Review

Multi-robot Coverage for Inspection of Offshore Wind Farms: A Review

Ashley Foster [1], Mario Gianni [2], Amir Aly [1], Hooman Samani [4] and Sanjay Sharma [1]

[1] University of Plymouth; ashley.foster@postgrad.plymouth.ac.uk; amir.aly@plymouth.ac.uk; sanjay.sharma@plymouth.ac.uk
[2] University of Liverpool; Mario.Gianni@liverpool.ac.uk
[4] University of the Arts London; h.samani@arts.ac.uk

Abstract: Offshore Wind Turbine (OWT) inspection research is receiving increasing interest as the sector grows worldwide. Wind farms are far from emergency services and experience extreme weather and winds. This hazardous environment lends itself towards unmanned approaches, reducing human exposure to risk. Increasing automation in inspections can reduce human effort and financial costs. Despite the benefits, research on automating inspection is sparse. This work proposes that OWT inspection can be described as a multi-robot coverage path planning problem. Reviews of multi-robot coverage exist, but to the best of our knowledge, none capture the domain-specific aspects of an OWT inspection. This paper conducts a scoping review on the current state of the art of multi-robot coverage to identify gaps in research relating to coverage for OWT inspection. To perform a qualitative study, the PICo (Population, Intervention, Context) framework was used. The retrieved works are analysed through three aspects of coverage approaches: environmental modelling; decision-making; and coordination. Based on the studies reviewed and the analysis conducted, candidate approaches are proposed for conducting structure coverage of an OWT. Future research would involve the use of adapting voxel-based ray tracing pose generation to UAVs and exploration, applying semantic labels to tasks to facilitate heterogenous coverage, and semantic online task decomposition to identify the coverage target at runtime.

Keywords: multi-robot; coverage; UAV; structure inspection; offshore wind

Citation: Foster, A.; Gianni, M.; Aly, Amir.; Samani, Hooman.; Sharma, Sanjay. Title. *Drones* **2023**, *1*, 0.
https://doi.org/

Received:
Revised:
Accepted:
Published:

1 Introduction

Offshore wind turbine inspection is an area of increasing interest with the increasing prevalence of wind power [1]. The relevance of renewable offshore energy sources has never been greater than at present [2]. Offshore wind has a number of benefits when compared to onshore wind turbines [3]. Wind farms offshore experience greater and more predictable wind speeds with reduced turbulence, ensuring a single OWT is more productive than an onshore counterpart. Additionally, offshore farms needn't compete with other land uses and are less likely to meet resistance from local communities. While there are significant benefits to offshore renewable wind energy, so too are there serious challenges to overcome. Dynamic loads from wind and waves, as well as saltwater, damage and degrade the turbine quicker than one onshore. Installation is significantly more expensive than onshore and, as will be discussed, Operation and Maintainance (O&M) operations are considerably more complicated. Within the already growing offshore wind sector, O&M is predicted to become the second largest sub-sector of the offshore renewable market in the UK by 2030, and potentially the largest in the 2040's [4]. O&M can be broken down into 4 subsections [4], details of these are shown in Figure 1. Maintenance is a particularly high-risk aspect of O&M, involving highly skilled technicians out in the field for long periods undertaking maintenance work on the turbines. These services can include rappelling to inspect or repair the blades and diving to inspect cabling, all the while being far away

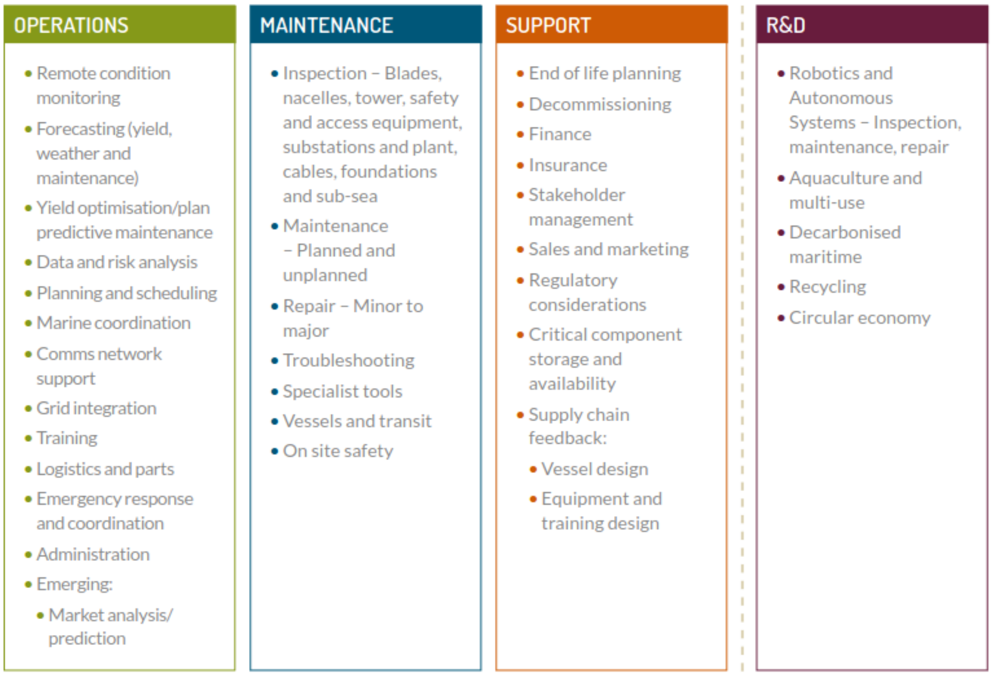

Figure 1. Key services that make up offshore wind O&M [4]

from any emergency assistance. The teams undertaking these operations are composed of disparate highly qualified technicians. Using traditional methods, a turbine inspection with 3 technicians will take 3-6 hours, allowing time for only 2-3 turbines to be inspected in a day [5]. Considering wind farms can often house hundreds of turbines, the cumulative labour time required for a single wind farm's regular inspections can commonly reach thousands of hours. It's both this financial cost and human risk that incentivises the use of robotic technologies. Commercial remotely operated robotic systems are now reasonably commonplace for Offshore inspections. Remotely operated underwater vehicles (ROVs) services facilitate inspections of the anchors, as well as sub-sea cabling [6][7][8][9][10]. Several companies offer Unmanned aerial vehicle (UAV) services for visual and thermal imaging inspections [11][12][13], and recently climbing robots have been made available for cleaning [14] and resurfacing OWTs [15]. Nordin et al. identified that individual unmanned vehicles have limited capacity to perform unmanned O&M for Offshore wind [16], rather the task lends itself to multi-robot systems. Five motivations for developing multi-robot systems were identified by Parker et al. [17]: 1) A task complexity too high for a single robot; 2) The task is inherently distributed; 3) The use of several less powerful robots is often less resource intensive than a single powerful robot; 4) Multiple robots can solve problems faster using parallelism; 5) using multiple robots increases robustness through redundancy. Notably, each of these could apply to offshore wind O&M. Indeed multi-robot approaches to Wind energy O&M have been researched, albeit overlooking critical factors such as communication challenges and harsh environmental conditions, necessary for real-world implementation [18][19][20]. Approaches to multi-robot navigation in extreme environments require mechanisms to minimise interference and spatial conflicts [21] else the system may perform unreliably. One interesting commonality in the aforementioned current research is their use of robotic heterogeneity. Parker defines robot heterogeneity as variety within robot behaviour, morphology, performance quality, size, and cognition [17] within a team. Certain inspections may require a heterogeneous team, while others may be performed faster with robots specialised for certain tasks. Considering a comprehensive inspection of an OWT (one covering the turbine's surface, the foundations, local cabling and turbine interior), a range of robots with varying morphologies, locomotion and sensing capabilities would be required. Variety with performance quality can also affect the quality of the inspection, a UAV fitted with a high-resolution camera would be able to capture footage of the same quality as one with a lower-resolution camera at a greater rate. Another possibility within a heterogeneous team is having robots collaborate in such a way, as to complete tasks impossible for just one. Jiang et al. provide an example of just this, with a UAV being used to deploy and retrieve a BladeBug to and from a wind turbine blade [22]. The authors made use of GNSS to position the UAV near the landing target, and then made use of lidar data to position itself for landing and deploying the bladebug on the blade. Reaching the Blade with the Bladebug would have been impossible alone, but is made possible via the UAV. Another example of such behaviour is using an Unmanned Surface Vehicle (USV) as a mothership for UAVs, with the USV serving as a "marsupial" robot. Fan et al. are concerned with the autonomous landing of a UAV on a USV using a fuzzy self-adaptive PID controller [23]. A marsupial relationship is also detailed by Miškovic et al, in which a USV relies on a UAV to localise itself with respect to a floating object needing tugging [24]. Zhang et al. describe a fully autonomous system for the recovery of fixed-winged UAVs, making use of an arresting cable and a net to safely land the UAVs [25].

The use of multi-robot teams for Offshore inspection is an area currently sparse of research. In this work, the focus is on visual inspection of the OWTs. It's common for the operators of the OWTs to request an inspection at the end of the warranty and regularly every three years after [26]. A typical inspection requires the capture of high-quality images of each side of the OWT blade (suction side, pressure side, leading edge, and trailing edge). Tower and nacelle inspections are sometimes also required and are concerned with identifying welding defects, coating issues and mechanical damage [27]. There maybe areas of particular interest such as the blades, although this can be seen as a variation of the problem. Further inspection operations use USVs or ROVs to inspect floating substructures of floating OWTs or the underwater cabling [28]. The ORE Catapult Levenmouth demonstration turbine detailed in Figure 2, built to facilitate OWT research, provides an example of the structure to be inspected. These operations all involve capturing images of all of an area of interest, the problem of ensuring the entirety of an area of interest is covered by a sensor's footprint is known as the coverage path planning problem.

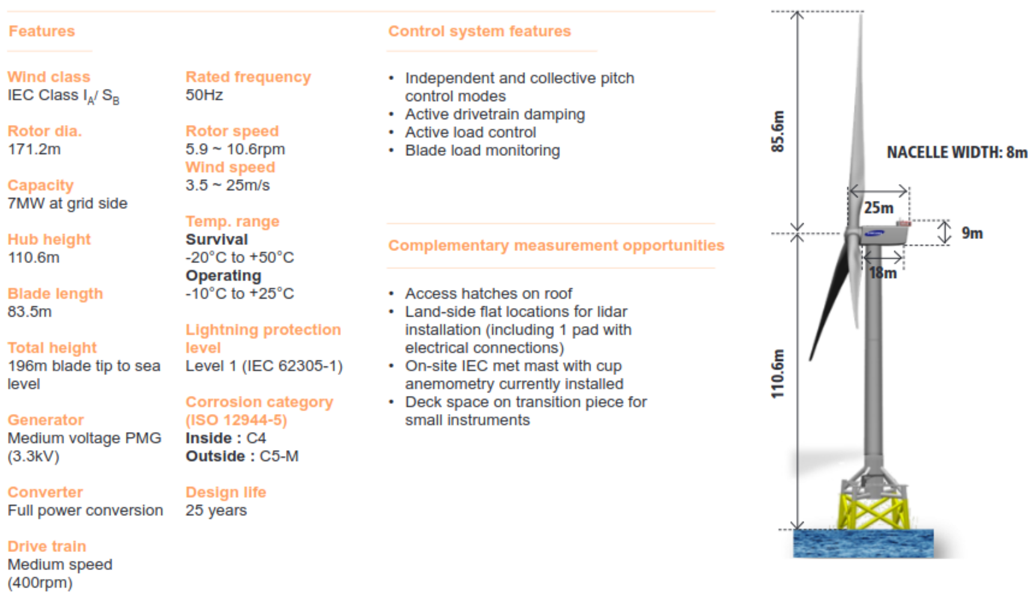

Figure 2. Specifications of the Levenmouth 7MW demonstration OWT [29]

Coverage path planning as defined by [30] is the problem of passing over all points in the target environment. While at the time coverage was mostly concerned with the

coverage of 2D planes, such as mowing a lawn [31] or vacuuming a floor [32], the definition of coverage has now expanded to include 3D environment and structure coverage. To this end Almadhoun et al. define coverage path planning as "a process of exploring or exhaustively searching a workspace, whether it a structure of interest or an environment and determining in the process the set of locations to visit while avoiding all possible obstacles" [33]. While reviews of the literature surrounding coverage more generally exist, such as Choset's inaugural survey on robotics for coverage [30] and Almadhoun et al. in their survey on multi-robot coverage path planning for model reconstruction and mapping [33]. However, these surveys do not focus on the domains representing offshore wind inspection, a specific variation of the problem characterised by its environmental lack of structure and sparseness. To the best of our knowledge, this work provides the first scoping literature review on the coverage problem and the first literature review on coverage of OWT inspections.
This paper is structured as follows: Section 2 details the methodology used to conduct the scoping literature review; Section 3 covers the approach to be undertaken for analysing the works retrieved from the literature search; Section 4 provides an analysis of approaches to environmental modelling used by the retrieved works; Section 5 covers the approaches to decision making and their applicability to offshore O&M; Section 6 is concerned with coordination approaches used in the literature and applicability; and finally the work will be concluded and future direction for the field will be discussed. The contributions of this work include the first systematic literature review of multi-robot coverage, following a strict systematic procedure novel to robotics. A taxonomy and discussion of current works were discussed, and several gaps in current research and avenues for the future were identified.

2 Methodology

A scoping review has been conducted to identify the research gaps in the literature on Multi-robot coverage for Offshore wind inspection [34]. To ensure the quality of this scoping review, the PRISMA 2018 checklist for scoping reviews has been followed [35]. The review has been structured according to the PICo framework for qualitative reviews, detailed in Table 1.

Table 1. PICo Definitions for Environmental Representations with Search Concepts

P	**I**	**Co**
Population	**Interest**	**Context**
Multi-robot systems	Coverage	Unknown and unstructured environments
	Search Concepts	
Multi-robot	Coverage	Unknown & unstructured
	Alternative Terms	
Multi-agent		Unknown Unstructured Extreme Real

Based on this framework, the following research question and sub-questions have been formulated after a brief review of the literature:

Table 2. Research Questions

Research Question
What is the most suitable framework for multi-robot coverage in domain applications resembling offshore wind inspection? **Sub-Questions:** • What is the most suitable environmental model for Multi-Robot Coverage in terms of suitability to domain applications resembling offshore wind inspection? • What is the most suitable multi-robot coverage decision-making approach for domain applications resembling offshore wind inspection? • What is the most suitable strategy to effectively coordinate a multi-robot system for domain applications resembling offshore wind inspection?

