Project Number 101017258

D2.5 Multi-Robot Collaborative Rescue Mission

Version 1.0
30 June 2023
Final

Public Distribution

University of Luxembourg

Project Partners: Aero41, ATB, AVL, Bonn-Rhein-Sieg University, Cyprus Civil Defence, Domaine Kox, FORTH, Fraunhofer IESE, KIOS, KUKA Assembly & Test, Locomotec, Luxsense, The Open Group, Technology Transfer Systems, University of Hull, University of Luxembourg, University of York

Project Partner Contact Information

Aero41 Frédéric Hemmeler Chemin de Mornex 3 1003 Lausanne Switzerland E-mail: frederic.hemmeler@aero41.ch	**ATB** Sebastian Scholze Wiener Strasse 1 28359 Bremen Germany E-mail: scholze@atb-bremen.de
AVL Martin Weinzerl Hans-List-Platz 1 8020 Graz Austria E-mail: martin.weinzerl@avl.com	**Bonn-Rhein-Sieg University** Nico Hochgeschwender Grantham-Allee 20 53757 Sankt Augustin Germany E-mail: nico.hochgeschwender@h-brs.de
Cyprus Civil Defence Eftychia Stokkou Cyprus Ministry of Interior 1453 Lefkosia Cyprus E-mail: estokkou@cd.moi.gov.cy	**Domaine Kox** Corinne Kox 6 Rue des Prés 5561 Remich Luxembourg E-mail: corinne@domainekox.lu
FORTH Sotiris Ioannidis N Plastira Str 100 70013 Heraklion Greece E-mail: sotiris@ics.forth.gr	**Fraunhofer IESE** Daniel Schneider Fraunhofer-Platz 1 67663 Kaiserslautern Germany E-mail: daniel.schneider@iese.fraunhofer.de
KIOS Maria Michael 1 Panepistimiou Avenue 2109 Aglatzia, Nicosia Cyprus E-mail: mmichael@ucy.ac.cy	**KUKA Assembly & Test** Michael Laackmann Uhthoffstrasse 1 28757 Bremen Germany E-mail: michael.laackmann@kuka.com
Locomotec Sebastian Blumenthal Bergiusstrasse 15 86199 Augsburg Germany E-mail: blumenthal@locomotec.com	**Luxsense** Gilles Rock 85-87 Parc d'Activités 8303 Luxembourg Luxembourg E-mail: gilles.rock@luxsense.lu
The Open Group Scott Hansen Rond Point Schuman 6, 5th Floor 1040 Brussels Belgium E-mail: s.hansen@opengroup.org	**Technology Transfer Systems** Paolo Pedrazzoli Via Francesco d'Ovidio, 3 20131 Milano Italy E-mail: pedrazzoli@ttsnetwork.com
University of Hull Yiannis Papadopoulos Cottingham Road Hull HU6 7TQ United Kingdom E-mail: y.i.papadopoulos@hull.ac.uk	**University of Luxembourg** Miguel Olivares Mendez 2 Avenue de l'Universite 4365 Esch-sur-Alzette Luxembourg E-mail: miguel.olivaresmendez@uni.lu
University of York Simos Gerasimou & Nicholas Matragkas Deramore Lane York YO10 5GH United Kingdom E-mail: simos.gerasimou@york.ac.uk nicholas.matragkas@york.ac.uk	

Document Control

Version	Status	Date
0.1	Document outline	15/03/2023
0.2	Initial draft	05/04/2023
0.3	First draft	19/06/2023
0.4	Internal reviews	25/06/2023
0.5	Internal reviews updates	29/06/2023
0.6	Final version for partner reviews	29/06/2023
0.7	Final version partner reviews updates	30/06/2023
0.8	Further final version partner reviews updates	30/06/2023
1.0	Final QA version	30/06/2023

Table of Contents

List of Figures

Acronyms

EU European Union

ExSce Executable Scenario

MPC Model Predictive Control

MRS Multi-Robot Systems

SESAME Secure and Safe Multi-Robot Systems

WP Work Package

EDDI Executable Digital Dependability Identities

UAV Unmanned Aerial Vehicle

GPS Global Positioning System

Executive Summary

This deliverable reports on “D2.5: Multi-Robot Collaborative Rescue Mission of the work package 2: Sensor Fusion and Collaborative Intelligence, within the context of the European Union (EU) Secure and Safe Multi-Robot Systems (SESAME) project. This presents a novel architecture to generate trajectories of a collaborative Multi-Robot Systems (MRS) performed. The generated trajectories can provide extra off-board sensor feedback to rescue robots under various distress types (e.g., cyber-attack, sensor malfunction). Also, when a cyber-attack or sensor malfunction affects one or more members of the team, non-attacked robots will generate a rescue trajectory providing further sensor feedback to the compromised robots and helping them to maintain their operational and safe state.

For this purpose, we outline the components of a collaborative rescue mission for MRS. For the planning part, we use an online trajectory optimization approach to compute the fastest trajectory, given the initial and final positions. The collaborative sensor fusion component provides a perception, that forms the essential elements of drone detection, position estimation, and semantic segmentation. The second part provides sensor fusion, which is central to collaborative sensor fusion. Using these components, we present a generic architecture for the rescue operation of the target drone with the sensor malfunction. All the corresponding rescue actions are also considered. Moreover, the required elements in the architecture, such as the switching mechanism and redundant components, are discussed. Furthermore, the algorithm and instructions to implement this architecture are given. The efficiency of the proposed architecture is investigated and evaluated by high-fidelity simulations in the Gazebo.

1 Introduction

In this section, the preliminary remarks on the project and the main research objectives of this deliverable are elaborated. Accordingly, the main research directions are identified as a set of detailed questions, which are mathematically formulated, and corresponding solutions are presented in the proceeding sections. Moreover, the relationship between the overall project aims and the proposed solution is established. Accordingly, the contributions and novelties are summarized.

Across the different civil domains where robots can support human operators, one of the areas where they can have more impact is rescue operations. In particular, MRS have the potential to significantly improve the efficiency of rescue personnel with faster response time [1], support in hazardous environments [2], or providing real-time mapping and monitoring of the area where an incident has occurred [3], among other possibilities.

Rescue operations can take significant advantage of supporting autonomous MRS. These can aid in mapping and situational assessment, monitoring and surveillance, establishing communication networks, or searching for victims. This deliverable provides an algorithm of multi-robot systems supporting rescue operations, with system-level considerations and focusing on the algorithmic perspectives for multi-robot coordination and perception. Autonomous robots have been playing increasingly important roles in civil applications in recent years [4].

Multi-Unmanned Aerial Vehicle (UAV) systems for civil applications (where rescue applications are a subset) are reviewed in [5] from the point of view of communication. A classification of technological trends and sensing modalities in UAVs for civil applications is available in [6]. UAVs for rescue operations are reviewed in [7], with a classification in terms of (i) sensing, (ii) system-level definitions, and (iii) operational environments. A study of MRS for rescue operations in [8] focuses on task allocation algorithms, communication modalities, and human-robot interaction for both homogeneous and heterogeneous multi-robot systems. While autonomous robots are being increasingly adopted for rescue missions, current levels of autonomy and safety of robotic systems only allow for full autonomy in the search part, but not for rescue, where human operators need to intervene. In general, the literature on MRS rescue operations with some degree of autonomy is rather sparse, with most results being based on simulations or simplified scenarios [9].

One of the most important capabilities in robotic systems is motion planning and execution, also comprising trajectory planning and tracking. A trajectory is defined as a time-parameterized motion reference (i.e., a geometric path with an associated timing law), and can, for instance, describe the motion reference of a mobile robot platform. Therefore, while in pure path planning the goal is to generate geometrically feasible (collision-free) paths, trajectory planning does not only consider geometrical feasibility but also kinematic and dynamic limits must be taken into account to generate a trajectory. Trajectory tracking is the capability of the robot to reach and follow the trajectory, i.e., the time-parameterized reference, at run-time. The challenges in trajectory planning and tracking are constrained by the motion manifold.

From this point of view, the rescue operation can be considered as an alternative trajectory planning at the moment the robot malfunctioning, satisfying predefined actions, as a safe and secure approach to recover/rescue the operation or the robot.

1.1 Document Purpose

This document is prepared in the context of the SESAME project. More precisely, it refers to "D2.5: Multi-Robot Collaborative Rescue Mission of Work Package (WP) 2: Sensor Fusion and Collaborative Intelligence. This reports the novel approach to generate trajectories of a collaborative MRS performed in Task 2.4. The generated trajectories can provide extra off-board sensor feedback to rescue robots under various distress types (e.g., cyber-attack, sensor malfunction).

To satisfy the above-mentioned objectives, we primarily incorporate two main components, i.e., trajectory planner and collaborative perception and sensor fusion.

Collaborative Perception and Sensor Fusion aim at a collective perception mechanism enabling safe and robust robot navigation. The main goal is for the robot to carry on with its mission in the presence of adverse environmental conditions, cyber- attacks or faults. In the context of this task, the scope of cyber-attacks or faults is considered as a sensor becoming unavailable, e.g., losing the signal of Global Positioning System (GPS). Similarly, the scope of the adverse environmental conditions is bounded to events that may perturb the state estimation of a robotic system, such as entering GPS-denied areas. Building on recent advances in deep learning-based object detection and semantic segmentation, perception algorithms provide automatic means to estimate the position of objects in a scene without additional aided fiducial markers. This can, for instance, enable an observer robot to visually estimate the position of the target robot. Sharing this estimation with the tracked robot could help it recover from a faulty GPS sensor for instance. Likewise, the observer robot can detect obstacles in the vicinity of the other robot and share their position with it. This could prove useful if the target robot's vision sensors were failing or scrambled, preventing it from sensing obstacles. On the other hand, online perception-aware trajectory generation considers the close environment information gathered by nearby robots using the collaborative sensor fusion approach developed in Task 2.3 and the mission goals. Every robot will be "monitored" by one or more robots, but not all of them. Also, when a cyber-attack or sensor malfunction affects one or more members of the team, non-attacked robots will generate a rescue trajectory providing further sensor feedback to the compromised robots and helping them to maintain their operational and safe state.

In summary, the overall objectives are as follows.

- Identifying the corresponding actions at the GPS denial moments.
- Structuring the architecture of the solution, taking into account the MRS settings and integrating both sensor fusion and planner components.
- Online trajectory generation approach that minimizes the uncertainty of MRS, to fulfil the overall common mission goals in offline/online, (de)centralized ways.
- Incorporating the environment information using Task 2.3.

Moreover, the proposed solutions are evaluated as

- Numerical and in-lab experimental evaluation of the planner, satisfying the metrics, the computational time/burden, and situational/perception awareness.
- Numerical and in-lab experimental evaluation of the sensor fusion component.
- Safety, security and quality assurance for a tentative trajectory.
- We provide the components in the form of algorithms with demonstrations of drones as an example.

Finally, it should be noted that the practical implementation and integration of use cases are sought in WP8. In this deliverable, we provide high-fidelity simulations and experiments to validate the proposed approaches as proof of concept. Furthermore, the potential applicability of the proposed approaches on the use cases is briefly motivated in Section 6.

1.2 Relationship to other Deliverables

Considering the above-mentioned points, In summary, the inputs to our component are:

- Robots model, parameters, dynamic and kinematic restrictions (Deliverable 3.2 and 3.4),
- Information of the sensors of the robots (Deliverable 3.2 and 3.4),
- Task plans information, including start and endpoints, the intermediate regions of interest, dependencies between the different robots, and temporal dependencies between the subtasks or dependencies between the states of the robot,

- Safety, security and quality assurance criteria (Deliverable 3.3 and D3.4),
- the GPS denial or malfunctioning flag to indicate the moment the rescue action is to be done.

Moreover, the expected outputs are

- Planned trajectories for each individual robot, including time-parameterized motion references for each robot, safety, security and quality assurance metrics were achieved for each planned trajectory,
- Robot commands in the form of actuator/driver command to the robotics platforms e.g. desired velocity commands,
- Algorithms of a Higher level centralized planner, Lower level decentralized planner and Lower level decentralized control,
- Non-attacked robots will generate a rescue trajectory providing further sensor feedback to the compromised ones,
- Numerical simulations and experimental studies on in-lab drones as an example.

Accordingly, the interfaces to exchange information with other components are identified as

- The dynamics and kinematics of the robots with the composable models,
- The sensors of the robots with the ExSce,
- Task plans with the Collaborative Intelligence, including, requirements, tasks, temporal constraints, start and endpoints,
- The MRS system structure and the possibility to exchange information among the robots or with a control station
- Complete situational awareness with Collaborative Perception,
- Safety, security and quality assurance metrics for a tentative trajectory with the EDDI,
- Robot-agnostic interface for the planned trajectories,
- Robot-agnostic interface for the actuator/driver commands to the robotics platforms.

1.3 Contribution and Novelties

One of the most important capabilities in robotic systems is motion planning and execution, also comprising trajectory planning and tracking. A trajectory is defined as a time-parameterized motion reference (i.e., a geometric path with an associated timing law), and can, for instance, describe the motion reference of a mobile robot platform. Therefore, while in pure path planning the goal is to generate geometrically feasible (collision-free) paths, trajectory planning does not only consider geometrical feasibility but also kinematic and dynamic limits must be taken into account to generate a trajectory. Trajectory tracking is the capability of the robot to reach and follow the trajectory, i.e., the time-parameterized reference, at run-time. The challenges in trajectory planning and tracking are manifold.

Our main focus is on MRS, where the trajectories of several robots need to be planned in a coordinated way, in general, to fulfil the overall common mission goals. The robots have to fulfil local tasks, such as the movement from a given start- to an endpoint, while for instance visiting different regions of interest in between and performing actions herein under temporal constraints. Furthermore, there might be dependencies between the robots and their dynamic states such as keeping a formation or avoiding collisions among each other. Therefore, task and trajectory planning are closely related and sometimes even done simultaneously. There are many approaches to performing MRS trajectory planning, mainly categorized as offline vs online, and centralized vs distributed/decentralized planning. The optimal approach to planning the trajectory depends mainly on the overall mission requirements, the MRS system structure and the possibility to exchange information among the robots or with a control station.

The trajectory planning is also ruled by the environment of the MRS and the model of the environment available for the planning, either in a centralized form or as partial models in single robots. The environment consists of static structures, static or dynamic objects and the free space in between. The modelling of the environment/situational awareness can be based on many different levels of abstraction and include different levels of uncertainty. The environmental model is continuously updated using mainly the sensing and perception of the different robots, equipped with different types of onboard sensors. Therefore, trajectory planning and tracking are closely linked to environmental modelling as well as the MRS sensing and perception capabilities.

However, the robotic motion does not only depend on the sensing and perception but the perceptual processing itself is influenced by intended actions (active perception). Recent research has extended active perception with support for partial or full mission "planification" by generating perception-aware path and trajectory plans. Advances in the area focus on minimising localisation uncertainty by simultaneously updating the path planning using the richness of texture information in the environment and on minimising state estimation uncertainty by computing the feasibility of trajectory-planning and trajectory-tracking based on the kinematic and dynamic models of the robot. In MRS, performing perception-aware trajectory planning in a collaborative but distributed way is a further challenge, strongly related to the coordination and communication schemes during the planning.

