnitude saturations.

Two regions are defined around the obstacles, namely the collision and safety regions. The collision region is a circle, which if entered is treated as a collision. The safety region is a new circle outside of the collision region, which is allowed (but not desirable) to enter, hence acting as a safety margin. The set of admissible velocities only include velocities which allow the vehicle to stop before entering the collision region, and is defined as:

$$V_a = \Big\{(u,r) \in \mathbb{R} \times \mathbb{R} \big| u \leq \sqrt{2\rho'(u,r)|\dot{u}_{\min}|} \\ \wedge |r| \leq \left\{ \begin{array}{ll} \sqrt{2\rho'(u,r)|\dot{r}_{\max}|} & , r < 0 \\ \sqrt{2\rho'(u,r)|\dot{r}_{\min}|} & , r \geq 0 \end{array} \right\}\,, \quad (12)$$

where $\rho'(u,r)$ represents the along-trajectory distance to the collision region at the next time instant the algorithm is run, given the velocity pair (u,r). Finally, the optimal velocity pair is selected through maximizing an objective function:

$$(u_d, r_d) = \underset{(u,r) \in V_r}{\text{argmax}}\, G(u,r; u_d', r_d'), \quad (13)$$

where $V_r = V_d \cap V_s \cap V_a$, and $G(u,r;u_d',r_d')$ is defined as:

$$G(u,r;u_d',r_d') = \alpha \cdot \text{yawrate}(u,r,r_d') + \beta \cdot \text{dist}(u,r) \\ + \gamma \cdot \text{velocity}(u,r,u_d'), \quad (14)$$

where u_d' and r_d' are inputs to the algorithm, generated by a line of sight (LOS) guidance system [12], and $\alpha, \beta, \gamma > 0$ are tuning parameters. The yawrate$(\cdot)$ and velocity$(\cdot)$ terms assign value to choosing a velocity pair close to the desired velocity (u_d', r_d'), while the dist$(\cdot)$ term motivate the algorithm to keep distance to obstacles based on the along-trajectory distance to the safety region. These are defined as:

$$\text{yawrate}(u,r,r_d') = 1 - \frac{|r_d' - r|}{\max\limits_{r \in V_r}(|r_d' - r|)}, \quad (15)$$

$$\text{velocity}(u,r,u_d') = 1 - \frac{|u_d' - u|}{\max\limits_{u \in V_r}(|u_d' - u|)}, \quad (16)$$

$$\text{dist}(u,r) = \frac{\bar{\rho}(u,r)}{\frac{1}{T_p}\int_0^{T_p} \|\boldsymbol{\chi}(u,r,t)\|_2 \,\text{dt}}, \quad (17)$$

where $\boldsymbol{\chi}(u,r,t)$ is the predicted vessel body velocity and $\bar{\rho}(u,r)$ represents the along-trajectory distance to the safety region, both given the velocity pair (u,r).

The maximization in (13) is performed discretely by uniformly sampling the dynamic window V_d, and removing the velocity pairs which are not elements of V_s and V_a. In this process, predicted trajectories for the velocity pairs are generated using a model of the vehicle closed-loop error dynamics. See [10] for more details on the modified DW algorithm.

There are several differences between the AUV application in [10] and the ASV platform described in Section 2:

1. The model of the Telemetron ASV is formulated in 2DOF using SOG and ROT, instead of 3DOF using surge, sway and yaw as in [10].
2. The control system requires a continuously differentiable velocity trajectory to employ acceleration feed-forward.
3. The Telemetron ASV is not well captured by the AUV model used in [10].

The first issue is handled by redefining the DW algorithm for SOG and ROT. This requires a way to model the sideslip of the vessel, to be able to simulate the vessel kinematics [8]. For these experiments, we assume that the sideslip is small enough to be neglected. During the identification experiments in [8], we observed that the sideslip stays below 10° when operating the ASV at moderate speeds without extreme maneuvers.