Table 3. Digital Libraries Used in Review

Digital Library	**Description**	**URL**	**Area of Focus**
IEEE Xplore	A digital library provides all IEEE publications as well as those from its publishing partners.	https://ieeexplore.ieee.org/	Computer science, electrical engineering and electronics.
The ACM Guide to Computing Literature	Association of Computing Machinery's Digital library provides all ACM publications and works from all major publishers.	https://dl.acm.org/	Computing and Information Technology
Scopus	Scopus covers 240 disciplines to ensure researchers, instructors, librarians and students have confidence that they are not missing out on the vital information they need to advance their research and scholarship.	https://www.scopus.com/	General
Web of Science	The Web of Science is a paid-access platform that provides access to multiple databases that provide reference and citation data from academic journals, conference proceedings, and other documents in various academic disciplines.	https://www.webofscience.com/wos/	General

An advanced search has been conducted in the databases IEEE Xplore, The ACM Guide to Computing Literature, Scopus, and Web of Science. Details of these libraries can be seen in Table 3. These four databases provide time-efficient access to a wide range of peer-reviewed publications.

From the research question, a query was formed, as shown in Table 4. Due to the nature of offshore wind inspection, a search query making use of the term "offshore" would have yielded no results due to the lack of current research.

It's the aim of this paper to identify works relevant to the OWT coverage problem, and to synthesise the knowledge from these works through the lens of the OWT coverage problem. To identify works relevant, albeit not specific to, OWT coverage, those works with "domain applications resembling offshore wind inspection" were identified. In examining which domain applications closely mirror offshore wind inspection, it's crucial to understand the unique characteristics of offshore wind inspection's environment. Offshore wind farms are vast and dispersed, composed of repeated turbines usually at regular intervals but not always so. The environment can be considered sparse and unstructured in that regard. Given the nature of the energy being captured by the turbines, these areas are also highly exposed and prone to unpredictable weather, hence we can consider the environment extreme. To ensure the works reviewed represented the state of the art, only those works after 2015 were considered, which was achieved through filtering on the individual databases:

Table 4. Search Query

Search Query
• (multi-robot* OR multi-agent*) AND (coverage) AND ((unstructured AND environment*) OR (unknown AND environment*) OR (extreme AND environment*) OR (real AND environment*))

Table 5. Criteria for Article Exclusion

Criteria Type	**Included**	**Excluded**
Coverage Control	Those works considering the coverage path planning problem	Those works considering the coverage control problem
Environmental structure	Those works considering environment resembling OTW inspection, namely one that is unstructured, unknown, extreme or real	Those works considering environment not fulfilling these criteria
Surveys	Any non-survey work was considered	Surveys

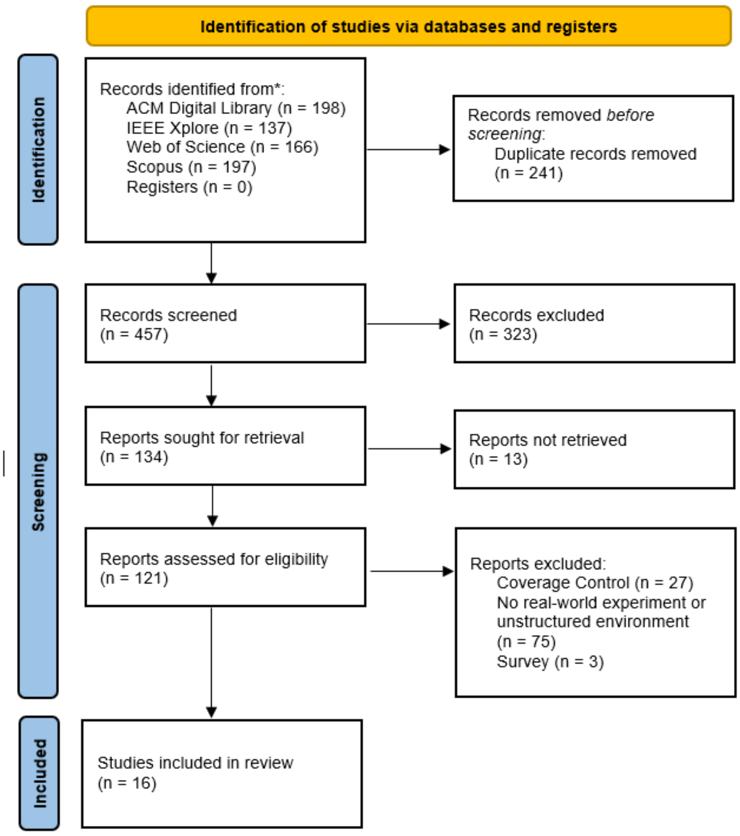

Figure 3. PRISMA Flowchart showing exclusion process

The exclusion criteria in table 5 were formed to remove those works not relevant despite not being excluded in screening.

The PRISMA flowchart in Figure 3 shows the number of records identified by the search strategy for each database. Initially, those works duplicated across the searches were removed. The screening process was carried out by removing those works whose title or abstract made no mention of "Coverage", "Multi-robot", or "Multi-agent". The works were then sought for retrieval, and if unavailable or required purchase was discarded. Finally, the literature-exclusion criteria were used to remove irrelevant works.

3 Analysis

The 16 studies included in the review were then analysed. In this section the process of analysis for these studies is detailed. As was detailed in the research subquestions in Table 2, three aspects of the coverage problem are considered: 1) Environmental modelling, 2) Decision making, and 3) Coordination.

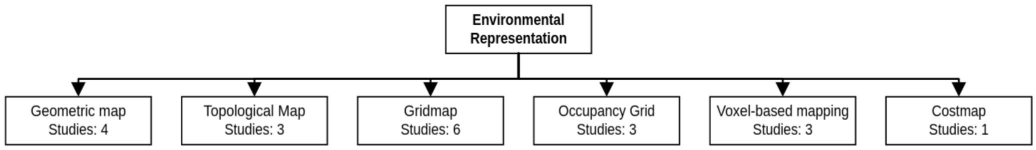

Figure 4. Environmental representation taxonomy with instances of surveys reviewed

Environmental modelling is concerned with the methods used by robots or a central planner to represent the environment and tasks within it. A taxonomy was constructed to systematically categorize and analyze the approaches featured in the studies. This taxonomy of environmental models includes the following categories: Geometric maps featured four studies; topological maps featured three studies; Gridmaps featured six studies; Voxel-based maps featured three studies; occupancy grids featured three studies; and Costmaps featured in one.

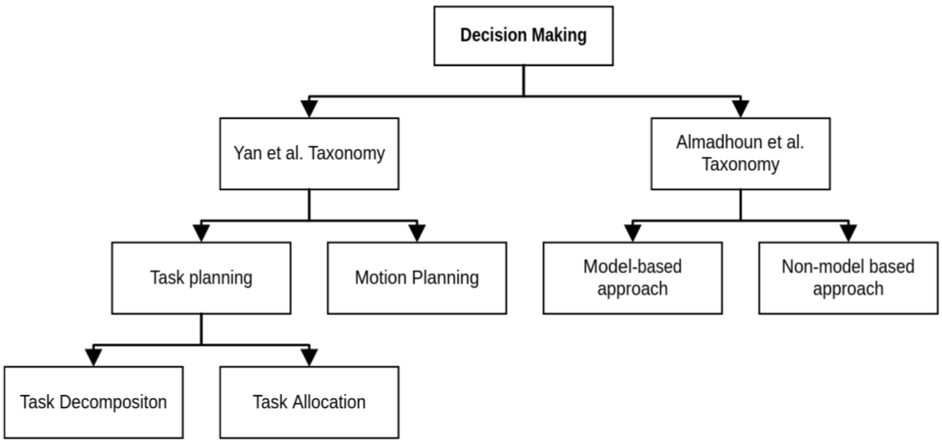

Figure 5. Aspects of decision making analysed including the Yan et al taxonomy [36] and the Almadoun et al. taxonomy [33]

Some approaches used multiple methods and therefore appear twice. Using the taxonomy and the details of the studies reviewed, those approaches judged most suitable for OWT inspections were identified and discussed.

Decision-making, as per our definition, is the collective choices defined by specified objectives made by a multi-robot system. The applicable studies are analysed using the model/non-model distinction proposed by Almadhoun et al. [33] and planning definitions from Yan et al. [36]. Almadhoun et al. identified a classification of approaches based on their assumed prior knowledge. Model-based approaches know the tasks and environmental structure before the coverage task. Non-model-based approaches forgo this assumption and require modelling of the environment during the task. Yan et al. identified three components that compose mobile multi-robot task planning approaches: task decomposition, task assignment, and motion planning. Task decomposition isn't always necessary depending on the prior knowledge, but refers to the decomposition of a multi-robot task into a set of single-robot tasks. In the case of coverage, the task decomposition takes the form of decomposing the environmental representation into robotic positions or poses as tasks. A task decomposition taxonomy was formed to analyse the approaches suitability to OWT coverage.

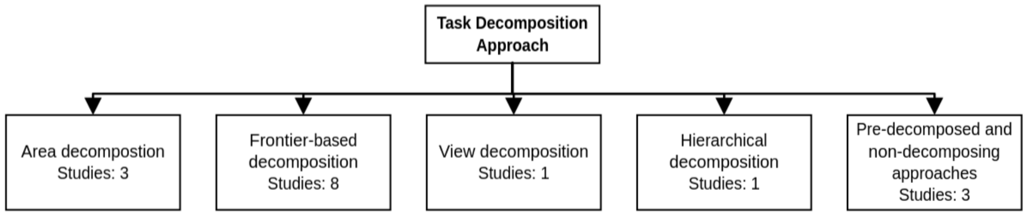

Figure 6. Task decomposition taxonomy with instances of surveys reviewed

Task allocation assumes the set of single robot tasks, and is concerned with how can the set of tasks be optimally assigned to the robots. Finally, motion planning is how, given the task assignments per robot, a path for the robots can be constructed to visit all tasks optimally. Motion planning could be considered a single-robot problem in regards to the order of visiting the tasks, but at a lower level necessitates collision avoidance between team members.

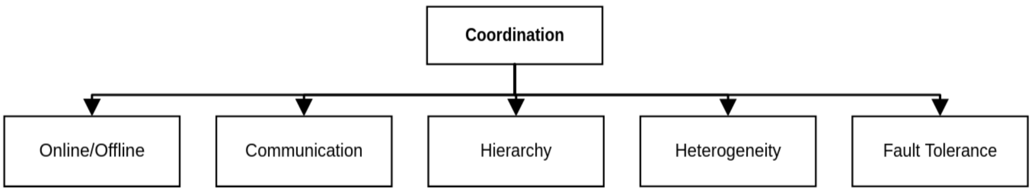

Figure 7. Aspects of coordination analysed

Farinelli et al. [37] consider coordination to be cooperation where team members consider other team members in their actions to increase system performance. Yan et al. [36] defined it as planning to deal with resource conflicts among team members. The aspects considered as coordination in this work are aspects related to potential resource and reliability issues that may arise from using real robots, Communication approaches, team hierarchies, fault tolerance, and robotic heterogeneity. Data relating to these aspects of the studies were extracted from the works where approaches were specified.

4 Environmental models

This section aims to answer the following sub-question:
What is the most suitable environmental model for Multi-Robot Coverage in terms of suitability to domain applications resembling offshore wind inspection?
Wind farms are sparse, unstructured, and dynamic environments. There are both the predictable dynamics of the rotation of the blades and unpredictability in the current yaw orientation of the hubs. The turbines are usually spread over a kilometre away from one another resulting in large sparse areas in an environmental model. There may be a degree of uncertainty in GPS localisation due to the multipath error resulting from signals reflecting off the turbines and the sea itself [38]. So in approaching this sub-question, we should

view the applicability of the models used through the lens of a multi-robot offshore wind inspection. To analyse the approaches used in the literature, and best identify those models most suited to domain applications resembling offshore wind inspection, a taxonomy of models was constructed. These classes of approaches were then described concerning the specific implementations, followed by a discussion of the applicability of the approaches reviewed to domain applications resembling offshore wind inspection. Burgard et al. [39] identified three main challenges in constructing or choosing environmental models: 1) such models should be compact 2) they should be task/application dependent 3) given they are constructed from sensor data, they should account for the uncertainty inherent in sensors and state estimation. The appropriate model for the offshore inspection task should consider these three factors.

Table 6. Environmental models used for works reviewed

Environmental Model	Work (Authors, Year)
Geometric map	Ball et al. 2015 [40] Masehian et al. 2017 [41] Karapetyan et al. 2018 [42] Tang et al. 2022 [43]
Topological map	Ball et al. 2015 [40] Karapetyan et al. 2018 [42] Kim et al. 2022 [44]
Gridmap	Kalde et al. 2015 [45] Song et al. 2015 [46] Perez-Imaz et al. 2016 [47] Sharma et al. 2016 [48] Zhang et al. 2019 [49] Yu et al. 2023 [50]
2D costmap	Ball et al. 2015 [40]
Occupancy grid	Colares and Chaimowicz 2016 [51] Bramblett et al. 2022 [52] Kim et al. 2022 [44]
Octomap	Dornhege et al. 2016 [53] Dong et al. 2019 [54]
Euclidean Signed Distance Field (ESDF) map	Bartolomei et al. 2023 [55]

4.1 Geometric Map

In some approaches, usually where the environment is known *a-priori*, a geometric map is used. In such approaches, the environment's shape, and obstacles within, are modelled as polygons, an example of which can be seen in Figure 8. In both Ball et al. [40] and Karapetyan et al. [42] the geometric map is known a priori and represents the environment needed to cover, and in both approaches the authors use Boustrophedon cell division to discretise the area into cells in a topological graph. Another method of discretisation comes via overlaying a grid on the model to form a gridmap, a rasterisation process, which was used by Tang et al.[43].

4.2 Topological map

Choset et al. [56] defines topological representations as aiming to represent environments with graph-like structures, with the nodes representing "something distinct" and edges representing the spatial relationship between nodes. The focus of topological maps is how different nodes, representing points of interest in the environment, are connected to each other, rather than the detailed geometric properties of the space. As such these representations are usually in the form of graphs composed of nodes with edges representing the interconnectivity of nodes, an example of which is given in Figure 9. The edges in a topological representation can be given semantic properties, such as a cost of traversal or directionality [57]. Topological maps are often contrasted with geometric maps, although as will be seen geometrics maps can be, and often are, decomposed to topological representations. This is the case in Ball et al. [40] and Karapetyan et al. [42] who both consider an

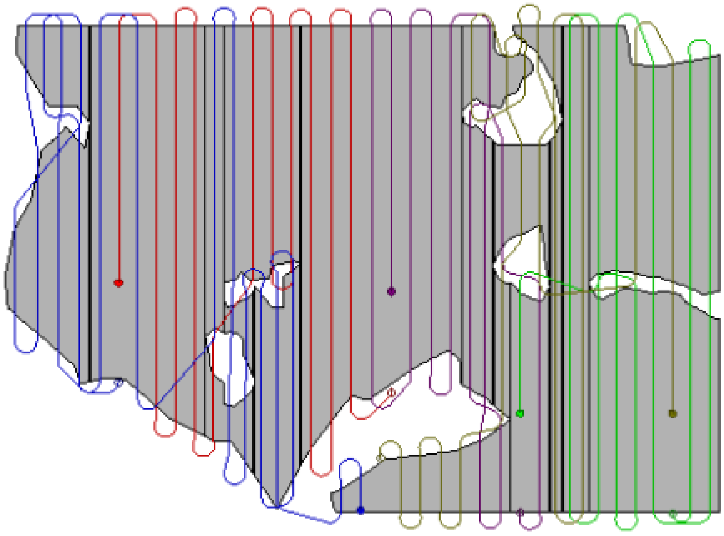

Figure 8. Geometric Map with robotic paths [42]

initial geometric map representation, and they use boustrophedon cell division which is described in greater detail in Section 5.1.1. The result of Boustrophedon cell divisions is a set of connected cells in the environment which take the form of a topological representation. An occupancy grid (Discussed in section 4.4) is used to generate "waypoints" to ensure sensor coverage of the environment by Kim et al. [44]. These tasks can then be viewed as nodes in a graph connected by edges. Topological maps are rarely considered a priori knowledge, rather another representation is decomposed into a topological map as in Kim et al. [44]. Due to their simplified abstract nature, they are better suited to global path planning with the specifics of path planning abstracted to an edge cost value.