Finally, additional challenges for the MRS trajectory planning and tracking arise from the requirements of safety and security. On the other hand, unauthorized access to the communication system of the MRS could be used to compromise the information that is exchanged between the robots during the trajectory planning, leading to a decrease in performance or even a failure or damage to the MRS or the environment. In our presented component, we provide a capability for MRS trajectory planning and tracking that takes the aforementioned challenges into account. We will develop a cascaded solution, providing online trajectory planning with continuous re-planning, and online trajectory tracking.

In our proposed approach, the trajectory planning part will include a high-level centralized planner, where the global MRS trajectory planning problem is formulated as one overall optimization problem computing the rough trajectories that each robot agent is to track. In a cascade and in a distributed way, each robotic agent will rely on a planner to compute its detailed trajectory to be tracked, based on the given rough one. The trajectory tracking will guarantee that the previously planned detailed trajectories are tracked with no deviations, providing the actuator/driver commands to the robotics platforms, e.g., desired velocity commands. We will apply Model Predictive Control (MPC) in a decentralized way, e.g., each robot independently tracks its trajectory. The following aspects will be considered at different levels on each component, i.e. planner/tracker: (1) collision avoidance with the structures, static or dynamic objects and the other robots, also including dynamic and stochastic models of the potential obstacles (2) kinematic and dynamic limitations of the robots, as well as their model uncertainties and the possibility for online parameter estimation/update, (3) capabilities and constraints of the different robotic sensors and the limits of the situational awareness algorithms to provide perception-aware planning, (4) safety, security, and quality assurance aspects, e.g. by including risk and trust in the planning, generated by the EDDI components at runtime. Herein, safety-related risk could be related to the environmental situation, the uncertainty of observations or the consequences of robotic actions.

The purpose of multi-robot cooperative state estimation is that the localization of a single robot can not only use its sensors but also fuse the sensors or localization results of other robots, received through the communication system. The MRS is considered a coupled system. The sensor information of different robots is connected to a network structure through the communication system. This structure requires high efficiency of calculation and high timeliness of communication. The benefits of collaborative state estimation are summarized as: (1) The state estimation of a single robot is improved by the information sent from other robots. For example, low-precision sensors on a single robot can benefit from high-precision sensors on other robots, and (2) Provide redundancy in case of sensor failure. For example, when a single robot performs Visual Inertial Odometry (VIO) localization, the camera or IMU suddenly fails, and its state estimation can be recovered by other robots through relative localization.

In the context of rescue operations, the collaborative sensor fusion is to "rescue" other robots that are affected by sensor malfunctioning or any type of cyber attack. There are two factors related to trajectory planning, i.e., a robot is performing a trajectory to keep other robots in the field of view, and a robot affected by an attack or sensor issue receives a rescue trajectory.

The main contributions are as follows.

- The trajectory planning problem will be conveniently split into an MRS high-level centralized part and a low-level distributed part, achieving a real-time operation and robust performance.
- The distributed trajectory planning and tracking components will be heterogeneous, i.e., different for each robotics platform, while the centralized trajectory planner will be generic and versatile to include all the targeted robotics platforms, e.g., different kinematic and dynamic models and restrictions. Also, the overall problem formulation is given on a generic dynamic, considering the different use cases with different robot types and the component is designed in a modular way to let each part usable with the least modification required.
- The information provided by the perception, sensing, and situational awareness components will be exploited at the previously mentioned three different levels, i.e., centralized planner, distributed planner, and distributed trackers.
- Safety and security aspects will be considered at the previously mentioned three different levels, i.e., centralized planner, distributed planner, and distributed trackers.
- We rely on camera images and a depth sensor of the detector robot. Using a neural network specifically trained to detect robots in images, we first identify the target robot in the image. Then, using a depth sensor, we measure the distance to the target robot.
- The distance information, we can estimate the relative position of the target robot in the detector robot's frame. This position is then projected into a global frame and shared with the target robot.

1.4 Document Structure

The rest of this deliverable is organized as follows. In Section 2, we review the state-of-the-art approaches to highlight the contributions of the proposed solution. In Section 3, the rescue operation is discussed in detail and all the required components are elaborated, including, trajectory planning and tracking component, collaborative sensor fusion component, sensor malfunction detection unit, network, switching and redundant components. Consequently, the architecture and algorithm to implement the rescue operation are presented. The results of the proposed architecture with its components are studied in Section 4. Then, the extension of this research is considered in Section 5. The potential applicability of the proposed approaches for use cases is described in Section 6. The concluding remarks are given in Section 7.

2 Related Work

This section discusses the related works and critically reviews similar approaches for MRS rescue operations. We also describe the main aspects and algorithms required for MRS coordination and planning in collaborative applications, towards rescue missions. These are key enablers of MRS capabilities in terms of exploration and navigation over large areas. We discuss this mainly from the point of view of cooperative multi-robot systems while focusing on their applicability for rescue missions. The main problems discussed in this section are the following:

- Communication: This plays a vital role in an MRS due to the need for coordination and information sharing necessary to carry out collaborative tasks.
- Multi-robot task allocation: distribution of tasks and objectives among the robots (e.g., areas to be searched, or positions to be occupied to ensure connectivity among the robots and with the base station)
- Path planning and area coverage: global path planning covers area coverage (generation of paths to entirely analyze a given area) and area partition (dividing the area between multiple robots). Local planning deals mainly with obstacle and collision avoidance, incorporating robot dynamics.
- Area exploration: coverage and mapping algorithms (or discover/ search for specific objects) in potentially unknown environments.
- Centralized multi-robot planning: decision-making on the actions of multiple robots by either gathering and processing data in a single node, from which decisions are distributed to others, or by achieving consensus through communication (often requiring agents to be aware of all others, and stable communication).
- Distributed multi-robot planning: algorithms enabling agents to make independent decisions individually or in subsets based only on their data or data shared by their neighbours. These do not necessarily need agents to be aware of the existence or state of all other agents in the system.

Over the past two decades, multiple international projects have been devoted to rescue robotics, often to work towards MRS solutions and the development of multi-modal sensor fusion algorithms. Some of the projects focus on the development of complex robotic systems that can be remotely controlled [10]. However, the majority of the projects consider MRS [11], and other projects consider collaborative robots. An early approach to the design and development of multi-UAV systems for cooperative activities was presented within the COMETS project (real-time coordination and control of multiple heterogeneous UAVs [12]. In terms of human-robot collaboration for rescue operations, one of the first EU-funded projects in rescue robotics, PeLoTe [13], designed mobile robots for rescue missions and developed a heterogeneous telematic system for cooperative (human-robot) rescue operations. Other international projects designing and developing autonomous multi-robot systems for rescue operations include the NIFTi EU project (natural human-robot cooperation in dynamic environments) [14], ICARUS (unmanned rescue) [11], TRADR (long-term human-robot teaming for disaster response) [15], or SmokeBot (mobile robots with novel environmental sensors for inspection of disaster sites with low visibility) [16]. Other projects, such as CENTAURO (robust mobility and dexterous manipulation in disaster response by full body telepresence in a centaur-like robot), have focused on the development of more advanced robots that are not fully autonomous but controlled in realtime [10].

In COMETS, the project aimed to design and implement a distributed control system for cooperative activities using heterogeneous UAVs. To that end, the project researchers developed a remote-controlled airship and an autonomous helicopter and worked towards cooperative perception in real-time [17]. In NIFTi, UAVs were utilized for autonomous navigation and mapping in harsh environments [14]. The focus of the project was mostly on human-robot interaction and on distributing information to human operators at different layers. Similarly, in the TRADR project, the focus was on collaborative efforts towards disaster response of both humans and robots [18], and on MRS planning [15]. In particular, the results of TRARD include a framework for the integration of UAVs in rescue missions, from path planning to a global 3D point cloud generator [19]. The project continued with the foundation of the German Rescue Robotics Center at Fraunhofer FKIE, where broader research is conducted, for example, in maritime rescue [20]. In ICARUS, project researchers developed mapping tools, middleware software for tactical communications, and a multi-domain robot command and control station [11]. While these projects focused on the algorithmic aspects of the rescue operation, and on the design of MRS, in Smokebot the focus was on developing sensors and sensor fusion methods for harsh environments [21]. A more detailed description of some of these projects, especially those that started before 2017, is available in [22].

In terms of international competition and tournaments, two relevant precedents in autonomous rescue operations are the European Robotics League (ERL) Emergency Tournament and the RoboCup Rescue League. In [23], the authors describe the details of what was the world's first multi-domain (air, land and sea) MRS rescue competition. A total of 16 international teams competed with tasks including (i) environment reconnaissance and mapping (merging ground and aerial data), (ii) search for missing workers outside and inside an old building, and (iii) pipe inspection with localization of leaks (on land and underwater). The RoboCup Rescue League, on the other side, was proposed in 1999 [24]. One of the ground robots utilized in the 2020 edition is described in [25], a full-scale rescue robot with a robot arm equipped with a gripper. Another set of major events featuring search and rescue robotics is the DARPA challenges. Humanoid robots [26] and human-robot coordination strategies [27] for rescue operations were presented in the 2013-2015 DARPA Robotics Challenge. The DARPA Subterranean (SubT) Challenge, running in 2018-2021, has shifted the focus towards underground MRS for rescue operations, with ground robots and UAVs collaborating in the tasks [28]. This challenge has demonstrated the versatility and significant increase of flexibility of heterogenous MRS [29], with robust UAV flight in inherently constrained environments [30], and ground robots able of navigating complex environments and long-term autonomy [31]. In 2020, due to the Covid-19 pandemic, the challenge moved to a fully virtual edition with realistic simulation-based environments [32].

2.1 Communication

In multi-agent systems, a mobile ad-hoc network (MANET) is often formed for wireless communication and routing messages between the robots. Owing to the changing characteristics in terms of wireless transmission in different physical mediums, different communication technologies are utilized for various types of robots. An overview of the main MRS communication technologies is available in [33], while a review on MANET-based communication for rescue operations is available in [34]. Collaborative MRS need to be able to communicate to keep coordinated but also need to be aware of each other's position to make the most out of the shared data [35]. Situated communication refers to wireless communication technologies that enable simultaneous data transfer while locating the data source [36]. Ubiquitous wireless technologies such as WiFi and Bluetooth have been exploited to enable localization [37]. These approaches have been traditionally based on the received signal strength indicator (RSSI) and the utilization of either Bluetooth beacons in known locations [38], or radio maps that define the strength of the signal of different access points over a predefined and surveyed area [39]. More recently, other approaches rely on angle-of-arrival [40], now built-in in Bluetooth 5.1 devices [41]. Ultra-wideband (UWB) technology has emerged as a more accurate and less prone to interference alternative to Wi-Fi and Bluetooth [42]. With most existing research relying on fixed UWB transceivers in known locations [43], recent works also show promising results in mobile positioning systems or collaborative localization [44]. A recent trend has also been to apply deep learning in positioning estimation [45]. From the point of view of multi-robot coordination, maintaining connectivity between the different agents participating in a rescue mission is critical. Connectivity maintenance in wireless sensor networks has been a topic of study for the past two decades [46]. In recent years, it has gained more attention in the fields of MRS with decentralized approaches [47]. Connectivity maintenance algorithms can be designed coupled with distributed control in multi-robot systems [48], or collision avoidance [49]. Xiao et al. have recently presented a cooperative multi-agent search algorithm with connectivity maintenance [50]. Similar works aiming at cooperative search, surveillance or tracking with multi-robot systems focus on optimizing the data paths [51] or fallible robots [52]. Another recent work in area coverage with connectivity maintenance is available in [53]. A comparison of local and global methods for connectivity maintenance of multi-robot networks from Khateri et al. is available in [54]. In environments with limited connectivity, building and maintaining communication maps with information about the coverage and reliability of communication in different areas brings evident benefits. To this end, Amigomi et al. have presented a method for updating communication maps in an online manner under connectivity constraints [55].

2.2 MRS Task Allocation

Search and rescue operations with MRS involve aspects including collaborative mapping and situational assessment, distributed and cooperative area coverage, or cooperative search. These or other cooperative tasks involve the distribution of tasks and objectives within the MRS. In a significant part of the existing multi-robot rescue literature, this is predefined or done in a centralized manner. Here, we discuss instead distributed multi-robot task allocation algorithms that can be applied to rescue operations. Distributed algorithms have the general advantage of being more robust in adverse environments against the loss of individual agents or when communication with the base station is unstable. A comparative study on task allocation algorithms for MRS exploration was carried out by Faigl et al. in [56], considering five distinct strategies: greedy assignment, iterative assignment, Hungarian assignment, multiple travelling salesman assignment, and MinPos. However, most of these approaches are often centralized from the decision-making point of view, even if they are implemented in a distributed manner. Decentralized task allocation algorithms for autonomous robots are very often based on market-based approaches and auction mechanisms to achieve consensus among the agents [57]. Both of these approaches have been extensively studied for the past two decades within the multi-robot and multi-agent systems communities. An auction-based approach aimed at optimizing a cooperative rescue plan within MRS rescue systems was proposed by Tang et al. [58]. In this work, the emphasis was also put on the design of a lightweight algorithm more appropriate for ad-hoc deployment in rescue scenarios. A different approach where a human supervisor was considered appears in [59]. Liu et al. presented in this work a methodology for task allocation in heterogeneous MRS-supporting rescue missions. By relying on a supervised system, the authors show better adaptability to situations with robot failures. The algorithm was tested under a simulation environment where multiple semiautonomous robots were controlled by a single human operator.

2.3 Path Planning

An essential part of autonomous rescue operations is path planning and area coverage. To this end, multiple algorithms have been presented for different types of robots or scenarios. Planning in rescue scenarios can pose additional challenges to well-established planning strategies for autonomous robots. In particular, the locations of victims trapped under debris or inside cave-like structures might be relatively easy to determine but significantly complex to access, thus requiring specific planning strategies. In [60], Suarez et al. present a survey of animal foraging strategies applied to rescue robotics. The main methods that are discussed are directed search (search space division with memory- and sensory-based search) and persistent search (with either predefined time limits or constraint optimization for deciding how long to persist on the search). With specialized robots being used for different scenarios (e.g., tracked robots or crawling robots), the ability of these robots to traverse different environments might not be known a priori. To address this issue, ML-based techniques that rely on online learning have been utilized to create cost maps of the environment in terms of ease of movement. Path planning algorithms can be part of area coverage algorithms or implemented separately for robots to cover their assigned areas individually. In any case, when area coverage algorithms consider path planning, it is often from a global point of view, leaving the local planning to the individual agents. A detailed description of path planning algorithms including approaches of linear programming, control theory, multi-objective optimization models, probabilistic models, and meta-heuristic models for different types of UAVs is available in [61]. While some of these algorithms are generic and only take into account the origin and objective position, together with obstacle positions, others also consider the dynamics of the vehicles and constraints that these naturally impose in local curvatures, such as Dubin curves. Area coverage and path planning algorithms take into account mainly the shape of the objective area to be surveyed. Nonetheless, several other variables are also considered in more complex algorithms, such as energy consumption, range of communication and bandwidth, environmental conditions, or the probability of failure. This data is not necessarily available a priori, and therefore it is also in the interest of the robots to collect data affecting the planning outcome while operating. The problem of maximizing the utility of data collection is called informative path planning (IPP) problem [62]. IPP approaches have been shown to outperform more traditional planning algorithms such as greedy algorithms and genetic algorithms [63]. The specific dynamics and capabilities of the robots being used can also be utilized to optimize the performance of the area coverage,

2.4 Robot Types

Mobile robots operating on different mediums necessarily have different constraints and a variable number of degrees of freedom. For local path planning, a key aspect to consider when designing control systems is the holonomic nature of the robot. In a holonomic robot, the number of controllable degrees of freedom is equal to the number of degrees of freedom defining the robot's state. In practice, most robots are non-holonomic, with some having significant limitations to their local motion. However, quadrotor UAVs, which have gained considerable momentum owing to their flexibility and relatively simple control, can be considered holonomic. Ground robots equipped with Omni wheel mechanisms and able to omnidirectional motion can be also considered holonomic if they operate on favourable surfaces [64]. The main limitations in robot navigation, and therefore path planning, in different mediums, can be roughly characterized by: (i) dynamic environments and movement limitations in ground robots; (ii) energy efficiency, situational awareness, and weather conditions in aerial robots; (iii) under actuation and environmental effects in surface robots, with currents, winds and water depth constraints; and (iv) localization and communication in underwater robots. Furthermore, these constraints increase significantly in rescue operations, with earthquakes aggravating the movement limitations of UGVs, or fires and smoke preventing normal operation of UAVs. Some emergency scenarios, such as flooded coastal areas, combine multiple of the above mediums making the deployment of autonomous robots even more challenging. A key parameter to take into account in autonomous robots, and particularly in UAVs, is energy consumption.

