# Optimisation of underwater acoustic sensor networks

Ronan Serre, David Dellong, Myriam Lajaunie, Hélène Pihan-Le Bars, Jean-Michel Boutonnier, Mathis Hamelotte, Laurent Lemarchand, Catherine Dezan

25-29 August 2024
inter.noise
NANTES FRANCE

# Optimisation of underwater acoustic sensor networks

Ronan Serré[1], David Dellong, Myriam Lajaunie, Hélène Pihan-Le Bars, Jean-Michel Boutonnier
Service Hydrographique et Océanographique de la Marine (Shom)
13 Rue de Châtellier, 29200 Brest, France

Mathis Hamelotte, Laurent Lemarchand[2], Catherine Dezan
Lab-STICC UBO
6 Avenue Le Gorgeu, 29200 Brest, France

**ABSTRACT**

*This contribution describes two methodologies based on evolutionary algorithms and Linear Programming to determine optimal configurations of underwater acoustic sensor networks maximising source detection capabilities, while accounting for the overall physical state of the marine environment. The optimisation framework is evaluated with the different optimisation algorithms. The relevance of various objectives, or metrics, are discussed, as well as specific search operators for crossover and mutation.*

## 1. INTRODUCTION

The oceans are the scene of intense economical and industrial activities. It is becoming even more competitive with recent military activities and the upcoming development of marine renewable energies [1]. All of these activities have a significant impact on marine ecosystems and must be carefully evaluated for any maritime planning. The marine environment is also a very effective medium for the propagation of acoustic waves over long distances, depending on the frequency, the source level, the static (bathymetry and seabed nature) and the dynamic (hydrology) oceanographic conditions [2]. Such conditions make acoustic sensors primary candidates for underwater investigations and the distribution of heterogeneous sensors with various sensitivities, frequency bandwidths or positioning can be optimized for detection, localization or monitoring purposes. In this context, Cheng *et al.* for instance, have used a binary particle swarm optimisation algorithm [3]. A list of objectives, or metrics, is usefully outlined however arranged in a fitness function, as a consequence of the optimisation algorithm being single-objective. The use of a fitness function might be cost effective but removes the possibility to emphasize a particular objective after the computation of the Pareto front as offered by multi-objective algorithm. Such approaches have been used in acoustics with for instance Galiana Nieves *et al.* [4] with a multi-objective genetic algorithm to optimise various parameters of a design problem and Huszty [5], with black-box optimisation algorithms to find optimum solutions for shape and placement related to problems in room acoustic applications. The research presented hereinafter is part of the RESSACH project, coordinated by Shom with research institutes IRENaV and Lab-STICC as partners. The RESSACH project includes numerical simulations of the environment

[1]ronan.serre@shom.fr

[2]laurent.lemarchand@univ-brest.fr

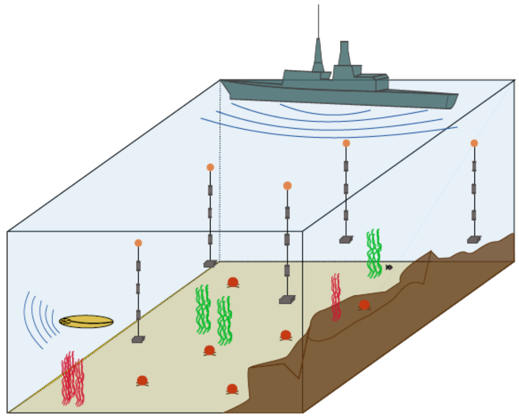


Figure 1: Schematic view of the mixed underwater acoustic networks.

with acoustic propagation, optimisation of heterogeneous underwater acoustic sensor networks for detection purposes and classification of acoustic events. Similarly to the work published by Cheng *et al.* [3], the objective of an optimisation of heterogeneous underwater acoustic sensor networks is finding optimal configurations of networks composed of various sensors such as, for example, ocean-bottom seismometers, acoustic moorings (referred to as anchors) equipped with hydrophones (referred to as nodes) and underwater autonomous vehicles, as illustrated in figure 1. In this context, the scope of this work is the use and evaluation of an optimisation framework for underwater acoustic sensor networks (UWASN) based on simplified test scenarios described in the next section.