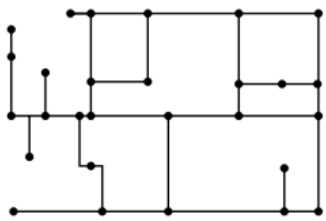

Figure 9. Topological Representation [56]

4.3 Gridmap

A gridmap is a grid of a specified dimension composed of squares of a certain size. Sometimes the size of the grid cells represents the size of the robot's footprint or sensor's footprint, such that visiting each cell would provide full coverage of the environment [45][48][49]. Other times the grid cell is used to discretise the possible positions [50] or to facilitate allocation of the environment to team members while still requiring coverage path inside the cell [47]. Contrasting with topological maps, with known dimensions and directional relations between cells, gridmaps provide an abstracted yet accurate modelling of the environment's geometry, However, assuming that each cell represents a task a gridmap can be considered both a metric map and a topological graph. Kalde et al. [45] give an example of encapsulating semantics in their gridmap through cell states. This can be seen in Figure 10. In their work the cells can be one of 4 states: Unknown cells, shown as question marks, represent those that have yet to be explored; Occupied cells, shown as black cells, represent static obstacles; animated cells represent robots e.g. R1 and humans e.g. H1; and free representing explored empty cells shown in white. Sharma et al. [48] use a similar representation. The model used in [49] is also described as a gridmap. Perez-imaz et al. [47] make use of a hexagonal gridmap rather than a square one. hexagonal grids facilitate diagonal movement with uniform distance between cells, as well as better approximating a circular sensor radius than a square allowing an environment to be represented with fewer cells. Song et al. [46] make use of a multi-resolution grid-based environmental model. At the smallest resolution the cells are the size of the sensor radius, and above that is a map of supercells composed of 4 cells. This multi-resolution grid is used to facilitate vehicles escaping local minima. While these simple multi-state grids are sufficient for the authors' uses, they don't take into account sensor uncertainty as per the three main challenges described before [39] and, therefore are unsuitable for real-world applications alone. Such representations are compact, however, and therefore are particularly useful for high-level planning.

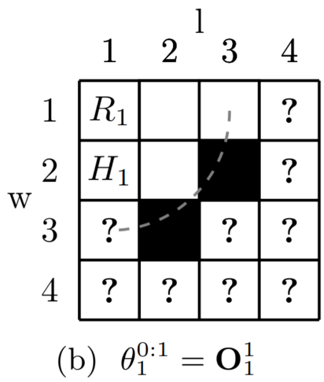

Figure 10. Four state gridmap [45]

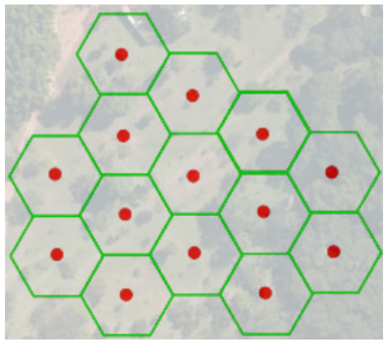

Figure 11. Hexagonal gridmap [47]

4.4 Occupancy grid

Occupancy grids are a common model used to tackle the uncertainty inherent in sensors. First proposed by Moravec & Elfes [58], the grid is composed of cells with values representing the probability of its occupancy with an obstacle. These cell occupancy probabilities were estimated as independent random variables which, while rarely if ever the case in the real world, simplifies computation. Numerous approaches now exist to relax the assumption of cell in-dependency [59]. Colares & Chaimowicz [51] make use of an occupancy map in their approach to exploration, this occupancy map is generated from their SLAM approach. In their approach, the occupancy grid is used to compute the costs of frontiers for the team members. Bramblett et al. [52] also make use of an occupancy grid representation, using recursive Bayesian estimation to update the cells given sensor measurements. In this case, the occupancy is once again used to identify frontier cells and exploration tasks are generated in areas of high uncertainty, facilitating complete sensor coverage of the environment. Occupancy grid representations are particularly useful in unknown environments, as they require no prior knowledge to form. In regards to OWT inspection, occupancy grids have three main areas of use. By forgoing the assumption of prior knowledge, occupancy grids can account for sensor uncertainty while facilitating the mapping of an unknown environment. The occupancy values can act as a component in an objective function to drive the team to explore uncertain areas. Occupancy grids can also be used to construct a costmap (see Section 4.6) for motion planning, providing a trade-off between traversing unknown areas and distance. The effect of the aforementioned cost map is a robot would have a degree of reluctance to traverse unknown areas due to potential obstacles or dead ends.

4.5 Voxel-based mapping

A voxel is a cell in a 3D grid, the term voxel being "an analogy to pixel" [60]. Voxel-based mapping represents the environment as a 3D grid composed of voxels. The simplest voxel representation is a 3D binary array, with 1 representing occupancy and 0 representing free space [61]. Two implementations of voxel-based mapping were reviewed: Octomap [53][54] and Euclidean Signed Distance Field (ESDF) map [55]. Octomap is a probabilistic framework for environmental modelling of 3D cell occupancy based on hierarchical octrees [62]. The hierarchical nature of the approach can reduce memory usage, and facilitate varying levels of environmental detail to be captured. Areas with fewer features or in which mapping is less critical can be mapped at a lower resolution, conserving memory and reducing the computational cost of future environmental decomposition. Conversely, those areas of particular interest can benefit from a higher resolution, allowing for more detailed and accurate mapping, and better-informed environmental decomposition. Dong et al. [54] make use of Octomap for their exploratory scanning, before decomposing it into a 2D occupancy grid to plan tasks. Dornhege et al. [53] provide a task planning algorithm working directly with Octomap. Octomaps is a very powerful approach to modelling 3D environments, and its availability as a ROS library has added to its popularity. An ESDF is used by Bartolomei et al [55] and described initially in [63]. This is a highly semantic voxel model, in which each voxel is linked to a data structure composed of the voxel's coordinates; the occupancy probability; the euclidiean distance to the nearest obstacle; whether the voxel has been observed; the voxels closest to itself; and information on the area sharing its closest obstacle. The authors describe the approaches' usefulness to UAV navigation as "what is truly useful is the information of free space, instead of obstacles." Voxel-based mapping approaches are, in their regular dimensions and regular directional relations (each voxel shares the same spatial relationship with their 6 neighbouring voxels), similar to the 2D grid-based representations discussed previously. However, unlike the 2D gridmap, they have little use outside of 3D coverage. While 2D coverage often involves visiting each cell in the environment once as in Zhang et al. [49], this is rarely the case in 3D coverage. Rather, 3D coverage tasks tend to involve the sensor coverage of either the whole environment [55] or sub-sections of the environment of particular interest [53]. Arguably, these tasks are particularly representative of an offshore wind inspection. Especially in the case of covering specific areas of interest within a 3D environment, which can represent turbines themselves or areas of the turbine of special interest such as the blades.

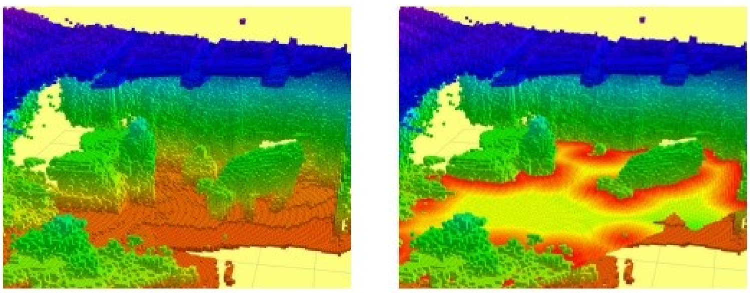

Figure 12. A occupancy grid model (left), and ESDF model (right) [63]

4.6 Costmap

Costmaps are grid-based representations, with the value of each cell expressing a cost of traversal. The cost of a cell, given as a numerical value, can represent a number of different attributes of traversing a given cell. In the work of Ball et al. [40], the attribute in question is a deviation from a high-level planned path and avoiding obstacles. They then make use of a searched-based lattice planner (SBLP) to generate paths that minimise the cost of traversal with respect to the high-level path and detected obstacles. Though not expressed explicitly, the ROS navigation stack uses a 2D costmap, so other works making use of ROS are very likely to make use of them also [64]. The ROS costmap builds an occupancy grid and based on the occupancy value of a cell increases the cost of traversal within a user-specified radius around the suspected obstacles. The effect of this is the path-planning algorithms will account for the costmap and select paths based on a tradeoff between distance and proximity to suspected obstacles. In regards to the coverage path planning problem, costmaps find their greatest utility in motion planning, facilitating the generation of paths between tasks while trading off the potential of obstacles with the distance to travel.

4.7 Discussion

Wind farms can be considered an extreme, sparse and unstructured environment. The environmental model used relates to the task being undertaken, OWT inspection, but there are variations on this task. The environmental modelling approaches one should select depend on a variety of aspects. These aspects can include whether a model is known prior, if the team is homogenous, and whether the blades are moving. This discussion attempts to map the suitability of the environmental model taxonomic classes to the inspection problem, also aware of the aforementioned characteristics of the environment[1]. An inspection of a turbine involves acquiring sensor data across the entire turbine, or at specific areas of interest such as the blades. Refer to Figure 2 for a basic description of the components of an OWT. This variant of the OWT inspection is a 3D structure inspection, a variation of coverage in which the aim is to ensure sensor coverage of either the entirety of a structure of interest or specific areas on said structure. The 3D nature of this task excludes the use of 2D environmental models, lending itself towards a Voxel-based model.

Considering the inspection is coverage of the structure, in a standard single-resolution voxel-based model the rest of the environment would modelled in the same detail as the turbine. The result of this single-resolution voxel-based model would be inefficient memory use and slower computation of the task decomposition and motion planning[2]. Therefore there is an incentive to make use of an adaptive resolution like that provided by Octomap [62]. In doing so the turbine can be given a detailed accurate voxel-based representation without also requiring a detailed model of the empty space around it. An additional benefit of an adaptive resolution for OWT inspection is allowing varying levels of coverage detail based on the specific turbine component being inspected.

One aspect of OWT inspection that may require a novel solution not seen in the works reviewed is the inspection of the moving blades. Blade inspection generally requires the turbine to stop, but there is a financial incentive to keep the turbine running during inspection. While Octomap is updatable and can represent dynamic environments there aren't any semantic labels attached to voxels to represent which blade is which, just a value to specify the probability the voxel is occupied. To ensure coverage of all the blades the environmental model would need to keep track of which blade is which. this could be achieved by applying a semantic label to the moving cluster of voxels that represents an individual blade, however, this presents challenges such as keeping track of which blade is which when not in view. In the literature reviewed, no modelling approach accounts for these moving tasks, as such this represents one avenue for future research.

4.7.1 Robotic heterogeneity

An unaddressed area is representing heterogeneous tasks, inspection tasks may require more than one class of robot. Different types of tasks or motion capabilities in a heterogeneous team would need to be represented in the environmental model. These requirements of certain capabilities could be represented semantically in topological models, by labelling edges based on traversal requirements or task nodes with information on the necessary capabilities to complete it. In order to semantically label the edges with these

1 extreme, sparse and unstructured.
2 Task decomposition and motion planning are discussed in Section 5.1.1 and Section 5.1.2 respectively

traversability requirements, a novel form of heterogeneous traversability analysis would need to be implemented, this is an open area of research.

5 Decision making

This section aims to provide an answer to sub-question 2 from Table 2:
What is the most suitable multi-robot coverage decision-making approach for domain applications resembling offshore wind inspection?
In this work, as mentioned previously, Decision-making, as defined by the authors of this paper, is the mechanism by which collective choices are defined by centralised or decentralised objectives. In order to systematically analyse the approaches' suitability to the domain, two taxonomies were applied. The first taxonomy relates to the amount of prior knowledge available to a system a-prori. Almadhoun et al. [33] in their survey on coverage path planning classified approaches assuming prior knowledge as being "model-based", and those without prior knowledge as being "non-model based". Model-based approaches assume a prior environmental model, a "known environment", whereas non-model-based approaches lack this initial knowledge. Knowledge of one's environment is a significant advantage, and as one would expect model-based approaches are usually better performing. However assuming prior knowledge of one's environment is a strong assumption, and this prior knowledge is not always available or accurate. The second taxonomy uses the planning definition from Yan et al. in order to analyse the works included in this review [36]. Yan et al. consider planning to be composed of two aspects, task planning and motion planning. Task planning can further be divided into two sub-aspects, task decomposition and task allocation, which are concerned with turning a multi-robot task into a set of single-robot tasks and then allocating these tasks to the team. Motion planning involves the generation of paths and trajectories for the team members to visit and complete all the tasks.

5.0.1 A-priori Knowledge

Almadhoun et al. [33] identified a dichotomy in approaches to coverage. Approaches can either have, or not have, an a-priori environmental model. The authors defined these groups of approaches as either non-model based and model based.

5.0.1.1 Non-model based

In the simplest sense, a non-model approach to coverage assumes nothing about the structure environment, it will facilitate coverage without requiring a prior environmental representation. Therefore, these approaches are often used when the environment is unknown or uncertain. Non-model-based approaches can be described using both the terms "exploration" [50] and "coverage of an unknown environment" [46]. There is a degree of ambiguity in the terms "Coverage" and "Exploration". Yamauchi [65] defined explorations as a problem of "Given what you know about the world, where should you move to gain as much new information as possible?". A commonality amongst papers concentrating on the exploration problem is that the approaches attempt to maximise the knowledge of an *a-priori* unknown environment. That is to say, exploration aims to model the environment, and approaches work to maximise the completeness of the model. On the other hand, coverage can be roughly split into two distinct problems: 1) covering an environment with a team of sensors' footprint in an optimal manner and 2) assigning spatially distributed tasks to a team of robots in an optimal manner. The former is often decomposed into the latter, and the latter is an instance of the multi-robot task allocation problem [66]. Coverage can be in an unknown environment without exploration. In Bramblett et al. a team of robots with limited communication range are tasked with exploring an unknown environment [52]. The authors consider an unknown environment with tasks, hence the problem requires both optimal full exploration and task allocation coverage. "Exploring an environment by repeatedly applying path planning algorithm at each instance of time" is a highly specific definition for exploration from Sharma et al. [48], characterising the online[3] nature of the exploration problem. The quality of sensor coverage is taken into account in Dong et als. [54] work, stating their problem as collaboratively exploring and mapping a scene such that scanning coverage and reconstruction quality is maximised, while the scanning effort is minimised. Of the works retrieved 8 are non-model based.