2.5 Heterogeneous MRS

SYSTEMS: Most existing approaches for MRS exploration or area coverage either assume that all agents share similar operational capabilities, or that the characteristics of the different agents are known a priori. Emergency deployments in post-disaster scenarios for the rescue of victims, however, require flexible and adaptive systems. Therefore, algorithms able to adapt to heterogeneous robots that potentially operate on different mediums and with different constraints (e.g., UAVs and UGVs collaborating in rescue scenarios) need to be utilized. In this direction, Mueke et al. presented a system-level approach for distributed control of heterogeneous systems with applications to rescue scenarios [65]. In general, we see a lack of further research in this area, as most existing projects and systems involving heterogeneous robots predefine how they are meant to cooperate. From a more general perspective, an extensive review of control strategies for collaborative area coverage in heterogeneous MRS was recently presented by Abbasi [4].

2.6 Reserach Questions

Research efforts have mainly focused on the design of individual robots autonomously operating in emergency scenarios, such as those presented in the European Robotics League Emergency Tournament. Most of the existing literature on MRS for rescue either relies on an external control centre for route planning and monitoring, on a static base station and predefined patterns for finding objectives or has predefined interactions between different robotic units. Therefore, there is a big potential to be unlocked through the wider adoption of distributed MRS. Key advances will require embedding more intelligence in the robots with lightweight deep-learning perception models, the design and development of novel distributed control techniques, and a closer integration of perception and control algorithms. Moreover, heterogeneous MRS have shown significant benefits when compared to homogeneous systems. In that area, nonetheless, further research needs to focus on interoperability and ad-hoc deployments of MRS. Based on the different aspects of MRS rescue that have been described in this survey, both at the system level and from the coordination and perception perspectives, we have summarized the main research directions where we see the greatest potential. Further development in these areas is required to advance towards wider adoption of MRS rescue.

2.6.1 Shared Autonomy

With the increasing adoption of MRS for rescue operations over individual and complex robots, the number of degrees of freedom that can be controlled has risen dramatically. To enable efficient rescue support from these systems without the need for a large number of rescue personnel controlling or supervising the robots, the concept of shared autonomy needs to be further explored. The applications of more efficient shared autonomy and control interfaces are multiple. For instance, groups of UAVs flying in different formation configurations could provide real-time imagery and other sensor information from a large area after merging the data from all the units. In that scenario, the rescue personnel controlling the multi-UAV system would only need to specify the formation configuration and control the whole system as a single UAV would be controlled in a more traditional setting. While some of the directions towards designing control interfaces for scalable homogeneous MRS are relatively clear, further research needs to be carried out in terms of conceptualization and design of interfaces for controlling heterogeneous robots. In these cases, owing to the variability of their operational capabilities and significant differences in the robot's dynamics and degrees of freedom, a shared autonomy strategy is not straightforward.

2.6.2 Operational Environment

Some of the main open research questions and opportunities are the following In the Urban area, we see the main opportunities and open challenges to be in (i) collaborative localization in GNSS-denied environments; (ii) collaborative perception of victims from different perspectives; (iii) ability to perform remote triage and establish a communication link between rescue personnel and victims, or to transport medicines and food; and (iv) more scalable heterogeneous systems with various sizes of robots (both UGVs and UAVs) capable of collaborative mapping and monitoring harsh environments or post-disaster scenarios. some of the most important challenges in Wilderness operations are the potentially remote and unexplored environments posing challenges to both communication and perception. Therefore, an essential step towards more efficient MRS operations in Wilderness scenarios is to increase the level of autonomy and the operational time of the robots. Long-term autonomy and embedded intelligence on the robots for decision-making without human supervision are some of the key research directions in this area in terms of MRS.

2.6.3 Deep Learning Methods

Deep-learning-based methods are flexible and can be adapted to a wide variety of applications and scenarios. Good performance, however, comes at the cost of enough training data and an efficient training process that is carried out offline. Other deep learning methods, particularly deep reinforcement learning (DRL), rely heavily on simulation environments for converging towards working control policies or stable inference, with training happening on a trial-and-error basis. Search and rescue robots are meant to be deployed in real scenarios where the conditions can be more challenging than those of more traditional robots. Therefore, an important aspect to take into account is the transferability of the models trained in simulation to reality. Recent years have seen an increasing research interest in closing the gap between simulation and reality in DRL. In the field of rescue robotics, a relevant example of the utilization of both DL and DRL techniques was presented by

Sampedro et al. [66]. The authors developed a fully autonomous aerial robot for rescue operations in which a CNN was trained for target-background segmentation, while reinforcement learning was utilized for vision-based control methods. Most of the training happened with a Gazebo simulation and ROS, and the method was tested also in real indoor cluttered environments. In general, and compared with other DL methods, DRL has the advantage that it can be used to provide an end-to-end model from sensing to actuation, therefore integrating the perception and control aspects within a single model. Bridging the gap between simulation and reality is thus another challenge in some of the current rescue robotic systems.

2.6.4 Heterogeneous MRS

Across the different types of rescue missions that have been discussed in this survey, the literature regarding the utilization of heterogeneous robots has shown the clear benefits of combining either different types of sensors, different perspectives, or different computational or operational capabilities. Nonetheless, most of the existing literature assumes that the identity and nature of the robots and how they communicate and share data are known a priori. Wider adoption and deployment of heterogeneous MRS, therefore, needs research to advance in the following practice areas: - Interoperability: flexible deployment of a variable type and number of robots for rescue missions requires the collaborative methods to be designed with wider interoperability in mind. there is still a lack of interoperability in terms of high-level planning and coordination for specific missions. In rescue robotics, these include collaborative search and collaborative mapping and perception. - Ad-hoc systems: closely related to the concept of interoperability in terms of high-level planning, wider adoption of MRS rescue systems requires these systems to be deployed in an ad-hoc manner, where the type or number of robots does not need to be predefined. This has been explored, to some extent, in works utilizing online planning strategies that account for the possibility of malfunctioning or missing robots. - Situational awareness and awareness of other robots: the wide variety of robots being utilized in rescue missions, and the different scenarios in which they can be applied, calls for the abstraction and definition of models., denning these scenarios but also how robots can operate with them. In heterogeneous MRS, distributed high-level collaborative planning requires robots to understand not only how can they operate in their current environment and what are the main limitations or constraints, but also the conditions of different robots operating in the same environment.

2.6.5 Active Perception

We have closed this survey by exploring the literature on active perception for MRS, where we have seen a clear lack of research within the rescue robotics domain. Current approaches for area coverage in rescue missions, for instance, mostly consider an a priori partition of the area among the available robots. Dynamic or online area partitioning algorithms are only considered either in the presence of obstacles or when the number of robots changes. Most of the works are based on either prior knowledge of the area or otherwise partitioning the search space in a mostly homogeneous manner. Therefore, there is an evident need for more efficient MRS Search strategies Active perception can be merged into current MRS rescue in multiple directions: actively updating and estimating the probabilities of victims' locations, but also with active SLAM techniques by identifying the most severely affected areas in post-disaster scenarios. In wilderness and maritime search and rescue where tracking of the victims might be necessary even after they have been found, active perception has the potential to significantly decrease the probability of missing a target. In general, we also see the potential of active perception within the concepts of human-robot and human-swarm cooperation, and in terms of increasing the awareness that robots have of victims' conditions. Regarding human-robot and human-swarm cooperation, active perception can bring important advantages in understanding the actions of rescue personnel and being able to provide more relevant support during the missions.

3 Proposed Architecture

In this section, we present the proposed architecture for the rescue operation of robots under various distress types (e.g., cyber-attack, sensor malfunction), using the components which have been designed in D2.3 Collaborative Sensor Fusion and, D2.4 Multi-Robot Monitoring Online Trajectory Generation. We tailor the components developed already in WP2 into one integrated scheme taking advantage of the benefits of individual ones. We, also, aim to design a generic scheme to have applicability for different "RESCUE" actions.

Collaborative sensor fusion is to facilitate the "rescue" of other robots that are affected by sensor malfunctioning or any type of cyber-attack. Trajectory planning is performed to keep other robots (s) in the field of view. Furthermore, a rescue action is performed for the robot malfunctioning. In this architecture, we have foreseen two roles for robots, namely, detector robot and target robot. Collaborative sensor fusion will be running on the detector robot to provide real-time detection, monitoring and state estimation of the target robot (the one which is likely to be malfunctioning). The main output is the estimated states of the target drone.

In summary, there are two factors related to trajectory planning, i.e., a drone is performing a trajectory to keep other robots in the field of view, and the robot affected by an attack or sensor issue receives a rescue action.

First, these components are briefly summarized and the rescue architecture is presented. It should be noted that the presented architecture is aimed at drone operations, however, the architecture itself is structured as generic as possible to be applicable to different robotic operations.

3.1 Collaborative Sensor Fusion

In this section, the collaborative sensor fusion algorithm is briefly summarized. Further details are presented in D2.3 Collaborative Sensor Fusion report. One of the primary objectives of this project is to robustly estimate the pose of a robot using its own sensors, but also using position estimates shared by other robots in its vicinity. More precisely, we will demonstrate the applicability of our algorithms using two robots: a detector robot, whose goal is to estimate the position of a target robot. Using this position estimation and its own sensors, the target robot estimates its position robustly. Using the extra information provided by the detector robot, the target drone can still estimate its state even if one of its positioning sensors were to fail. To estimate the position of the target robot, we rely on camera images and a depth sensor of the detector robot. Using a neural network specifically trained to detect robots in images, we first identify the target robot in the image. Then, using a depth sensor, we measure the distance to the target robot. Leveraging this distance information, we can then estimate the relative position of the target robot in the detector robot's frame. This position is then projected into a global frame and shared with the target robot.

As detailed in the introduction, we apply these algorithms to drones, hence, some of the method subsections will provide some application-specific details, yet they should translate well to other types of robots. Figure 1 provides an overview of this objective when applied to drones.

The second goal of this task is to provide scene understanding to the robots. To do so, we propose two different approaches. A classical approach based on scene segmentation. And an object detection-based approach. In the first one, semantic segmentation, we train a neural network to provide high-level context to the robot. In this case, we build two different algorithms. One provides information regarding the occupancy of the space in front of the drone, i.e. it detects obstacles and free space in front of the drone. The other labels image region, in the case of the drone, we segment the pixels that belong to the ground or a wall for instance. On a rover, this information could be used to detect unsafe areas, such as avoiding stairs.

In the second one, we apply the same algorithms as the robot detection except, this time, we detect different obstacles that must be avoided. This has numerous advantages, first, it comes at no extra computational cost, and the inference time of the object detector does not change that much when the number of detected objects increases. Hence, adding to it the ability to detect humans or other relevant obstacles is only increasing the cost of the tracking part which is computationally cheap compared to running a network, and can easily be

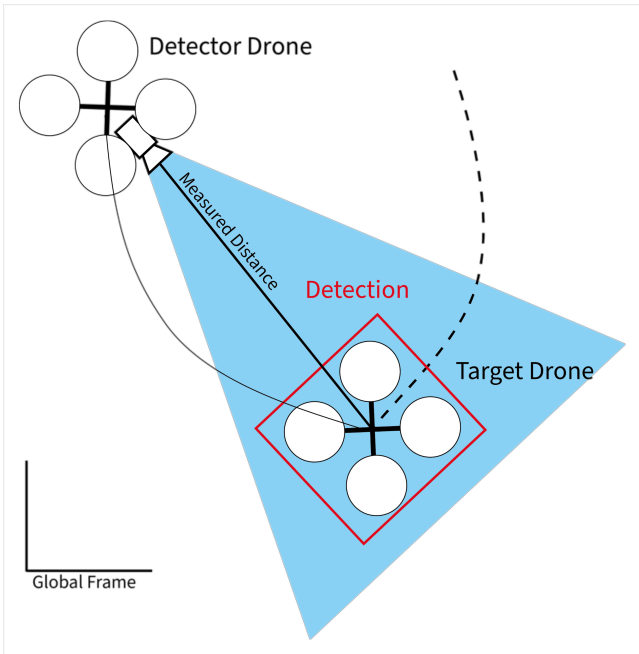

Figure 1: Overview of the sensor fusion task.

threaded. Second, since it can be reduced to a 3D bounding box, this information is lightweight and can easily be shared across a fleet of robots. This is particularly true when compared to a map. Moreover, when done in a distributed fashion, the reconstruction of 3D maps is particularly heavy both in terms of computing and for the network. While the information provided by the object detection is not as rich as a 3D map, it is significantly lighter, making it better suited to our use cases. Also, as mentioned in the introduction, we consider that an up-to-date global map already exists and that the only changes that occurred are related to obstacles that can be identified using object detection.

Unlike the previous scenario example, we apply our obstacle detection algorithms to rovers, to show that our algorithms are not tied to drones and can easily be deployed on other systems. In fact, all the examples shown here are running on the same code when it comes to vision-based state estimation.

To perform the detection, we use the smallest version of yolov5, the S version. Unlike the other larger variants of yolov5, this smaller version has nine convolutional layers with 3×3 kernel layers and six pooling layers with 2×2 kernel layers. The final output of our network is a tensor of size $13 \times 13 \times 30$. These modifications make its memory footprint smaller and significantly reduce its inference time. This makes it ideal for embedded applications. The drawback is a decrease in accuracy. However, the performance drop remains acceptable because we only seek to detect drones. Compared to the original algorithm, we only detect a single class, drones, instead of 80.

The collaborative sensor fusion part has been encapsulated into docker and proven to be reliable working providing accurate estimation.