The second issue is tackled by changing the way we generate the velocity pairs. Instead of sampling the dynamic window, we define a dynamic acceleration window:

$$A_d = \Big\{(\dot{U}, \dot{r}) \in \mathbb{R} \times \mathbb{R} \Big| \dot{U} \in [\dot{U}_{\min}, \dot{U}_{\max}] \\ \wedge \dot{r} \in [\dot{r}_{\min}, \dot{r}_{\max}]\big\}\,, \quad (18)$$

which we sample to obtain a list of acceleration pairs. We then define an acceleration trajectory as a piecewise linear trajectory. For a SOG acceleration sample $\dot{U}'$, the trajectory is defined as:

$$\dot{U}(t) = \left\{ \begin{array}{ll} \dot{U}_0 + \frac{\dot{U}' - \dot{U}_0}{T_{act}} t & , 0 \leq t < T_{act} \\ \dot{U}' & , T_{act} \leq t < T_{acc} \\ \dot{U}' - \frac{\dot{U}'}{T_{act}}(t - T_{acc}) & , T_{acc} \leq t < T_{acc} + T_{act} \\ 0 & , otherwise, \end{array} \right. \quad (19)$$

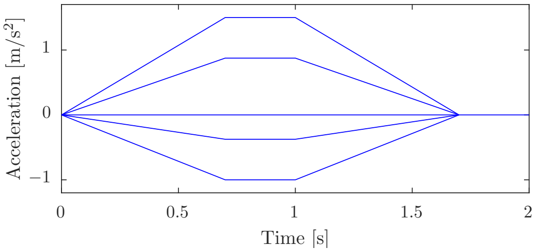

Figure 2. SOG acceleration trajectories for 5 samples in A_d.

where T_{act} is the time allowed for changing the actuator settings and T_{acc} is the time for acceleration, which should be equal to the sample time of the algorithm to allow for continuous accelerations. A collection of 5 SOG acceleration trajectories with $\dot{U}_0 = 0\,\mathrm{m/s^2}$, $T_{act} = 0.7\,\mathrm{s}$ and $T_{acc} = 1\,\mathrm{s}$ is shown in Fig. 2. Trajectories for ROT acceleration is obtained in a similar way.

The third issue is handled by simply using a kinematic model to obtain velocity and position trajectories given the acceleration trajectories. This relies on the velocity controller being able to track the desired trajectory, without too much bias introduced from external disturbances. Based on the experimental validation of the velocity controllers in [8], we believe that they will be able to quite closely track the desired velocity, and therefore justify this approach. In addition, the DW algorithm will be run at a quite high rate, which reduces the impact of external disturbances.

In addition, it has been found that using only the distance to the safety region in the distance term (17) of the objective function (14) may lead to the vessel being trapped in the safety region [21], [7], essentially disabling the COLAV aspect of the algorithm. To improve on this, an alternative distance term including the amount of a trajectory residing inside the safety region is proposed in [21]. This distance term does, however, rely on a part of the trajectory residing outside of the safety region, which may not be the case if the safety region is large. We therefore interchange the $\mathrm{dist}(\cdot)$ term with a more complex one similar to the one in [21], but also including the distance to the collision region:

$$\mathrm{dist}(U,r) = \kappa \frac{\bar{\rho}(U,r) + \rho(U,r)}{\frac{1}{T}\int_0^T U \mathrm{d}t} + (1-\kappa)\frac{\sum_{n=1}^{N} \frac{\lambda(U,r,n)}{\sqrt{n}}}{\sum_{n=1}^{N} \frac{1}{\sqrt{n}}}. \tag{20}$$

Here, the function $\lambda(U, r, n)$ is 1 if point n of the predicted trajectory resides inside the safety region, and 0 otherwise, while $\kappa \in (0, 1)$ is a tuning parameter. The variables $\bar{\rho}(U, r)$ and $\rho(U, r)$ are the along-trajectory distance to the safety and collision regions, respectively, both given the velocity pair (U, r). For more details, see [10] and [21].