5.0.1.2 Model-based

An approach can be said to be model-based if it assumes a prior environmental model, an assumption that simplifies the task decomposition [33]. Ball et al. assume a known geometric map for using multiple modified John Deere TE Gator for crop spraying [40]. This geometric map is then decomposed into multiple sub-regions through boustrophedon cellular decomposition. An approach initially assuming environmental bounds given by a set of vertexes representing the bounds of an area of interest, this is then decomposed to a hexagonal grid is given by Perez-imaz et al. [47]. Karapetyan et al. consider a known geometric model that is then decomposed to task areas via boustrophedon cellular decomposition [42]. Kim et al. [44] in their approach also assumes a search region of an arbitrary shape. Zhang et al. assume the prior model of the environment is in the form of a simple binary gridmap of free cells and obstacles, a very common representation in offline coverage problems [49]. Finally, Tang et al. consider a known geometric model of the environment. However, due to random dynamic interference, their approach cannot be computed offline [43]. A prior model of the environment can facilitate prior planning and optimal solutions to the task decomposition and path planning problems. However solutions that rely to heavily on the prior knowledge of the environment may struggle with the uncertainty of a real world implementation, especially in areas with high uncertainty like offshore wind farms. 8 of the works retrieved where model based.

5.1 Planning

Planning is defined as "the task of coming up with a sequence of actions that will achieve a goal" by Yan et al. [36]. Planning for a mobile multi-robot system can be divided into task planning and motion planning. Task planning is a problem of how tasks should be divided among the team, while motion planning is concerned with devising paths in order to facilitate locomotion to, and completion of, said tasks.

5.1.1 Task Planning

Yan et al. [36] defined Task planning as the problem of "which robot should execute which task". They then proposed to split task planning into two further categories, task decomposition and task allocation. Task decomposition is how a multi-robot problem can be split into single-robot tasks, and task allocation is how best to assign these single-robot tasks to the robotic team. The works in this review are grouped by the task decomposition method used:

5.1.1.1 Area decomposition

The works discussed in this section share in common the decomposition of a 2D plain into a set of geometric shapes representing coverage areas to be assigned.

In Ball et al. [40], the initial representation is in the form of a geometric map. The task is decomposed using a boustrophedon cell decomposition, as first described in Choset's work [68]. The boustrophedon cell decomposition algorithm takes a known geometric model and decomposes it into a topological representation composed of uneven cells

[3] Online in the sense of robotic planning indicates the plan is generated at runtime, offline plans are generated before the execution

Table 7. Task planning approaches

Paper	Task Decomposition	Task Allocation
Ball et al. (2015) [40]	Boustrophedon cell division	Not Described
Kalde et al. (2015) [45]	Frontiers and Humans identified as potential tasks	Greedy allocation on a cost matrix
Song et al. (2015) [46]	Pre-decomposed	Pre-allocated
Colares and Chaimowicz (2016) [51]	All frontier cells as tasks	Optimal frontier based on a cost function
Dornhege et al. (2016) [53]	Set of optimal views	Greedy allocation, or set cover solution with TFD solver
Perez-Imaz et al. (2016) [47]	Hexagon grid	K-means clustering
Sharma et al. (2016) [48]	Pre-decomposed	Pre-allocated
Masehian et al. (2017) [41]	Hierarchy of decompositions	Allocated based on other classes of robots identifying tasks
Karapetyan et al. (2018) [42]	Boustrophedon cell division or DCS path splitting	Not Described
Dong et al. (2019) [54]	Set of optimal frontier views	K-means clustering
Zhang et al. (2019) [49]	DARP	DARP
Bramblett et al. (2022) [52]	All frontier cells as tasks for exploration; Tasks discovered in exploration	K-means clustering, Auctioning, and Optimal frontier based on a cost function
Kim et al. (2022) [44]	Frontier cells based on the uncertainty of neighbours	Heterogeneous k-means clustering
Tang et al. (2022) [43]	N/A	N/A
Bartolomei et al. (2023) [55]	Exploration: Clustered frontiers[67] Collection: Uncovered trails	Exploration: Optimal frontier based with minimal cost Optimal trail based with minimal cost
Yu et al. (2023) [50]	N/A	N/A

based on the models' geometry. This approach works by running a vertical line along the geometric model, and when an obstacle bisects line two the current cell will be closed and two new cells will be created. The result of this is several cells that can be covered in a boustrophedon motion (back and forth). The resulting cells are allocated to the robotic team, but the details of this are not given.

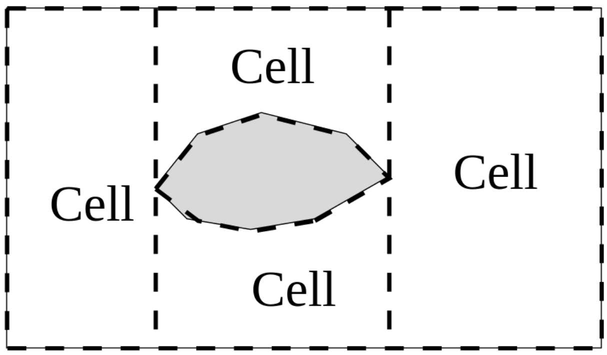

Figure 13. Boustrophedon Cell Decomposition [68]

A UAV coverage for first-response rescue and recovery with UAVs was implemented by Perez-imaz et al [47]. Hexagonal decomposition was used to decompose the task, this worked by overlaying the hexagon over the known geometric environment, with the hexagon size representing the sensor range. The tasks are allocated using K-means clustering, and each hexagon within a Graph is formed. While the approach considers a multi-robot team, the real-world experiments carried out only used a single UAV.

A purely offline approach is considered by Karapetyan et al. [42] in their approach to Autonomous surface vehicle coverage. The approach takes two approaches to environmental decomposition, Boustrophedon cell division as used by Ball et al. work discussed earlier [40], and a Dubins coverage solver (DCS). The DCS splits the environment into several passes to form a graph, outputting a Hamiltonian path. In Dubins coverage with route clustering this Hamiltonian path is then split between the team. Another approach Dubins Coverage with Area clustering segments the environment with Boustrophedon cell division, clusters cells together, and then uses the DSC to create the tasks. Task allocation isn't discussed.

All the works reviewed in this section consider the coverage footprint and the sensing platform to be inseparable, an individual robot has a sensing footprint of a specified size centred on itself. In a 3D structure inspection, this isn't the case, rather the sensor footprint will always be separate from the sensing platform. While the relevance of decomposing a 2D space into several regions to be covered doesn't have an obvious application to OWT inspection, it's worth considering the extendability of the reviewed approaches to the 3D structure inspection problem. One possible avenue for this is the decomposition of 3D space into a set of assignable regions. Both Darp and Boustrophedon cell divisions rely on 2D geometry to decompose the environment so extending them may not be simple. As the problem we're considering is with sensor coverage of a structure, segmenting the environment without consideration of the structure to be sensed would likely result in sub-optimal solutions. An alternative approach is to use these task decompositions as a component of a larger task decomposition approach. In the case of the OWT this could be the use of an area segmentation method to decompose the structure surface into continuous sections that can be assigned to the robots within the team Following this, the coverage problem can be seen as a set of single robot coverage problems of the assigned regions.

5.1.1.2 Frontier-based decomposition

The concept of frontier-based exploration was first introduced by Yamauchi [65] in 1997. These approaches harness environmental uncertainty to generate tasks or viewpoints iteratively, allowing exploration or coverage with limited knowledge of the environment. Viewpoints are usually selected based on some cost function aiming to maximise the reduction in uncertainty upon moving to it.

Kalde et al. [45] consider the problem of exploring an unknown environment with wheeled robots. The author's approach to this problem is frontier-based iterative planning with human guidance. At each planning interval, the environment is first decomposed through identifying tasks, either frontiers or humans. In this work, humans can assist robots in navigation by leading them. The work makes use of a parametric heuristic to equilibrate the frontier tasks and the human tasks. This parametric heuristic takes the form of a "Mixed Cost model", a cost value is computed for each agent-task assignment in a cost matrix. The cost function is formed from two components. A distance component is simply the distance for the robot to traverse to a task. A penalty component is composed of a time penalty and an orientation penalty, the time penalty being the time elapsed since the frontier's discovery, and the orientation being the smallest angle between the robot's orientation and the direction of the frontier or orientation of the human. Given the cost matrix, two greedy approaches were used, one fully decentralised, and one locally coordinated.

Colares & Chaimowicz [51] consider an instance of the exploration problem using a frontier-based approach. The task is decomposed by considering all known frontier cells' potential targets. For task allocation, a three-component cost factor was used in a distributed fashion. The first component is an "Information factor", which quantifies the potential information gained for visiting a cell based on its neighbours. A distance component was used, with two variables to change the behaviour by favouring close or distant frontiers. Finally, a coordination factor penalises selecting a frontier close to a known neighbouring robot. Given this, the optimal frontier is selected for each robot. The approach was implemented with two Pioneers 3AT wheeled robots to explore an indoor environment with success.

Another exploration approach is considered by Dong et al. [54]. This was implemented in an indoor environment with a team of up to six turtlebots. The authors consider an

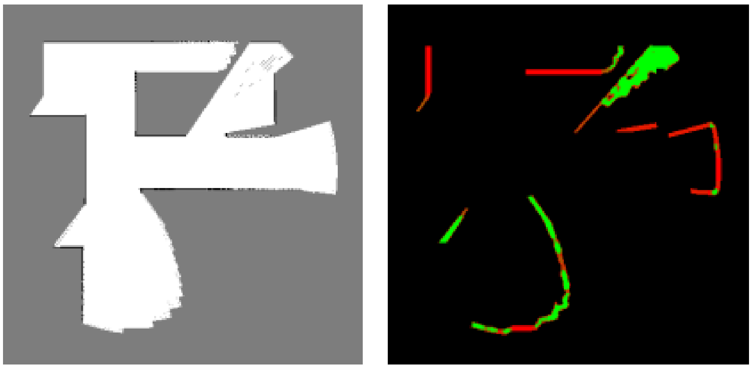

Figure 14. Colares & Chaimowicz's Frontier detection [51]

octomap representation projected on the floor plane and use the uncertainty to decompose the task. The approach also uses a validness map, which gives the possible poses of a sensor. The voxel positioned on frontiers is sorted by uncertainty in a priority queue, and then the validness maps are considered to find poses that rays pass through the voxel. The pose with the optimal validness is taken, where validness is composed of the deviation from 0 degrees and a function of the optimal distance. On the view being selected, all voxels within its view are removed from the queue. This process repeats until a specified number of views are generated. As for the task assignment, the problem is viewed as an Optimal Mass transport problem. This problem is formulated and then discretised to an objective function with three components to be minimised. A compactness component penalises spatial scattering of assigned tasks, a distance component minimises travelling cost, and a capacity component ensures robots can complete only some tasks within their capacity for a given interval. This is then optimised through a modified k-means clustering algorithm was used. Zhang et al. [49] consider area coverage using UAVs with mobile charging stations. The continuous area is initially split into tasks through a gridmap decomposition. They then make use of the Divide Areas based on Robot's Initial Positions algorithm (DARP) modified to avoid discontinuities via edge detection. This effectively allocates areas of the grid for coverage. The authors made use of crazyflie UAVs adapted for mobile charging, and wheeled mobile charging stations.

As discussed previously, Bramblett et al. [52] were concerned with the problem of coverage of tasks in an unknown environment, and as such planning has to occur for both tasks. For exploration, a Sobel operator is used on the occupancy grid to identify frontiers based on the gradient between known and unknown space. Those frontiers representing obstacles are discarded. Naive to and in tandem with the edge detection, the unknown environment is clustered using K-means clustering for each robot. These are auctioned to the robots in a centralised manner. The robots then act in a greedy manner using a cost function that favours closer tasks, but those tasks outside of the robot's assigned task area a penalised by distance from the task area. In regards to the coverage aspect, the tasks are "Decomposed" from the environment in the sense that they're discovered during exploration. The decision logic for coverage is given in Figure 15. This Search involves seeking a robot that didn't rendezvous, it is likely in such scenarios that the robot found a task. Exploitation is the act of working on a task. The authors implemented the approaches on three Husarion ROSbot 2.0 UGVs.

Kim et al. start with a geometric representation of the environment[44]. Their approach considers a degree of heterogeneity, in the sense that team members have different sensor ranges. The tasks are generated based on the smallest sensor range while grouping unknown frontier cells. Task assignment is treated as a clustering problem. They extend K-means clustering into their Heterogeneous clustering algorithm. This clustering algorithm considers both the spatial proximity of two agents and the weighted distance based on the sensing ability of the specific robot.

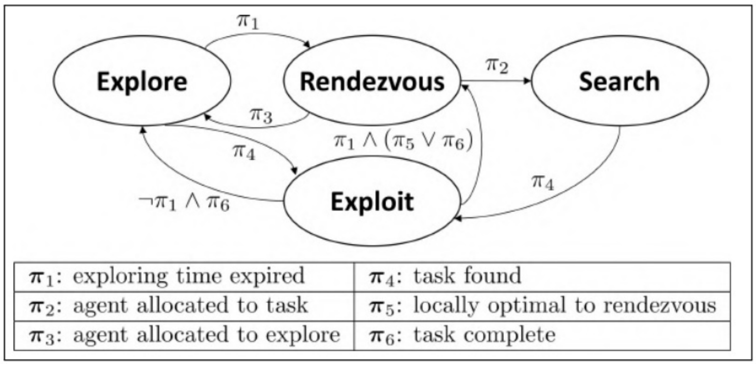

Figure 15. Bramblett Decision logic [52]

Finally, Bartolomei et al. discuss the exploration of forests with a team of UAVs [55]. This method has two modes for the robots in the team, exploration and collection. Exploration is as expected, focused on obtaining new knowledge of the environment, while the collection is concerned with cleaning up the "trails" of unexplored areas left by exploration. Exploration tasks are decomposed from clustered frontiers. The clustering algorithm used isn't specified. Given the cluster centroid, candidate views in a cylinder focused on the centroid are considered. The view with the highest coverage of the cluster is then selected as the optimal view for that cluster. A set of these views with optimal views over the clusters from the decomposed tasks for the task allocation. These clusters also undergo a classification, with those representing trails being semantically classified as such based on their isolation. A mechanism for declaring areas of interest for a team member takes place between two robots if they are inside communication range. This area of interest is used by the robot to select tasks from the previously discussed set of tasks. The assignment for exploration is based on a cost function with four components, a distance and a change of direction component, a label component to penalise trails, and a component to encourage those views near the area of interest. As for those team members assigned as collectors, they cover trails through a cost function only considering distance and proximity to the area of interest.