3.2 Trajectory Planning

In this section, the trajectory planning algorithm is briefly summarized. Further details are presented in D2.4 Multi-Robot Monitoring Online Trajectory Generation. It is worth noting that the planner algorithm has been structured to be as generic as possible to be "Plug and Play" to some extent.

The trajectory generation component, preferably running off board, will take the position of the target robot to generate a trajectory to "RESCUE" it. Therefore, the planner will use the estimated states not measured ones.

Furthermore, it generates another trajectory for the detector robot to keep the target drone within the field of view.

From the higher level planner, the waypoints W_i of each robotic agent, for $i \in \{1, ..., K\}$, are determined and sent to the lower level decentralized planner, where the trajectory planning for each robot is obtained as the solution to an optimization problem. It is guaranteed that the robotic agent passes through the corresponding waypoints. The trajectory planning is mathematically formulated as

$$x_{d,T}(t) = \underset{x_T(t)}{\operatorname{argmin}}\, J\left(x_T(t)\right), \tag{1}$$

subject to

$$\begin{aligned}
J\left(x_T(t)\right) &= t_f\left(x_f, x_T(t)\right) - t_0,\\
x_T(t_0) &= x_0,\\
x_T(t_f) &= x_f,\\
x_T(t_i) &= W_i,\\
\dot{x}_T(t) &= f_T(x(t), u(t)),\\
y_T(t) &= x_{T,est}(t),\\
x_L \leq x_T&(t) \leq x_U,\\
u_L \leq u&(t) \leq u_U,\\
p_L \leq p&(t) \leq p_U,\\
R_k \leq \|x&(t)_T - x_{obs,k}(t)\|,
\end{aligned} \tag{2}$$

where, T represents the target robot, $x_T(t)$ is the dynamics state vector, $x_{T,est}(t)$ is the estimated states of the robot, t_f and t_0 are final and initial operation times, respectively, x_f and x_0 are final and initial states, respectively, $u(t)$ is the control command, $t_0 \leq t_i \leq t_f$, for $i \in \{1, ..., K\}$, are increasing time sequence, W_i is the state of the i^{th} waypoint, $f(x(t), u(t))$ is the dynamics equations governing the motion of the robot, and $y(t)$ is the measurements vector. X_L and X_U denote the element-wise lower and upper bound vectors on the variable vector $X(t)$. Finally, $x_{obs,k}(t)$ and R_k represent the position and safety radius of k^{th} obstacle. In fact, $R_k \leq \|x(t) - x_{obs,k}(t)\|$ imposes the constraint to keep the state of the robot outside of the safety sphere around the obstacle.

To solve the optimization (1), there are many approaches, as reviewed in Section 2. However, as we analyzed the existing works, the main issue with the real-time implementation and fast optimization is the implementation of the constraints (2). Specifically, the inequality constraints might cause issues in the convergence of the solver. Accordingly, it is really crucial how we implement the inequality constraints. As discussed later, we are going to use the Legendre pseudospectral method to transcribe the continuous optimization problem into an equivalent discrete one. The trajectory is represented by a number of Legendre functions passing through a number of collocation points. Then, the position of these collocation points is as the optimization variables. Therefore, the constraints are to be applied to these collocation points.

To apply the obstacle avoidance constraint $R_k \leq \|x_T(t) - x_{obs,k}(t)\|$ to the collocation points, the common condition to be checked is the distance of the points from the obstacle to be greater than the safety radius.

For the target robot, the trajectory planning optimization problem is formulated as

$$x_{d,D}(t) = \underset{x_D(t)}{\operatorname{argmin}}\, J\left(x_D(t)\right), \tag{3}$$

Algorithm 1 Perception aware MRS trajectory planning and tracking

1: Design $x_d(t)$, by optimizing $J(t)$, subject to
- Passing through the waypoints (defined based on the decomposed task),
- Identified dynamics of the robot and estimated states,
- Actuator limit,
- Keeping distance by the estimated obstacle position,
- Retaining in /avoiding from some areas or points (perceptive information),
- Satisfying the safety and security metrics.

2: Design $u(t)$, by taking into account the currently estimated states of the robot and designed $x_d(t)$,

subject to

$$\begin{aligned}
J\left(x_D(t)\right) &= t_f\left(x_f, x_D(t)\right) - t_0, \\
x_D(t_0) &= x_0, \\
x_D(t_f) &= x_T - R_{saf}\frac{x_T - x_D}{\|x_T - x_D\|}, \\
\dot{x}_D(t) &= f(x(t), u(t)), \\
y(t) &= x_D(t) + \epsilon_x(t), \\
p(t) &= P(x_T(t), x_D(t)) \\
x_L &\leq x_D(t) \leq x_U, \\
u_L &\leq u(t) \leq u_U, \\
p_L &\leq p(t) \leq p_U, \\
R_{saf} &\leq \|x_T - x_D\|,
\end{aligned} \tag{4}$$

It is worth noting that for the detector drone trajectory planning, it is assumed that no obstacle is obstructing the target drone from the FoV of the detector drone.

where D represents the detector robot, R_{saf} is the safety radius from target robot and $p(t)$ is the perceptive index governed by the equation $P(.)$. In this formulation, we define $p(t)$ as the field of view of the detector robot, while we aim to keep the target robot within the field of view. This is achieved by pointing the detector robot towards the detector robot.

3.3 Sensor Malfunction

In the context of this work, the scope of cyber-attacks or faults is considered as a sensor becoming impaired, e.g., losing the GPS positioning. Similarly, the scope of the adverse environmental conditions is bounded to events that may perturb the state estimation of a system, such as entering GPS-denied areas. Building on recent advances in deep learning, perception algorithms estimate the position of objects in a scene. This can for instance enable an observer robot to visually estimate the position of the target robot. Sharing this estimation with the tracked robot could help it recover from a faulty GPS sensor for instance. Likewise, the observer robot can detect obstacles in the vicinity of the other robot and share its position with it. On the other hand, online perception-aware trajectory generation considers the close environment information gathered by nearby robots using the collaborative sensor fusion approach developed in Task 2.3 and the mission goals. Target robots will be “monitored” by one or more robots, but not all of them. Also, when a cyber-attack or sensor malfunction affects one or more members of the team, non-attacked robots will generate a rescue trajectory providing further sensor feedback to the compromised robots and helping them to maintain their operational and safe state.

3.4 Triggering and Switching Mechanisms

It is assumed that at the moment the sensor malfunction happens, there already exists a safety component to detect this malfunction. This, consequently, is announced to either the onboard controller or ground control station via a flag. in our proposed architecture we presume this detection mechanism is already available. Accordingly, a mechanism is triggered to switch the measurements from the onboard sensors of the target robot to the estimated ones from the detector robot. For this purpose, it is required to steer the detector robot to the vicinity of the target robot and keep it in its field of view. In case the detector robot is far away, it is needed to have a redundancy procedure to keep the target drone safe, between the moments of the switching. For example, in the case of drones, we can switch to altitude mode to keep them at their position. This redundancy component is inevitable and has to be identified for the target robot type. Without loss of generality, in the proposed architecture we assume that the detector robot has the target robot in its field of view from the beginning and provides the target robot's estimated states.

3.5 Rescue Actions

The corresponding rescue actions, given the sensor malfunctioning of the target robot, are usually determined based on the severity of the issue as well as the nature of the robotic operation. The rescue actions can be categorized, namely, as

- Continue operation: In this case, even though the malfunctioning has been already identified, the accuracy of the estimated states is high, or the operation itself is not that hazardous. Therefore, the target robot continues its operation using the estimated states provided by the detector robot.
- Holding at its position: In this case, it is not possible to terminate the operation right after the detected malfunction. Moreover, it is not safe to continue the operation. Therefore, the corresponding action is to control and hold the target robot in its own position.
- Steering back to Home and land (Return to base): In this case, with the aid of an estimated position, the target drone is controlled and navigated back to a predefined position (home), and either held there or immediately lands.

3.6 Rescue Planning Architecture

After the above-mentioned considerations, in this section, we present the proposed rescue planning architecture, schematically. It should be noted that the main concern of this architecture is the drone operation. However, it can be readily adopted for a variety of different robot types. The overall architecture is shown in Figure 2.

In the proposed architecture, there are three main components, i.e., target drone, detector drone, and ground control station. The collaborative sensor fusion component is running on the detector drone. On the other hand, the trajectory planning component for both detector and target drones is running on the ground control station computers for faster computational power.

The communication is on the shared network of these components. Practically, communication happens using Telemetry. However, to have faster and more reliable communication, the data transfer can be a shared Wi-Fi network. Regardless of the network architecture, the drones and ground control stations will be communicating using MAVROS and MAVLink nodes. This is illustrated in Figure 3.

As illustrated in Figure 3, as soon as the sensor malfunction is detected, the flag is sent to the switching mechanism for the detector drone planning. In this case, the detector drone is sent to the latest position of the target drone and detects the target drone via the image processing component. Then, the position and way angle of the detector drone is controlled via the trajectory planning component to keep the target drone within the field of view (FoV). This is shown in Figure 4. Consequently, the sensor fusion component provides the

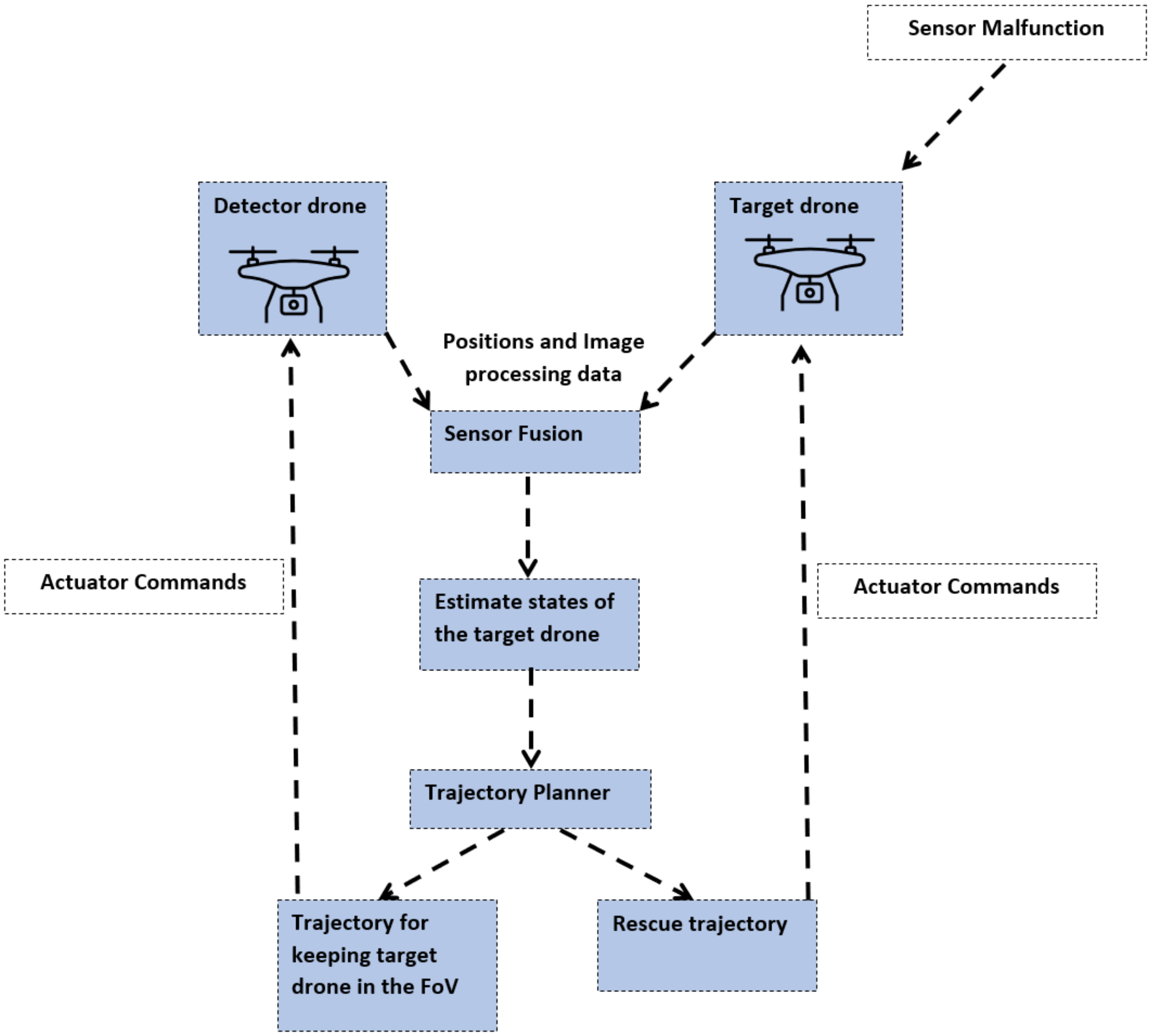

Figure 2: Overall architecture of rescue planning.

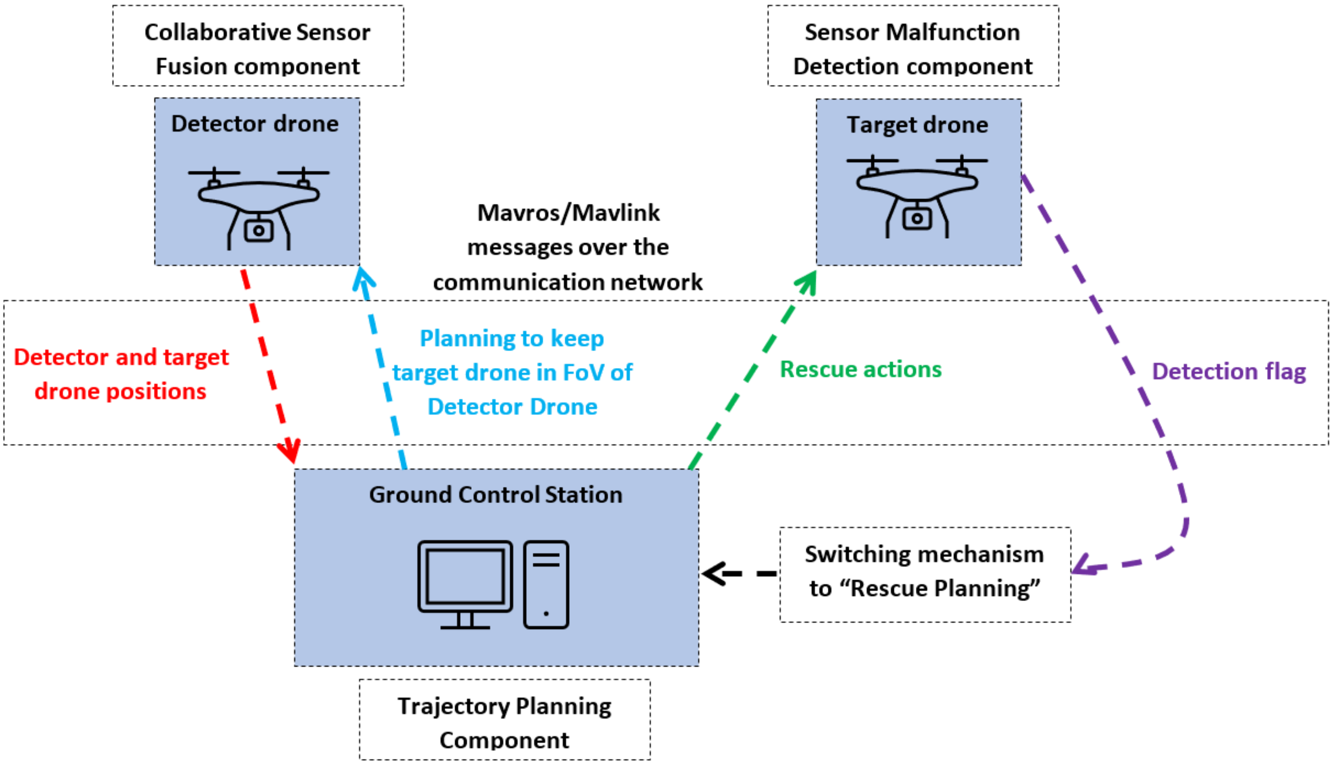

Figure 3: Netwrok for MRS rescue mission.png

estimated position of the target drone to the ground control station. It is worth noting from the malfunction was detected till the position estimates are provided, a redundancy mechanism is switched to keep the target drone at its position. However, in our evaluation, without loss of generality, we assume from the beginning of the operation, even before the malfunction happens, the detector is monitoring the target drone. So, we use the estimated states of the target drone for its trajectory planning. As soon as the malfunction happens, the ground control station provides the corresponding rescue actions.

