Distributed coverage optimization for a fleet of unmanned maritime systems

Geert De Cubber1, Rihab Lahouli2, Daniela Doroftei1, Rob Haelterman2

1 Royal Military Academy, department of Mechanics, Avenue De La Renaissance 30, 1000 Brussels, Belgium  
2 Royal Military Academy, department of Mathematics, Avenue De La Renaissance 30, 1000 Brussels, Belgiuml

ABSTRACT

Unmanned maritime systems can provide important benefits for maritime law enforcement agencies for tasks such as area surveillance and patrolling, especially when they are able to work together as a coordinated team. In this context, this paper proposes a methodology that optimizes the coverage of a fleet of unmanned maritime systems, and thereby maximizes the chances of noticing threats. Unlike traditional approaches for maritime coverage optimization, which are also used for example in search and rescue operations when searching for victims at sea, this approaches takes into consideration the limited seaworthiness of small unmanned maritime systems, compared to traditional large ships, by incorporating the danger level in the design of the optimizer .

Section: RESEARCH PAPER

**Keywords:** unmanned maritime systems, multi-agent systems, maritime surveillance, distributed coverage optimization

**Citation:** Geert De Cubber, Rihab Lahouli, Daniela Doroftei, Rob Haelterman, Distributed coverage optimization for a fleet of unmanned maritime systems, Acta IMEKO, vol. A, no. B, article C, Month Year, identifier: IMEKO-ACTA-A (Year)-B-C

**Section Editor:** name, affiliation

**Received** month day, year; **In final form** month day, year; **Published** Month Year

**Copyright:** This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Funding:** The research presented in this paper has been funded by the Belgian Royal Higher Institute for Defense, in the framework of the DAP19/08 (MarSur) project and by the Flemish Agency for Innovation and Entrepreneurship, in the framework of the Blue Cluster project SSAVE (HBC.2019.0045).

**Corresponding author:** Paul P. L. Regtien, e-mail: [paul@regtien.net](mailto:paul@regtien.net)

1. Introduction

An ever-increasing percentage of the world population is living in coastal areas. A downside of this evolution is that also more and more criminals are turning their attention to our seas and oceans to carry out illegal activities. Examples are drug smuggling, human trafficking, illegal fishery, border infringements, etc. The problem for law enforcement agencies is that patrolling and surveilling all these vast ocean surfaces with traditional means (large manned vessels) is impossible from an economic and operational point of view.

Unmanned Maritime Systems (UMS) can potentially provide maritime law enforcement agencies with a valuable tool for increasing their capabilities related to maritime surveillance. Obviously, UMS are not the sole answer, and just one part of a much wider maritime situational awareness toolkit [1], encompassing also satellite monitoring [2], manned and unmanned aerial assets [3] with advanced analytics solutions, allowing to turn the data gathered by all these agents into information and knowledge.

One of the main capabilities the UMS need to possess is the capability to operate together as a well-coordinated group or team, working together towards a higher-level goal such as maritime surveillance. However, the practical deployment of these novel smaller-scale UMS requires the careful consideration of several aspects related to the operational requirements of the end users [4], the interoperability between the different systems [5] and towards the design of the surveillance architecture. As an example, the classical approaches towards distributed patrol and surveillance [6], [7], [8] by manned systems generally do not take into consideration the effects of small waves (which are irrelevant for larger ships, but very important for small UMS).

In this paper, we will therefore propose a novel methodology for the real-time control of a fleet of two to ten UMS. The presented methodology is casted as a distributed coverage optimization problem, with as specificity that the danger-level for the UMS of turning over is effectively estimated in function of the potential trajectories and taken into consideration for the choice of the optimal movement strategy. As a result, optimal safe trajectories for all the agents of the fleet can be planned.

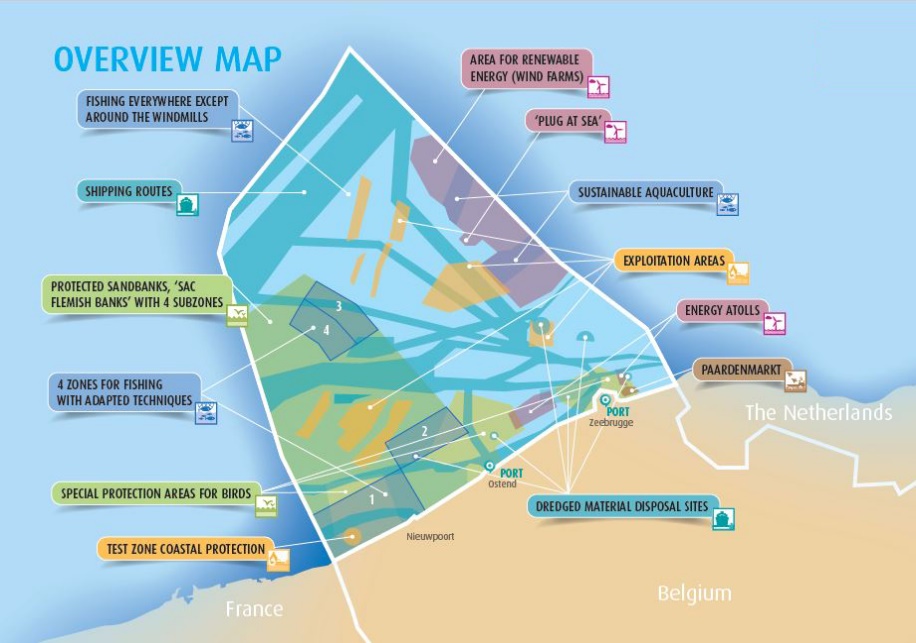


Figure 1. Maritime Spatial Plan of the Belgian territorial waters, showing the very dense occupation of the Belgian territorial waters by different actors and for different economic activities. This paper considers the surveillance and patrol of the off-shore wind farms, which is indicated on the map as the area with a red overlay (Source: Belgian Federal public service Health)

We validate the proposed approach in simulation in an application scenario [9] connected to the surveillance of the Belgian off-shore windmill parks. The Belgian territorial waters are a very densely populated maritime area, with tightly reserved spaces for all actors, as presented in Figure 1, and it is important that all actors stay within the delimited zones. For the windmill farms (area shaded in red on Figure 1), this often presents a problem, as other users (fishermen, pleasure yacht sailors, ...) penetrate this zone without permission. In order to police and enforce the interdiction zones, there is thus a need to patrol this area on the maritime border with the Netherlands, which measures about 10 km by 30 km.