## 2. METHODOLOGY

To evaluate the optimisation framework used for UWASN configurations, three scenarios have been designed based on the objective and constraint used. For the three scenarios, the optimisations were performed using five anchors (with each anchor equipped with one node) of a single type, at one single immersion depth (100 m), and with a frequency bandwidth representative of industrial applications (between 100 Hz and 10 kHz), on a domain discretised into $102 \times 60$ points. Scenario 1 is designed with the objective to maximise the number of points covered by at least one node. Scenario 2 is designed with the objective to maximise the number of points covered by at least three nodes. Scenario 3, similarly to scenario 1, is designed with the objective to maximise the number of points covered by at least one node with addition of a non-collinearity constraint.

### 2.1. Metrics

In the different scenarios, the objective function to be maximised in the optimisation framework is defined as the number of points that are covered by at least a certain number of nodes. A point is labelled as covered by a node if for that node, the value of the data input (the transmission losses, see section 2.2) at that point is below 100 dB (with a reference pressure underwater of 1 $\mu$Pa), in the frequency bandwidth of the node. The points covered by a node then represent all the positions in space at which a node can detect a source. In the third scenario, a constraint function is added, designed to avoid collinearity on nodes in the UWASN. Constraint is assigned to a network if at least one sub-network of three anchors (among any combination possible) is aligned, or collinear. A sub-network is collinear if one of the angles constitutive of the triangle formed by the three anchors on the longitude-latitude plane (that is, ignoring the vertical dimension represented by the immersion) is below 15°. This parameter is purely arbitrary and not based on any physical analysis but simply acts as a control parameter. On real networks deployed at sea, it might be in fact necessary to control collinearity (either by avoiding it or by enforcing the alignment of anchors) for example for source localisation purposes.

### 2.2. Data input

The input data are the transmission losses obtained from underwater acoustic simulations performed with an in-house simulation tool and stored in a file. With reference to an input source level, the transmission loss could translate into a signal-to-noise ratio. The simulation tool combine parabolic methods for frequencies lower 300 Hz (using RAM software [6]) and ray tracing methods for frequencies higher than 300 Hz (using Bellhop software [7]). The transmission losses are computed only up to 300 km away from the point at which the acoustic simulation is being run. In the simulation tool, the background noise was estimated from a statistical distribution of maritime traffic based on AIS database while the bathymetry, the seabed nature and the sound speed profiles were taken from the Shom database with a five minute-arc resolution. Each point of the discretised domain can be considered as a potential receptor and in the same way, every point of the discretised domain around a receptor can be considered as a potential source of noise.

### 2.3. Design space

The area selected for the evaluation is set in the Bay of Biscay, between longitude 8.95° W and longitude 0.54° W and between latitude 43.54° N and latitude 48.46° N, approximately. It is discretised with 102 points in longitude and 60 points in latitude, corresponding to an average cell size of 13 km in longitude and 18 km in latitude. Six deployment zones have been designed which define the points constituting the design space for the optimisation algorithms. Each of these are a sub-domain of the $102 \times 60$ grid. The deployment zones are depicted in figure 2 with bathymetry and sound speed profiles extracted at the centre of each areas of the deployment zones. Two deployment zones are in deep water ("DZ0" and "DZ1"), two on the continental shelf ("DZ2" and "DZ3") and two in the transition zone ("DZ4" and "DZ5"). Some of the deployment zones are defined as one single area ("DZ0", "DZ2" and "DZ4") while some others are defined as three ("DZ1" and "DZ3") or two ("DZ5") distinct areas.

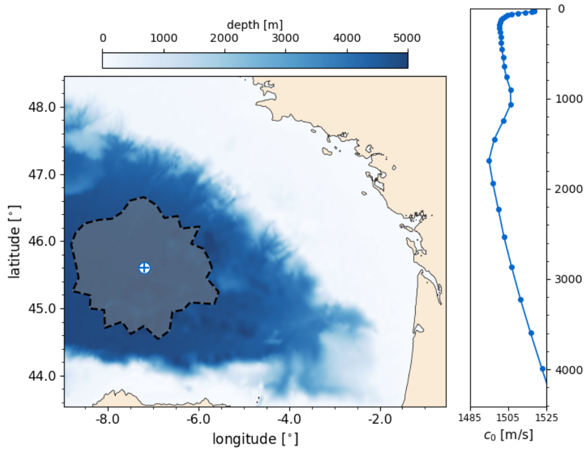
## 3. OPTIMISATION FRAMEWORK

The optimisation of the acoustic network relies on the metrics defined in subsection 2.1. Either combinatorial optimisation techniques can be used to compute optimal solutions with the best values for the metrics, or meta-heuristics such as evolutionary algorithms can be employed to obtain *good* solutions with objective function values close to the optimal ones. Furthermore, if the metrics are antagonist, it is not possible to optimise them simultaneously, and trade-offs are to be found. We use two classes of approaches for optimisation:

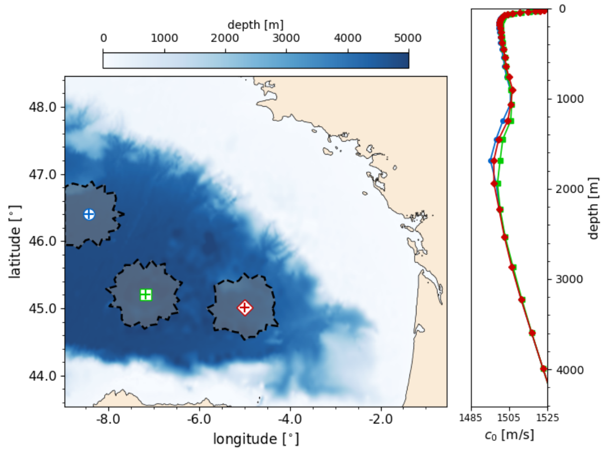
- Mixed Integer Linear Programming (MILP) [8] allows to solve optimally single objective problems. The computational effort is sometimes important for large testcase instances, but even in these cases, a bound on the optimal value can be obtained.

- Multiple Objective Evolutionary Algorithms (MOEA) [9] such as NSGA-II [10] are flexible meta-heuristics allowing simultaneous optimisation of more than one objective. The counterpart of this flexibility is the fact that optimal solutions are not guaranteed.

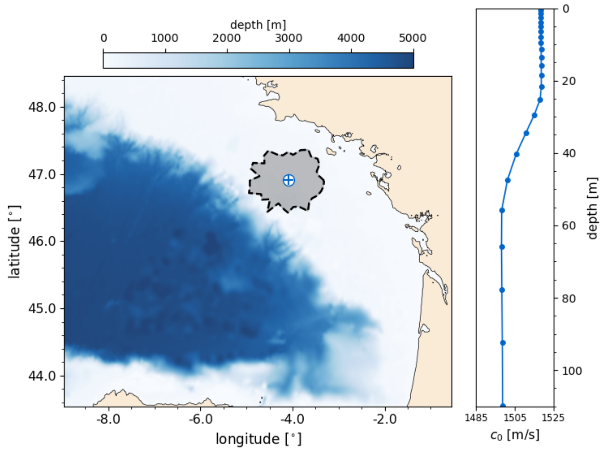
Applications of these classes of methods are detailed hereinafter in the context of underwater acoustic networks optimisation.



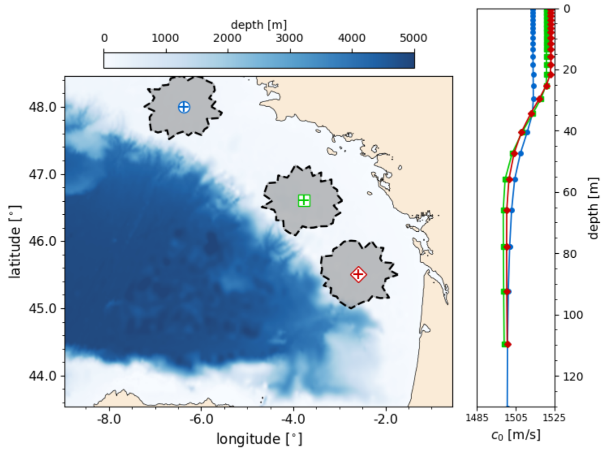

(a) test case "DZ0"