Considering the corresponding rescue actions, we let the generic trajectory planning, as presented in Section 3.2, be applicable and we just switch the final position to the home/base position. Then, the target drone is held there or landed, as shown in Figure 5. If the immediate landing is requested, the trajectory planning code is terminated and the drone lands. Furthermore, the operation can be continued if it is requested using the estimated states.

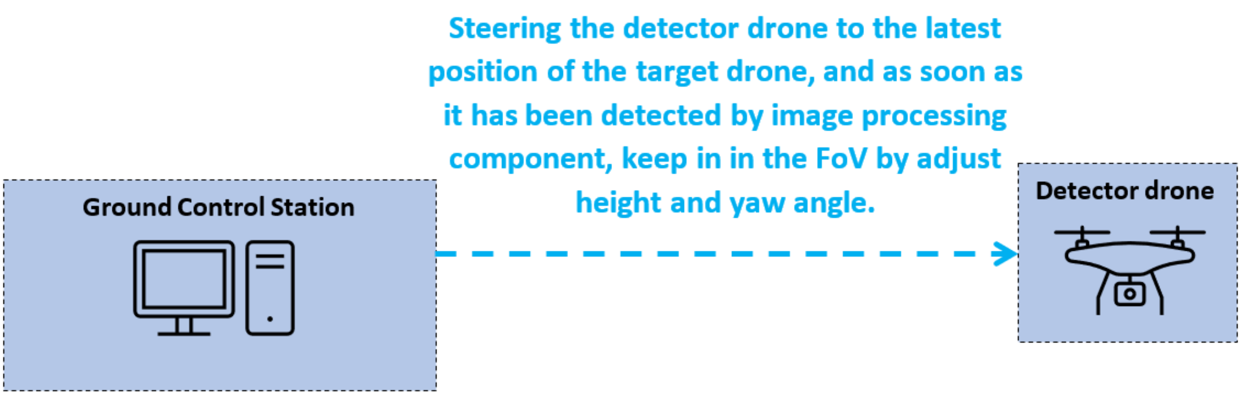

Figure 4: Detector drone monitoring trajectory planning

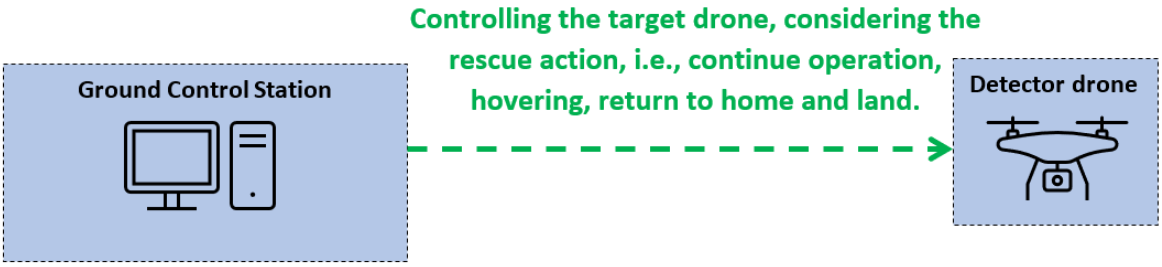

Figure 5: Target drone rescue trajectory planning

In Algorithm 2, the collaborative rescue operation is summarized.

Algorithm 2 MRS Collaborative Rescue Mission

1: Running the sensor malfunction detection unit on the target drone
2: Sending the detection flag to the ground control station.
3: Activating the switching mechanism. in case the detection flag is on.
4: Activating the redundant component on the target drone to keep it at its current position.
5: Recording the latest position of the target drone and commanding it to the detector drone.
6: Planning the trajectory for the detector drone and navigating it to the vicinity of the target drone.
7: Running the image processing unit to detect the target drone.
8: Controlling the detector drone height and yaw angle to keep the target drone within the FoV.
9: Estimating the target drone states using the sensor fusion component.
10: Identifying the corresponding rescue actions.
11: Designing the rescue action, for the target drone, either by change of the final position, landing, or continuation of the operation.
12: Running trajectory planning for both detector and target drones.

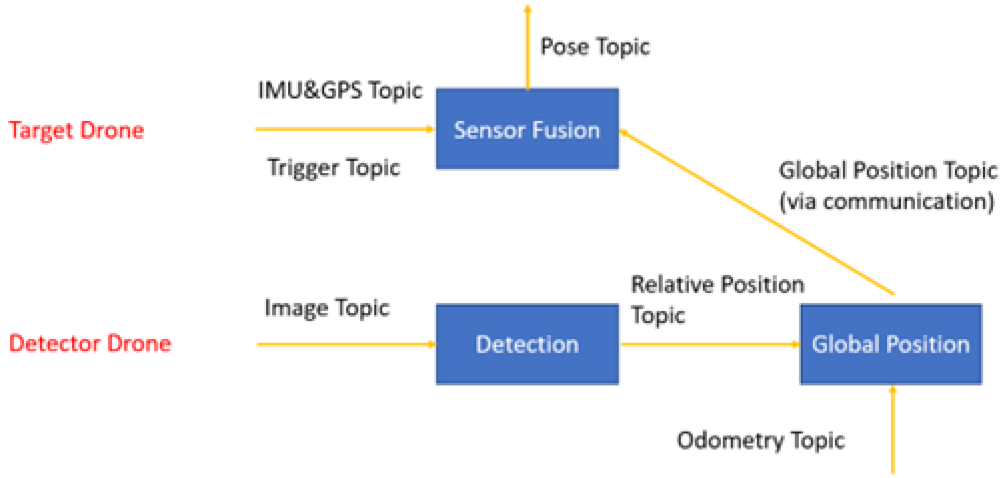

Figure 6: Message flow for the specific rescue mission.

4 Results and Discussion

In this section, we evaluate the presented architecture with its components in the high-fidelity Gazebo simulations. Here, we have made the following assumptions:

Assumption 1: The individual robot has a private internal state estimator. The sensors used by the robot could be different with different robot platforms. Their odometry systems share the same global frame.

Assumption 2: If sensors are good, the robot can publish specific odometry topics. If all the sensors are bad except IMU during specific timeslots, then the robot needs another robot to provide position information.

Assumption 3: If the target robot experiences such failure, we use the detector robot to provide the position of the target robot, which is calculated by the odometry of the detector robot and inferred relative position from the detector robot. The position of the target robot is sent by the detector robot and will be fused with the IMU of the target robot. Abstract message flows for the specific rescue mission are shown in Figure 6.

We test Collaborative Sensor Fusion (CSE) in the gazebo simulation environment. The EKF approach developed in Task 2.3 is used as an example. When GPS is normal, the target drone outputs the fusion results of GPS and IMU. When GPS fails, the target drone outputs the fusion result of the observation position and IMU. The observation position is provided by the detector drone. As shown in Figure 7, we manually disable GPS during the middle period of the path (60s to 120s) to simulate the trigger signal indicating the abnormal GPS. At this time, the observation position output by the detector drone is obtained by adding Gaussian noise to the ground-truth position of the target drone. Two types of Gaussian noise are generated, one is the smaller 10cm, and the other is the larger 20cm. It can be seen that the trajectory error of the target drone is at cm level when using high-accuracy GPS. When using the observation position, the trajectory error is at dm level. The increase in error is expected due to the use of low accuracy observation position. We have achieved the goal of ensuring the position of the target drone without divergence.

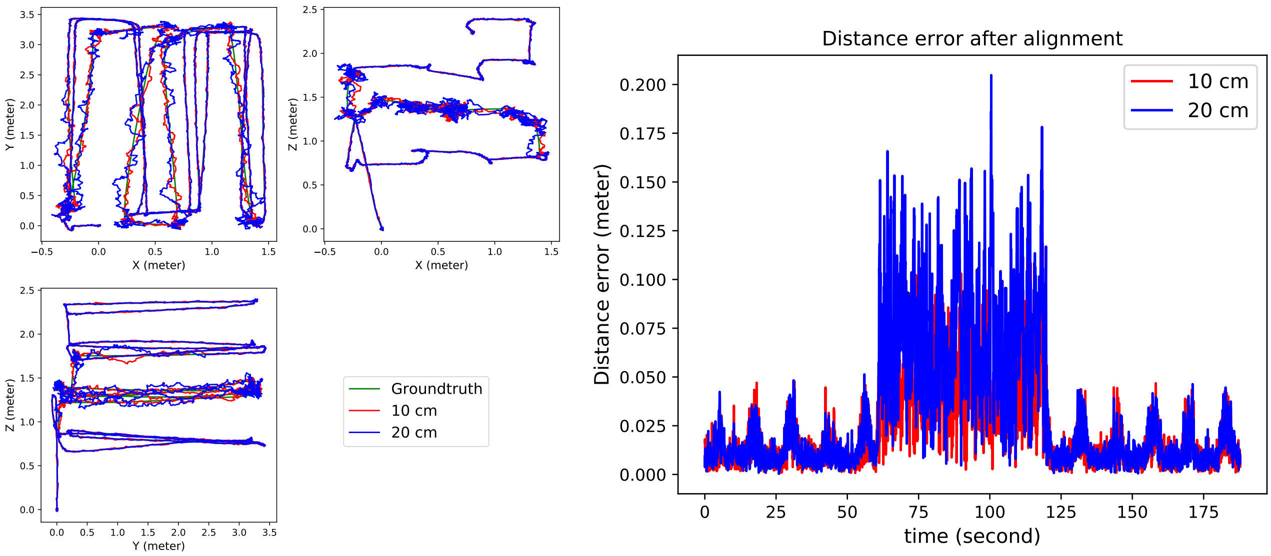

Figure 7: Left: Different views of the aligned trajectories from the CSE module evaluated with different levels of simulated position noise. Right: The distance error of the aligned trajectories evaluated with different levels of simulated position noise.

Now, the evaluation of the trajectory planner with the docker simulation model, for integration of the sensor fusion is given here.

In Figures 8, 9 and 10 the simple waypoints navigation is considered. It is obvious that the target drone is detected by the detector drone. Then, the position of the target drone is controlled and navigated through the takeoff and waypoints. Moreover, we have the holding command at each point for 3 seconds. This functionality might be useful for some use cases, e.g., vineyard pesticide spraying.

In Figure 11, simple trajectory planning for normal operation integration with the docker model is illustrated. As it is obvious, the target drone is navigated between the initial and final points, avoiding the moving obstacle.

Now, the rescue operation is illustrated in Figures 12-15. In Figure 12 and 13 the normal trajectory planning operation with avoiding the obstacle is started. Then, as shown in Figure 14, the rescue flag is activated and for the corresponding action, the drone is navigated to the home position. This is achieved by changing the final point in the trajectory optimization to the home position. Consequently, after the drone reaches the home position, it is held there and land. It is worth noting that during the rescue operation, obstacle avoidance is also achieved.

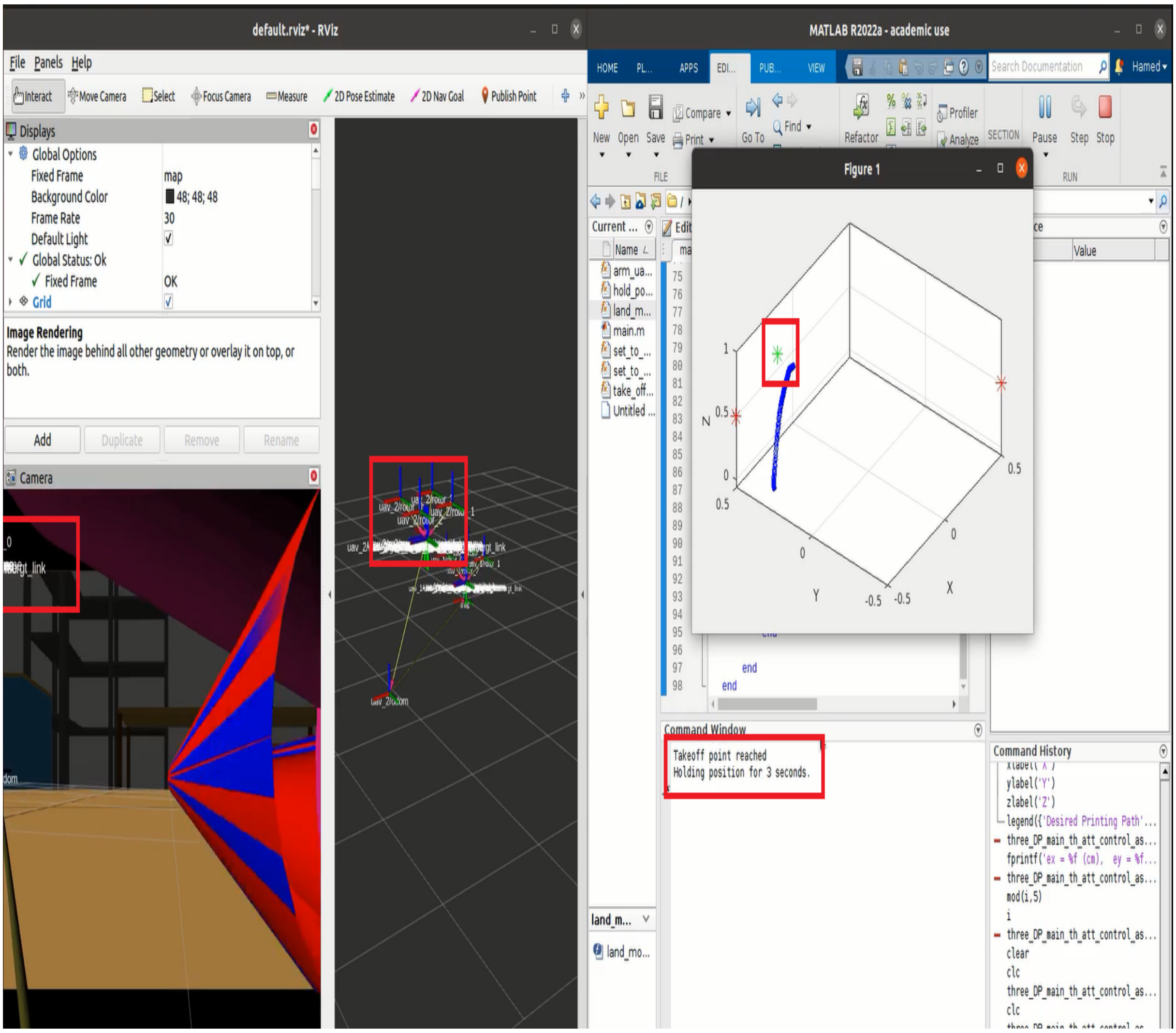

Figure 8: Waypoint navigation for reaching the takeoff point (green star) and holding.