1. Previous work

Multi-agent robotic coverage optimization is a research topic which has received a lot of attention in recent years, as more and more robotic assets are being deployed and thus also the need for finding strategies to optimize the coordination among these agents increases.

A first distinction to be made between the different methodologies is based upon the type of agents that is taken into consideration. On one hand, there are approaches that tackle *swarms* of a high number of less intelligent agents [10]. Swarm approaches generally make use of some form of ant colony optimization algorithm [11] for solving the coverage problem. On the other hand, there are *multi-agent* approaches that deal with a lower amount of more intelligent agents, which is the case in our application.

A second important distinction between methodologies is based upon the assumption which is made related to the connectivity between the different agents. If continuous broadband access between the agents is assumed, then all agents can get perfect localization and sensor data from one another and then the approaches are often based on some kind of global optimization approach [12], with the capability to adapt to a time-dependent environment [13]. Even though it has been shown that finding a globally optimal solution for the coverage maximization of a multi-agent fleet is an NP-hard problem [14], it is possible to come quite close to this solution within real-time constraints [15], [16], however, this requires intelligent strategies to guide the optimization process (see more discussion on this subject later).

If, on the other hand, unreliable network connections are assumed, then the agents cannot rely on a global planner and a local optimization is required. This also entails that a distributed approach is required which still allows for timely coordination between the different agents within the system, as proposed by Xin et al. in [17].

Our methodology adopts a hybrid approach. Conceptually, it is based on a global optimization, but which is executed separately by each of the agents, taking into consideration the latest known data from the other agents. We use spatio-temporal memories to track and predict the localization and sensor data from the other agents, in order to cover up communication delays and breakdowns. Obviously, these estimations are not perfect, but in this way the optimization scheme tries to adopt the best of both kind of approaches.

Within the robotics community, most attention has been focused on providing solutions to the multi-agent coverage optimization problem for unmanned ground vehicles, but there are certainly also approaches that consider unmanned aerial vehicles [18]. However, for maritime systems, the research domain is less developed. Fabbri et al presented in [19] a path and decision support system for maritime surveillance vessels, based on multi-objective optimization algorithms that see to find an optimal trade-off among several mission objectives. While the concepts are similar, this paper focuses on a high-level decision support system for large manned vessels. In our application, we are interested in developing a solution for small-scale unmanned patrol vessels, which means that the requirements and constraints are very different.

As discussed before, finding a globally optimal solution for the coverage maximization of a multi-agent fleet is an NP-hard problem [14]. This implies that the algorithms scale traditionally badly for an increasing amount of agents. As the hybrid planner we propose here features a mix between global and local optimization aspects, it also suffers from this drawback of global planning systems. In order to remedy this problem, researchers have proposed particle optimization [20], as proposed by Han et. al. or Grey Wolf Optimization methodologies, as first introduced by Mirjalili et. al. in [21] and later improved for distributed coverage optimization problems by Wang et. al. in [22]. Basically, all these methodologies aim to intelligently prune the number of candidate positions that have to be investigated in order to limit the number of computations to be performed. Developing further on these ideas, we also propose an optimization strategy to select quickly the high-likeliness candidate positions, thereby limiting the computation time.

1. Methodology
   1. Overall framework

The proposed methodology draws inspiration from behaviour-based control frameworks [23], where multiple behaviours actively work together to control the robot, or in this case the UMS. The main problem in behaviour-based control is how to synergize the different individual behaviours into a consistent and optimal global behaviour of the robotic agent. This requires the choice of so-called weight parameters that lead to the expected global behaviour. However, finding these weight parameters is a non-trivial task. Therefore, we propose to use an optimization scheme to find the optimal weights, taking into consideration two objectives: a) increasing the global coverage (and thereby increasing the acquisition of new knowledge about the environment), b) minimizing the danger level (and thereby minimizing the chance for the vessel to capsize).

A major design issue for the development of such an optimization scheme is that the weight parameters to be optimized are subject to a large amount of environmental factors, such as the visibility, the wave height, etc. Therefore, we adopted a dual approach.

* **In an offline learning stage**, depicted by Algorithm 1, we repeatedly run an optimization process in order to find the optimal weight parameters for multiple environmental conditions:

|  |  |
| --- | --- |
|  | (1) |

With the following parameters:

* + the weight parameters to be optimized
  + the number of agents
  + the position of the agents in a metric grid
  + the orientation of the agents in radians
  + the visibility in meters. This is a function of the sensorial visibility (which is considered to be static, as the sensor package of the UMS does not change throughout a mission) and the meteorological visibility, which is dynamic, as the weather conditions may change throughout a mission.
  + the field of view of the sensors on board of the UMS in radians. In this implementation, the sensors are always assumed to be front-facing (but the field of view can be set to 360°).
  + the maximum velocity (in meters per second) that can be obtained by the UMS
  + the maximum turning rate (in radians per second) that can be obtained by the different UMS
  + the wave height in meters
  + the wave orientation in radians
  + an obstacle map which is expressed as a probability density function expressing the probability of finding an obstacle
  + a dimensionless parameter regulating the relative importance of coverage maximisation and the minimisation of the risk for capsizing.

The parameters of the optimization function and the function itself are further explained in section 3.3. For this optimization process, we used a quite classic Nelder-Mead simplex algorithm [24]. This process typically takes a very long time (a few days, depending on the granularity / resolution requested). For this reason, section 3.4 introduces an accelerated optimization scheme. At the end of this process, the resulting data is stored in a database for later retrieval (during the online stage).

* **In an online stage**, we retrieve the correct weight parameters for the environmental conditions at hand from the database and apply these directly to the same optimization function used before, as depicted by Algorithm 2.