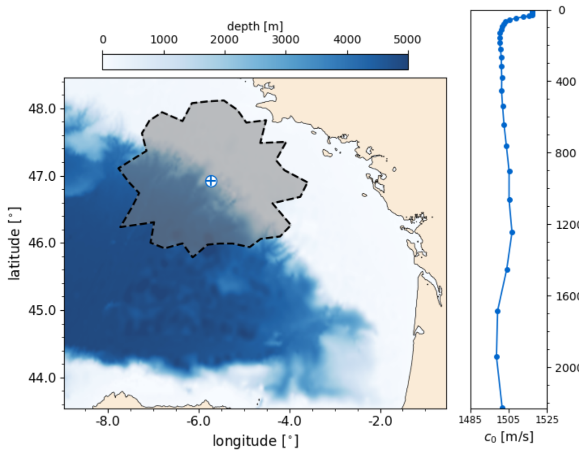

(b) test case "DZ1"



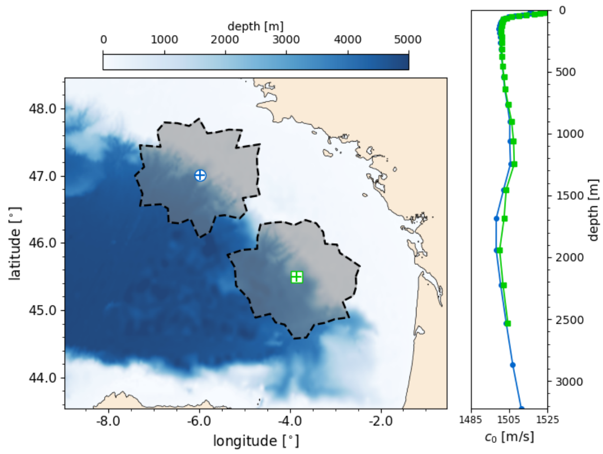

(c) test case "DZ2"




(d) test case "DZ3"




(e) test case "DZ4"




(f) test case "DZ5"

Figure 2: Deployment zones, bathymetry and sound speed profiles (corresponding colors and marker symbols between the sound speed profiles and their location on the map).

### 3.1. MILP approaches

MILP (Mixed Integer Linear Programming) is a mathematical optimization method that solves combinatorial problems where decision variables are either integers or real values, while satisfying linear constraints and optimizing linear objective functions. It is a powerful tool in operations research and mathematical modeling, enabling precise representation of discrete decision-making processes. MILP efficiently explores the solution space to find optimal solutions. We propose the following MILP formulation tailored to the single-objective version of the underwater acoustic networks problem. This MILP formulation is to be solved using standard solvers such as CPLEX [11]. Data used by the MILP:

- $I$: the list of all influence points.
- $D$: the list of deployment points.
- $M$: the list of all types of sensors.
- $P$: $P \subset I$, representing the list of reachable points.
- $C_{ij}$: states the list of covered points where $\forall i \in P$, $\forall j \in D$, $C_{ij} = 1 \Leftrightarrow j$ covers $i$.

The goal of this MILP formulation is to maximize the covered reachable area while ensuring each reachable point is covered by at least $n$ sensors, and limiting the number of deployed sensors of each model $m \in M$ (or type) to a predefined value $s_m$. Two sets of decision variables are used. $x_{jm}$ represents a binary decision variable indicating whether a sensor of model $m \in M$ is placed at deployment point $j \in D$, where $x_{jm} = 1$ if a sensor $m$ is deployed at that point. These variables represent the solution network found. Additionally, $y_i$ is introduced as a binary decision variable, with $y_i = 1$ indicating that reachable point $i$ is covered by at least $n$ sensors. These variables allow to define the covering objective function:

$$\text{maximize} \sum_{i \in P} y_i \tag{1}$$

This objective function (Equation 1) is associated to the following constraints. Equation 2 ensures that a point is covered by $n$ sensors and Equation 3 limits the number of sensors of different models according to their availability:

$$\forall i \in P \qquad n \cdot y_i \leqslant \sum_{j \in D, m \in M} C_{ij} \cdot x_{jm} \tag{2}$$

$$\forall m \in M \qquad \sum_{j \in D} x_{jm} \leqslant s_m \tag{3}$$

If required, non collinearity of sensors can be guaranted by the following constraints:

$$\forall \text{ colinear points } j_1, j_2, j_3 \in D \qquad \sum_{m \in M} x_{j_1 m} + \sum_{m \in M} x_{j_2 m} + \sum_{m \in M} x_{j_3 m} \leqslant 2 \tag{4}$$

Note that Equation 4 could lead to a very large number of constraints ($C^3_{|D|}$) when the size of the deployment area increases. These constraints are thus not included when solving the problem at first. Non-collinearity is checked on the solution and if three sensors are collinear in this solution, the corresponding constraint is added to the MILP. The corresponding points are removed from the design space and the MILP is solved again. This process is repeated as long as the solution includes collinear nodes.