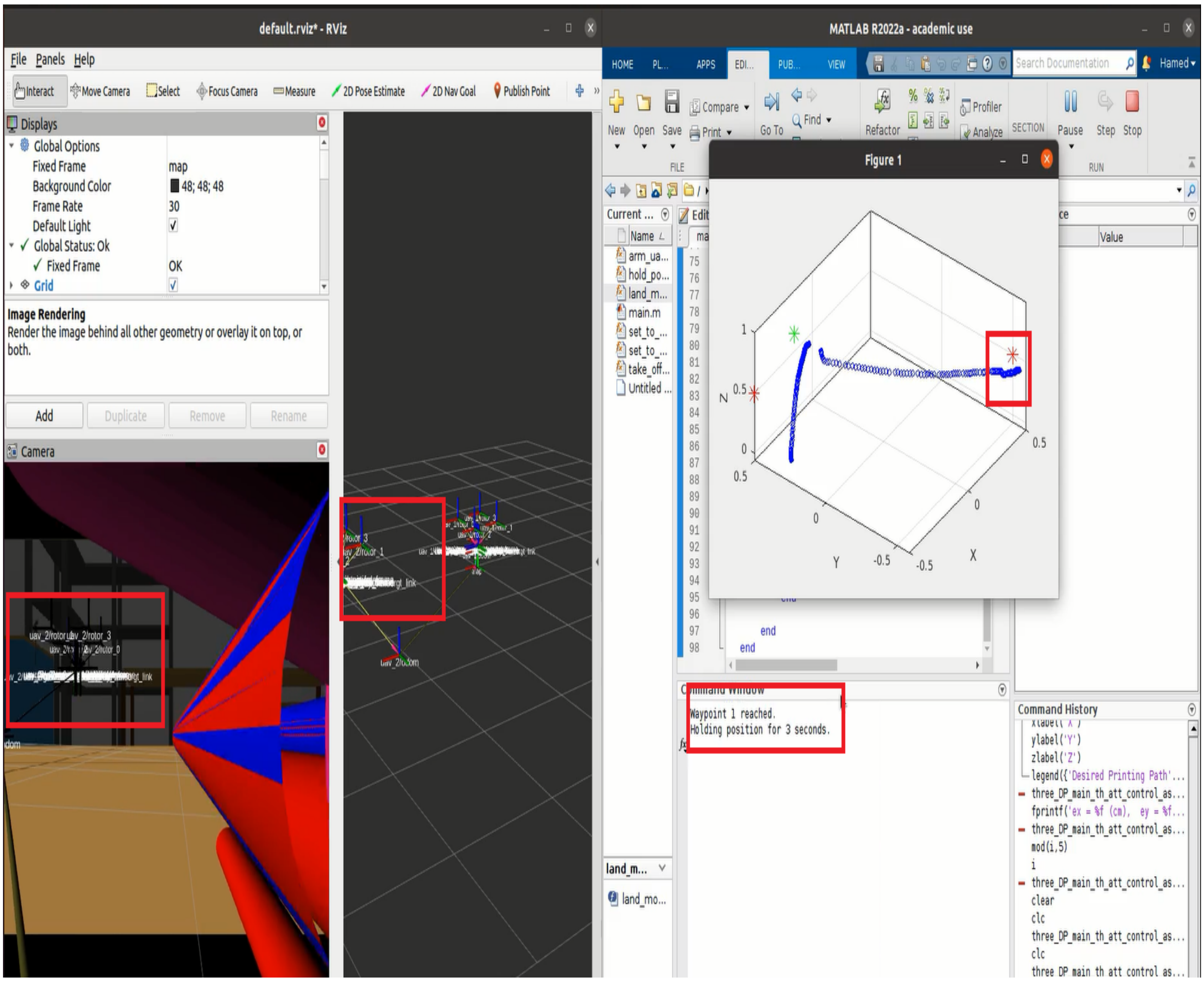

Figure 9: First Waypoint (red star) navigation and holding.

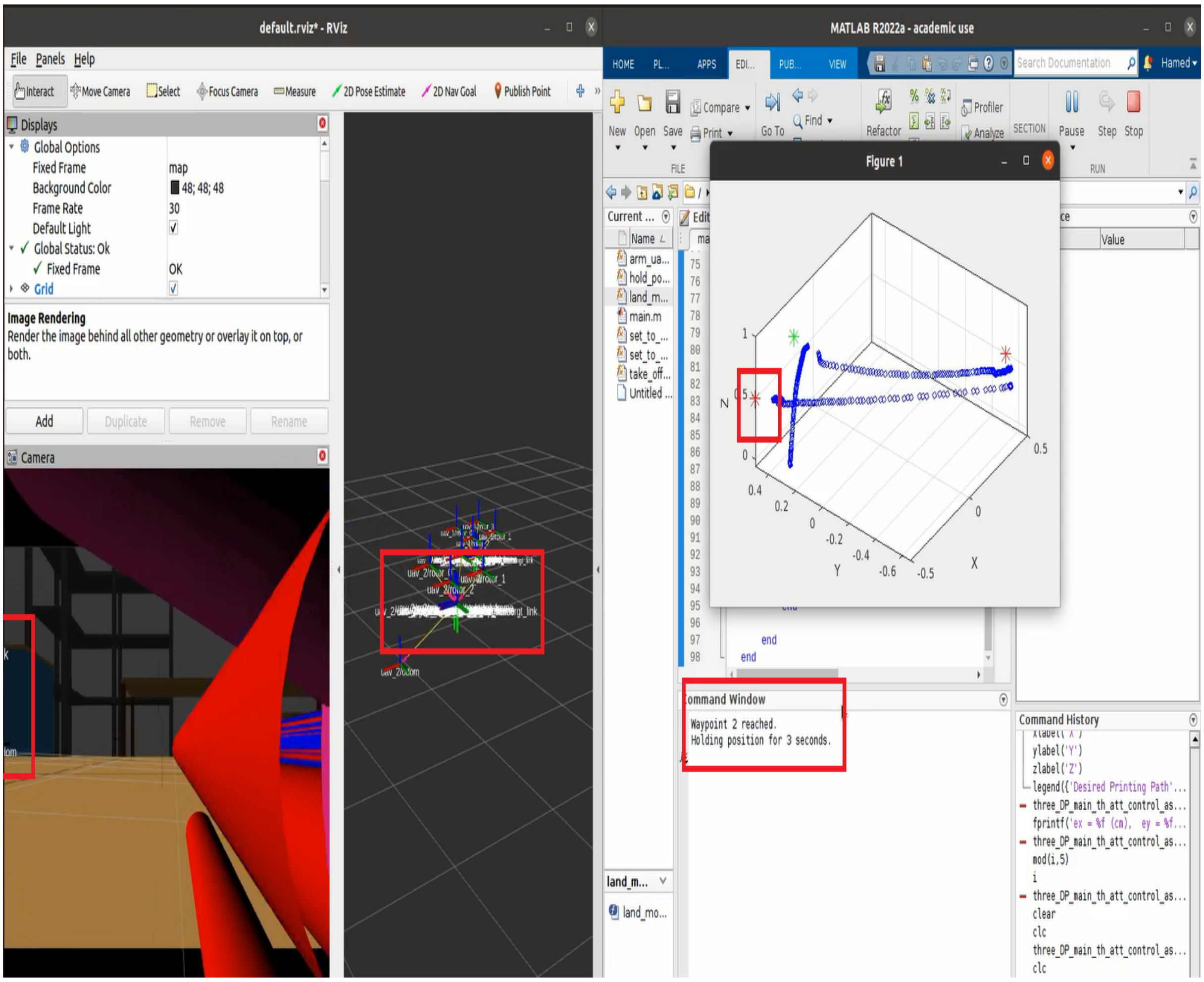

Figure 10: Second Waypoint (red star) navigation and holding.

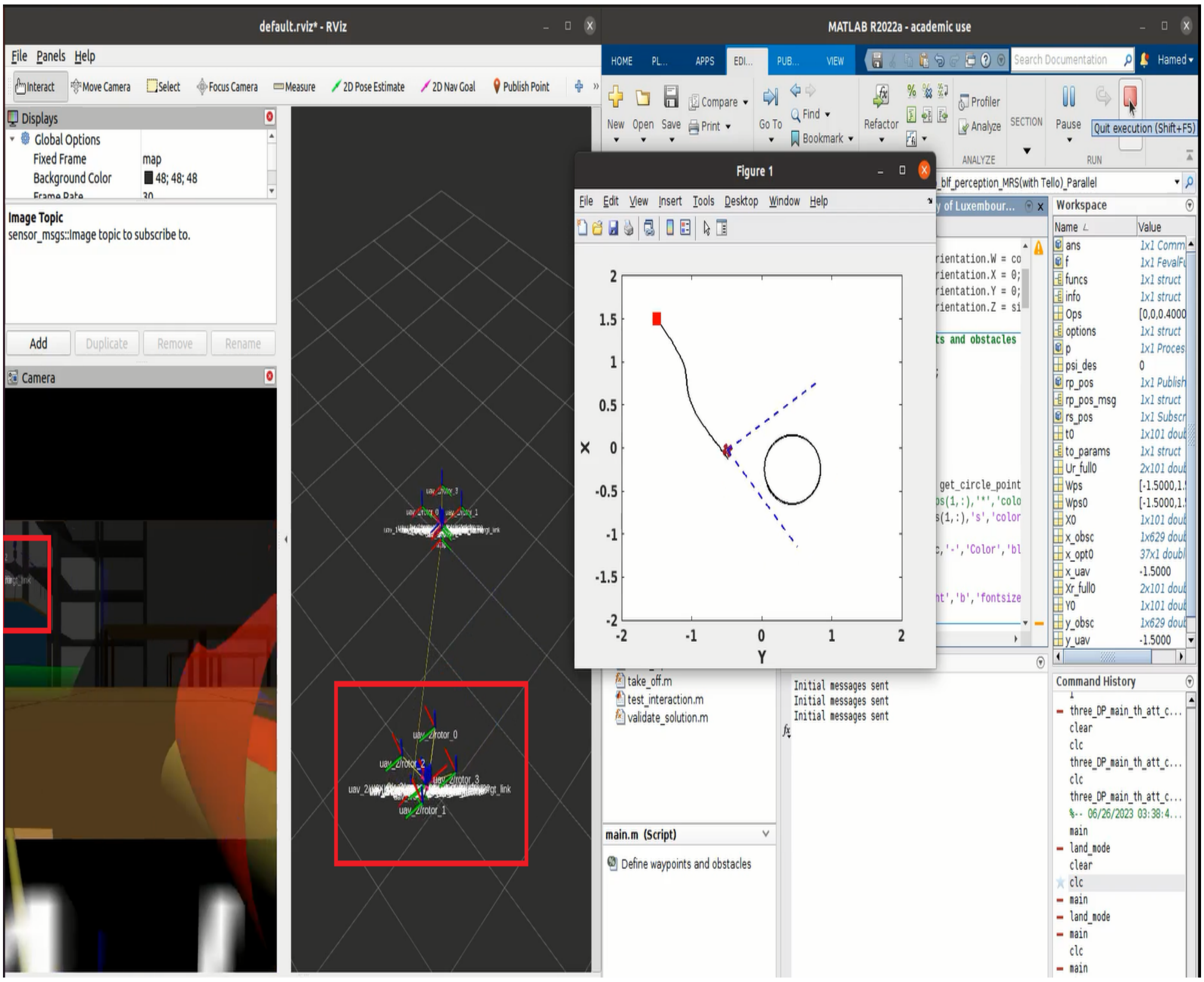

Figure 11: Trajectory planning for normal operation with obstacle avoidance.

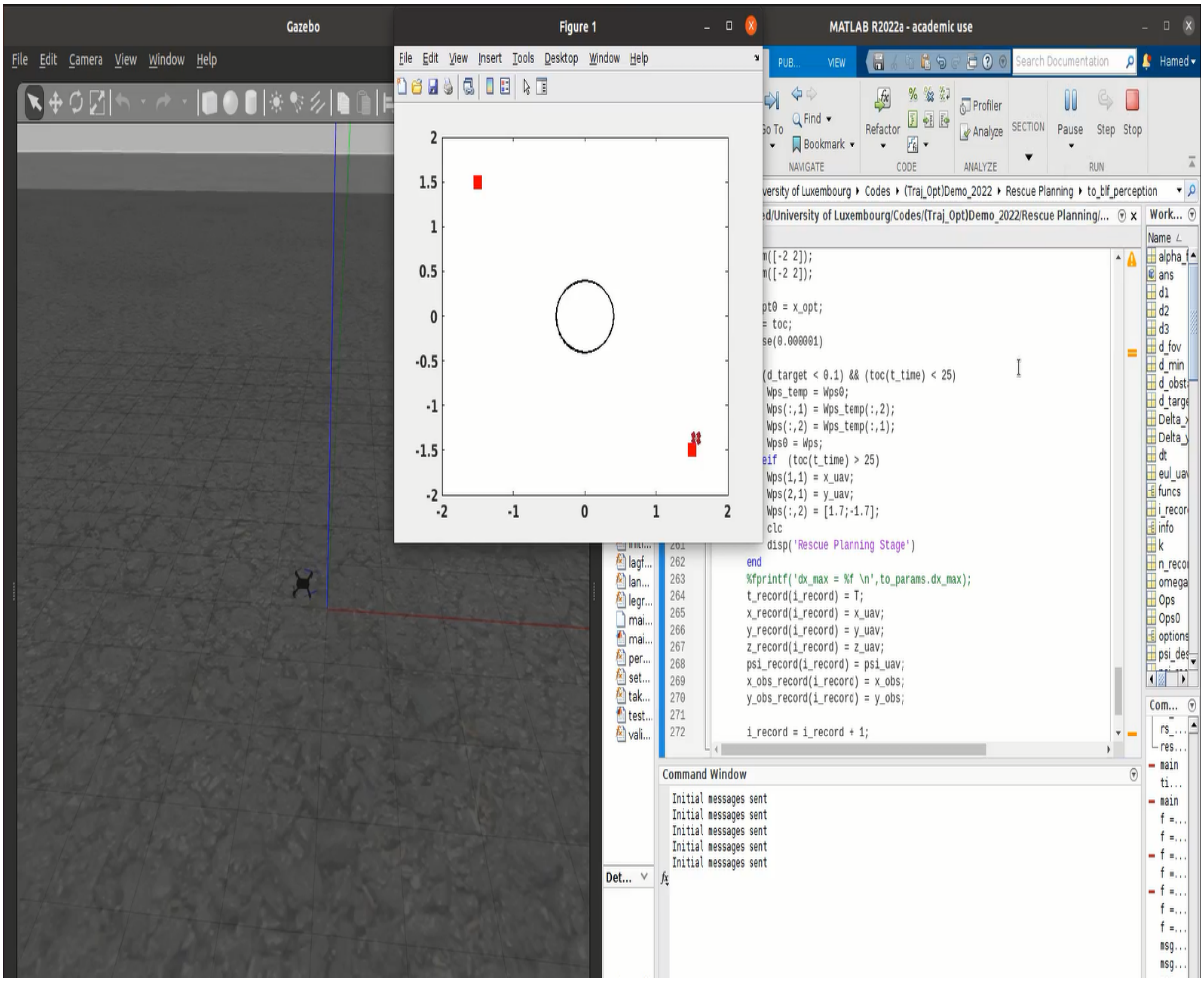

Figure 12: Rescue operation; reaching the initial point.