In the experiments we ran the DW algorithm every second using 225 m and 350 m as the collision and safety region sizes, respectively. Further tuning parameters are shown in Table 3.

5. Target tracking and collision avoidance interface

Successfully closing the loop between the target tracking and COLAV systems requires that these two modules communicate in some way. The target tracking module has to provide the COLAV module with estimated target tracks in some manner. This can be posed as a software design problem, with many possible solutions. One way of enabling communication would be to implement the target tracking and COLAV functionality in one integrated module, handling communication implicitly within this module by sharing state. This is flexible and has the advantage of requiring very little explicit design, but at the same time makes the implementation less modular, harder to test and more difficult to develop and maintain as a team. An explicit interface has the additional flexibility of abstracting which sensor/sensors the target estimates originate from, and what kind of COLAV functionality the estimates are used for.

Table 3. COLAV tuning parameters for the experiments.

Parameter	Value	Comment
T_p	30 s	Planning horizon length
T_{act}	0.7 s	Actuator ramp time
T_{acc}	1.0 s	Acceleration time
α	1.0	Yawrate function weight
β	4.0	Distance function weight
γ	1.0	Velocity function weight

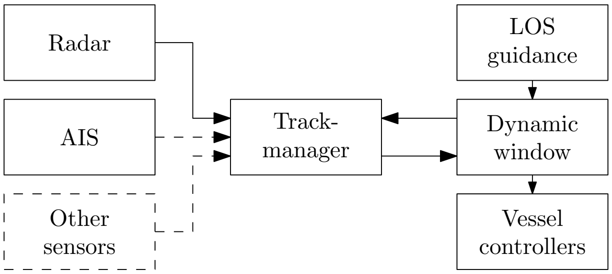

Figure 3. Diagram illustrating the target tracking and COLAV interface.

Within the ROS software framework, there are two main ways of communicating between separate software modules. The first is an asynchronous publish-subscribe mechanism, and the second is a synchronous request-response mechanism. We have chosen to use the latter model, such that the COLAV module requests a list of all known targets at specific times, i.e. a list of target trajectories that are discretized in time. This places the burden of track management, prediction and possibly interpolation in the target tracking module. All these functions are arguably naturally encapsulated within a target tracking framework. The interface is shown in Fig. 3.

6. Closed-loop collision avoidance experiments

The combined tracking and COLAV system was tested in the Trondheimsfjord on the 15th to 19th of May 2017.

Scenario description

During the experiments, we tested a number of head-on situations with a target vessel under our control. The target vessel was a 17 foot motorboat constructed in glass fibre and equipped with a radar reflector to improve the visibility on the radar, as shown in Fig. 4. The target vessel was also equipped with an emulated AIS transponder using an

Figure 4. The 17 foot long target vessel, equipped with a radar reflector to improve radar visibility.

Android phone to transmit the vessel position, course and speed in the AIS format over the mobile network to the processing platform onboard the ASV. We also performed some COLAV experiments using the emulated AIS, but in the experiments presented here we only used it for ground truth. The experiments were performed with a constant desired SOG of 4 m/s, while the guidance system attempts to follow a straight line towards the initial position of the target vessel. The target vessel was manually steered at a speed of approximately 5 m/s, attempting to keep a constant course towards the initial position of the ASV. The scenarios were initiated with a distance of at least 900 m between the ASV and the target vessel.

