Collision-avoidance capabilities reduction after a cyber-attack to the navigation sensors

G Longo[a], M Martelli[b*], E Russo[a], R Zaccone[b]

[a]Dept. of Informatics, Bioengineering, Robotics, and Systems Engineering (DIBRIS), Polytechnic School of Genoa University, Genova, Italy.
[b]Dept. of Electrical, Electronic, Telecommunications, Naval Architecture and Marine Engineering (DITEN), Polytechnic School of Genoa University, Genova, Italy.
*Corresponding Author. e-mail: michele.martelli@unige.it

Synopsis

The interest in autonomous navigation for surface ships has arisen over the years for both naval vessels and commercial ships. In literature, several pieces of research addressed the challenges of developing new guidance laws, stable and robust control algorithms, methodologies to increase situational awareness, and collision avoidance algorithms. Current papers base their outcome on blind trust in the information coming from navigation sensors. However, sensors can be subjected to malfunctioning, or even worse, cyber-attackers may hijack their data. This last scenario will be more and more common in the near future and represents a dangerous threat that future generation ships must face. In this work, a grey-box approach has been used to predict the outcomes of a state-of-the-art collision avoidance algorithm. Based on that, the attack consists of injecting fictitious targets as the input of the collision avoidance to force the ship to follow a predetermined and malicious track. A set of dedicated simulations using a ship simulator is carried out, and the effects of such a cyber-attack on the automatic collision avoidance system are shown. Based on the obtained results, at the end of the paper, effects on the evasive route generations are analysed and deeply discussed.

Keywords: Autonomous Ship, Collision Avoidance, Cyber-range, Cyber-attack

1 Introduction

Autonomous navigation is a major focus of nowadays research. While engineers deal with autonomous navigation and control strategies, lawyers and insurance companies manage the controversies associated with the existence of unmanned autonomous vehicles. The third open research debate is related to cybersecurity. With the increase of digital and interconnected systems, this aspect becomes crucial. In Kavallieratos et al. (2018), authors identify and categorize systems that make up an autonomous ship and analyze the ship's cybersecurity. The problem of the cybersecurity is handled by international bodies and register issuing guidelines based on the risk analysis International Maritime Organization (2017), since at current stage, with an immature, fast-changing and not standardised rules can be difficult to apply. In Tam and Jones (2018), the authors present a model-based risk assessment and the need for a specific framework that differs from the automotive field. A cyber attack is not a futurist problem but is still present in the maritime community. In fact, in Meland et al. (2021) reports an overview of 46 maritime cybersecurity incidents from the last decade (2010-2020). Of course, the risk depends on the level of autonomy of the ship defined by International Maritime Organization (2018), as reported in Tusher et al. (2022). Cybersecurity is also of great interest to the ports industry and shore-based infrastructures, de la Peña Zarzuelo (2021).

Thinking of the future, many cyber-attack scenarios might involve autonomous ships: for instance, an autonomous ship might be lured near a fixed threat such as minefields, or in an area that exposes it to other threats, such as collisions or grounding. Moreover, thinking about future piracy, the ship could be forced to steer in a dangerous area to be seized to obtain a ransom. The attacker can also unnecessarily lengthen a trip to damage goods or make them arrive late. Lastly, a ship can be forced to bypass the territorial sea or navigate in off-limit areas. All these possible scenarios can have severe consequences, and the near-future automation designers should face with. The main idea of the paper is to understand how realistic these scenarios are. For such a reason, in this work, a grey-box approach has been used to predict the outcomes of a state-of-the-art collision avoidance algorithm. Based on that, the attack consists of injecting fictitious targets as the input of the collision avoidance to force the ship to

Authors' Biographies

Giacomo Longo received his B.Sc. (2019), M.Sc. (2021) in Computer Engineering from the University of Genoa. He is currently a PhD student in the national PhD program in Artificial Intelligence for Security. His research interests revolve around digital twins, virtualisation, and maritime cyber security.

Michele Martelli received his B.Sc., M.Sc. and PhD degrees in Naval Architecture and Marine Engineering from Genoa University (Italy) in 2006, 2009 and 2013. From 2014 to 2016, as a post-doc, he worked on two research projects dealing with the autonomous capabilities of ships and small crafts. He joined as Assistant Professor at the DITEN Department, University of Genoa, in 2016. In 2019 he has been appointed as Associate Professor in the same Department. Since the beginning of his PhD, he has been involved in several national and international projects either as a researcher or as the principal investigator; he published over 60 peer-reviewed articles. He is a reviewer for several high-ranked journals and part of several international scientific committees.

Enrico Russo received his M.Sc.in Computer Science and Ph.D. in Computer Science and Systems Engineering at University of Genoa in 2001, and 2021. He joined as Assistant Professor at DIBRIS, University of Genova, in 2021. His research activity is focused on Cyber Range systems, digital twins and maritime cyber security.