The value of frontier-based approaches is their application in coverage when knowledge of the environment is limited, and there are uncertainties in regions of the map. When considering the applicability of these methods to the OWT inspection problem, it's important to take into account prior assumptions. If the entire environment is considered known a-priori frontier-based approaches have few clear benefits. Frontier-based approaches can work in a decentralised manner, potentially performing better where communication may be limited. In situations where the turbine's location may be uncertain such as with floating turbines in areas with large currents, there could be value in using a frontier-based approach, ideally while still accounting for the known general geometry of OWT.

5.1.1.3 View decomposition

While there's only one example of this decomposition approach, it proves to be one of the most applicable. View decomposition attempts to, for a known structure in a known location, find a set of views that optimally cover the surface of the said structure with the sensor footprint

Dornhege et al.[53] tackle a coverage search problem with a team of wheeled robots. The authors consider an Octomap environmental representation with a known search set of voxels. For each voxel in the search set several random vectors are generated, and then ray tracing is used to find a set of grid cells that represent possible sensor states along the vector. The corresponding grid cells from the ray tracing are used to increment a utility function for the grid cells, this is done for all cells in the search set to create a utility map across all accessible sensor states. Those states over a given utility threshold are added to a set of useful sensor states. For task allocation, the problem can then be considered a set cover problem, given a set representing and search set and a set of sets representing those observers' cells for a sensor position, finding the minimal set of sensor positions that cover the environment. Dornhege et al. used a variant of the planner Temporal Fast Downward to solve this set cover problem as a planning problem. Alternatively, the authors used a greedy approach in which the views were selected for each robot iteratively based on the cost. The cost in the greedy approach can either balance the view utility and the travel time or be the travel time. This task allocation method also ensures a high-level path plan.

While this approach may be the most readily applicable to the OWT inspection problem, it would need modification for this use case. The authors consider only wheeled robots and choose the possible sensor states based on this assumption, for the OWT inspection, USV and UAV would be necessary. USV sensor positions have similar specifications to wheeled robots, being limited to surface level, but UAVs are not bound to the surface. The UAV's capability to reach almost any position in space would possibly make the computational complexity of the approach Dornhege et al. [53] infeasible. The authors also don't account for camera orientation and distance concerning the surface of the structure to be inspected. To apply a method similar to theirs to an OWT inspection would need to account for sensing quality by ensuring the sensor is positioned to capture useful information, which can be achieved by requiring a certain proximity and orientation to the surface being captured. Applying these requirements would benefit the computational complexity of the solution by reducing the number of possible sensor states to those that fulfil the requirements. This approach also has no accounting for uncertainty in the environment, requiring a full a-priori model and no dynamics.

5.1.1.4 Hierarchical Decomposition

Masehian et al. [41] give an interesting hierarchical, heterogeneous approach to coverage of an environment with limited sensing capabilities. In this approach, the tasks for some robots are decomposed as robotic tasks for others. In their approach, there are three classes of robots, each with a different sensing capability and differing behaviours. A Quadridirectional robot with four quadridirectional sensors is not assigned tasks as such, rather it initially starts a boustrophedon motion across the environment. The quadridirecital robot will identify obstacle and wall boundaries, this represents the task decomposition for the second robot class, the Boundary follow robot equipped with a radial sensor. The assignment for a boundary follow robot class is the robot with minimal distance. This robot will follow the boundary and if a sensed point doesn't align with the last two points a task will be created for the last robot in the hierarchy. The Gap robot can identify gaps between obstacles within a radius and is hence assigned the potential gaps identified by the boundary follower.

While this work addresses a very specific case, it touches on an interesting aspect of the OWT inspection problem. There is potential when addressing the OWT inspection to utilise heterogeneity of capabilities to increase the quality of inspection, while this isn't necessary for the problem as defined by us, it could be of practical use in industry and act as a variant of the inspection problem. That is to identify areas of interest, identifying a damaged area, from a distance with a suitable sensor, and then using a team member with a different sensor to elaborate on the identified damage by moving closer. This behaviour, while not identical, resembles the approach used by Masehian et al. [41] in the generation of a hierarchy of tasks based on the sensing capabilities of other team members.

5.1.1.5 Pre-decomposed and non-decomposing approaches

Some works reviewed didn't consider the decomposition of the environment, and others like the reinforcement learning approaches don't view task decomposition as a separate problem from motion planning, as such the specifics of these will be discussed in greater detail in the motion planning section.

Song et al. [46] focused on the use of AUVs for full sensor coverage. In their approach, they assumed the environment is pre-decomposed in sub-regions, and initial task allocation isn't considered. Rather the work focuses on motion planning and fault tolerance, the former discussed in Section 5.1.2, and the latter discussed in Section 6.

In Sharma et al. the environment is considered both pre-decomposed and task areas are pre-assigned [48].

In Tang et al. [43] a worker station approach to coverage is given. The environment is decomposed into a grid, but the resulting cells cannot be viewed as tasks. The authors use a reinforcement learning approach, so tasks are not allocated as such. The reinforcement learning approach's action space is concerned with the linear and angular velocity of a single robot, so this will be discussed in section 5.1.2. The approach was implemented using a skid-steer wheeled robot as a station and two differential-driven wheeled robots as workers.

5.1.2 Motion planning

As previously described motion planning is concerned with devising paths to facilitate locomotion to, and completion of, the previously planned tasks. In some cases motion planning alone is used without task allocation, for example greedily covering a geometric map may not have discrete tasks, only motion planning. Path planning is defined by Kavraki & LeValle [69] as finding a collision free path from an initial pose to a goal pose. The problem being considered here closer resembles the multi-goal path planning problem proposed by Wurll et al. [70], finding a collision free path connecting a set of goal poses while minimising some cost function. Solutions often consist of two tiers, a global and a local planning. Global planning approaches solve the multi-goal path planning problem at a higher level, sometime forgoing consideration of collision altogether. While local planning closer resembles the traditional path planning problem, concerned with a path from an origin to a goal while avoiding collision and minimising some cost.

Table 8. Summary of Motion Planning Approaches

Paper	Motion Planning Approach
Ball et al. (2015) [40]	Search-based lattice planner with a local pure pursuit controller
Kalde et al. (2015) [45]	Potential field on a gridmap
Song et al. (2015) [46]	Generalized Ising model with local and global navigation mechanisms
Colares and Chaimowicz (2016) [51]	Not specified beyond iterative task selection
Dornhege et al. (2016) [53]	Single TSP problem solved with Temporal Fast Downward planner or Lin-Kernighan heuristic. Or a single greedy approach split for the number of robots
Perez-Imaz et al. (2016) [47]	Dijkstra's algorithm on a hexagonal graph with lawnmower pattern
Sharma et al. (2016) [48]	Directional motion and nature-inspired algorithms
Masehian et al. (2017) [41]	Different policies for different robotic classes: boustrophedon motion, Boundary following, and guide path following with obstacle avoidance
Karapetyan et al. (2018) [42]	Dubins coverage solver with TSP problem solving
Dong et al. (2019) [54]	Christofides algorithm for TSP approximation with path smoothing
Zhang et al. (2019) [49]	Spanning tree coverage algorithm
Bramblett et al. (2022) [52]	A* path planning algorithm with iterative frontier-based tasks
Kim et al. (2022) [44]	Genetic algorithm for TSP problem with A* algorithm and B-spline for path computation
Tang et al. (2022) [43]	Reinforcement learning with multi-layer perception for policy network. Action space being angular and linear velocity
Bartolomei et al. (2023) [55]	Trajectory generation integrated with task allocation
Yu et al. (2023) [50]	Reinforcement learning with a Multi-tower-CNN based Policy decentralised. The action space represents a global goal. Local navigation is achieved with A* algorithm

An example of this distinction between global and local path planning is given by Ball et al. [40]. In this work, the global planner makes use of a search-based lattice planner to find the best path considering both the cost of motion primitives and minimising the cost of traversing a costmap while avoiding obstacles. A local pure pursuit controller is used for the global planner path if followed optimally, using two PI controllers to minimize the error in the robot position and the global planner path. If a collision is detected in the global path, the local pure pursuit controller can reject it, and ensure the global planner recomputes a new path. Kalde et al. [45] describe their motion planning as done using a potential field propagated on the gridmap. Another two-level approach to motion planning is considered by Song et al. [46]. As discussed earlier the authors consider a multi-layer grid representation, and their local navigation works on the lowest level of this grid. They describe their navigation as being based on a generalized Ising model. The cells within the Ising model have one of three states, obstacle, explored, and unknown. Local potential energy is formed from the state of the cell and its neighbours. A component of the local potential energy is a constant potential energy field that encourages back-and-forth motion for coverage. For each robot, the target is therefore the cell with the highest energy potential. But it may be the case that a robot could get caught in a local minima. The authors account for this eventuality with a global navigation mechanism. The global navigation works on a coarser grid than the local navigation, using a low-dimensional probability vector to restore environmental information for the coarser grid. Then much as with local navigation, a target is selected and navigated towards, until local navigation is possible. Colares & Chaimowicz's work doesn't discuss the specific motion planning approach implemented beyond the iterative task selection previously described [51]. Dornhege et al. approach path planning by treating it as a set of single travelling salesman problems (TSP)[4], given the result of their set cover problem. The authors solve the TSP problem for each subset using either a Temporal Fast Downward planner [71] or a Lin-Kernighan heuristic [72]. An alternative method the authors use is extending the single robot greedy allocation, by taking a single greedy plan for the environment and splitting this path for the number of robots. These approaches give a global path plan for traversing a topological graph, but details on path planning to account for the structure of the environment itself are sparse. Given the topological graphs decomposed and clustered from the hexagonal graph by Perez Imaz et al. [47], the authors ensure a lawnmower pattern within the hexagons by using parallel lines intersecting with the hexagon to create nodes and then using Dijkstra's algorithm to generate the path. An optimal angle of the path for each hexagon is found to minimise the complete coverage distance. In Sharma et al. [48] the authors have split the environment into several task areas to be covered. Until the entire areas for a robot are covered, an iterative path planning approach is taken. The robot at any given iteration will randomly choose one of two motion policies: Directional motion or a Nature-inspired algorithm. Directional motion has two variations: A directional scattering effect moves in the direction of a "cluster head" selected randomly to encourage exploration; and a zig-zag search effect, in this approach the cluster head is chosen dynamically, ensuring more random and less directional motion. The authors modified Particle swarm optimisation, Bacteria foraging algorithm, and Bat algorithm for their multi-robot exploration. The particle swarm optimisation algorithms were found to perform optimally for exploration. The motion planning in Masehian et al. [41] takes into account the heterogeneous nature of the team involved. Each of the three robotic classes has a different motion planning policy. The quadridirectional robots use boustrophedon motion to cover the task area. During boustrophedon motion the robot may get trapped in a corner, this is resolved by referring to its observation history and finding any gaps it may have passed since its last row, and will backtrack to this point and continue in the direction of that gap. The boundary follower robot follows the boundary of an obstacle or wall. This is done by considering a band, representing an optimal distance from the obstacle, around the edge of said obstacle that the robot should stay in. The Gap robot's motion planning can be considered a classical

[4] The travelling salesman problem is a well-known mathematical and computer science problem that can be summarised as "Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

path planning problem, given a task, to find the optimal path from the current location to the path. The robot considers a guide path, being the direct line from the robot's position to the task. On the gap robot sensing an obstacle along the guide path, the robot will randomly decide whether to go left or right. around the obstacle. Karapetyan et al. [42] involve using the Dubins coverage solver from Lewis et al. [73], an approach that solving the TSP problem for generated rows while accounting for dubins constraints. A TSP problem is also considered by Dong et al. [54] after selecting some tasks for a given robot. For this purpose, the authors used the Christofides algorithm to calculate a TSP approximation, the path of which is then smoothed. Examples of paths over an iteration are shown in Figure 16.

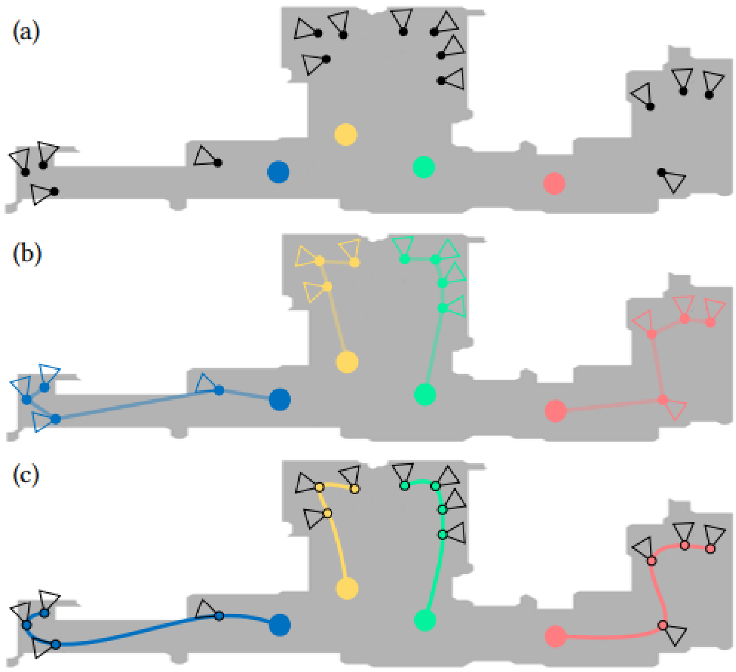

Figure 16. Paths over an iteration in the work of Dong et al. [54] a) The decomposed viewpoints and the robot starting positions b) The multiple TSP paths from the robots c) The smoothed paths for the robots

Zhang et al. [49], previously having decomposed the environment into a set task of task areas, produces a coverage path common to offline coverage of gridmap environments a Spanning tree coverage algorithm. Spanning tree coverage capitalises on the grid structure by grouping sets of four cells together, and considering these supercells to find a spanning tree. This spanning tree can then be traversed by the robots in the team, forming a cycle across the entire environment. Bramblett et al. [52] consider the iterative frontier-based task, as discussed earlier. To navigate to these tasks they use the A* path planning algorithm. Kim et al. [44] used a topological graph, as discussed earlier, and now considers the TSP problem. Their approach to the TSP problem is a genetic algorithm [74]. The path between the points in the TSP solution is then computed using an A* algorithm, and a spline function is taken of that using B-spline. Tang et al. [43] is concerned with the use of reinforcement learning coverage, as was stated earlier, this doesn't have task planning as such. The authors describe their multi-agent reinforcement learning problem as a "Decentralized Partially Observable Markov Decision Process". They follow a centralized training and decentralized execution paradigm. The observation space consists of the robots' information, information from its sensors, and information from those robots within communication range. The action space consists of linear and angular velocities. Their reward function is composed of four components. A completion reward is given for finishing coverage, and the second component approximates worker capacity, giving a negative reward if the energy capacity of a robot is beneath a threshold. During the training phase, robots can continue coverage with depleted capacity. The third reward is a negative reward for collisions, and the final component is a constant negative reward to encourage an optimal time. The authors use a Multi-layer perception for their policy network. Path planning is computed with the task allocation by Bartolomei et al. [55], and the trajectories are then generated using the approach proposed by Zhou et al. [75]. Another reinforcement learning approach is considered in Yu et al. [50]. For this purpose the authors make use of an Asynchronous variation of the Multi-Agent Proximal Policy Optimization algorithm [76]. The task is modelled as a decentralized partially observable Semi-Markov decision process. A Multi-tower-CNN based Policy is used for each agent. The action space is a global goal, but atomic actions enact this goal using the A* algorithm. A three-component reward function is used, a coverage reward proportional to the discovered area, a success reward when a threshold coverage is achieved, and an overlap penalty for repeating coverage. The team members communicate extracted features from a CNN local feature extractor to one another.