$$\begin{cases} P_1 = [i_{11}, i_{12}, i_{13}, i_{14}, i_{15}] \\ P_2 = [i_{21}, i_{22}, i_{23}, i_{24}, i_{25}] \end{cases} \rightarrow \begin{cases} C_1 = [i_{11}, i_{12}, i_{13}, i_{24}, i_{25}] \\ C_2 = [i_{21}, i_{22}, i_{23}, i_{14}, i_{15}] \end{cases}$$

Figure 3: Schematic representation of the in-house crossover process developped and integrated in the pymoo framework. Two parents $P_1$ and $P_2$ exchange part of their genes to produce two children $C_1$ and $C_2$.

| **Parameter** | **Value** |
|---|---|
| population size | 40 |
| number of offsprings | 20 |
| number of generations | 30 |
| mutation probability | 80 % |
| mutation rate | 50 % |

Table 1: Parameters used in the MOEA approach.

### 3.2. MOEA approaches using NSGA-II

Evolutionary algorithms (EA) are meta-heuristics that explore a solution space by maintaining a set of solutions (a population) and iteratively modifying these solutions using search operators. Unary operators are called mutations and binary operators are called crossovers. The former transforms a solution into a new one. The later produce a set of new solutions (offspring) by mixing two parents solutions. All operators try to produce solutions with better objective function values. Single objective EA extract the best known solution from a final population after a set of iterations (generations). MOEA maintain diversity of solution values in the population and extract at the end a set of goods trade-off solutions that do not dominate each other. A solution dominates when all of its objective function values are better. EA and MOEA are specialised for a given optimisation problem by customising existing or inventing new crossover and mutation operators, acting on a representation of each solution. For the MOEA approach, the Non-dominated Sorting Genetic Algorithm II [10] (NSGA-II) was chosen based on its popularity and availability in many MOEA frameworks, such as pymoo [12], and its constraint handling capabilities. However, specific *ad-hoc* crossover and mutation operators have been designed to keep a better control over the solutions and make sure theey stay integers and inside the variable space. The crossover operator takes two parents, randomly selects a position in the variable space and exchanges the second part of the variables. An illustration of the process is depicted in figure 3. The mutation operator randomly selects a mutation choice (among which "replace" a gene, "add" or "substract" a number to a gene, or "do nothing") with a probability. The part of the population that is affected by the mutations is defined by the mutation rate. Based on preliminary convergence tests, the parameters used in the MOEA approach are outlined in table 1.

## 4. RESULTS ANALYSIS

The optimal solutions obtained from both the MILP and the MOA with NSGA-II approaches are compared and analysed in this section. One parameter used in the analysis is the gap which represents the loss of quality of the NSGA-II approach compared to the MILP approach. In most cases the optimal solution is obtained by MILP in less than 30 minutes, which is our execution time limit and was set according to the computational efforts deployed with the NSGA-II approach. In terms of computation times, their direct comparison is not straightforward. The two methodologies were computed on different machines and in the case of the MOEA approach, the transmission losses file is read at each generation while it is read only once in the MILP approach.

In scenario 1, MILP computes the optimal solution for all test cases in less than 1 s while NSGA-II runs for several minutes. For the benchmark, an average of 30 % loss of quality is observed in terms of coverage ratio for NSGA-II. The MILP approach is clearly preferable for this single-objective version of the optimisation problem. In order to explore trade-offs between the covering properties of a solution network and its size, we take benefit from the NSGA-II multi-objective optimisation capabilities. Figure 4 shows the Pareto fronts obtained while optimising both the number of sensors and the number of covered points. We applied this technique to the test case "DZ2" with both NSGA-II and MILP. The covering metric is expressed here as the minimisation of the percentage of reachable points that are not covered by the optimised solution network. Thus, most interesting networks are those close to $(0,0)$ point in figure 4. In order to obtain the exact Pareto front with the MILP approach, the problem is solved by varying the number of sensors limit up to obtain the full coverage of the reachable points when solving associated MILP series ($\epsilon$-constraint method [9]). The standard hypervolume metric [9] was used for comparing both Pareto fronts quality values and the loss of quality for the NSGA-II front is of 22.87 % as compared to the exact front value obtained with MILP. The MILP approach results show that 67 sensors are used for covering all reachable points and that only 31 of them are sufficient for covering around 90 % of the reachable area.