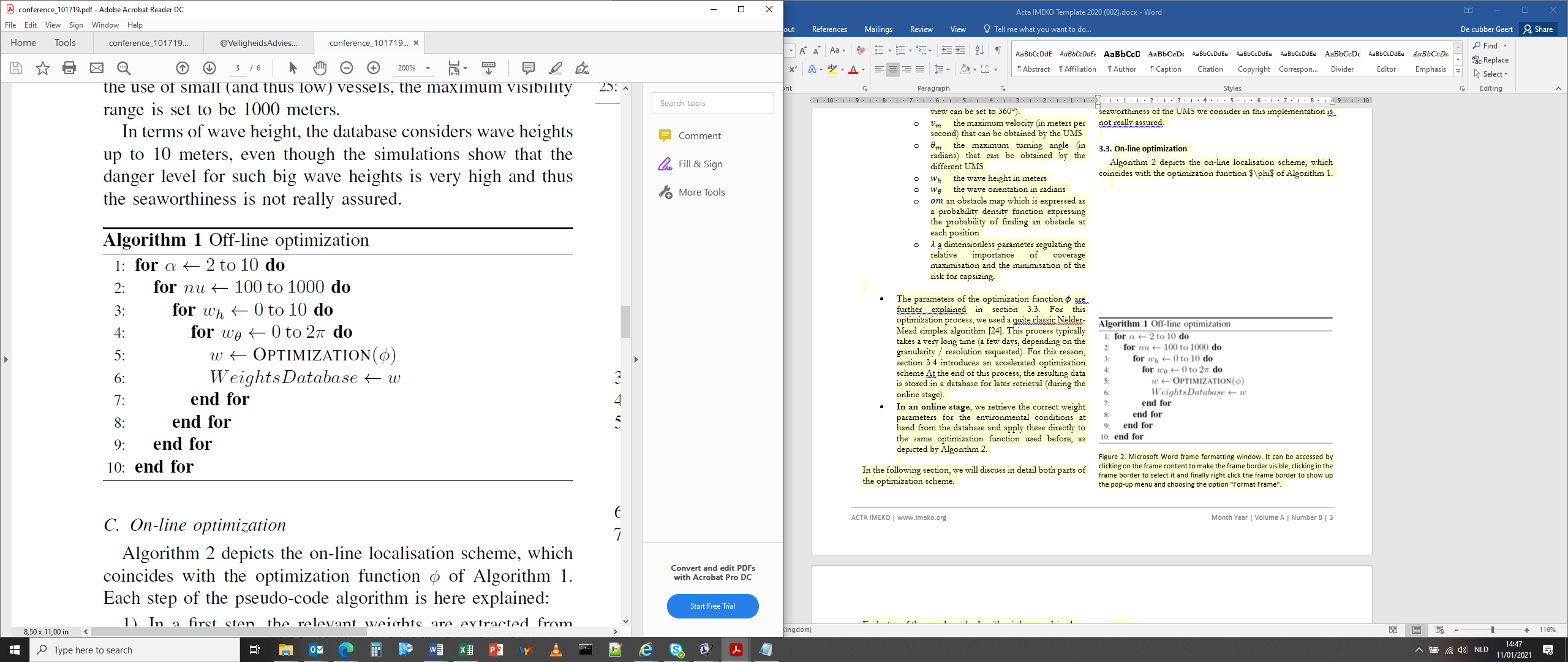
In the following section, we will discuss in detail both parts of the optimization scheme.

* 1. Off-line optimization

Algorithm 1 depicts the off-line optimization scheme. As explained, its objective is to fill a database containing for each possible combination of environmental factors the optimal weight parameters. In this implementation, we focus on four main factors that have experimentally shown to have an important impact on the choice of the different weight parameters: the number of assets (), the visibility (), the wave height () and the wave direction ().

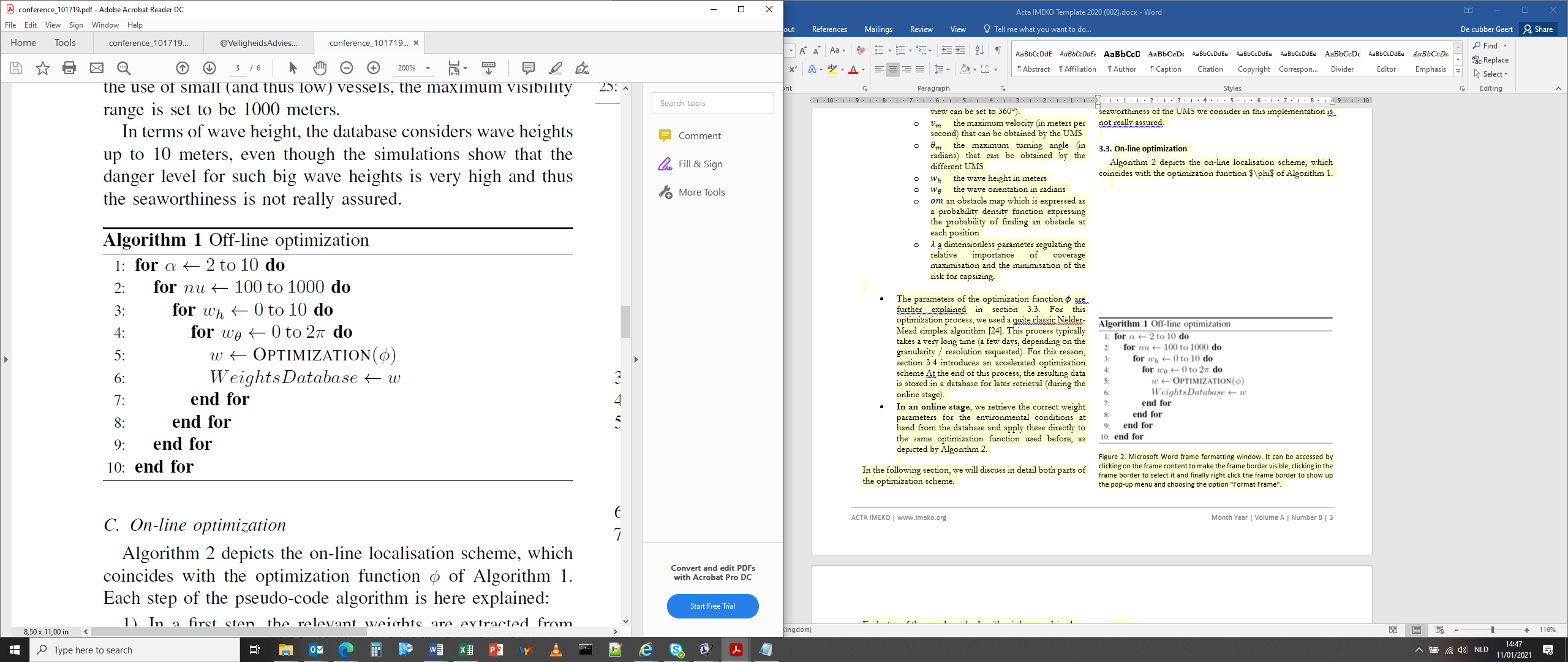
Concerning the Number of Assets , we consider fleets of 2 to 10 unmanned systems. The reason why this does not scale up further is that the methodology relies on an analysis of the localization and sensor data from all other assets. The methodology aims to predict the outcome of moving in a number of directions for each of these assets with is an problem. As a result, increasing the number of assets above 10 leads to prohibitively long computation times, certainly for the non-optimized version of the algorithm (see section 3.4 for a discussion).