5.2 Discussion

When discussing the applicability of decision-making approaches to domain applications resembling OWT inspection, an initial question would be whether model-base or model-free approaches are better suited to the task. In an OWT inspection, the orientation of the turbine is not tracked, multipath error makes GPS positioning unreliable[38], and in the case of floating OWTs, the entire structure can excur up to 35% of the depth of the mooring system [28]. A case could be made that a model-free approach would better account for the uncertainty of the turbines' pose. As was previously proposed in Section 4.7, the OWT inspection task lends itself to 3D environmental models, with semantics labels for a search set representing the structure to be inspected.

Given a 3D voxel-based environmental model with a subset of voxels representing the search set, what planning approaches should be used to provide sensor coverage of the search set with a team of robots? The answer to this question depends on the prior knowledge assumed. Given the full environmental model a priori, a task decomposition approach such as Dornhege et al may be used [53]. Dornhege et al. were concerned with wheeled robots however, and the set of reachable voxels along the ground, applying the same approach to UAVs OWT inspections would have a much larger set of reachable voxels greatly increasing the computational cost of the set-cover solution. One solution in lessening this problem is to apply bounds on which voxels are considered, not based on voxel reachability, but on proximity to the search set. As for task allocation, the approaches in the literature are quite limited, approaches like greedy allocation or K-means clustering would work, but may not provide near-optimal solutions. As for motion planning, given the assigned tasks for the team members, an open TSP approximation should be computed over the task assigned, generating a high-level path plan [53]. To follow this plan a 3D costmap in the form of an ESDF should be used to prevent collisions with obstacles [55], and the path considers a tradeoff between the length and the cost. None of the task allocations takes into account the dynamics of the OWT environment, it may be the case that strong winds may increase the time to reach a given task, and this could be accounted for in task cost. Additionally, none of the approaches considered disruption of performing the task itself, say a strong gust of wind or a wave disrupted the image capture process for a given robot, this would need to be reassigned to the team dynamically.

As was previously discussed, an accurate model of the environment can be unrealistic in an OWT inspection due to the mobility of floating OWTs and the dynamic nature of the nacelle yaw and blade rotation. It may be necessary to treat the task as an exploration problem, following a similar approach to task decomposition as Dong et al. [54]. As with the approach of Dornhege et al., the issue with the approach of Dong et al. is its assumption of wheeled robots. Another issue is the approach's focus on exploring an entire environment rather than a search set of interests. This represents an area of future research, if the search set isn't known a priori (as in Dornhege et al. [53]) task decomposition requires the inference of the search set from sensor information (it requires the team to identify the OWT as the search set in an online manner). Assuming such an approach to identify the search set at each iteration, a frontier-based approach can be applied to generate tasks at the frontier of the known search set to identify more OWT. Task allocation and motion planning can be achieved in much the same way as for model-based approaches. As the general geometry of the turbine is predictable, this could be utilised to allow for these tasks to be generated. This coverage of an unknown environment with significant prior knowledge, a structure-informed coverage of an unknown environment, is an interesting area of research and as far as the authors are aware hasn't received attention so far.

6 Coordination

This section aims to provide an answer to sub-question 3 from Table 2: *What is the most suitable strategy to effectively coordinate a multi-robot system for domain applications resembling offshore wind inspection?* Coordination has many definitions in the literature. Farinelli et al. [37] consider coordination to be cooperation in which team members take actions in consideration of other team members "in such a way that the whole ends up being a coherent and high-performance operation". Yan et al. [36] defined it as multi-robot planning to deal with resource conflicts, be that conflicts in space, tasks or communication media. Cao et al. [77] define coordination as "Given some task specified by a designer, a the multiple-robot system displays cooperative behaviour if, due to some underlying mechanism (i.e., the "mechanism of cooperation"), there is an increase in the total utility of the system". While it is true to say that in all the approaches discussed, the team members coordinate to increase the utility of the system, this section will focus on those coordination mechanisms necessary to resolve issues brought about by the dynamics of a task environment, or the online nature of an approach. The decision-making, even if aware of, and therefore coordinating with other team members is discussed in the section prior. We will discuss here the necessary communication mechanisms required to facilitate this coordinated decision-making. To succinctly evaluate the coordination mechanism in the works reviewed, they were charted in Table 9. This notes whether the works take an online or offline approach to planning, details of communication, whether the teams are heterogeneous or have an inter-team hierarchy, and whether fault tolerance is considered. These categories provide context to how the team members coordinate to complete their tasks.

6.1 Collaboration in decentralised Planning

In centralised works, a central planner assumes knowledge of the state of the environment and dictates tasks based on this global view. Such an approach is powerful, and can find optimal solutions if feasible, but is very rarely feasible. Communication issues, or failure of the planner, have a catastrophic effect on online coverage with a centralised planner. Hence, many approaches attempting to perform coverage with communication restraints will implement distributed approaches to the problem. Additionally, decentralised approaches can handle a larger number of robots by distributing the computation across the robots. In our review, eight works were identified to use decentralised planning. As discussed earlier, Kalde et al. [45] consider a decentralised frontier-based approach. The robots share a map and their locations. With the locations of the robots shared. A task robot cost matrix is formed by each robot using the map, however, the matrix only considers those robots local to the computing robot in order to optimise the assignment. While centralised task allocation is considered by Song et al. [46], a decentralised approach was taken to handle unequal task sizes. In this approach, once a robot completes its initially assigned task, it starts a cooperative game with those robots local to it in need of help. Cooperative games are one of the state-based potential games described by Marden [78]. In their case, the cooperative game is said to be made up of coalitions of two robots, each with a payment balancing the distance to the task of the receiving robot and the remaining uncovered cells

Table 9. Table of data extracted for coordination research question

Literature	Online /Offline	Communication	Hierarchy	Heterogeneity	Fault tolerance
Ball et al. (2015) [40]	Online	Extrinsic	No	Homogenous	Not discussed
Kalde et al. (2015) [45]	Online	Extrinsic	No	Homogenous	Not discussed
Song et al. (2015) [46]	Online	Extrinsic	Dynamic Hierarchy	Homogenous	Not discussed
Colares and Chaimowicz (2016) [51]	Online	Extrinsic	No	Homogenous	Not discussed
Dornhege et al. (2016) [53]	Offline	Extrinsic	No	Homogenous	Not discussed
Perez-Imaz et al. (2016) [47]	Online	None	No	Homogenous	Yes
Sharma et al. (2016) [48]	Online	None	No	Homogenous	Not discussed
Masehian et al. (2017) [41]	Online	Extrinsic	Yes	Heterogeneous	Not discussed
Karapetyan et al. (2018) [42]	Offline	None	No	Homogenous	Not discussed
Dong et al. (2019) [54]	Online	Extrinsic	No	Homogenous	Not discussed
Zhang et al. (2019) [49]	Online	Extrinsic	No	Heterogeneous	Not discussed
Bramblett et al. (2022) [52]	Online	Extrinsic	No	Homogenous	Not discussed
Kim et al. (2022) [44]	Online	Extrinsic	No	Heterogeneous	Yes
Tang et al. (2022) [43]	Online	Extrinsic	No	Heterogeneous	Not discussed
Bartolomei et al. (2023) [55]	Online	Extrinsic	No	Homogenous	Not discussed
Yu et al. (2023) [50]	Online	Extrinsic	No	Homogenous	Yes

in the task. Given this, the optimal coalition is selected by the initiating robot, and it will assist in the completion of the receiving robot's task. This process is shown in Figure 17

In Colares and Chaimowicz's work, a single utility function taking into account robot positions is used to coordinate the decision-making [51]. The robots communicate implicitly through a camera identifying the robots' positions and poses in the locale. After this implicit identification, the robots share their maps and pose information, and the initial robot communicates an estimated pose for the spotted robot relative to itself. Using this information, map stitching is used to combine the map information for both robots. The cost function to assign tasks for a given robot is performed in a decentralized manner, with a cost function composed of the value of a task based on its neighbours; the distance to the task; and most relevant to this section, a coordination factor disincentivizing allocation of task close to identified neighbouring robots. Bramblett et al. [52] consider exploration and task coverage in an unknown environment, under the constraint of limited communication range. Therefore the team is required to rendezvous intermittently to share environmental and task information. For the exploration phase K-means clustering is used to assign task areas to robots whenever they're able to communicate. The clusters are auctioned in a centralized manner. A rendezvous mechanism is used whenever all robots are connected, it finds a valid navigable point with minimal distance to the centroids of the robots' partitions. After some time, the robots will navigate back to this rendezvous point to share information. If a task is discovered during exploration, a rendezvous policy representing the cost of rendezvousing is formed from the potential path to the rendezvous and the unknown space it passes through balanced with the subtraction of the global expected path length from the length of the path explored and the task length. The approach used by Tang et al. [43] for coordination takes the form of using two classes of robots for the coverage problem. The authors consider a worker station approach to coverage, the workers have limited energy, while the stations have unlimited energy and the ability to replenish the workers. They consider this problem as a multi-agent reinforcement learning problem. The robots can communicate and use this communication to form their observation of the environment. The observation space is composed of three components: Zero-range observations are the position velocity and energy of the agent; Perception range observations provide information about obstacles and agents within the perception range; and communication

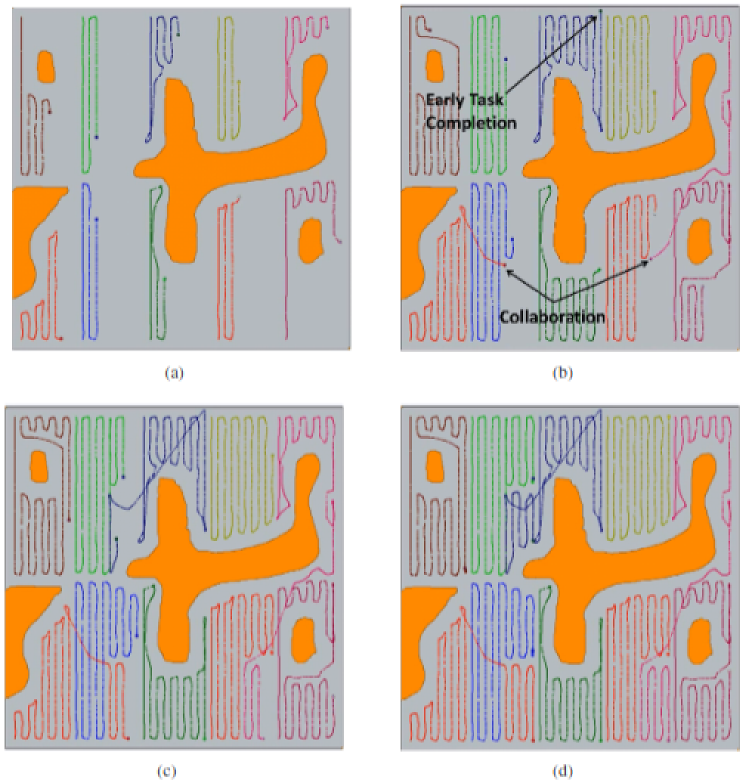

Figure 17. Kalde et al. cooperative game collaboration [45]

range observations include information about the agents within the communication range. The authors make use of centralized training decentralized execution (CTDE), in which their critic network has full knowledge of the environment state, but the actions of an individual agent are based on local observations. Additionally, a two-stage curriculum is used for training, with a simple environment of one actor and one station used initially until convergence, followed by an environment with two stations and four workers [79]. Multi-layer perception policy networks were shared between robots of the same class, but differ between the worker and the stations to account for their differing abilities. A visualization of their deep reinforcement learning pipeline is given in Figure 18. In the work of Bartolomei

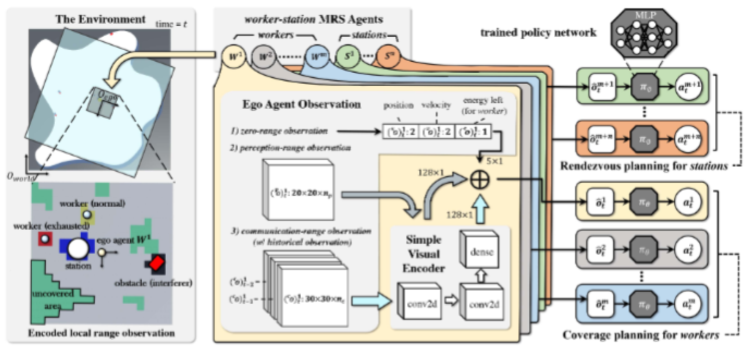

Figure 18. Tang et al. deep reinforcement learning pipeline [43]

et al. [55], a team of robots complete exploration with roles, exploration, and collection. The members of the team can vary roles based on the needs of the team, exploration involves seeking large patches of unexplored frontier, while collection prioritizes small unexplored sections of the map surrounded by covered areas. The robots, by standard, take the role of an explorer, but given a threshold number of disjointed unexplored regions close to the robot, it will switch to collector mode. Another approach that considers the problem of multi-robot reinforcement learning for exploration is considered by Yu et al. [50]). The authors noticed that the existing literature primarily focused on agents acting in a fully synchronous manner, and this is a problematic assumption for real-world adoption. As such they use an asynchronous Multi-agent Proximal policy optimisation approach to training. Each robot has its policy network, therefore behaviour will vary between members. To better facilitate communication between the robots, a CNN was used for feature extraction on the local environmental map, and these features were then shared between members of the team. To further facilitate collaboration, the reward function takes into account the overlap between the coverage of the robot and the rest of the team, discouraging repeated coverage of the same area.