Raphael Zaccone has a BSc, MSc and PhD in Naval Architecture and Marine Engineering. He received his PhD in 2017 from the University of Genoa, Italy. After two years of post doc, he joined the DITEN Department of the University of Genoa as Assistant Professor in 2019. His main research interests deal with ship autonomous navigation, collision avoidance and route planning, as well as ship propulsion control and simulation. He published over 30 peer-reviewed articles in international scientific journals and conferences.

follow a predetermined and malicious track. Then an evaluation of the "effectiveness" of the proposed approach in modifying a ship's route is carried out. In particular, the consequences of trajectory modifications have been analysed by using a ship simulator and carrying out about 100000 scenarios. The paper is structured as follows. The collision avoidance module is described in Section 2. The attacker model and description of the attack are shown in Section 3 and in Section 4, respectively. The implementation is reported in Section 5 while the results are in Section 6. Conclusions and recommendations are drawn in Section 7.

2 Collision avoidance

The collision avoidance capabilities of the autonomous surface vessel are ensured by a Guidance and Navigation Controller (GNC) system. Figure 1 presents the architecture of the autonomous navigation bridge. The Integrated Navigation System (INS) network collects data from the onboard sensors to provide situational awareness of the surrounding environment (see Section 3.1), including fixed obstacles or other ships. The GNC system is composed of a path planning module, a Collision Avoidance Algorithm (CAA) module, and a track-keeping module. The INS shares information about the surrounding obstacles and the vessel's position, speed, and heading with the CAA module. The path planning module is then required to compute a collision-free route based on the available information. The route is updated at a fixed rate with up-to-date obstacle information: each time, the motion planning module recomputes the optimal path. To ensure collision avoidance and COLREG compliance, the path planning algorithm predicts the future motion of the moving obstacles: periodic re-planning allows updating the obstacle data with any unpredictable changes. Eventually, the route waypoints feed the track-keeping module, ensuring that the vessel follows the computed route.

The motion planner is based on the RRT* algorithm to provide COLREG-compliant evasive manoeuvres, taking into account the manoeuvring capabilities of the vessel (Zaccone et al., 2019). The algorithm has been extensively tested in simulation in realistic scenarios (Zaccone and Martelli, 2020; Zaccone, 2021). The presented architecture is designed for open water navigation, therefore the own vessel is supposed to move at a medium to high speed and a reasonable distance from the other ships in the scenario, in the order of a minimum of some ship lengths, or at least twice the tactical diameter. Moreover, the evasive manoeuvres do not require a reduction of the own vessel's speed. Such an assumption is reasonable since steering is usually considered the first choice solution for collision avoidance in everyday ship conduction.

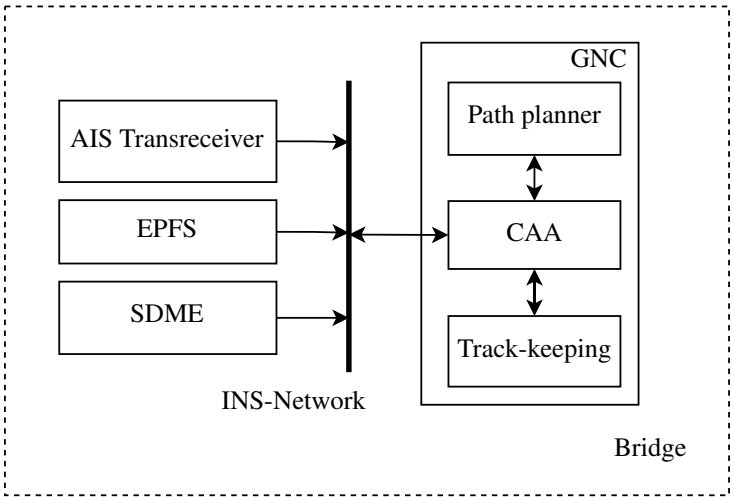


Figure 1: An overview of the autonomous navigation bridge architecture

The CAM operates according to the following rules:

- If target ships are detected, the CAA tries to perform a COLREG-compliant CAM, keeping a safety distance of about 1 nautical mile and limiting the route elongation;
- If no COLREG-compliant CAMs are possible, the CAA tries to perform a non-COLREG-compliant manoeuvre respecting the safety distance;
- If no CAMs are possible, the algorithm progressively reduces the safety distance until a feasible CAM is found.

3 Attacker model

This section introduces the assumptions and capabilities of the attackers.