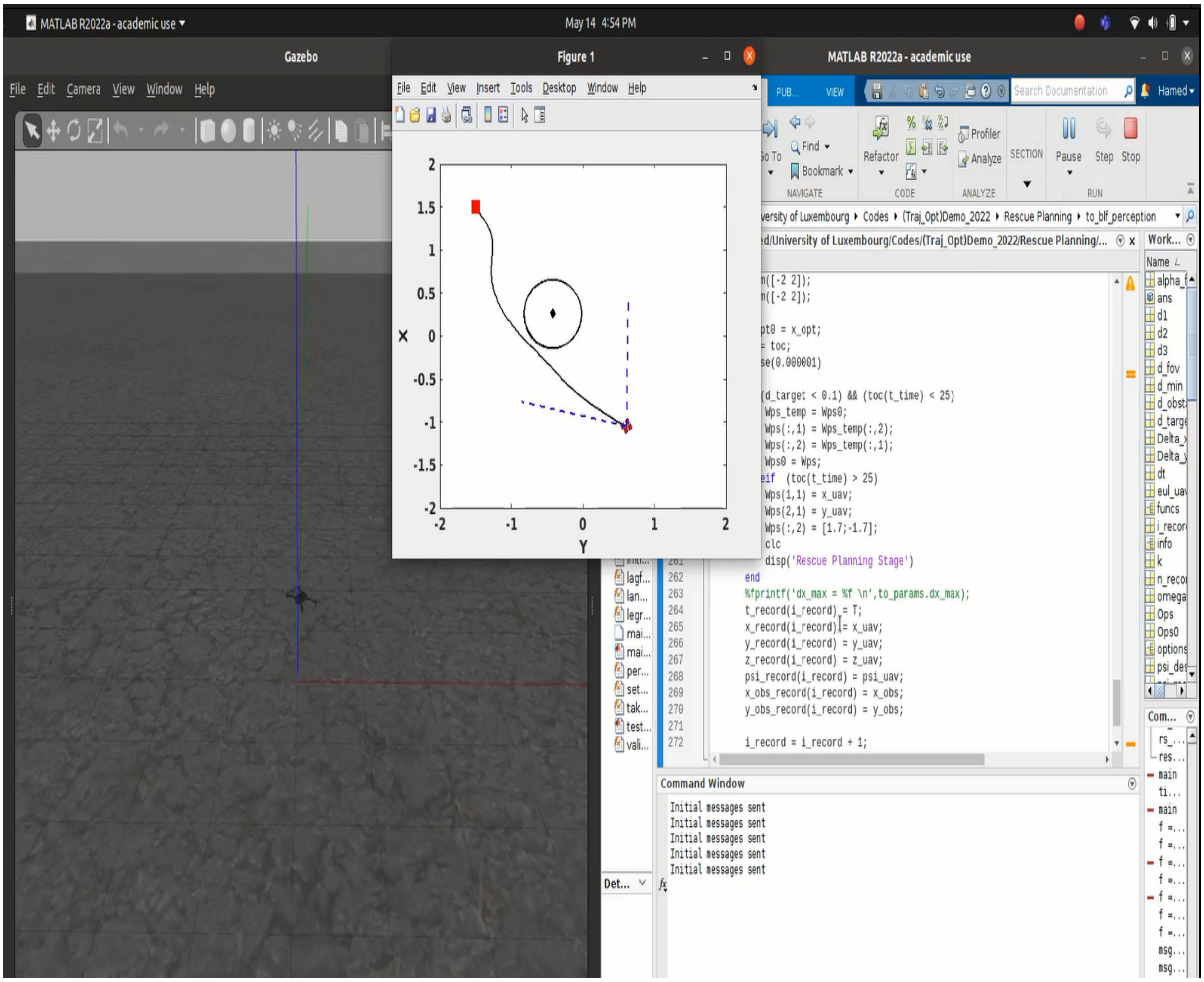

Figure 13: Rescue operation; starting the normal operation.

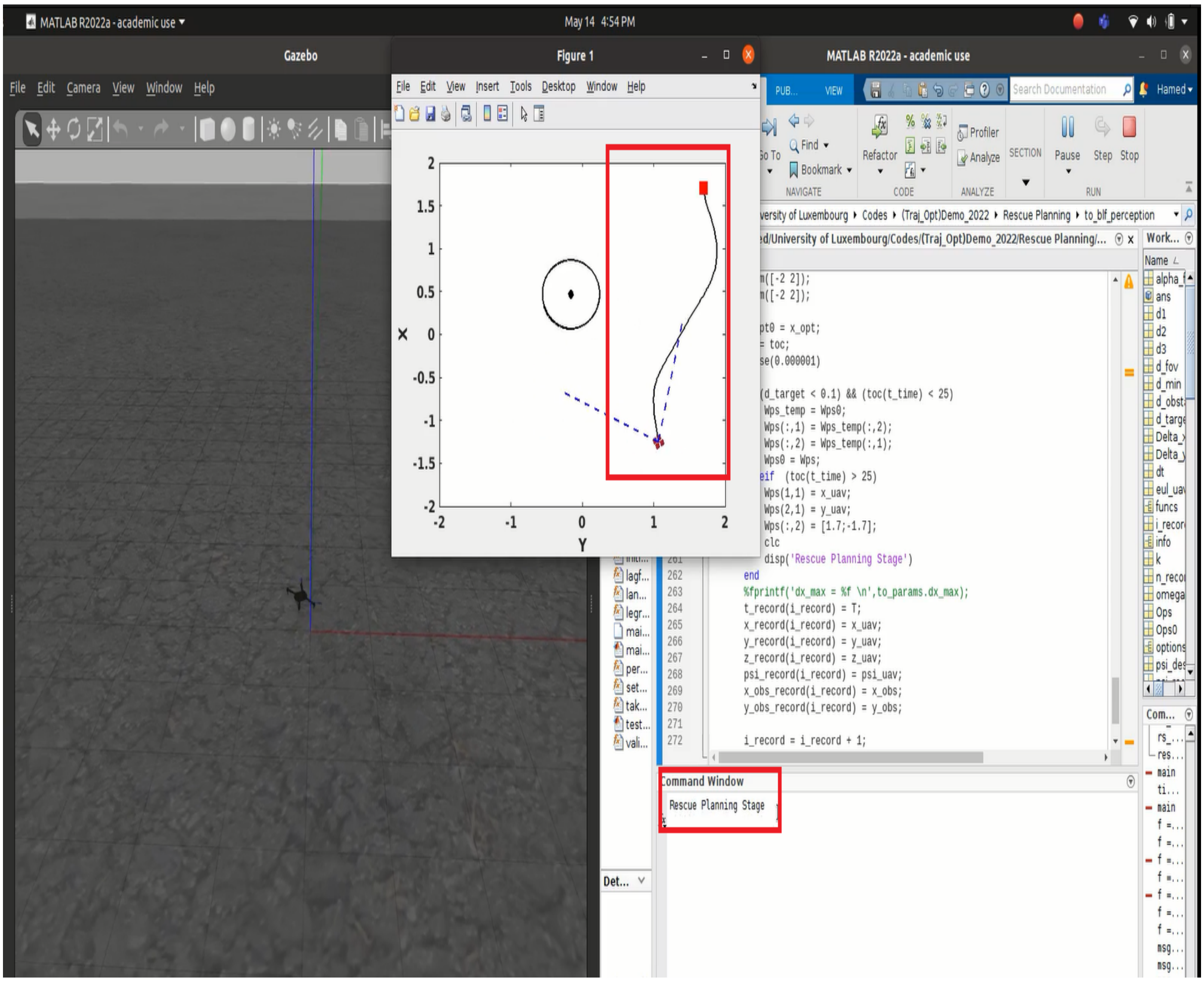

Figure 14: Rescue flag activated and final point has changed.

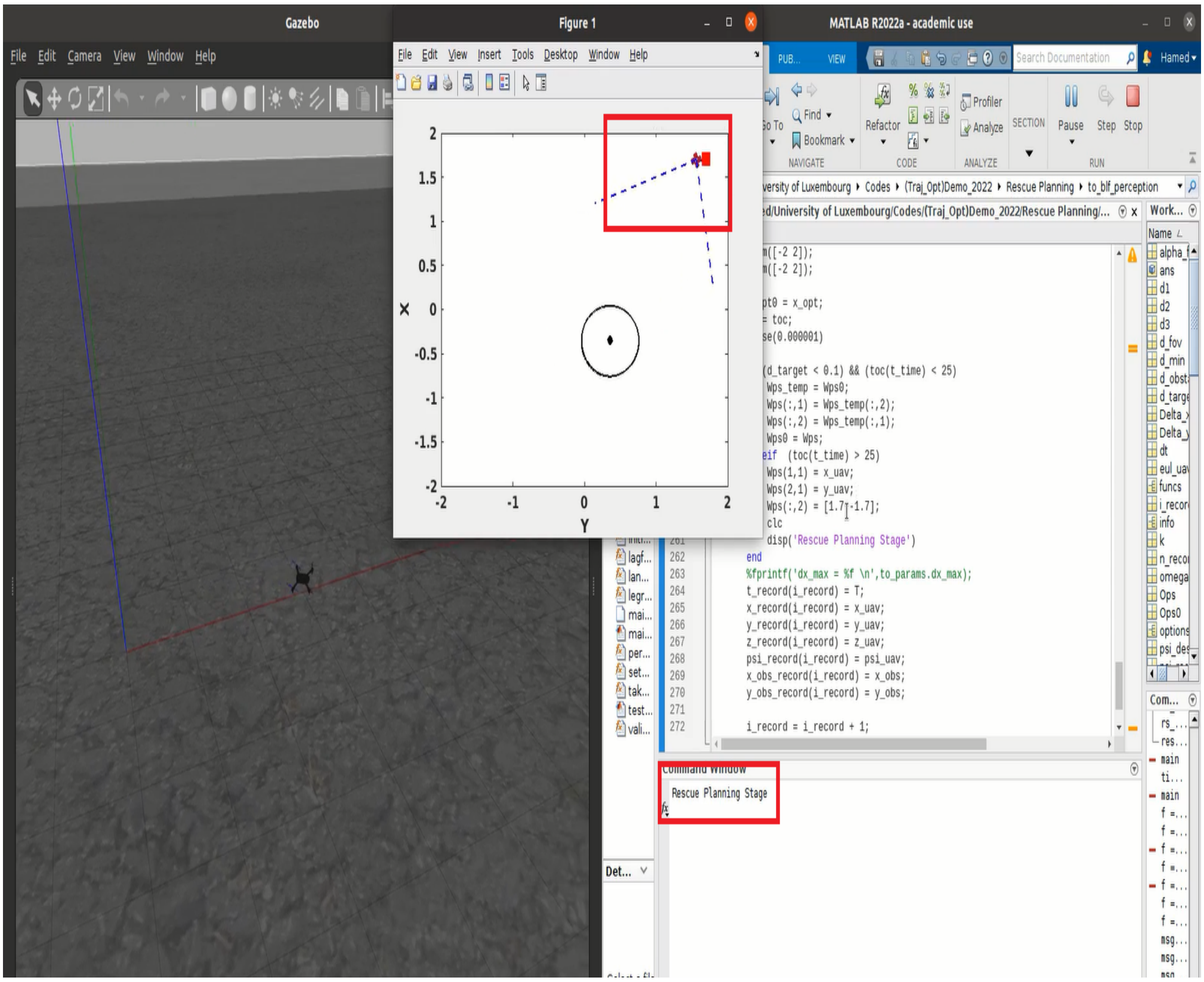

Figure 15: Navigation of target drone to the rescue position and holding there.

5 Future Work

In the next steps of this research, aligning with the SESAME project objectives, we conduct the following steps to accomplish the aims for autonomous safe navigation of MRS.

- Integration of all components developed in WP2.
- Integration of actual malfunction detection unit with the rescue planning unit.
- Design of reliable switching and redundant mechanisms to use the detector drone for monitoring the target drone only at the time malfunction happens.
- Integration of ASTC with trajectory optimization to have both planning and tracking components be executed at the same time, for rescue actions.
- Experimental evaluation of the integrated solution, with potential application on the use cases.

6 Potential Applicability for Use Cases

In this section, the potential applicability of the proposed approaches for different use cases is briefly motivated.

- Use Case 2: Disinfecting Hospital Environments using Robotic Teams: The ground mobile robot can take advantage of the proposed planners, given the implemented constraints in the algorithm. The constraints might represent the area the robot is to visit or avoid. More importantly, the presented results can be directly obtained for the ground robot, as we have fixed the altitude of the drone which can represent a planar motion similar to the ground robot. Also, the fastest trajectory can lead to fast disinfection to avoid intervention with other personnel in the hospital environment. More importantly, whenever, the localization of one robot is compromised the other robot, equipped with a camera, can accomplish the rescue operation and navigate the compromised robot to the base station.
- Use Case 3: Power Station Inspection using Autonomous Multi-Robot Systems: Here, the obtained results can be directly used for this use case, as we have evaluated the proposed algorithm on drones. Furthermore, the initial and final inspection points can be fed into the planning approach to find a safe trajectory to be followed for autonomous inspection. The rescue operation can be directly applied to this use case, for safe inspection operation.
- Use Case 4: Autonomous Pest Management in Viticulture: The monitoring drone is to fly over the farm to detect the plants which need to be sprayed or the locations of obstacles. Therefore, the proposed rescue algorithm 2 can have a great impact on the safe operation of the multi-drones.

7 Conclusions

This document outlined the components for a collaborative rescue mission for MRS. For the planning part, we used an online trajectory optimization approach to compute the fastest trajectory, given the initial and final positions. The collaborative sensor fusion component described a perception, which forms the essential elements of drone detection, position estimation, and semantic segmentation. The second part detailed sensor fusion, which is central to collaborative sensor fusion. Using these components, we presented a generic architecture for the rescue operation of the target drone with the sensor malfunction. All the corresponding rescue actions were considered. Furthermore, we discussed the required elements in the architecture, such as the switching mechanism and redundant components. Furthermore, we provided the algorithm and instructions to implement this architecture. The efficiency of the proposed architecture was investigated and evaluated by high-fidelity simulations in Gazebo.