6.2 Communication

Nordin et al. [16] identified several issues with communication in offshore wind turbine environments: there's likely to be no cellular network due to the distance from land; normal satellite communication has a high latency that would hinder online planning; and although there now exists real-time satellite communication, in the form of Inmarsat SwiftBroadband satellite service, may be hindered regardless due to proximity to the towers [80]. The authors' proposal is the use of USV to connect to the satellite service positioned away from the towers, which may then communicate with the UAVs through an ad-hoc Wi-Fi network. Communication was classified into two categories by Matric [81]. Direct communication is purely communicative, transmitting data from one agent to another or a central planner. Indirect communication is based on observation, a robot could for example sense the tracks of another, communicating the fact an area has been visited. While all online cooperative approaches in this review make use of explicit communication, some additionally make use of implicit communication. In the approach of Ball et al. [40] for Broadacre agriculture, the real-world implementation uses a 3G mobile data connection to the internet for communication between the robots and a central planner using a ROS middle-ware. A map is shared between the robots in Kalde et al., though the communication mechanism for doing this isn't described [45]. Song et al. [46] make use of the player/stage simulator, which allows modules to communicate through TCP. Colares and Chaimowicz [51] used ROS as a middleware for their real-world experiments. Communication isn't discussed in Dornhege et al. [53]. The approach of Perez-imaz et al. [47] has robots communicating their position with a central planner as an approach to fault tolerance, this communication is achieved again through ROS. Masehian et al. [41] consider communication between the robots and the central planner to be of unlimited bandwidth, assuming ideal conditions. Karapetyan et al. [42] assume no communication capabilities, with a purely offline approach. Dong et al. [54] consider communication between a central planner. Zhang et al. [49] also consider a centralized planner, though details of the implementation are sparse. Bramblett et al.[52] make use of a disk constraint to simulate communication range constraints. Kim et al. [44] make use of robofleet for communication [82], with communication used for fault detection in the team. Tang et al. [43] also considered the communication range.

6.3 Fault Tolerance

Fault tolerance is a crucial aspect of building robust multi-robot systems. Multi-robot systems provide inherent redundancy, by allowing other robots to complete the tasks previously assigned to the faulty robot. A reality of working outside of simulation is that eventually, failure will occur. In the reviewed work robotic failure was explicitly discussed in two of the works, that of Perez-imaz et al. [47] and Kim et al.[44]. Other approaches, such as frontier-based exploration approaches, might have some inherent robustness to failure as a result of iterative planning. In Perez-imaz et al. [47] when a robotic failure occurs the hexagon cells can be reallocated to members of the team. Kim et al. Similarly, when a robot failure is detected the system recomputed the coverage task decomposition with the smaller team size [44]. Of note with this is this will result in repeated coverage of areas already covered by the failed robot. The reallocation mechanism used by Kim et al. is shown in Figure 19. Both these works focus on recomputing offline task allocations.

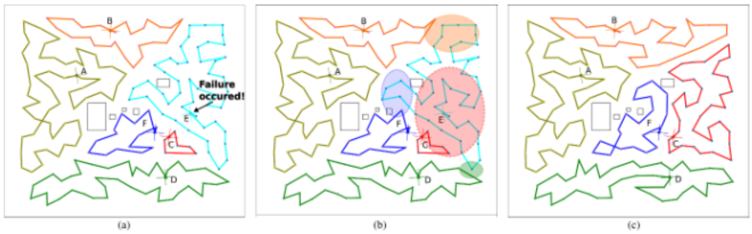

Figure 19. Kim et al. fault tolerance reallocation [44] a) The robot with the cyan path experiences failure b) The waypoints are reassigned to neighbouring robots c) The TSP paths for the new allocations are computed ensuring coverage

6.4 Discussion

It's hard to imagine a robust real-world coordination framework using only the approaches discussed in this work. Such a coordination framework would need to account for communication downtime, robotic failures, and possibly heterogeneous capabilities. One noticeable trend, albeit with a small sample size, is a focus on reinforcement learning approaches in recent years. Yu et al.[50] note that reinforcement learning approaches, when compared to traditional planning approaches, can effectively produce complex strategies and after training proves computationally inexpensive. Regardless, the majority of literature considers planning-based approaches. There seems to be potential for future research in both classes of algorithms. Fault tolerance and dealing with communication constraints are open avenues for research, with only two of the reviewed works explicitly considering faults [44][50]. Coordination for heterogeneity has received very little focus from the research community, with current works concentrating on worker-station relations ships [43][49], or sensing range and locomotion speed[44]. Masehian provided a highly specific case of robot mapping with different forms of sensor [41], but beyond that, there has been no work focusing on robots' semantic capability in regards to completing tasks or traversing the environment. Coverage of tasks with semantic requirements for both completion and area traversal by heterogeneous teams is still an open area of research. Dynamic environments were of little focus in these works, with only Kalde et al. [45] considering such, with mobile obstacles. Dynamic environments are a potentially interesting area of research for OWT inspection due to the mobile nature of wind turbines, even more so for floating offshore wind. Another area of interest that hasn't been considered is dynamic tasks. Considering the problem of covering a single OWT, the task of visually covering the blades may not be at one static coordinate. If the wind turbine is in operation, the coverage task will be constantly moving predictably. Such a dynamic coverage task also provides some future direction.

7 Future Work

None of the works reviewed in this paper would enable coverage for OWT inspections alone. A comprehensive multi-robot coverage system would require the combination and extension of existing techniques. Several potentially useful aspects of the reviewed approaches have been identified in the previous sections. This section will attempt to synthesise identified approaches and limitations into several areas of future research for OWT inspection.

7.1 Task decomposition with areas of interest

One aspect of OWT inspection coverage that hasn't been addressed in the existing research is coverage with varying required degrees of quality. In an OWT inspection, certain sections of the turbine may require greater focus than other sections. Usually, the tower is of less interest than the turbine blades. To address this, it would be necessary to select and extend an existing environmental representation to account for varying coverage requirements across the structure. One method for achieving this would be through a bespoke semantic label applied to sections of the environment. Assuming a voxel-based representation, this may be a property for each voxel that specifies, for example, a required proximity for coverage. This semantic label would then need to be accounted for when decomposing the task in a set of views, only considering a voxel covered if a view fulfils the requirement encapsulated within it. An alternative approach is the use of multiple resolutions depending on the degree of interest in a section. This wouldn't inherently apply proximity requirements, but it would ensure more thorough coverage within a specified region. This could be achieved through Octomap [62], and would allow the use of the approach of Dornhege et al. [53] without modification. Combining these two approaches may prove even more efficient in ensuring both thorough and high-quality sensor coverage. This all assumes the area of interest is known a-priori however. To identify areas of interest in an unknown environment, some form of semantic area detection would be necessary, maybe through object detection techniques.

7.2 Limited knowledge approach

An interesting area of research is the possibility of using the known geometry of the turbines in an otherwise unknown environment. The geometry of a turbine will always be available before an inspection, and intuitively, an approach should be able to exploit this knowledge. None of the reviewed works considered the case of geometric structural knowledge in an otherwise unknown environment. The most obvious use-case for this is in floating OWT inspection, where the turbines drifted from the centre of the moorings, but just because the turbine has moved a certain amount, doesn't mean the environment is now completely uncertain. This could be achieved by considering a problem of two layers, exploration within a small sub-area of the environment to localise the turbine, and then model-based coverage of the now-known structure.

7.3 Dynamic Tasks

In all the reviewed works, the area or structure to be covered was static. By having a moving structure such as OWT blades, task planning and motion planning would be significantly complicated, and a novel environmental representation would be necessary to represent the moving tasks. One possible solution for the blades is to use one team member to constantly observe and track the blade's positions, and then use other team members to complete the coverage to the required proximity and quality. This problem identified one key issue with using a voxel-based representation alone, in that voxels tend not to represent semantic objects but just occupancy, so when the physical object moves some mechanism would be necessary to ensure any label is transferred to the new voxel representing that physical object.

7.4 Limited Communication

As was discussed by Nordin et al. [16], communication is an issue in the OWT environment. While most of this work has been focused on UAV coverage of turbines, it's the case that UAV batteries are currently limited, and any feasible implementation would require the use of USVs for UAV deployment. As Nordin et al suggested, the use of a USV may also play a role in solving communication for OWT coverage. This slightly resembles the worker-station approach of Tang et al. [43]. An approach that strategically places a USV distant enough from the turbines for satellite communication interference from the turbines while providing a temporary wireless network for the UAVs in the team may solve this problem. This would require a new approach to planning, accounting for USV placement, and possibly requiring a rendevous mechanism with UAVs working outside of the network and then returning.

7.5 Heterogeneous Sensing/Locomotion Capabilities

Heterogeneity among robots was lightly touched on in the reviewed work, but to fully harness the capabilities of a diverse team new planning approaches would be necessary. Sensing heterogeneity can be implemented in the sense of team members with different sensor specifications, such as some members with cameras specialised for close-up photography, or carrying thermal cameras. Alternatively, there is homogeneity in locomotion capabilities, where some robots may fly, like UAVs, and some can't and are limited to the surface such as USVs. If tasks are going to be shared between these members, the capabilities should be taken into account. The capabilities of the team members should be considered through all aspects of the OWT coverage problem. The environmental representations should encapsulate the requirements of both tasks and the traversal between them. Task decomposition should derive the requirements for a task from the information at hand. Tasks should only be allocated to robots able to complete them, and the capabilities of the robots should be accounted for when grouping tasks. Finally, motion planning should plan paths and trajectories aware of the capabilities and the kinematics of the robot being planned for.

7.6 UAV structure coverage

None of the approaches considered 3D sensor coverage with UAVs, a necessity for the OWT coverage task. Those approaches that did approach the 3D structural inspection assumed a prior model and used offline planning. None of the approaches considered heterogeneous tasks or locomotion capabilities, which would be essential for heterogeneous structure coverage. If blade coverage is to be performed while the OWT is in use, it would be necessary to represent the blade as a moving task and track the blade's position, none of the studies reviewed were relevant to this. As to tackle the 3D sensor coverage with UAVs assuming a prior knowledge of the environment, one may use the ray-tracing voxel-based task generation as in the work of Dornhege et al. [53]. Rather than the reachable voxels being along the ground, it would be necessary to ensure the sensor is a certain distance from the turbine surface, without this the set of reachable voxels would be very large, and therefore inefficient for computation. This same approach could be extended to perform in an online exploratory manner, however, 3D structure exploration wasn't in the reviewed work.

8 Conclusion

In this work, a scoping review of the literature on Multi-robot coverage concerning OWT inspection was carried out. The PRISMA 2020 methodology was detailed to standardize the review process, along with the PICo framework for forming and modelling the research questions. These approaches for standardizing the review process are rarely used in computer science and even more so in robotics literature. However, such systematic processes are essential for providing a scientific review that the reader could repeat themselves and obtain the same or representative data. The works retrieved were then systematically analyzed for the formed research questions and discussed. This work applies not only to OWT inspection scenarios but should also apply to those scenarios resembling offshore wind inspection. It's important to note that coverage planning algorithms are far from the only hurdle to putting autonomous offshore inspections into practice, and the coverage path planning structure inspection would be considered one component of a larger system. As of writing this work drone battery durations would not be sufficient to enable their use alone from shore. To enable the long-term autonomy required for wind farm inspections, an approach to charging drones in the field would be necessary such as that proposed by Han et al. [83], in which drones will launch from a USV with the capability of charging the drones when necessary. Several areas for future research were suggested. Decomposing the coverage task concerning areas of particular interest would facilitate more detailed coverage, allowing focus on areas of the turbine most prone to failure or where failure is most critical. The use of existing knowledge of the turbine geometry without further knowledge of placement or pose, is particularly applicable to floating OWTs. Dynamic tasks, where tasks might move within the environment, and the importance of keeping track of covered and uncovered moving structures. Addressing the limitations of communication around the large OWT structures which may affect satellite communication. Considering Heterogeneous capabilities in the team, both in sensing and in locomotion, and hence facilitating complex planning for teams aware of capabilities. And finally extending existing surface robot voxel-based approaches to UAVs while minimising the computational complexity due to the large accessible space.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. UKRI. Harnessing offshore wind. https://www.ukri.org/news-and-events/responding-to-climate-change/topical-stories/harnessing-offshore-wind/, 2021.
2. GWEC Global Offshore Wind Report 2022. https://gwec.net/wp-content/uploads/2022/06/GWEC-Global-Offshore-Wind-Report-2022.pdf, 2022.
3. Desalegn, B.; Gebeyehu, D.; Tamrat, B.; Tadiwose, T.; Lata, A. Onshore versus offshore wind power trends and recent study practices in modeling of wind turbines' life-cycle impact assessments. *Cleaner Engineering and Technology* **2023**, *17*, 100691. https://doi.org/https://doi.org/10.1016/j.clet.2023.100691.
4. ORE Catapult. Offshore Wind Operations & Maintenance: A £9bn per year opportunity by 2030 for the UK to seize. https://ore.catapult.org.uk/?orecatapultreports=offshore-wind-operations-maintenance-9bn-year-opportunity-2030-uk-seize, 2021.
5. Fenstermaker. Wind Turbine Drone Inspections. https://blog.fenstermaker.com/wind-turbine-drone-inspections/, 2022.
6. ECA Group. Windfarm inspection and monitoring by Rov. https://www.ecagroup.com/en/solutions/windfarm-inspection-and-monitoring-rov.
7. Film-Ocean. Renewables - film-ocean. https://www.film-ocean.com/renewables.
8. Balmore Inspection Services. Offshore Wind Farm Underwater Drone Survey Services across Scotland. https://balmoreuav.co.uk/offshore-wind-farm-inspection/, 2021.
9. Atlantas Marine. Offshore Inspection Services. https://www.atlantasmarine.com/offshore-inspection-services/, 2022.
10. Systems, B.A. A.IKANBILIS - designed for cost effective underwater inspections. https://beex.sg/hauv.
11. Aero Enterprise. Inspection Service at offshore wind turbines. https://aero-enterprise.com/services/offshore-wind-turbines/.
12. Force Technology. Drone inspection of wind turbines – on- and offshore. https://forcetechnology.com/en/services/inspection/drone-inspection-of-wind-turbines-onshore-and-offshore.
13. Iberdrola. We are committed to using drones to inspect and maintain wind farms. https://www.iberdrola.com/innovation/drones-wind-farms.
14. INNVOTEK. Amphibian™. https://innvotek.com/amphibian/, 2022.
15. ROBOTICS, R. Leading edge repair. https://roperobotics.com/index.php/leading-edge-repair/, 2021.
16. Nordin, M.H.; Sharma, S.; Khan, A.; Gianni, M.; Rajendran, S.; Sutton, R. Collaborative Unmanned Vehicles for Inspection, Maintenance, and Repairs of Offshore Wind Turbines. *Drones* **2022**, *6*. https://doi.org/10.3390/drones6060137.
17. Parker, L.E.; Rus, D.; Sukhatme, G.S. Multiple mobile robot systems. *Springer handbook of robotics* **2016**, pp. 1335–1384.
18. Franko, J.; Du, S.; Kallweit, S.; Duelberg, E.; Engemann, H. Design of a multi-robot system for wind turbine maintenance. *Energies* **2020**, *13*, 2552.
19. Bernardini, S.; Jovan, F.; Jiang, Z.; Watson, S.; Weightman, A.; Moradi, P.; Richardson, T.; Sadeghian, R.; Sareh, S. A multi-robot platform for the autonomous operation and maintenance of offshore wind farms. *Autonomous Agents and Multi-Agent Systems (AAMAS) 2020* **2020**, pp. 1696–1700.
20. Jovan, F.; Bernardini, S. Multi-Robot Coordination in Operations and Maintenance of Off Shore Wind Farms with Temporal Planning. In Proceedings of the Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS2021), Guangzhou, China, 2021, pp. 2–13.
21. Freda, L.; Gianni, M.; Pirri, F.; Gawel, A.; Dubé, R.; Siegwart, R.; Cadena, C. 3D multi-robot patrolling with a two-level coordination strategy. *Autonomous Robots* **2019**, *43*, 1747–1779.
22. Jiang, Z.; Jovan, F.; Moradi, P.; Richardson, T.; Bernardini, S.; Watson, S.; Weightman, A.; Hine, D. A multirobot system for autonomous deployment and recovery of a blade crawler for operations and maintenance of offshore wind turbine blades.