Experimental results

The first experiment was performed using the PDAF tracking states for vessel prediction. From Fig. 5, we see that the ASV traveled too close to the target vessel, finally entering the collision region, which caused the experiment to be aborted for safety reasons. The reason for this was the large variations in course and speed estimates of the target ship, which when used in an NCV model results in large variations in the predicted future trajectory of the target vessel. This further results in that the ASV travels closer than it should to the target ship, as the DW algorithm in some iterations believes that the target vessel moves in another direction than the actual direction of travel. Finally, the target vessel is so close that there is no option to avoid entering the collision region, failing the experiment. It is quite apparent from the estimated target course in Fig. 6 that the predicted target trajectory will fluctuate a lot. This is also confirmed by Fig. 7 showing the distance at closest point of approach (DCPA) for the target vessel, noting that this is dependent on which trajectory the DW algorithm chooses. One should also note that when entering the collision region, all velocity pairs are considered inadmissible, in practice deactivating the DW algorithm.

The second experiment was performed using the retrodicted tracking states for vessel prediction. In this case, the ASV avoided the collision region, rendering the experiment as a

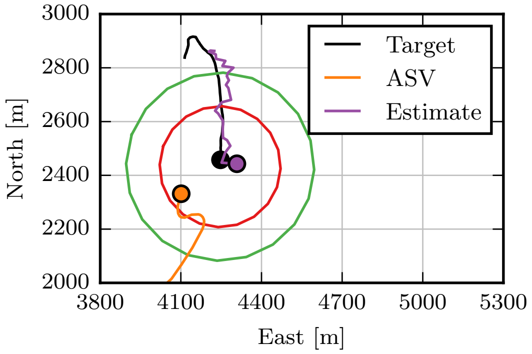

Figure 5. Experiment 1: North-east trajectory of the ASV and the target vessel, together with the PDAF position estimate used in the experiment. The experiment was aborted since the ASV traveled too close to the target vessel, moving into the collision region shown in red. The green circle show the safety region.

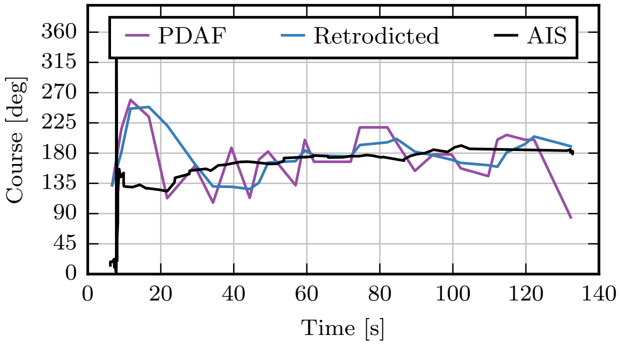

Figure 6. Experiment 1: Course estimate of the target vessel, with and without retrodiction. In this experiment, the PDAF estimate was used for predicting the target trajectory.

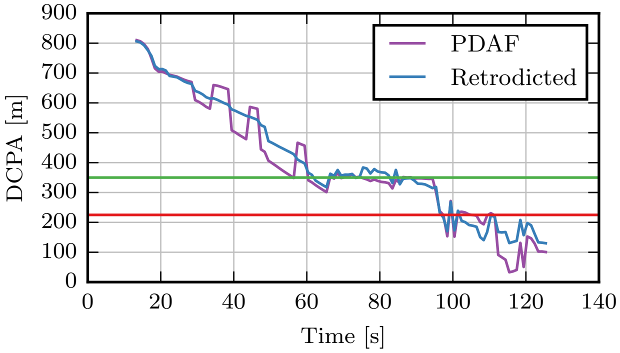

Figure 7. Experiment 1: DCPA for the target vessel. The collision and safety regions are shown in red and green, respectively.

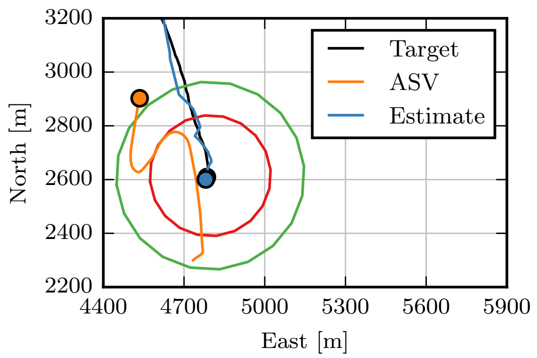

Figure 8. Experiment 2: North-east trajectory of the ASV and the target vessel, together with the retrodicted position estimate used in the experiment. The collision and safety regions are shown in red and green, respectively.