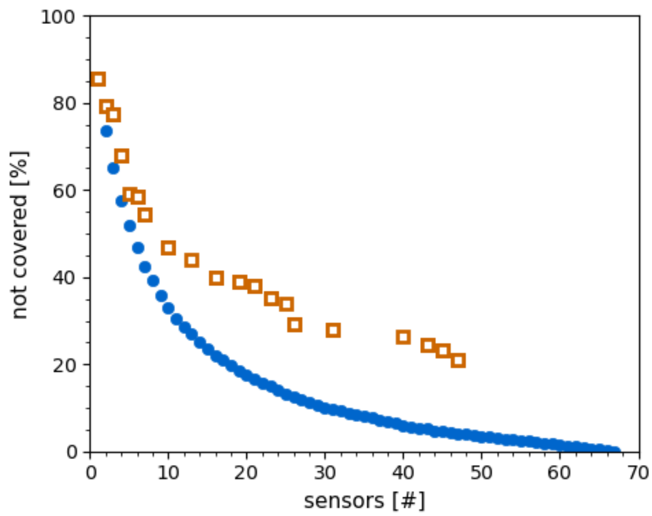



Figure 4: Pareto fronts for test case "DZ2" in a corresponding multi-objective version of scenario 1.

In scenario 2, with points covered by at least three nodes, the computational time for MILP reaches the execution time limit of 30 minutes while the NSGA-II execution times remain around 15 minutes. The results are summarised in table 2. MILP values are optimal except when a gap to best known bound on optimal value is specified in % in the covered points column of table 2. The average gap between the two approaches for the testcases in Table 2 is of 36.57%. However, with its limited execution times, NSGA-II becomes a more favourable option.

In scenario 3, with points covered by at least one node with a non-collinearity constraint, the same computational costs were observed than for scenario 1. The results are summarised in table 3. The number of forbidden collinear sub-networks, or forbidden triplets, is added to table 3.

| Test Case | Total points | Covered by MILP | Covered by NSGA-II | Gap (%) |
|---|---|---|---|---|
| DZ0 | 623 | 164 | 136 | 17.1 |
| DZ1 | 438 | 164 | 59 | 64.0 |
| DZ2 | 152 | 67 | 53 | 20.9 |
| DZ3 | 458 | 67 (4.17 %) | 50 | 25.4 |
| DZ4 | 903 | 80 (20.47 %) | 49 | 38.8 |
| DZ5 | 923 | 379 (17.97 %) | 37 | 53.2 |

Table 2: Comparison between MILP and NSGA-II on the six deployment zones for scenario 2. Gap to best known bound on optimal value specified in % in the column of the points covered by MILP.

A large number of sub-networks are eliminated when the deployment zones are made of multiple areas.

| Test Case | Covered by MILP | Forbidden Triplets | Covered by NSGA-II | Gap (%) |
|---|---|---|---|---|
| DZ0 | 411 | 1 | 305 | 25.8 |
| DZ1 | 386 | 159 | 236 | 38.9 |
| DZ2 | 226 | 15 | 173 | 23.5 |
| DZ3 | 277 | 1134 | 124 | 55.2 |
| DZ4 | 382 | 11 | 221 | 42.1 |
| DZ5 | 338 | 226 | 202 | 40.2 |

Table 3: Comparison between MILP and NSGA-II on the six deployment zones for scenario 3.

The gaps of quality for single objective of covering between the MILP and NSGA-II for the three scenarios are summarised in figure 5. No clear correlation can be found with the bathymetry (whether in deep water or on the continental shelf), or the number of areas composing the deployment zones. To illustrate some concluding remarks and in order to see some results from an underwater acoustics perspective, figure 6 shows the covered points and the optimal node positions found by MILP and NSGA-II in the case of the test case "DZ5" for the three scenarios. The depicted covered areas show the extent at which a node can detect a potential source if the signal transmission loss does not exceed 100 dB. Interestingly, optimal configurations of nodes are found in most cases on or near the continental shelf. Physical interpretation should be done carefully as immersion was imposed on the anchor nodes but a possible explanation can be outlined. Most of the noise sources accounted for in the transmission loss calculations come from ships towards the Ouessant traffic rail and near the surface. The occurrence of a surface channel with bottom reflections may sustain acoustic propagation over longer distances in the relatively shallow waters of the continental shelf, for immersions between 100 m and the surface. The proposed optimal configurations of the UWASN on the continental shelf is then probably better suited to detect sources which mainly consist in ship noise.