Journal of Field Robotics **2023**, *40*, 73–93, [https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.22117]. https://doi.org/https://doi.org/10.1002/rob.22117.

23. Fan, Y.; Ma, J.; Wang, G.; Li, T. Design of a heterogeneous marsupial robotic system composed of an USV and an UAV. In Proceedings of the 2016 Eighth International Conference on Advanced Computational Intelligence (ICACI), 2016, pp. 395–399. https://doi.org/10.1109/ICACI.2016.7449858.
24. Mišković, N.; Bogdan, S.; Nad, D.; Mandić, F.; Orsag, M.; Haus, T. Unmanned marsupial sea-air system for object recovery. In Proceedings of the 22nd Mediterranean Conference on Control and Automation, 2014, pp. 740–745. https://doi.org/10.1109/MED.2014.6961462.
25. Zhang, H.; He, Y.; Li, D.; Gu, F.; Li, Q.; Zhang, M.; Di, C.; Chu, L.; Chen, B.; Hu, Y. Marine UAV–USV marsupial platform: System and recovery technic verification. *Applied Sciences* **2020**, *10*, 1583.
26. ORE Catapult. UAV Approaches to Wind Turbine Inspection. https://ore.catapult.org.uk/wp-content/uploads/2019/03/Cyberhawks-Approach-to-UAV-Inspection-Craig-Stout-ORE-Catapult.pdf, 2019.
27. Exo Inc. Wind Turbine Structural Tower Inspection. https://exoinc.com/wind-turbine-structural-tower-inspection.
28. BVG Associates. Guide to a Floating Offshore Wind Farm. https://guidetofloatingoffshorewind.com/, 2023.
29. ORE Catapult. Levenmouth 7MW demonstration offshore wind turbine. https://ore.catapult.org.uk/wp-content/uploads/2018/08/Catapult-Specification-and-Service-Summary-Sheet-Levenmouth-turbine.pdf, 2018.
30. Choset, H. Coverage of known spaces: The boustrophedon cellular decomposition. *Autonomous Robots* **2000**, *9*, 247–253.
31. Arkin, E.M.; Fekete, S.P.; Mitchell, J.S. Approximation algorithms for lawn mowing and milling. *Computational Geometry* **2000**, *17*, 25–50.
32. Ulrich, I.; Mondada, F.; Nicoud, J.D. Autonomous vacuum cleaner. *Robotics and autonomous systems* **1997**, *19*, 233–245.
33. Almadhoun, R.; Taha, T.; Seneviratne, L.; Dias, J.; Cai, G. A survey on inspecting structures using robotic systems. *International Journal of Advanced Robotic Systems* **2016**, *13*, 1729881416663664, [https://doi.org/10.1177/1729881416663664]. https://doi.org/10.1177/1729881416663664.
34. Peters, M.D.; Godfrey, C.; McInerney, P.; Munn, Z.; Tricco, A.C.; Khalil, H. Scoping reviews. *Joanna Briggs Institute reviewer's manual* **2017**, *2015*, 1–24.
35. Tricco, A.; Lillie, E.; Zarin, W.; O'Brien, K.; Colquhoun, H.; Levac, D.; Moher, D.; Peters, M.; Horsley, T.; Weeks, L.; et al. PRISMA extension for scoping reviews (PRISMA-ScR): Checklist and explanation. *Annals of Internal Medicine* **2018**, *169*. https://doi.org/10.7326/M18-0850.
36. Yan, Z.; Jouandeau, N.; Cherif, A.A. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems* **2013**, *10*, 399.
37. Farinelli, A.; Iocchi, L.; Nardi, D. An analysis of coordination in multi-robot systems. In Proceedings of the SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme-System Security and Assurance (Cat. No. 03CH37483). IEEE, 2003, Vol. 2, pp. 1487–1492.
38. McGraw, G.A.; Groves, P.D.; Ashman, B.W. Robust positioning in the presence of multipath and NLOS GNSS signals. *Position, navigation, and timing technologies in the 21st century: integrated satellite navigation, sensor systems, and civil applications* **2020**, *1*, 551–589.
39. Burgard, W.; Herbert, M.; Bennewitz, M. World Modeling. In *Springer handbook of robotics*; Siciliano, B.; Khatib, O.; Kröger, T., Eds.; Springer: Oxford, 2008; chapter 45, pp. 1136–1152.
40. Ball, D.; Ross, P.; English, A.; Patten, T.; Upcroft, B.; Fitch, R.; Sukkarieh, S.; Wyeth, G.; Corke, P. Robotics for sustainable broad-acre agriculture. *Springer Tracts in Advanced Robotics* **2015**, *105*, 439 – 453. Type: Conference paper, https://doi.org/10.1007/978-3-319-07488-7_30.
41. Masehian, E.; Jannati, M.; Hekmatfar, T. Cooperative mapping of unknown environments by multiple heterogeneous mobile robots with limited sensing. *Robotics and Autonomous Systems* **2017**, *87*, 188 – 218. Type: Article, https://doi.org/10.1016/j.robot.2016.08.006.
42. Karapetyan, N.; Moulton, J.; Lewis, J.S.; Quattrini Li, A.; O'Kane, J.M.; Rekleitis, I. Multi-robot Dubins Coverage with Autonomous Surface Vehicles. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, pp. 2373–2379. https://doi.org/10.1109/ICRA.2018.8460661.
43. Tang, J.; Gao, Y.; Lam, T.L. Learning to Coordinate for a Worker-Station Multi-Robot System in Planar Coverage Tasks. *IEEE Robotics and Automation Letters* **2022**, *7*, 12315–12322. https://doi.org/10.1109/LRA.2022.3214446.
44. Kim, M.; Gupta, R.; Sentis, L. CONCERTS: Coverage Competency-Based Target Search for Heterogeneous Robot Teams. *Applied Sciences (Switzerland)* **2022**, *12*. Type: Article, https://doi.org/10.3390/app12178649.
45. Kalde, N.; Simonin, O.; Charpillet, F. Comparison of classical and interactive multi-robot exploration strategies in populated environments. *Acta Polytechnica* **2015**, *55*, 154 – 161. Type: Article, https://doi.org/10.14311/AP.2015.55.0154.
46. Song, J.; Gupta, S.; Hare, J. Game-theoretic cooperative coverage using autonomous vehicles. In Proceedings of the 2014 Oceans - St. John's, OCEANS 2014, 2015. Type: Conference paper, https://doi.org/10.1109/OCEANS.2014.7003082.
47. Perez-imaz, H.I.A.; Rezeck, P.A.F.; Macharet, D.G.; Campos, M.F.M. Multi-robot 3D coverage path planning for First Responders teams. In Proceedings of the 2016 IEEE International Conference on Automation Science and Engineering (CASE), 2016, pp. 1374–1379. https://doi.org/10.1109/COASE.2016.7743569.

48. Sharma, S.; Shukla, A.; Tiwari, R. Multi robot area exploration using nature inspired algorithm. *Biologically Inspired Cognitive Architectures* **2016**, *18*, 80 – 94. Type: Article, https://doi.org/10.1016/j.bica.2016.09.003.
49. Zhang, P.; Xu, S.; Zhang, W.; Dong, W. A Cooperative Aerial Inspection System with Continuable Charging Strategy. In Proceedings of the 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2019, pp. 770–777. https://doi.org/10.1109/ROBIO49542.2019.8961597.
50. Yu, C.; Yang, X.; Gao, J.; Chen, J.; Li, Y.; Liu, J.; Xiang, Y.; Huang, R.; Yang, H.; Wu, Y.; et al. Asynchronous Multi-Agent Reinforcement Learning for Efficient Real-Time Multi-Robot Cooperative Exploration. *arXiv preprint arXiv:2301.03398* **2023**.
51. Colares, R.G.; Chaimowicz, L. The next Frontier: Combining Information Gain and Distance Cost for Decentralized Multi-Robot Exploration. In Proceedings of the Proceedings of the 31st Annual ACM Symposium on Applied Computing, New York, NY, USA, 2016; SAC '16, pp. 268–274. event-place: Pisa, Italy, https://doi.org/10.1145/2851613.2851706.
52. Bramblett, L.; Peddi, R.; Bezzo, N. Coordinated Multi-Agent Exploration, Rendezvous, & Task Allocation in Unknown Environments with Limited Connectivity. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2022, pp. 12706–12712. https://doi.org/10.1109/IROS47612.2022.9981898.
53. Dornhege, C.; Kleiner, A.; Hertle, A.; Kolling, A. Multirobot Coverage Search in Three Dimensions. *Journal of Field Robotics* **2016**, *33*, 537 – 558. Type: Article, https://doi.org/10.1002/rob.21573.
54. Dong, S.; Xu, K.; Zhou, Q.; Tagliasacchi, A.; Xin, S.; Nießner, M.; Chen, B. Multi-Robot Collaborative Dense Scene Reconstruction. *ACM Trans. Graph.* **2019**, *38*. Place: New York, NY, USA Publisher: Association for Computing Machinery, https://doi.org/10.1145/3306346.3322942.
55. Bartolomei, L.; Teixeira, L.; Chli, M. Fast Multi-UAV Decentralized Exploration of Forests. *IEEE Robotics and Automation Letters* **2023**.
56. Choset, H.; Lynch, K.; Hutchinson, S.; Kantor, G.; Burgard, W.; Kavraki, L.; Thrun, S. Principles of Robot Motion: Theory, Algorithms, and Implementation ERRATA!!!! 1 **2003**.
57. Christensen, H.I.; Hager, G.D. Sensing and estimation. *Springer Handbook of Robotics* **2016**, pp. 91–112.
58. Moravec, H.; Elfes, A. High resolution maps from wide angle sonar. In Proceedings of the Proceedings. 1985 IEEE International Conference on Robotics and Automation, 1985, Vol. 2, pp. 116–121. https://doi.org/10.1109/ROBOT.1985.1087316.
59. Thrun, S. Learning occupancy grid maps with forward sensor models. *Autonomous robots* **2003**, *15*, 111–127.
60. Foley, J.D. *Computer graphics: principles and practice*; Vol. 12110, Addison-Wesley Professional, 1996.
61. Fisher, R.B.; Konolige, K. Range Sensor, 2008.
62. Hornung, A.; Wurm, K.M.; Bennewitz, M.; Stachniss, C.; Burgard, W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous robots* **2013**, *34*, 189–206.
63. Han, L.; Gao, F.; Zhou, B.; Shen, S. FIESTA: Fast Incremental Euclidean Distance Fields for Online Motion Planning of Aerial Robots. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE Press, 2019, p. 4423–4430. https://doi.org/10.1109/IROS40897.2019.8968199.
64. Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; Ng, A. ROS: an open-source Robot Operating System. In Proceedings of the Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics, Kobe, Japan, 2009.
65. Yamauchi, B. A frontier-based approach for autonomous exploration. In Proceedings of the Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'. IEEE, 1997, pp. 146–151.
66. Korsah, G.A.; Stentz, A.; Dias, M.B. A comprehensive taxonomy for multi-robot task allocation. *The International Journal of Robotics Research* **2013**, *32*, 1495–1512, [https://doi.org/10.1177/0278364913496484]. https://doi.org/10.1177/0278364913496484.
67. Cieslewski, T.; Kaufmann, E.; Scaramuzza, D. Rapid exploration with multi-rotors: A frontier selection method for high speed flight. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp. 2135–2142. https://doi.org/10.1109/IROS.2017.8206030.
68. Choset, H.; Pignon, P. Coverage path planning: The boustrophedon cellular decomposition. In Proceedings of the Field and service robotics. Springer, 1998, pp. 203–209.
69. Kavraki, L.E.; LaValle, S.M. Motion planning. In *Springer handbook of robotics*; Springer, 2016; pp. 139–162.
70. Wurll, C.; Henrich, D.; Wörn, H. Multi-goal path planning for industrial robots **1999**.
71. Eyerich, P.; Mattmüller, R.; Röger, G. Using the Context-Enhanced Additive Heuristic for Temporal and Numeric Planning. In Proceedings of the Proceedings of the Nineteenth International Conference on International Conference on Automated Planning and Scheduling. AAAI Press, 2009, ICAPS'09, p. 130–137.
72. Helsgaun, K. An effective implementation of the Lin–Kernighan traveling salesman heuristic. *European Journal of Operational Research* **2000**, *126*, 106–130. https://doi.org/https://doi.org/10.1016/S0377-2217(99)00284-2.
73. Lewis, J.S.; Edwards, W.; Benson, K.; Rekleitis, I.; O'Kane, J.M. Semi-boustrophedon coverage with a dubins vehicle. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp. 5630–5637. https://doi.org/10.1109/IROS.2017.8206451.
74. Sedighpour, M.; Yousefikhoshbakht, M.; Mahmoodi Darani, N. An effective genetic algorithm for solving the multiple traveling salesman problem. *Journal of Optimization in Industrial Engineering* **2012**, *4*, 73–79.

75. Zhou, B.; Gao, F.; Wang, L.; Liu, C.; Shen, S. Robust and efficient quadrotor trajectory generation for fast autonomous flight. *IEEE Robotics and Automation Letters* **2019**, *4*, 3529–3536.
76. Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; Wu, Y. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems* **2022**, *35*, 24611–24624.
77. Cao, Y.U.; Kahng, A.B.; Fukunaga, A.S. Cooperative mobile robotics: Antecedents and directions. *Robot colonies* **1997**, pp. 7–27.
78. Aumann, R.J.; Dreze, J.H. Cooperative games with coalition structures. *International Journal of game theory* **1974**, *3*, 217–237.
79. Bengio, Y.; Louradour, J.; Collobert, R.; Weston, J. Curriculum learning. In Proceedings of the Proceedings of the 26th annual international conference on machine learning, 2009, pp. 41–48.
80. inmarsat. Launch of Inmarsat SwiftBroadband unmanned aerial vehicle service to provide operational capability boost. https://www.inmarsat.com/en/news/latest-news/government/2017/launch-inmarsat-swiftbroadband-unmanned-aerial-vehicle-service-provide-operational-capability-boost.html, 2017.
81. Mataric, M.J. Interaction and intelligent behavior **1994**.
82. Sikand, K.S.; Zartman, L.; Rabiee, S.; Biswas, J. Robofleet: Secure Open Source Communication and Management for Fleets of Autonomous Robots . In Proceedings of the Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, 2021, pp. 406–412. https://doi.org/10.1109/IROS51168.2021.9635830.
83. Han, Y.; Ma, W. Automatic Monitoring of Water Pollution based on the Combination of UAV and USV. In Proceedings of the 2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT), 2021, pp. 420–424. https://doi.org/10.1109/ICEICT53123.2021.9531204.