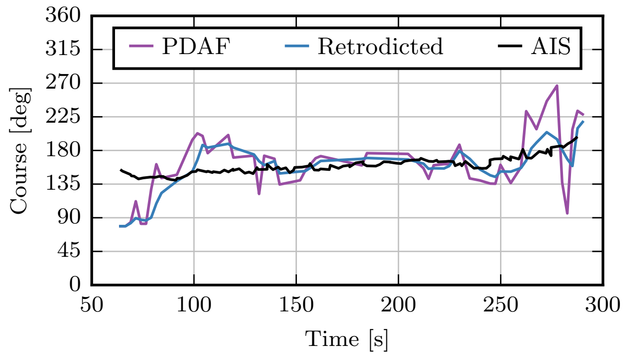

Figure 9. Experiment 2: Course estimate of the target vessel, with and without retrodiction. In this experiment, the retrodicted estimate was used for predicting the target trajectory.

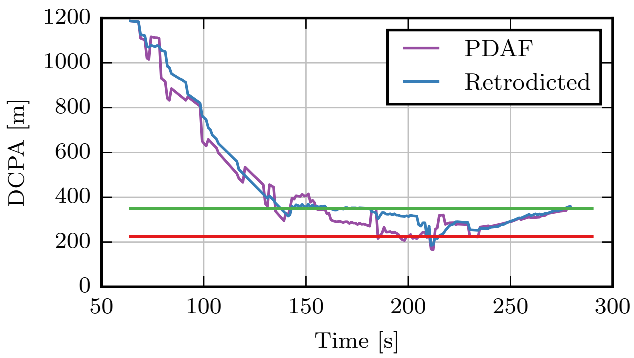

Figure 10. Experiment 2: DCPA for the target vessel. The collision and safety regions are shown in red and green, respectively.

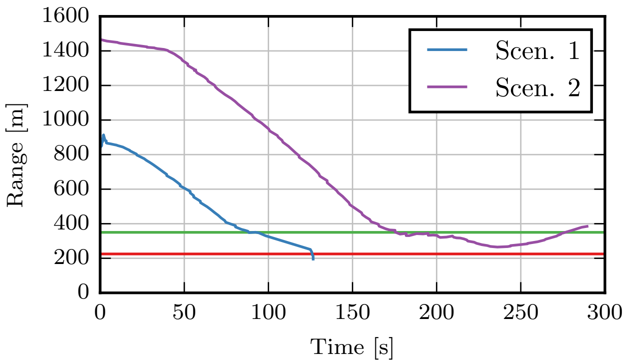

Figure 11. Distance to the target vessel for both experiments, with the collision and safety regions shown with red and green lines.

success. From Fig. 8, we see that the ASV successfully avoids the target vessel, before returning towards the path specified by the LOS guidance system. Figures 9 and 10 show the target course estimate and DCPA for the second experiment, showing the same trends as for the first experiment.

Fig. 11 shows the distance between the ASV and the target vessel for both experiments. It is clear that using a retrodicted target course and speed estimate improves the performance of the closed-loop COLAV system.

7. Conclusion

We have experimentally tested a closed-loop COLAV system consisting of a radar-based tracking system using PDAF and a COLAV system based on the DW algorithm. The system successfully avoided collision with a target vessel when using a retrodiction filter to generate smooth estimates of the target vessel course and speed, but failed when not applying a filter. This highlights the requirements for the input to the COLAV system. The tracking system should hence be able to provide smooth estimates of the target course and speed, but the COLAV system should also be able to handle inaccurate estimates of target vessels to increase the robustness of the closed-loop system. In practice, the DW algorithm became deactivated when the ASV entered the collision region, since this causes all velocity pairs to be considered inadmissible. The COLAV system should steer the ASV to avoid this situation, but variations in the tracking estimates can still put the ASV in this position, revealing an important shortcoming of the DW algorithm.