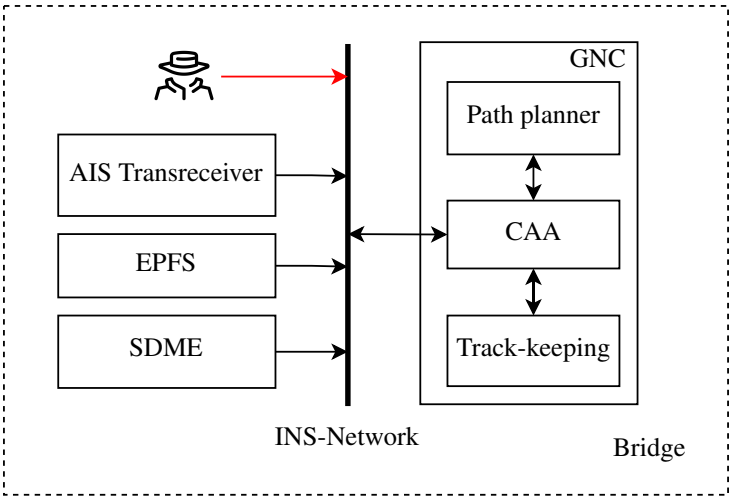


Figure 2: Entrypoint of the attacker inside of the INS

3.1 Assumptions

Figure 1 depicts the network hosting an INS (International Maritime Organization, 2007) and an excerpt of data sources related to connected sensors. In particular:

i) an AIS transreceiver (International Telecommunication Union, 2014) picks up radio signals and transmits over the network the position reports of other AIS-equipped vessels;

ii) the EPFS and SDME sensors report the position and speed of the own ship, respectively.

The above data sources represent the inputs of the GNC for outputting its route waypoints and the cross-track error. All the transmissions are expected to leverage the standard NMEA protocol (International Electrotechnical Commission, 2016). Moreover, it is assumed that a ship under the guidance of a CAA will try to meet two basic objectives.

- COLREG-compliant guidance (International Maritime Organization, 1972).
- A minimal deviation w.r.t. the original route whenever a CAM is required.

Eventually, the attackers are supposed to have already gained access to the ship and are connected to its INS network.

3.2 Capabilities

Under the assumption that attackers are connected to the INS network, they can operate like any other equipment and interact by reading and writing NMEA data. This condition enables different capabilities as follows.

- Observe the position (as reported by EPFS), speed (as sensed by SDME), and planned route waypoints (as planned by GNC) of the ship under attack.
- Gain an approximate knowledge of some properties (see Section 4) of the CAA in use.
- Inject into the INS fake ships with a predetermined position and speed, i.e., forge and transmit AIS messages by leveraging the shared communications bus nature of the INS.

Moreover, attackers can rely on highly predictable outcomes of some manoeuvres. This capability depends on the assumption that the CAA operates according to the COLREG. For instance, *Rule 14* states that whenever two vessels are approaching with nearly reciprocal courses, they both must perform a starboard turn. This rule allows adversaries to induce a starboard turn of the vessel under attack by injecting a fake ship that reproduces to the CAA the conditions prescribed above.

4 Attack description

This section details the attack against CAAs. The goal of the attackers is to lure the autonomous ship to a predetermined malevolent location. Briefly, the attack comprises two distinct phases: *reconnaissance* and *exploitation*

During the reconnaissance phase, attackers analyse the CAA in response to various predetermined conditions. As a result, the attackers obtain the insights needed to characterise a model of it. The use of the generated model is twofold. First of all, it allows attackers to maximise their probability of success by understanding which CAMs are possible. Then, during the exploitation phase, attackers can leverage it to calculate the trajectory of a to-be-injected fake target ship which is subsequently fed to the INS.

4.1 Reconnaissance phase

The reconnaissance phase is divided into two steps, the first aimed at identifying the expected behaviour, namely *profiling of baseline conditions*, and the second at characterising the CAA, namely *estimating obstacle detection range and CAA deviation*. Details are given in the following subsections.

4.1.1 Profiling of baseline conditions

The first reconnaissance step aims at establishing a baseline to correctly distinguish between normal navigation and CAMs. For that purpose, attackers monitor the ship's cross-track error while underway. This activity allows them to establish the maximum deviation from the straight-line course to the next waypoint under normal conditions. The attackers will then classify any future course alteration exceeding the found threshold T_h as belonging to a CAM.

$$\text{IsCAMStart}(\mathbf{X}_n, \mathbf{X}_{n-1}, \ldots, \mathbf{X}_{n-k}) \coloneqq \bigvee \begin{cases} \mathbf{X}_{i_h} \geq \mathbf{X}_{i-1_h} \forall i = n, \ldots, n-k+1 \bigwedge \mathbf{X}_{i_h} - \mathbf{R}_{i_h} \geq T_h \forall i \\ \mathbf{X}_{i_h} \leq \mathbf{X}_{i-1_h} \forall i = n, \ldots, n-k+1 \bigwedge \mathbf{R}_{i_h} - \mathbf{X}_{i_h} \geq T_h \forall i \end{cases} \tag{1}$$

Equation 1 allows attackers to classify whenever an observation of k ship positions $\mathbf{X}_i = \langle \mathbf{X}_{i_h}, \mathbf{X}_{i_v} \rangle$ belongs to the beginning of a CAM. Each of the cases is a conjunction between a monotonicity constraint on horizontal components $\mathbf{X}_{i_h}$ of the positions and a threshold on the deviation w.r.t. the original route horizontal position $\mathbf{R}_{i_h}$.