Concerning the visibility, due to the fact that we consider the use of small vessels (which thus have a minimal height and cannot look too far over the waves), the maximum visibility range is set to be 1000 meters.



Algorithm 1. Off-line optimization scheme.

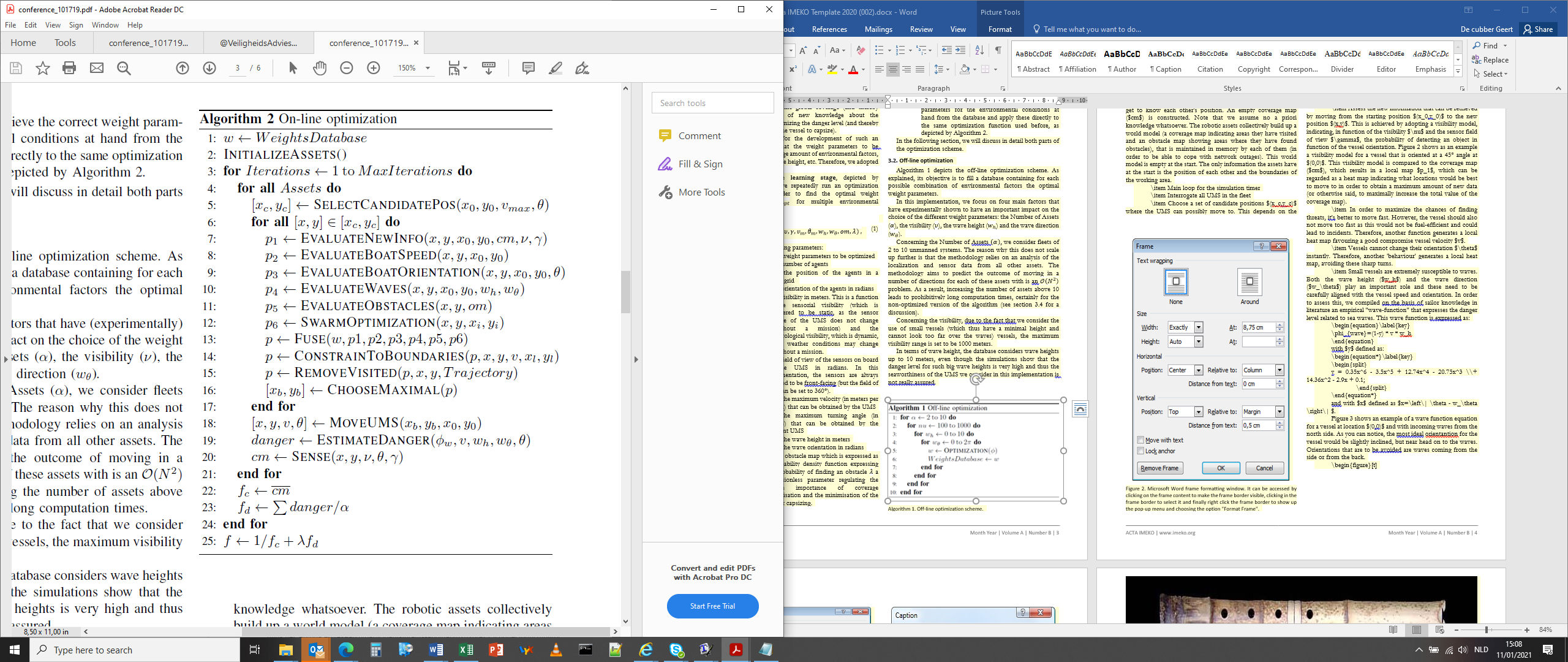
In terms of wave height, the database considers wave heights up to 10 meters, even though the simulations show that the danger level for such big wave heights is very high and thus the seaworthiness of the UMS we consider in this implementation is not really assured.



Algorithm 1. Off-line optimization scheme.

* 1. On-line optimization

Algorithm 2 depicts the on-line localisation scheme, which coincides with the optimization function of Algorithm 1.



Algorithm 1. On-line optimization scheme.

Each step of the pseudo-code algorithm is explained here in detail:

1. In a first step, the relevant weights are extracted from the database. In case no exact match can be found, an interpolation is performed taking into consideration the closest matching conditions in the database built up during the off-line stage.
2. The assets perform an initial communication to get to know each other's position. An empty coverage map () is constructed. Note that we assume no a priori knowledge whatsoever. The robotic assets collectively build up a world model (a coverage map indicating areas they have visited and an obstacle map showing areas where they have found obstacles), that is maintained in memory by each of them (in order to be able to cope with network outages). This world model is empty at the start. The only information the assets have at the start is the position of each other and the boundaries of the working area.
3. Main loop for the simulation timer
4. Interrogate all UMS in the fleet
5. Choose a set of candidate positions where the UMS can possibly move to. This depends on the starting position and orientation and on the maximum velocity of the UMS.
6. Explore all possible candidate positions
7. Assess the new information that can be retrieved by moving from the starting position to the new position . This is achieved by adopting a visibility model, indicating in function of the visibility and the sensor field of view , the probability of detecting an object in function of the vessel orientation. The adopted visibility model assumes a mix of infrared, visual and LIDAR-based sensing and draws upon the heuristically established sensor models established by Lahouli et al. in [25] (for infrared sensors) and Balta et al. in [26] (for visual and LIDAR sensors). Figure 2 shows as an example a visibility model for a vessel that is oriented at a 45° angle at the position . This visibility model is compared to the coverage map (), which results in a local map , which can be regarded as a heat map indicating what locations would be best to move to in order to obtain a maximum amount of new data (or otherwise said, to maximally increase the total value of the coverage map).
8. In order to maximize the chances of finding threats, it's better to move fast. However, the vessel should also not move too fast as this would not be fuel-efficient and could lead to incidents. Therefore, another function generates a local heat map favouring a good compromise vessel velocity .
9. Vessels cannot change their orientation instantly. Therefore, another 'behaviour' generates a local heat map, avoiding these sharp turns.
10. Small vessels are extremely susceptible to waves. Both the wave height () and the wave direction () play an important role and these need to be carefully aligned with the vessel speed and orientation. In order to assess this, we compiled on the basis of sailor knowledge in literature an empirical "wave-function" that expresses the danger level related to sea waves. This wave function is expressed as:

|  |  |
| --- | --- |
|  | (2) |

with defined as:

|  |  |
| --- | --- |
|  | (3) |

Figure 3 shows an example of a wave function equation for a vessel at location and with incoming waves from the north side. As you can notice, the most ideal orientation for the vessel (highest value of ) would be slightly inclined, but near head on to the waves. Orientations that are to be avoided (lowest value of ) are waves coming from the side or from the back.

1. Vessels should not run into detected obstacles. Therefore, the UMS collectively create and share an obstacle map () and steer away from items on this map.
2. It is of no use that multiple agents of the fleet investigate the same area. Therefore the swarm optimization behaviour seeks to keep adequate distances between all of the agents.
3. The different local heat maps are combined into a single map , using the weights that were calculated before (in the off-line step).
4. An extra check is performed in order to ensure that the UMS do not stray away from the designated surveillance area.
5. An extra check is made in order to avoid revisiting recent locations. This is required not only to speed up the convergence, but also to avoid to get stuck in local minima. Therefore, a trajectory memory is maintained and checked for pruning the local heat map .
6. On the local heat map , the optimal position is located.
7. All possible positions are now checked.
8. The vessel is steered towards the optimal position.
9. The danger level for moving to this new position is estimated, based upon the wave function. The danger level is here defined as:
10. The UMS performs an update of its sensing cycle, which will lead to an update of the coverage map, as new information is obtained.
11. End of the iteration over all agents.
12. The mean coverage score is recorded
13. The total (summed) danger score is recorded. For reasons of normalisation, it is divided by the number of assets .
14. End of the temporal loop.
15. We need to maximize the coverage, while minimizing the danger level. Therefore, the objective function to be minimized is defined as:

|  |  |
| --- | --- |
|  | (4) |

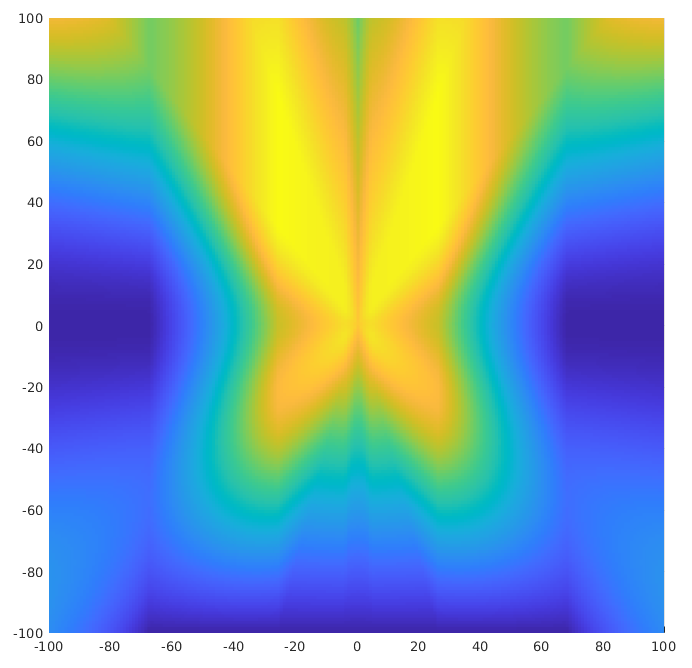


Figure 2. Wave function for a vessel at location and with incoming waves from the north side..

The first term of the objective function ensures that the coverage is maximized, while the second term ensures that the danger level is minimized. The parameter regulates the relative importance accorded to both aspects. This parameter is dependent on the type of vessel used. For smaller UMS, sea waves present a much higher risk, so should be higher. For larger vessels, can be reduced in order to maximize the coverage mapping quicker.

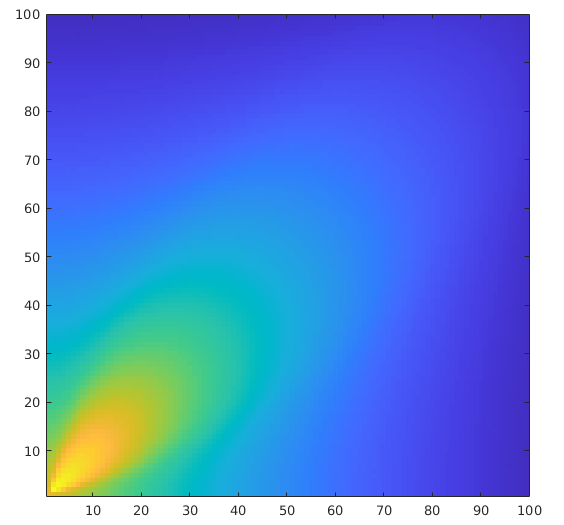


Figure 2. Visibility model for a vessel that is oriented at a 45° angle at the position

* 1. Computational speed optimization

As can be noticed from the definition of Algorithm 2, the computation time will rise exponentially with the number of unmanned systems considered. Not only do all of these assets require a separate evaluation (step 4 in Algorithm 2), but also the complexity of many of the subprocesses (e.g. the swarm optimization) rises with the number of agents. Next to this aspect, the main culprits for the long computation time are the number of candidate positions that has to be evaluated (step 5 in Algorithm 2) and the sometimes high number of iterations required for convergence of the Nelder-Mead simplex algorithm, used for solving Equation 1.