Future work will investigate the potential for improvements in speed and course estimates through multiple model filtering, fixed-lag smoothing techniques and/or nonlinear motion models. The results also motivate for modifications to the DW algorithm, or the development of a new algorithm more robust to variations in the tracking estimates. An attractive topic is also to combine the reactive COLAV system with the deliberate COLAV algorithm in [9].

Acknowledgments

This work was supported by the Research Council of Norway through project number 244116 and the Centres of Excellence funding scheme with project number 223254. The authors would like to express great gratitude to Kongsberg Maritime and Maritime Robotics for providing high-grade navigation technology and the Telemetron vessel at our disposal for the experiments.

REFERENCES

[1] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, 1995.

[2] R. A. Best and J. Norton, "A new model and efficient tracker for a target with curvilinear motion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 3, pp. 1030–1037, 1997.

[3] E. F. Brekke and E. F. Wilthil, "Suboptimal Kalman filters for target tracking with navigation uncertainty in one dimension," in *2017 IEEE Aerospace Conference*, Big Sky, MT, USA, 2017, pp. 1–11.

[4] A. N. Cockcroft and J. N. F. Lameijer, *A Guide to the Collision Avoidance Rules*. Elsevier, 2004.

[5] O. E. Drummond, "Target tracking with retrodicted discrete probabilities," in *Signal and Data Processing of Small Targets*, vol. 3163. International Society for Optics and Photonics, 1997, pp. 249–269.

[6] L. Elkins, D. Sellers, and W. R. Monach, "The autonomous maritime navigation (AMN) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *Journal of Field Robotics*, vol. 27, no. 6, pp. 790–818, 2010.

[7] B.-O. H. Eriksen, "Horizontal collision avoidance for autonomous underwater vehicles," Master's thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2015.

[8] B.-O. H. Eriksen and M. Breivik, "Modeling, identification and control of high-speed ASVs: Theory and experiments," in *Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles*, T. I. Fossen, K. Y. Pettersen, and H. Nijmeijer, Eds. Springer International Publishing, 2017, pp. 407–431.

[9] ——, "MPC-based mid-level collision avoidance for ASVs using nonlinear programming," in *Proc. of IEEE CCTA*, Kohala Coast, Hawai'i, USA, 2017.

[10] B.-O. H. Eriksen, M. Breivik, K. Y. Pettersen, and M. S. Wiig, "A modified dynamic window algorithm for horizontal collision avoidance for AUVs," in *Proc. of IEEE CCA*, Buenos Aires, Argentina, 2016.

[11] A. L. Flåten and E. F. Brekke, "Rao-blackwellized particle filter for turn rate estimation," in *2017 IEEE Aerospace Conference*, Big Sky, MT, USA, 2017, pp. 1–7.

[12] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Ltd, 2011.

[13] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, pp. 23–33, 1997.

[14] A. Harati-Mokhtari, A. Wall, P. Brooks, and J. Wang, "Automatic identification system (AIS): Data reliability and human error implications," *Journal of Navigation*, vol. 60, no. 3, pp. 373–389, 2007.

[15] T. Kirubarajan and Y. Bar-Shalom, "Kalman filter versus IMM estimator: when do we need the latter?" *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1452–1457, 2003.

[16] Y. Kuwata, M. T. Wolf, D. Zarzhitsky, and T. L. Huntsberger, "Safe maritime autonomous navigation with COLREGS, using velocity obstacles," *IEEE J. Oceanic Eng.*, vol. 39, no. 1, pp. 110–119, 2014.

[17] X.-R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[18] Ø. A. G. Loe, "Collision avoidance for unmanned surface vehicles," Master's thesis, Norwegian University of Science and Technology (NTNU), Trondheim, Norway, 2008.