4.1.2 Estimating obstacle detection range and CAA deviation

As prescribed by COLREG, each ship's CAA route must keep a safe distance from other vessels to avoid collisions. Measuring how the distance from other vessels influences the ship's route by avoiding collision allows attackers to constrain which areas are reachable as a consequence of the attack.

$$\text{SD}(O) \coloneqq \lambda(\mathbf{X}, \mathbf{s}, \mathbf{X}_o, \mathbf{s}_o) \tag{2}$$

The first problem faced by the attackers is finding an approximate relation λ between the CAA desired safe distance SD, the ship position $\mathbf{X}$, and speed $\mathbf{s}$ whenever faced with a vessel O that is deemed to be in a collision. Its symbolic representation is given in Equation 2.

$$\mathbf{R}(O) \coloneqq \xi(\mathbf{X}_t, \mathbf{X}, \mathbf{s}, \mathbf{X}_o, \mathbf{s}_o) \quad i.i.f. \quad ||\mathbf{R}_i - \mathbf{X}_o|| \leq \text{SD}(O) \text{ for any } i \tag{3}$$

Then, the attackers must estimate which waypoints $\mathbf{R}(O)$ the CAA heuristic ξ generates once it finds a vessel O on a collision course along the route. Equation 3 presents the mathematical formulation of this second problem. In particular, correctly estimating the codomain of ξ allows attackers to understand which waypoints are feasible to be reached during the attack.

4.2 Exploitation phase

The exploitation phase comprises two steps aiming at planning and executing the rogue ship injection. During the first step, the attackers must find where to inject the fake ship to lure the victim ship towards the desired waypoint $\mathbf{X}_a$.

$$O(\mathbf{X}_a) \coloneqq \Gamma(\lambda, \xi) \tag{4}$$

Equation 4 summarises the above step. Briefly, it synthesises a heuristic Γ for calculating the properties of a rogue vessel O, dependent on the inferred CAA behaviour λ and ξ so that the target position $\mathbf{X}_a$ will belong to the CAM for O.

Finally, in the second step, attackers inject the generated vessel O via rogue NMEA sentences (Balduzzi et al., 2014) sent to the INS network.

5 Attack Implementation

This section details an implementation of the attack described above. For the sake of shortness, the described implementation focuses on Rule 14 of COLREG. However, the proposed approach is general and can leverage other scenarios covered by the COLREGs.

5.1 Reconnaissance phase

This phase consists in observing the behaviour of the victim ship and traffic trajectories between pairs of waypoints emitted by the CAA, namely segments. P_s and P_e identify the start and end points of each segment, respectively. In particular, the attackers overhear the INS network to identify segments and, for each one, determine the speeds and positions of the ship under attack and other vessels. Then, they classify segments into two categories.

1. No traffic has been encountered along the way.

2. A possible head-on situation has unfolded during the segment.

They discard segments that do not fall into these two categories. Once the number of gathered segments of both categories is deemed sufficient, the attackers can start building their adversarial CAA model. Recalling the equations presented in Section 4.1, the techniques used to calculate T_h (Equation 1), λ (Equation 2), and the codomain of ξ (Equation 3) are briefly described.

5.1.1 Estimation of T_h

To estimate T_h, the attackers calculate for each acquired sample belonging to the first category the point-line distance between the victim ship position $\mathbf{X}$ and the $P_e - P_s$ line. Then, they build a distribution of the found value. T_h represents the sample mean plus three times the found standard deviation.

5.1.2 Estimation of λ

λ is estimated to be a linear function of the victim ship speed. For that purpose, the attackers calculate for every segment belonging to the second category the minimum distance to traffic ships and the speed $\mathbf{s}$ at which the victim ship was going at that moment of the closest encounter. Then, they use the above samples to calculate via least squares regression a slope m and intercept q. q allows them to estimate the safe distance function as $\lambda = m \cdot |\mathbf{s}| + q$.

5.1.3 Estimation of the codomain of ξ

Attackers can estimate the co-domain of ξ by considering the point-line distance measurement for each segment belonging to the second category, as in the case of T_h. The above measurements are then normalised with respect to their position among the $P_e - P_s$ line to construct a virtual *boundary* of the CAM. Eventually, they use the boundary to constrain the positions of their target point $\mathbf{X}_a$.

5.2 Exploitation phase

In this phase, the attackers estimate the properties of the ship to be injected, i.e. the shape of Γ and inject the rogue ship into the INS.

5.2.1 Estimation of Γ

For estimating Γ, attackers can leverage a *gradient boosting decision trees* (Ke et al., 2017) regressor. In particular, its training relies on segments belonging to the second category to estimate, given a specific victim ship position $\mathbf{X}$, the properties below.