The adopted optimization methodology is mostly geared towards an intelligent pruning of the candidate positions, as follows:

* In a first phase, we select only a very limited subset of the original candidate positions. This is performed by downscaling the selected candidate positions on a lower-scale metric grid.
* In a second phase, the analysis (as described in Algorithm 2, lines 6 to 17) is performed on the downscaled candidate positions.
* Around the winning position, a local subgrid is defined at the original resolution.
* For all candidate positions within this local subgrid, the analysis (as described in Algorithm 2, lines 6 to 17) is performed again and a final winning position is chosen.

The double loop may seem to add extra complexity and computation time, but in practice, it avoids the evaluation of many candidate positions that do not have a viable chance of being selected. Indeed, as the local maps are mostly continuous functions, it makes sense to evaluate them first at a lower resolution and then to scale up. Furthermore, we optimized the convergence settings of the Nelder-Mead simplex algorithm to match the candidate position pruning approach.

1. Validation
   1. Quantitative validation

For the validation of the proposed approach, we chose the application of the surveillance of the Belgian off-shore windmill parks, which means that an area of around 10 km x 30 km needs to be patrolled. However, the proposed methodology would for example also be very useful for a maritime search and rescue scenario [27] or a fishery control scenario.

In order to validate the methodology, we compared it to five state of the art solutions:

* **Random search**, where each agent adopts a completely random movement pattern
* **Distributed random search**, where the search area is subdivided in equal parts and each agent adopts a random search pattern within the designated subzone
* **Lawnmower search**, where each agent uses a movement pattern typically adopted by robotic lawnmowers: moving in straight lines and turning a random amount of degrees when coming near the boundaries
* **Distributed lawnmower search**, where the search area is subdivided in equal parts and each agent adopts a lawnmower search pattern within the designated subzone
* **Distributed Greek patterns**. This is the search and surveillance approach typically adopted by manned vessels and it has been proven to be very efficient for rapid area coverage. Moreover, by subdividing the search area and distributing the search tasks among multiple agents, this approach is quite well suited for maritime coverage optimization. Figure 4 shows an example of the Greek pattern.

Figure 3. Microsoft Word caption insertion window. It can be accessed by right clicking on the picture or table and selecting "Insert Caption" from the pop-up menu.

One disadvantage of all these state of the art approaches is that they do not take into consideration the danger posed by the waves on the vessel, which is an integral part of the proposed solution.

In order to further validate the optimization scheme, we compared also the results from a non-optimized, **nominal** version (using a static initial guess for the weights parameter ) with the optimized approach.

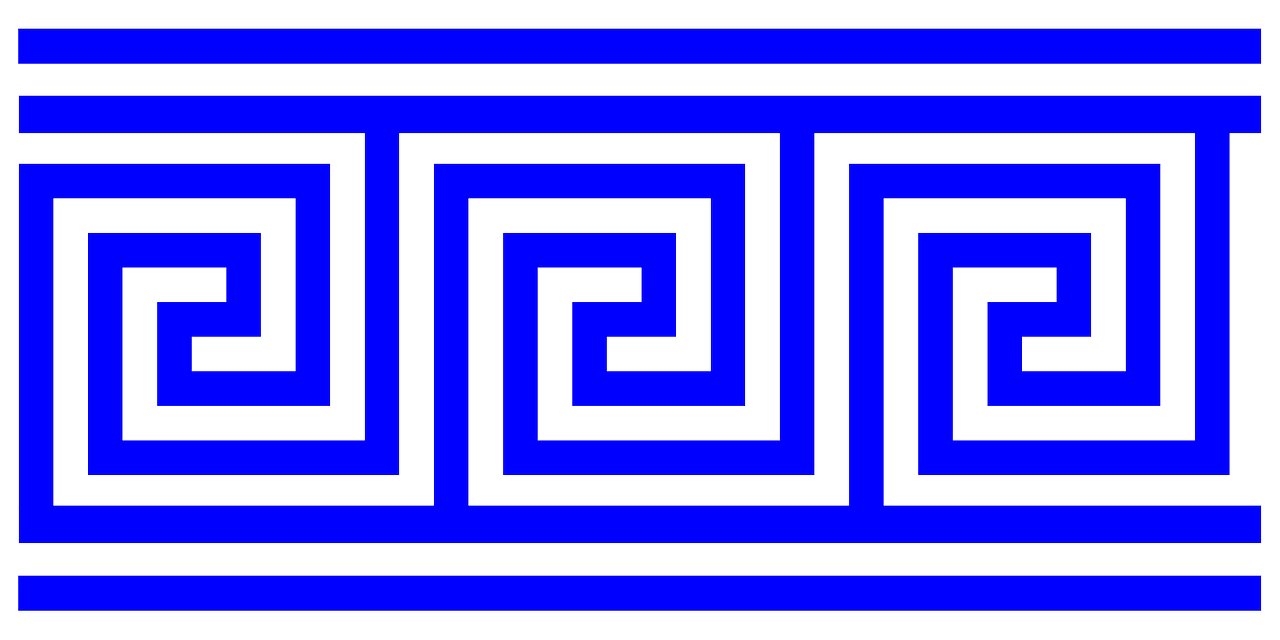


Figure4. Example of the Greek pattern.

Figure 4. Microsoft Word frame formatting window. It can be accessed by clicking on the frame content to make the frame border visible, clicking in the frame border to select it and finally right click the frame border to show up the pop-up menu and choosing the option "Format Frame".

C:\Users\De Cubber.G\AppData\Local\Temp\Temp1_Conference-LaTeX-template_10-17-19.zip\Conference-LaTeX-template_10-17-19\coverage.tif

Figure 5. Evolution of the relative coverage of a surveillance area using seven different approaches.

Figure 5 presents the results in terms of coverage in a simulation with 4 agents present. It can be clearly noted that the presented approach (denoted as **optimal** and indicated in dark red) achieves the highest overall coverage. Without using weight optimization, the Distributed Greek Patterns approach outperforms our baseline nominal approach here. All other approaches achieve a performance which is far lower. It can also be noticed that the coverage results are not always monotonically increasing. This can be explained by the fact that at each iteration, the existing coverage data is “aged” (in practice, the coverage map is multiplied by 0.99) in order to represent the fact that older data has become less valid. The result is that with a limited number of agents, it becomes very difficult to maintain a high overall coverage score.