[19] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: an open-source robot operating system," in *Proc. of the IEEE ICRA Workshop on Open Source Robotics*, Kobe, Japan, 2009.

[20] M. Schuster, M. Blaich, and J. Reuter, "Collision avoidance for vessels using a low-cost radar sensor," in *Proc. of the 19th IFAC World Congress*, Cape Town, South Africa, 2014, pp. 9673–9678.

[21] E. Serigstad, "Hybrid collision avoidance for autonomous surface vessels," Master's thesis, Norwegian University of Science and Technology (NTNU), Trondheim, Norway, 2017.

[22] The Norwegian Mapping Authority, "CPOS: Positioning," https://kartverket.no/posisjonstjenester/cpos/, accessed: 2017-10-13.

[23] E. F. Wilthil and E. F. Brekke, "Compensation of navigation uncertainty for target tracking on a moving platform," in *19th International Conference on Information Fusion (FUSION)*, Heidelberg, Germany, 2016, pp. 1616–1621.

[24] E. F. Wilthil, A. L. Flåten, and E. F. Brekke, "A target tracking system for ASV collision avoidance based on the PDAF," in *Sensing and Control for Autonomous Vehicles: Applications to Land, Water and Air Vehicles*, T. I. Fossen, K. Y. Pettersen, and H. Nijmeijer, Eds. Springer International Publishing, 2017, pp. 269–288.

BIOGRAPHY

Bjørn-Olav H. Eriksen *is a PhD candidate at the Department of Engineering Cybernetics at NTNU, where he also obtained his MSc in Engineering Cybernetics in 2015. He specializes in collision avoidance for autonomous surface vehicles, currently working in the project "Sensor fusion and collision avoidance for autonomous surface vehicles".*

***Erik F. Wilthil** is a PhD candidate at the Department of Engineering Cybernetics at NTNU, where he also obtained his MSc in Engineering Cybernetics in 2015. He specializes in target tracking and navigation for autonomous surface vehicles, currently working in the project "Sensor fusion and collision avoidance for autonomous surface vehicles".*

***Andreas L. Flåten** is a PhD candidate at the Department of Engineering Cybernetics at NTNU, where he also obtained his MSc in Engineering Cybernetics in 2015. He specializes in multisensor fusion for collision avoidance, currently working in the project "Sensor fusion and collision avoidance for autonomous surface vehicles".*

***Edmund F. Brekke** has an MSc (2005) in Industrial Mathematics and a PhD (2010) in Engineering Cybernetics, both awarded by NTNU. His PhD research covered methods for target tracking using active sonar. This research was conducted at the University Graduate Center at Kjeller in collaboration with Kongsberg Maritime. After his PhD studies, Brekke worked as a postdoctoral research fellow at the Acoustic Research Lab at NUS in Singapore, before becoming an Associate Professor in Sensor Fusion at NTNU in 2014. Brekke's main research interests lie in Bayesian estimation, with applications in target tracking and autonomous navigation. Brekke is project manager of the competence-building research project "Sensor fusion and collision avoidance for autonomous surface vehicles" funded by the Research Council of Norway, Kongsberg Maritime, DNV GL and Maritime Robotics.*

***Morten Breivik** has an MSc (2003) and a PhD (2010) in Engineering Cybernetics from NTNU. He is currently Head of NTNU's Department of Engineering Cybernetics, and is also an associate researcher at the Centre for Autonomous Marine Operations and Systems (NTNU AMOS). Breivik has previously worked as an assistant professor and researcher at NTNU, a scientific advisor for Maritime Robotics, and as a principal engineer and department manager in applied cybernetics at Kongsberg Maritime. His research interests include nonlinear and adaptive motion control of unmanned vehicles in general and autonomous ships in particular. Breivik is also currently a member of the Norwegian Board of Technology.*