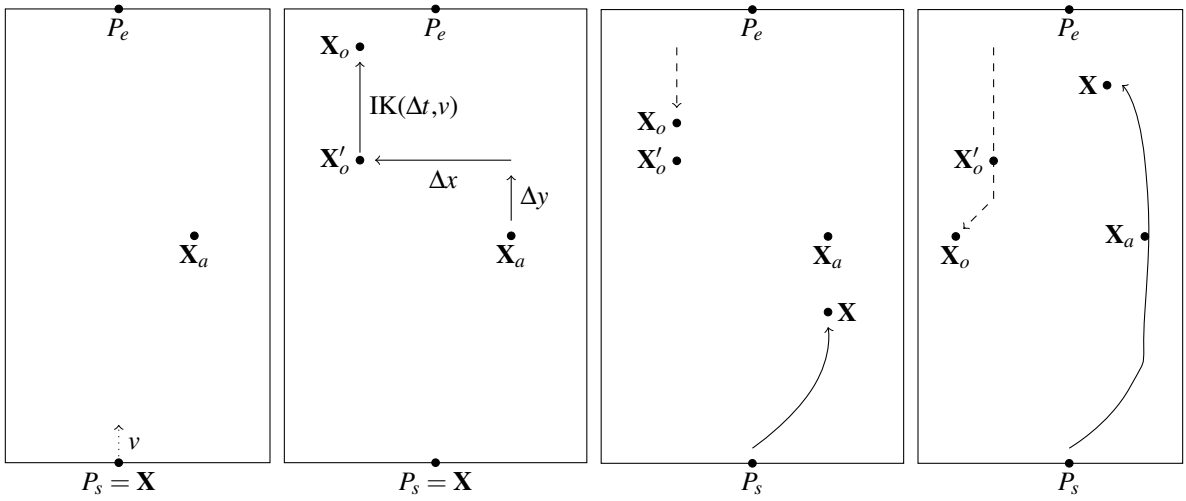
- The horizontal separation (Δx) between $\mathbf{X}$ and the head-on traffic ship.
- The vertical separation (Δy) between $\mathbf{X}$ and the head-on traffic ship.
- The time (Δt) since the victim ship has crossed P_s.

Figure 3 represents how the injection of a rogue ship evolves during an attack. In the initial state (see Figure 3a), the victim ship is navigating towards P_e at a constant speed v. $\mathbf{X}_a$ identifies the target chosen from within the codomain of ξ where the attackers want to lure the victim ship.

To inject the rogue ship (see Figure 3b), the attackers need to calculate its starting position $\mathbf{X}_o$. In particular, they can obtain Δx, Δy, and Δt by feeding $\mathbf{X}_a$ to the regressor. Then, they calculate $\mathbf{X}'_o$ by adding the resulting offsets to $\mathbf{X}_a$. As the regressor returns a position after a time of Δt, the attackers can compute the initial $\mathbf{X}_o$ by solving an Inverse Kinematics (IK) problem with Δt and v.

After calculating the initial placement, the attackers must evolve its position by imposing a trajectory that ensures the rogue ship crosses $\mathbf{X}'_o$ with the Δt and v used for solving IK. To this aim, they fix the final trajectory as a straight line at speed v (see Figure 3c).

Finally, the injected ship leads the victim to perform a CAM that forces the crossing at $\mathbf{X}_a$ as desired (see Figure 3d). To make the manoeuvre more realistic and malicious intents less evident, the attackers impose the rogue ship to engage too in its CAM and then proceed again with a straight line trajectory.



(a) Initial state (b) Rogue ship injection (c) Evolution (d) Closing phase

Figure 3: Evolution of the injected ship

6 Results

6.1 Simulation scenario

Figure 4 depicts the simulating scenario used to analyse the attack implementation against the GNC described in Section 2.

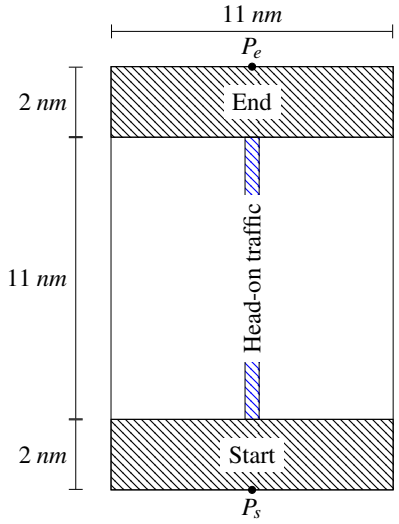


Figure 4: The environment for the experiments.

Briefly, the autonomous own-ship starts from P_s, to reach a destination P_e located 15 *nm* ahead. The scenario evolves in discrete timesteps of 5*s*, in which the own-ship autonomously manoeuvres (max rotation $0.2\,^\circ/s$) along the waypoints generated every 30 *s* by the CAA.