These results can be expected, as the random search and lawnmower search approaches are quite simplistic methodologies, whereas the Greek Patterns has a proven track record for these kinds of applications. Still, using weight optimization, our proposed methodology succeeds in achieving a higher coverage score.

However, the major strength of our approach can be witnessed by also considering Figure 6, which indicates the danger level of executing a mission using each of the approaches. The blue portion of the bar chart indicates the mean danger level, whereas the red portion indicates the maximum danger level attained during a particular mission. Obviously, both are important to assess the risk of incidents. It can be clearly noted that both the nominal and the optimal proposed methodology achieve a danger level that is significantly lower than the other approaches. Moreover, for the optimal approach, there is little difference between the mean and the maximum danger levels, indicating that the methodology succeeds in keeping the risk at a constant and low level.

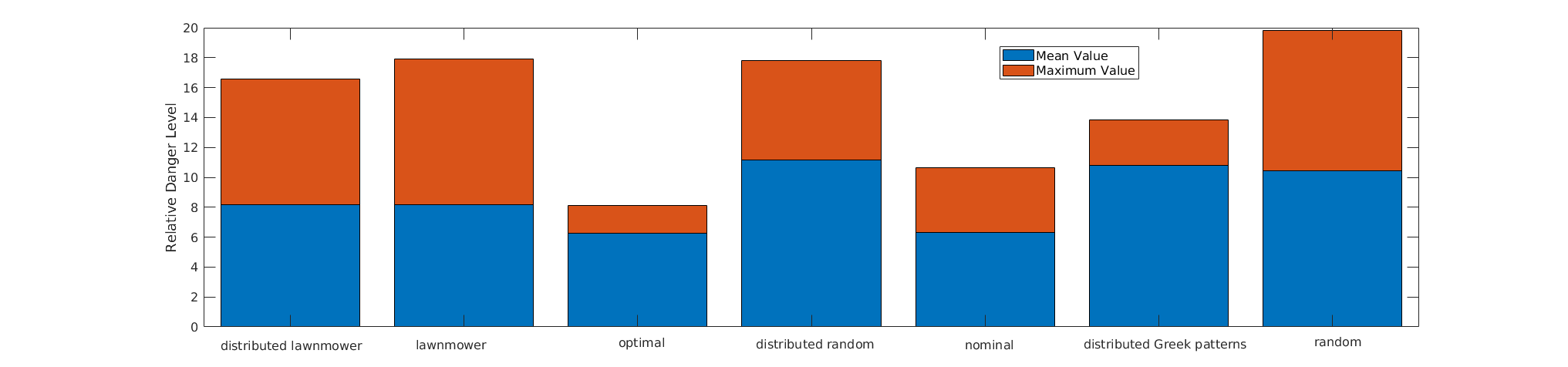


Figure 6. Relative danger level for executing a maritime surveillance mission using seven different approaches.

* 1. Scaling and timing

In order to assess the effects of the speed optimization methodology that was explained in section 3.4, Figure 7 shows the evolution of the processing time, relative coverage and relative danger level for the proposed approach with and without applying the candidate location pruning methodology.

As is clearly indicated on Figure 7, the computation time is drastically reduced by the incorporation of the candidate position pruning methodology. In general, the accelerated approach is about 9 times faster than the baseline approach. This clearly shows that the proposed methodology of first analysing a limited set of points and then focusing a detailed analysis of a dense point set only on a small area has a highly beneficial impact on the global processing time. This enables as well to use the methodology for an increased amount of assets, even though the global algorithm still scales slightly more than .

An important aspect to assess was of course whether there would be no loss of quality using the accelerated approach. In order to evaluate this, we recorded also the coverage mapping and danger levels for both methodologies and for different numbers of agents, as also shown on Figure 7.

To our surprise, the accelerated approach performed even better than the baseline approach, even though the differences are certainly not very large. This may seem counter-intuitive at first sight, because the accelerated approach evaluates less candidate positions and has thus – compared to the baseline approach – always a higher risk of being trapped in some local minima.

We investigated this phenomenon and found out that it is caused by better convergence properties of the accelerated approach. Indeed, as the accelerated approach requires fewer time-consuming local map evaluation steps, the Nelder-Mead simplex optimization algorithm achieves to obtain (slightly) lower values for the optimization function when using the accelerated method. The ‘side-effect’ of the speed optimization as described in section 3.4 is thus also an improved (higher) coverage mapping and a slightly improved (lower) danger level.

C:\Users\De Cubber.G\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\4337CZNV\timing.tif

Figure 7. Evolution of the processing time, mean relative coverage and mean relative danger level for the proposed approach with and without applying the candidate location pruning methodology.

1. Conclusions

In this paper, we have presented an approach towards distributed coverage optimization for a maritime surveillance application. The approach is based upon a mix of off-line learning and on-line optimization. In order to remedy the traditional problem related to the excessive processing time for multi-agent global planning methodologies, we have also proposed an approach for multi-scale selection of the candidate locations.

The methodology was validated by comparing it in simulation to multiple state of the art approaches. This comparison shows that the proposed approach performs well in terms of coverage mapping and very well in terms of minimizing the capsizing danger for the small unmanned vessels. Moreover, the validation of the performance of the accelerated approach on multi-agent systems showed that the computation time can be drastically reduced, while also increasing the coverage mapping performance.

A next step will be to implement and test the system on real-life Unmanned Maritime Systems, that are planned to be sent out to patrol the Belgian territorial waters.

Acknowledgement

The research presented in this paper has been funded by the Belgian Royal Higher Institute for Defense, in the framework of the DAP19/08 (MarSur) project and by the Flemish Agency for Innovation and Entrepreneurship, in the framework of the Blue Cluster project SSAVE (HBC.2019.0045).