## 5. CONCLUSION

This paper presents two approaches for optimising underwater acoustic sensor networks (UWASN) in order to maximise detections. Collinearity between the sensors can be controlled to improve source localisation capabilities. In the context of this contribution, the main objective function is the area covered with one or more sensors for each point in the deployment zone. More objectives such as the number of sensors used can be optimised simultaneously. In this case, a

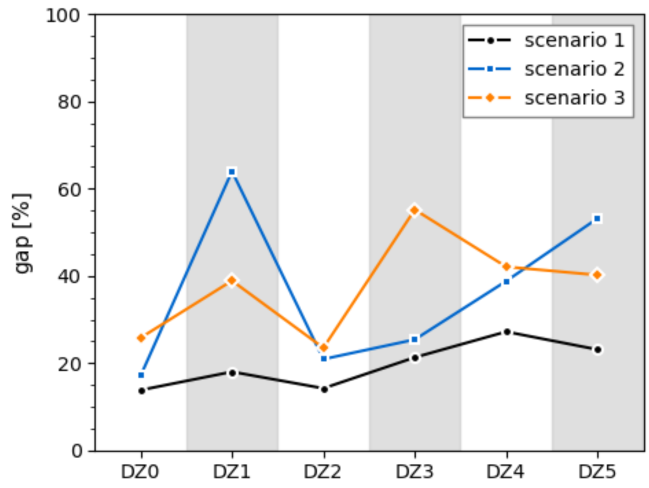



Figure 5: Loss of quality (gap) of the NSGA-II approach compared to the MILP approach for the three scenarios. Background filled with grey for the deployment zones made of multiple areas.

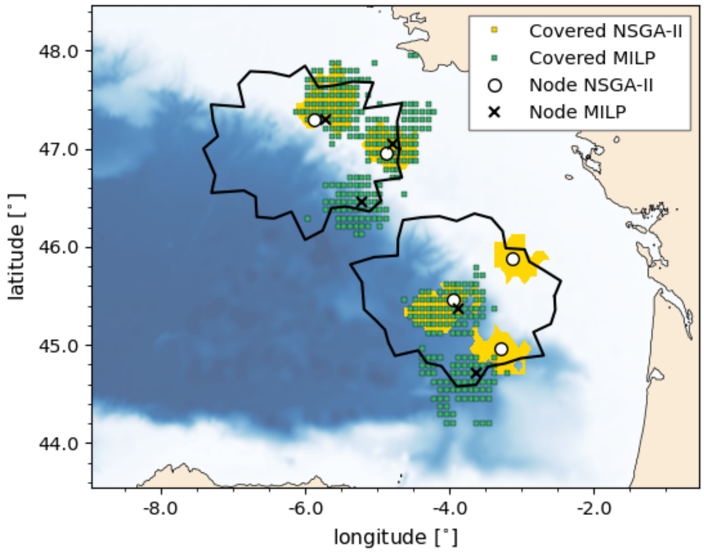
Pareto front is produced, corresponding to different network configurations that correspond to trade-offs for the two metrics. The MILP approach provides optimal values in reduced execution time for the simplest version of the problem (single objective, one sensor per point for coverage, no collinearity checking), but its execution time increases strongly when collinearity or when two objectives are considered. The MOEA approach with NSGA-II is not efficient for the simplest problem solving but remains more stable in execution times (around 15 minutes) for all versions of the optimisation process. Furthermore, even if MILP approach is flexible, it is limited by the linear constraints and functions it can handle. Furthermore, even if multi-objective optimisation is possible with MILP, it is not a native procedure such as the NSGA-II algorithm or other MOEA provide. The later is thus more promising for more complex versions of UWASN optimisations with more realistic objectives.