COLREG rule 14 has been simulated by placing a head-on ship between 2 and 13 nautical miles ahead of the own-ship, among the $P_e - P_s$ line with a random shift in the horizontal direction of at most 500*m*. The simulation reproduced 100 no traffic situations and 1000 head-on situations. According to their capabilities to listen for the INS traffic (see Section 3.2), the attackers recorded the victim and traffic ship trajectories at each timestep of every scenario.

6.2 Reconnaissance phase

The recorded trajectories represent the dataset required by the reconnaissance phase. From this dataset, T_h for IsCAMStart has been calculated as the mean plus three times the standard deviation of the cross-track error observed whenever no traffic ship is present. Figure 5 highlights that the GNC under attack does not exhibit a deviation exceeding $T_h = 59.8$m in no-traffic conditions.

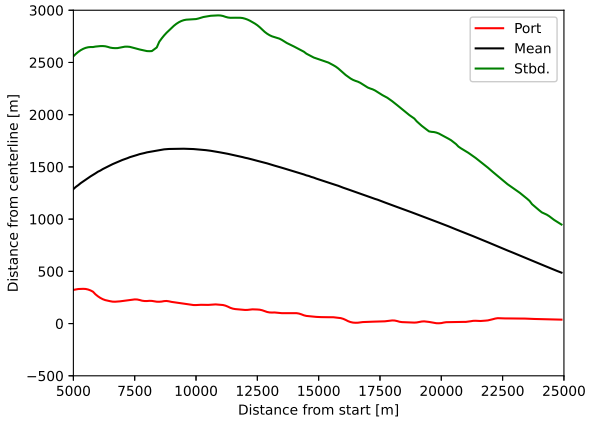


Figure 7: Derived bounds of the CAM.

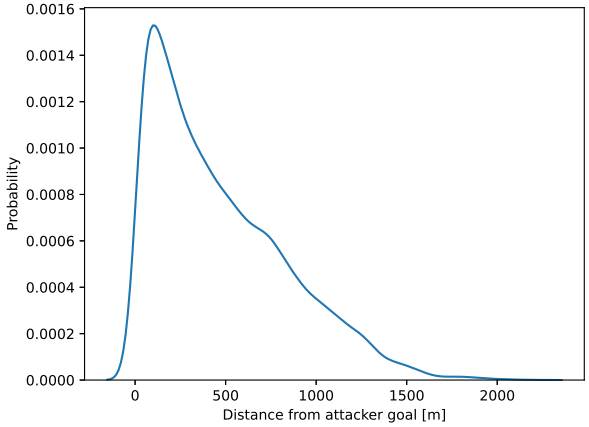


Figure 8: Distribution of minimum distances between $\mathbf{X}$ and $\mathbf{X}_a$

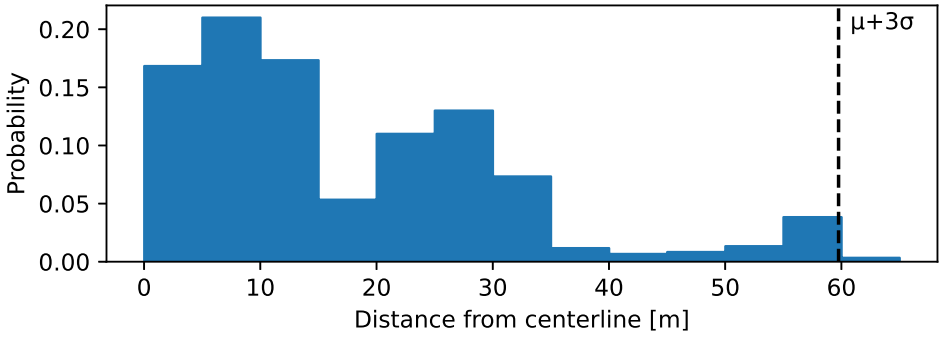


Figure 5: Cross track error distribution during no-traffic experiments.

Then, IsCAMStart has been leveraged to sample the minimum separation between the victim and the traffic ship. During the experiment, the speed of the own-ship has negligible effects on the separation distance. For this reason, λ (see Section 4.1.2) was approximated with a constant value. Figure 6 shows the obtained distribution. For the GNC under attack, the mode of the sampled distribution (2130m) is chosen due to its high positive skewness.

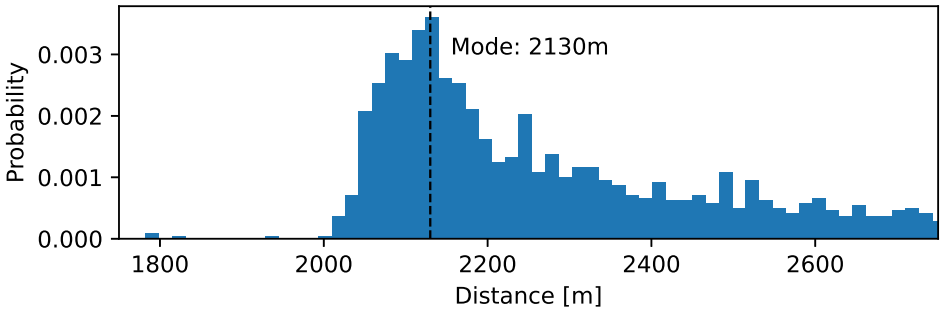


Figure 6: Distribution of minimum obstacle separations.