References

1. L. Snidaro, I. Visentini, and K. Bryan, Fusing uncertain knowledge and evidence for maritime situational awareness via markov logic networks, Information Fusion, vol. 21, pp. 159 – 172, 2015.
2. E. Schwarz, D. Krause, H. Daedelow, and S. Voinov, Near real time applications for maritime situational awareness, in Deutscher Luft- und Raumfahrtkongress 2015, pp. 1–5, November 2015.
3. R. Csernatoni, Constructing the eu’s high-tech borders: Frontex and dual-use drones for border management, European Security, vol. 27, no. 2, pp. 175–200, 2018. M. Fazio, S. L. Rota, Metrology on stamps, Phys. Educ. 30 (1995) pp. 289-297.
4. D. Doroftei, A. Matos, G. De Cubber, Designing Search and Rescue Robots towards Realistic User Requirements, Advanced Concepts on Mechanical Engineering (ACME), 2014,
5. D.S. López, G. Moreno, J. Cordero, J. Sanchez, S. Govindaraj, M.M. Marques, V. Lobo, S. Fioravanti, A. Grati, K. Rudin, Konrad, M. Tosa, A. Matos, A. Dias, A. Martins, J. Bedkowski, H. Balta, G. De Cubber, Interoperability in a heterogeneous team of search and rescue robots, Search and Rescue Robotics - From Theory to Practice, 2017, Rijeka, InTech
6. C. Yan, T. Zhang, Multi-robot patrol: A distributed algorithm based on expected idleness. International Journal of Advanced Robotic Systems. 13, 2016.
7. D. Aksaray, K. Leahy, C. Belta, Distributed Multi-Agent Persistent Surveillance Under Temporal Logic Constraints, IFAC-PapersOnLine, Volume 48, Issue 22, Pages 174-179, 2015.
8. M. Baseggio, A. Cenedese, P. Merlo, M. Pozzi, L. Schenato, Distributed perimeter patrolling and tracking for camera networks. 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, 2010, pp. 2093-2098.
9. G. De Cubber and R. Haelterman, Optimized distributed scheduling for a fleet of heterogeneous unmanned maritime systems, in International Symposium for Measurement and Control in Robotics (ISMCR), 2019.
10. M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, Swarm robotics: a review from the swarm engineering perspective, Swarm Intelligence, vol. 7, pp. 1–41, Mar 2013.
11. M. Dorigo and T. Stutzle, Ant Colony Optimization: Overview and Recent Advances, pp. 311–351. Cham: Springer Int. Publishing, 2019.
12. A. Pierson, L. C. Figueiredo, L. C. Pimenta, and M. Schwager, Adapting to sensing and actuation variations in multi-robot coverage, Int. J. of Robotics Research, vol. 36, no. 3, pp. 337–354, 2017.
13. L. C. A. Pimenta, M. Schwager, Q. Lindsey, V. Kumar, D. Rus, R. C. Mesquita, and G. A. S. Pereira, Simultaneous Coverage and Tracking (SCAT) of Moving Targets with Robot Networks, pp. 85–99. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.
14. C. Gao, Y. Kou, Z. Li, A. Xu, Y. Li, and Y. Chang, Optimal multirobot coverage path planning: Ideal-shaped spanning tree, Mathematical Problems in Engineering, vol. 2018, p. 3436429, Sep 2018.
15. A. C. Kapoutsis, S. A. Chatzichristofis, and E. B. Kosmatopoulos, Darp: Divide areas algorithm for optimal multi-robot coverage path planning, J. of Intelligent & Robotic Systems, vol. 86, pp. 663–680, Jun 2017.
16. J. Cortes, Coverage optimization and spatial load balancing by robotic sensor networks” IEEE Transactions on Automatic Control, vol. 55, no. 3, pp. 749–754, 2010.
17. B. Xin, G. Gao, Y. Ding, Y. Zhu, and H. Fang, Distributed multi-robot motion planning for cooperative multi-area coverage, in Int. Conf. on Control Automation (ICCA), pp. 361–366, 2017.
18. J. Valente, J. Del Cerro, A. Barrientos, D. Sanz, Aerial coverage optimization in precision agriculture management: A musical harmony inspired approach, Computers and Electronics in Agriculture, vol. 99, pp. 153 – 159, 2013.
19. T. Fabbri, R. Vicen-Bueno, R. Grasso, G. Pallotta, L. M. Millefiori, and L. Cazzanti, Optimization of surveillance vessel network planning in maritime command and control systems by fusing metoc ais vessel traffic information, in OCEANS 2015 - Genova, pp. 1–7, 2015.
20. X. Han, S. Li, X. Pang, Coverage Optimization Algorithm of Wireless Sensor Network. In: Jin D., Lin S. (eds) Advances in Future Computer and Control Systems. Advances in Intelligent and Soft Computing, vol 159. Springer, Berlin, Heidelberg.
21. S. Mirjalili, S. M. Mirjalili, A. Lewis, Grey Wolf Optimizer, Advances in Engineering Software, Volume 69, 2014, Pages 46-61.
22. Z. Wang, H. Xie, Z. Hu, D. Li, J. Wang, W. Liang, Node coverage optimization algorithm for wireless sensor networks based on improved grey wolf optimizer, Journal of Algorithms & Computational Technology, January 2019
23. D. Doroftei, G. De Cubber, E. Colon, and Y. Baudoin, Behavior based control for an outdoor crisis management robot, in Int. Ws. on Robotics for Risky Interventions and Environmental Surveillance, 2009.
24. J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, Convergence properties of the nelder-mead simplex method in low dimensions., SIAM J. Optimization, vol. 9, no. 1, pp. 112–147, 1998.
25. I. Lahouli, E. Karakasis, R. Haelterman, Z. Chtourou, G. De Cubber, A. Gasteratos, R. Attia, Hot spot method for pedestrian detection using saliency maps, discrete chebyshev moments and support vector machine, IET Image Processing, 12, 7, 1284-1291, 2018, IET Digital Library
26. H. Balta, J. Velagic, G. De Cubber, W. Bosschaerts, B. Siciliano, Bruno, Fast Statistical Outlier Removal Based Method for Large 3D Point Clouds of Outdoor Environments, 12th IFAC Symposium on robot control - SYROCO 2018
27. G. De Cubber, D. Doroftei, K. Rudin, K. Berns, D. Serrano, J. Sanchez, S. Govindaraj, J. Bedkowski, and R. Roda, Search and rescue robotics - from theory to practice, 2017.