**ACKNOWLEDGEMENTS**

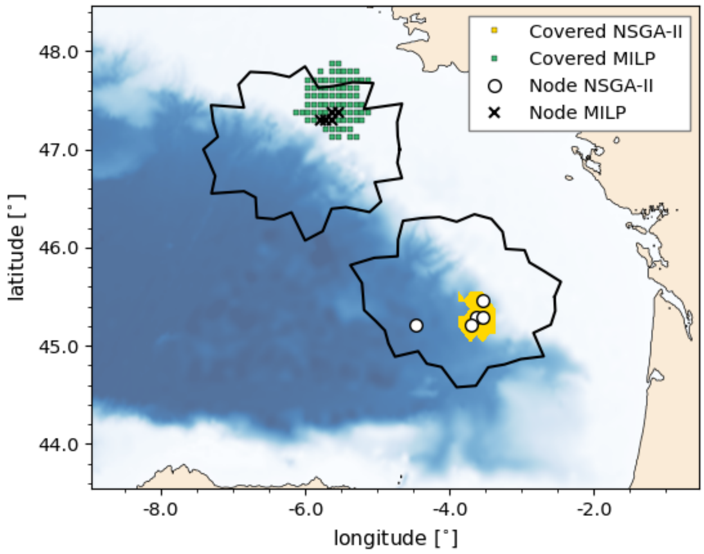
This work was partially funded by the ANR Astrid RESSACH under grant number ANR-22-ASTR-0030 and an IsBlue grant.

**REFERENCES**

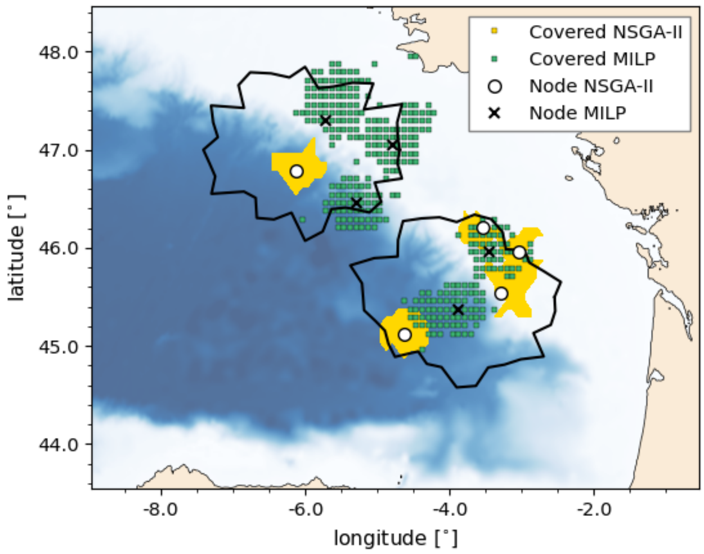
1. UICN France. Development of renewable marine energies and the preservation of biodiversity. Synthesis for decision-makers. Paris, France. ISBN 978-2-918105-81-7, Union internationale pour la conservation de la nature, 2019.
2. R. J. Ulrick. *Principles of underwater sound*. McGraw-Hill, Inc., 3rd edition, 1983.
3. E. Cheng, L. Wu, F. Yuan, C. Gao, and J. Yi. Node selection algorithm for underwater acoustic sensor network based on particle swarm optimization. *IEEE Access*, 7:164429–164443, 2019.
4. J. Galiana Nieves, J. Redondo, D. Benítez Aragón, D. Ramírez-Solana, and J. M. Herrero. Evolutionary optimization processes for acoustic applications where size matters. *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, 265(3):4125–4135, 2023.
5. C. Huszty. Multi-objective optimization for designing acoustic reflectors. *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, 253(1):6951–6957, 2016.
6. M. D. Collins. User's guide for ram versions 1.0 and 1.0p.




(a) scenario 1




(b) scenario 2




(c) scenario 3

Figure 6: Optimal solutions and covered points from the two approaches for test case "DZ5".

https://www.siplab.fct.ualg.pt/models/ram/manual.pdf. Last accessed 2024-03-22.

7. M. B. Porter. The bellhop manual and user's guide: preliminary draft. http://oalib.hlsresearch.com/Rays/HLS-2010-1.pdf. Last accessed 2024-03-22.
8. Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA, 1997.
9. Carlos A Coello Coello. *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.
10. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. Fast elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
11. IBM ILOG CPLEX Optimization Studio. IBM CPLEX user manual. https://www.ibm.com/docs/en/icos/12.8.0.0?topic=cplex-users-manual. Last accessed 2024-03-28.
12. J. Blank and K. Deb. Pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.