Eventually, IsCAMStart allowed sampling the minimum and maximum horizontal route elongation for a given vertical position along the $P_e - P_s$ line. As depicted in Figure 7, the GNC under attack bounds area of action for the adversaries to a maximal horizontal elongation of approximately 3000m.

6.3 Exploitation phase

After terminating the reconnaissance phase, the regressor on was trained 1510085 datapoints belonging to the scenarios containing a CAM according to IsCAMStart. Then, 96780 synthetic scenarios were generated and exe-

cuted. Each scenario runs in a $100x100$m square contained within the bounds of Figure 7. At least 10 experiments for each distinct square have been ensured. The injected ship's initial position and movement comply with the description given in Section 5.2.1, i.e., it shares the same manoeuvrability of the victim (max rotation $0.2\frac{deg}{s}$) and performs a 30° starboard turn as its CAM.

Figure 8 shows the resulting minimum distance distribution between $\mathbf{X}$ and $\mathbf{X}_a$. The GNC under attack has a sample mean of 470m with a standard deviation of 371m. Figure 9 presents the minimum distance between $\mathbf{X}$ and $\mathbf{X}_a$ as a function of its position among squares of size $300x1000$m (each square contains 300 samples). Figures 7 and 9 show that the best area for the attackers' waypoints is the closest to a genuine CAM, i.e., close to the black line in Figure 7. A possible limitation affecting such behaviour is the sampling procedure used for training the regressor that considers only Rule 14 of COLREG. The above condition biases predictions of the regressor toward what it has already seen and gives insufficient coverage of cases close to the centerline. Moreover, Figure 7 highlights that the attackers can lure the autonomous ship within a hundred meters from their intended target for 23 squares. Instead, extreme values were unreachable by the attacker. This last property stems from the tendency of the CAA to generate CAMs with minimal elongation w.r.t. original route (see Section 3).

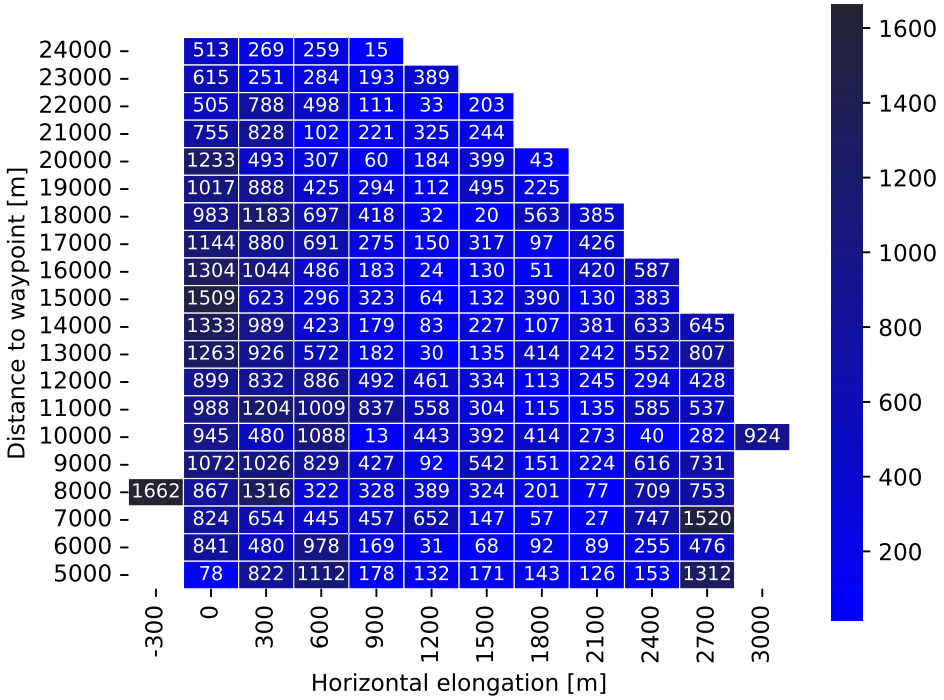


Figure 9: Minimum distances between $\mathbf{X}$ and $\mathbf{X}_a$

7 Conclusions

In this paper, the risk of a cyber-attack on the collision avoidance module has been shown and potential consequences have been explored. The trained algorithm was able to lure the ship to a predetermined position by misleading the heuristic-based collision avoidance algorithm with a fake target. A systematic analysis has been carried out to better understand the degree of threat and the maximum elongation. Despite the promising results, some improvements might be considered to further improve the attackers' outcomes and capabilities. For instance, augmenting the algorithm with heuristics capable of choosing the correct regressor depending on $\mathbf{X}_a$ would allow the method to generalise better with $\mathbf{X}_a$ located further apart from the mean CAM. In addition, multiple repetitions of the presented approach during the same attack could allow adversaries to reach $\mathbf{X}_a$s located further away from the centerline.

The increasingly open access to navigation data nowadays can make this scenario realistic, by allowing attackers to perform their reconnaissance phase from public data sources. As countermeasures, the navigation system should be fed by data coming from navigation different sensors such as cameras, RADAR and LiDAR and combine them with data fusion algorithms. To conclude, the recommendation is that, in addition to the data fusion techniques, proper tools need to be studied and deployed in order to early detect the hijack of both sensors and data, for example as proposed in (Kougiatsos et al., 2022; Maestre et al., 2021).

8 Acknowledgements

The research activities have been carried out thank to the SHIL (Ship Hardware In the Loop) research infrastructure, co-funded by Regione Liguria, University of Genova and DLTM under the program POR FESR LIGURIA 2014-2020 ASSE 1 "Research and Innovation (OT1)" Action 1.5.1 Notice "Support for research infrastructures considered critical / crucial for regional systems".

References

Balduzzi M, Pasta A, Wilhoit K. 2014. A Security Evaluation of AIS Automated Identification System. In: Proceedings of the 30th Annual Computer Security Applications Conference; New York, NY, USA. Association for Computing Machinery. p. 436–445. ACSAC '14.

de la Peña Zarzuelo I. 2021. Cybersecurity in ports and maritime industry: Reasons for raising awareness on this issue. Transport Policy. 100:1–4.

International Electrotechnical Commission. 2016. 61162-1 Maritime Navigation and Radiocommunication Equipment and Systems—Digital Interfaces—Part 1: Single Talker and Multiple Listeners.

International Maritime Organization. 1972. Convention on the international regulations for preventing collisions at sea.

International Maritime Organization. 2007. Adoption of the revised performance standards for integrated navigation systems (ins). MSC.252(83).

International Maritime Organization. 2017. Guidelines on Maritime Cyber Risk Management. Circular MSC-FAL.1/Circ.3.

International Maritime Organization. 2018. Framework for the regulatory scoping exercise for the use of maritime autonomous surface ships (mass). MSC.100/20.

International Telecommunication Union. 2014. Technical characteristics for an automatic identification system using time division multiple access in the vhf maritime mobile frequency band. M.1371-5.

Kavallieratos G, Katsikas S, Gkioulos V. 2018. Cyber-attacks against the autonomous ship. In: Computer security. Springer; p. 20–36.

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu TY. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, editors. Advances in Neural Information Processing Systems; vol. 30. Curran Associates, Inc.

Kougiatsos N, Negenborn RR, Reppa V. 2022. Distributed model-based sensor fault diagnosis of marine fuel engines. IFAC-PapersOnLine. 55(6):347–353. Available from: https://doi.org/10.1016/j.ifacol.2022.07.153.

Maestre JM, Velarde P, Ishii H, Negenborn RR. 2021. Scenario-based defense mechanism against vulnerabilities in lagrange-based DMPC. Control Engineering Practice. 114:104879. Available from: https://doi.org/10.1016/j.conengprac.2021.104879.

Meland P, Bernsmed K, Wille E, Rødseth Ø, Nesheim D. 2021. A retrospective analysis of maritime cyber security incidents. TransNav: International Journal on Marine Navigation and Safety of Sea Transportation. 15.

Tam K, Jones K. 2018. Cyber-risk assessment for autonomous ships. In: 2018 international conference on cyber security and protection of digital services (cyber security). IEEE. p. 1–8.

Tusher HM, Munim ZH, Notteboom TE, Kim TE, Nazir S. 2022. Cyber security risk assessment in autonomous shipping. Maritime Economics & Logistics:1–20.

Zaccone R. 2021. Colreg-compliant optimal path planning for real-time guidance and control of autonomous ships. Journal of Marine Science and Engineering. 9(4).

Zaccone R, Martelli M. 2020. A collision avoidance algorithm for ship guidance applications. Journal of Marine Engineering & Technology. 19(sup1):62–75.

Zaccone R, Martelli M, Figari M. 2019. A colreg-compliant ship collision avoidance algorithm. In: IEEE European Control Conference - ECC2019. p. 2530–2535.

Glossary

COLREG Convention on the International Regulations for Preventing Collisions at Sea (International Maritime Organization, 1972).

Acronyms

AIS Automatic Identification System.
ARPA Automatic Radar Plotting Aid.
CAA Collision Avoidance Algorithm.
CAM Collision Avoidance Manoeuvre.
EPFS Electronic Position Fixing System.
GNC Guidance and Navigation Controller.
INS Integrated Navigation System.
NMEA National Marine Electronics Association.
SDME Speed and Distance Measuring Equipment.