References

[1] S. Mehmood, S. Ahmed, A. S. Kristensen, and D. Ahsan, "Multi criteria decision analysis (mcda) of unmanned aerial vehicles (uavs) as a part of standard response to emergencies," in *4th International Conference on Green Computing and Engineering Technologies*, p. 31, Gyancity International Publishers, 2018.

[2] B. L. Luk, D. S. Cooke, S. Galt, A. A. Collie, and S. Chen, "Intelligent legged climbing service robot for remote maintenance applications in hazardous environments," *Robotics and Autonomous Systems*, vol. 53, no. 2, pp. 142–152, 2005.

[3] S. Brenner, S. Gelfert, and H. Rust, "New approach in 3d mapping and localization for search and rescue missions," *CERC2017*, p. 105, 2017.

[4] J. P. Queralta, J. Taipalmaa, B. C. Pullinen, V. K. Sarker, T. N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, and T. Westerlund, "Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision," *Ieee Access*, vol. 8, pp. 191617–191643, 2020.

[5] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2624–2661, 2016.

[6] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges," *Ieee Access*, vol. 7, pp. 48572–48634, 2019.

[7] S. Grogan, R. Pellerin, and M. Gamache, "The use of unmanned aerial vehicles and drones in search and rescue operations–a survey," *Proceedings of the PROLOG*, pp. 1–13, 2018.

[8] S. Grayson, "Search & rescue using multi-robot systems," *School of Computer Science and Informatics, University College Dublin*, pp. 1–14, 2014.

[9] A. V. Nazarova and M. Zhai, "The application of multi-agent robotic systems for earthquake rescue," *Robotics: Industry 4.0 Issues & New Intelligent Control Paradigms*, pp. 133–146, 2020.

[10] T. Klamt, D. Rodriguez, L. Baccelliere, X. Chen, D. Chiaradia, T. Cichon, M. Gabardi, P. Guria, K. Holmquist, M. Kamedula, *et al.*, "Flexible disaster response of tomorrow: Final presentation and evaluation of the centauro system," *IEEE robotics & automation magazine*, vol. 26, no. 4, pp. 59–72, 2019.

[11] G. D. Cubber, D. Doroftei, K. Rudin, K. Berns, D. Serrano, J. Sanchez, S. Govindaraj, J. Bedkowski, and R. Roda, "Search and rescue robotics-from theory to practice," 2017.

[12] A. Ollero, S. Lacroix, L. Merino, J. Gancet, J. Wiklund, V. Remuß, I. V. Perez, L. G. Gutiérrez, D. X. Viegas, M. A. G. Benitez, *et al.*, "Multiple eyes in the skies: architecture and perception issues in the comets unmanned air vehicles project," *IEEE robotics & automation magazine*, vol. 12, no. 2, pp. 46–57, 2005.

[13] M. Kulich, J. Kout, L. Preucil, R. Mazl, J. Chudoba, J. Saarinen, J. Suomela, A. Halme, F. Driewer, H. Baier, *et al.*, "Pelote-a heterogeneous telematic system for cooperative search and rescue missions," in *The IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai 2004*, p. 8, 2004.

[14] G.-J. M. Kruijff, I. Kruijff-Korbayová, S. Keshavdas, B. Larochelle, M. Janíček, F. Colas, M. Liu, F. Pomerleau, R. Siegwart, M. A. Neerincx, *et al.*, "Designing, developing, and deploying systems to support human–robot teams in disaster response," *Advanced Robotics*, vol. 28, no. 23, pp. 1547–1570, 2014.

[15] A. Gawel, C. Del Don, R. Siegwart, J. Nieto, and C. Cadena, "X-view: Graph-based semantic multi-view localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1687–1694, 2018.

[16] P. Fritsche, S. Kueppers, G. Briese, and B. Wagner, "Radar and lidar sensorfusion in low visibility environments.," in *ICINCO (2)*, pp. 30–36, 2016.

[17] J. Gancet, G. Hattenberger, R. Alami, and S. Lacroix, "Task planning and control for a multi-uav system: architecture and algorithms," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1017–1022, IEEE, 2005.

[18] J. De Greeff, T. Mioch, W. Van Vught, K. Hindriks, M. A. Neerincx, and I. Kruijff-Korbayová, "Persistent robot-assisted disaster response," in *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 99–100, 2018.

[19] H. Surmann, R. Worst, T. Buschmann, A. Leinweber, A. Schmitz, G. Senkowski, and N. Goddemeier, "Integration of uavs in urban search and rescue missions," in *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 203–209, IEEE, 2019.

[20] J. Güldenring, L. Koring, P. Gorczak, and C. Wietfeld, "Heterogeneous multilink aggregation for reliable uav communication in maritime search and rescue missions," in *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 215–220, IEEE, 2019.

[21] G. Wei, J. W. Gardner, M. Cole, and Y. Xing, "Multi-sensor module for a mobile robot operating in harsh environments," in *2016 IEEE SENSORS*, pp. 1–3, IEEE, 2016.

[22] G. De Cubber, D. Doroftei, R. Roda, E. Silva, S. Ourevitch, A. Matos, K. Berns, D. Serrano, J. Sanchez, J. Bedkowski, *et al.*, "Chapter introduction to the use of robotic tools for search and rescue," 2017.

[23] A. F. Winfield, M. P. Franco, B. Brueggemann, A. Castro, M. C. Limon, G. Ferri, F. Ferreira, X. Liu, Y. Petillot, J. Roning, *et al.*, "eurathlon 2015: A multi-domain multi-robot grand challenge for search and rescue robots," in *Towards Autonomous Robotic Systems: 17th Annual Conference, TAROS 2016, Sheffield, UK, June 26–July 1, 2016, Proceedings 17*, pp. 351–363, Springer, 2016.

[24] H. Kitano, S. Tadokoro, I. Noda, H. Matsubara, T. Takahashi, A. Shinjou, and S. Shimada, "Robocup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research," in *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, vol. 6, pp. 739–743, IEEE, 1999.

[25] K. Junge, R. Slaney, J. Jiang, P. Woo, B. Zhang, Z. Lambert, and J. Suchánek, "Robocup rescue 2020 team description paper cur rescue," *Univ. Cambridge, Cambridge, UK, Tech. Rep*, 2019.

[26] M. Spenko, S. Buerger, and K. Iagnemma, *The DARPA robotics challenge finals: humanoid robots to the rescue*, vol. 121. Springer, 2018.

[27] S. Kohlbrecher, A. Romay, A. Stumpf, A. Gupta, O. Von Stryk, F. Bacim, D. A. Bowman, A. Goins, R. Balasubramanian, and D. C. Conner, "Human-robot teaming for rescue missions: Team vigir's approach to the 2013 darpa robotics challenge trials," *Journal of Field Robotics*, vol. 32, no. 3, pp. 352–377, 2015.

[28] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanskỳ, D. Heřt, M. Petrlík, T. Báča, V. Spurnỳ, *et al.*, "Darpa subterranean challenge: Multi-robotic exploration of underground environments," in *Modelling and Simulation for Autonomous Systems: 6th International Conference, MESAS 2019, Palermo, Italy, October 29–31, 2019, Revised Selected Papers 6*, pp. 274–290, Springer, 2020.

[29] V. S. Saska, T. Báca, M. Petrlık, T. Krajnık, and T. Roucek, "'darpa subt stix qualification submission: Ctu-cras," *Dept. Cybern., CTU-CRAS, Czech Tech. Univ. Prague, Prague, Czechia, Tech. Rep*, 2019.

[30] M. Petrlík, T. Báča, D. Heřt, M. Vrba, T. Krajník, and M. Saska, "A robust uav system for operations in a constrained environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2169–2176, 2020.

[31] Y.-W. Huang, C.-L. Lu, K.-L. Chen, P.-S. Ser, J.-T. Huang, Y.-C. Shen, P.-W. Chen, P.-K. Chang, S.-C. Lee, and H.-C. Wang, "Duckiefloat: a collision-tolerant resource-constrained blimp for long-term autonomy in subterranean environments," *arXiv preprint arXiv:1910.14275*, 2019.

[32] A. Koval, C. Kanellakis, E. Vidmark, J. Haluska, and G. Nikolakopoulos, "A subterranean virtual cave world for gazebo based on the darpa subt challenge," *arXiv preprint arXiv:2004.08452*, 2020.

[33] I. Jawhar, N. Mohamed, J. Wu, and J. Al-Jaroodi, "Networking of multi-robot systems: architectures and requirements," *Journal of Sensor and Actuator Networks*, vol. 7, no. 4, p. 52, 2018.

[34] S. S. Anjum, R. M. Noor, and M. H. Anisi, "Review on manet based communication for search and rescue operations," *Wireless personal communications*, vol. 94, no. 1, pp. 31–52, 2017.

[35] A. W. Stroupe, M. C. Martin, and T. Balch, "Distributed sensor fusion for object position estimation by multi-robot systems," in *Proceedings 2001 ICRA. IEEE international conference on robotics and automation (Cat. No. 01CH37164)*, vol. 2, pp. 1092–1098, IEEE, 2001.

[36] K. Støy *et al.*, "Using situated communication in distributed autonomous mobile robotics.," in *SCAI*, vol. 1, pp. 44–52, Citeseer, 2001.

[37] L. Kanaris, A. Kokkinis, A. Liotta, and S. Stavrou, "Fusing bluetooth beacon data with wi-fi radiomaps for improved indoor localization," *Sensors*, vol. 17, no. 4, p. 812, 2017.

[38] J. Wisanmongkol, L. Klinkusoom, T. Sanpechuda, L.-o. Kovavisaruch, and K. Kaemarungsi, "Multipath mitigation for rssi-based bluetooth low energy localization," in *2019 19th International Symposium on Communications and Information Technologies (ISCIT)*, pp. 47–51, IEEE, 2019.

[39] W. Sun, M. Xue, H. Yu, H. Tang, and A. Lin, "Augmentation of fingerprints for indoor wifi localization based on gaussian process regression," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10896–10905, 2018.

[40] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pp. 269–282, 2015.

[41] N. B. Suryavanshi, K. Viswavardhan Reddy, and V. R. Chandrika, "Direction finding capability in bluetooth 5.1 standard," in *Ubiquitous Communications and Network Computing: Second EAI International Conference, Bangalore, India, February 8–10, 2019, Proceedings 2*, pp. 53–65, Springer, 2019.

[42] W. Shule, C. M. Almansa, J. P. Queralta, Z. Zou, and T. Westerlund, "Uwb-based localization for multi-uav systems and collaborative heterogeneous multi-robot systems," *Procedia Computer Science*, vol. 175, pp. 357–364, 2020.

[43] J. P. Queralta, C. M. Almansa, F. Schiano, D. Floreano, and T. Westerlund, "Uwb-based system for uav localization in gnss-denied environments: Characterization and dataset," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4521–4528, IEEE, 2020.

[44] C. M. Almansa, W. Shule, J. P. Queralta, and T. Westerlund, "Autocalibration of a mobile uwb localization system for ad-hoc multi-robot deployments in gnss-denied environments," *arXiv preprint arXiv:2004.06762*, 2020.

[45] A. Khan, S. Wang, and Z. Zhu, "Angle-of-arrival estimation using an adaptive machine learning framework," *IEEE Communications Letters*, vol. 23, no. 2, pp. 294–297, 2018.

[46] D. Tian and N. D. Georganas, "Connectivity maintenance and coverage preservation in wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 6, pp. 744–761, 2005.

[47] L. Sabattini, N. Chopra, and C. Secchi, "Decentralized connectivity maintenance for cooperative control of mobile robotic systems," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1411–1423, 2013.

[48] L. Sabattini, C. Secchi, N. Chopra, and A. Gasparri, "Distributed control of multirobot systems with global connectivity maintenance," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1326–1332, 2013.

[49] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 2659–2664, IEEE, 2016.

[50] H. Xiao, R. Cui, and D. Xu, "Cooperative multi-agent search using bayesian approach with connectivity maintenance," *Assembly Automation*, vol. 40, no. 1, pp. 76–84, 2020.

[51] Q. Zhu, R. Zhou, and J. Zhang, "Connectivity maintenance based on multiple relay uavs selection scheme in cooperative surveillance," *Applied Sciences*, vol. 7, no. 1, p. 8, 2016.

[52] C. Ghedini, C. H. Ribeiro, and L. Sabattini, "A decentralized control strategy for resilient connectivity maintenance in multi-robot systems subject to failures," in *Distributed Autonomous Robotic Systems: The 13th International Symposium*, pp. 89–102, Springer, 2018.

[53] L. Siligardi, J. Panerati, M. Kaufmann, M. Minelli, C. Ghedini, G. Beltrame, and L. Sabattini, "Robust area coverage with connectivity maintenance," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 2202–2208, IEEE, 2019.

[54] K. Khateri, M. Pourgholi, M. Montazeri, and L. Sabattini, "A comparison between decentralized local and global methods for connectivity maintenance of multi-robot networks," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 633–640, 2019.

[55] F. Amigoni, J. Banfi, N. Basilico, I. Rekleitis, and A. Quattrini Li, "Online update of communication maps for exploring multirobot systems under connectivity constraints," in *Distributed Autonomous Robotic Systems: The 14th International Symposium*, pp. 513–526, Springer, 2019.

[56] J. Faigl, O. Simonin, and F. Charpillet, "Comparison of task-allocation algorithms in frontier-based multi-robot exploration," in *Multi-Agent Systems: 12th European Conference, EUMAS 2014, Prague, Czech Republic, December 18-19, 2014, Revised Selected Papers 12*, pp. 101–110, Springer, 2015.

[57] T. G. Tadewos, L. Shamgah, and A. Karimoddini, "On-the-fly decentralized tasking of autonomous vehicles," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 2770–2775, IEEE, 2019.

[58] J. Tang, K. Zhu, H. Guo, C. Gong, C. Liao, and S. Zhang, "Using auction-based task allocation scheme for simulation optimization of search and rescue in disaster relief," *Simulation Modelling Practice and Theory*, vol. 82, pp. 132–146, 2018.

[59] Y. Liu, M. Ficocelli, and G. Nejat, "A supervisory control method for multi-robot task allocation in urban search and rescue," in *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–6, IEEE, 2015.

[60] J. Suarez and R. Murphy, "A survey of animal foraging for directed, persistent search by rescue robotics," in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp. 314–320, IEEE, 2011.

[61] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, 2020.

[62] J. Binney, A. Krause, and G. S. Sukhatme, "Informative path planning for an autonomous underwater vehicle," in *2010 IEEE International Conference on Robotics and Automation*, pp. 4791–4796, IEEE, 2010.

[63] Y. Wei and R. Zheng, "Informative path planning for mobile sensing with reinforcement learning," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pp. 864–873, IEEE, 2020.

[64] R. Damoto, W. Cheng, and S. Hirose, "Holonomic omnidirectional vehicle with new omni-wheel mechanism," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 1, pp. 773–778, IEEE, 2001.

[65] K. Muecke and B. Powell, "A distributed, heterogeneous, target-optimized operating system for a multi-robot search and rescue application," in *Modern Approaches in Applied Intelligence: 24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2011, Syracuse, NY, USA, June 28–July 1, 2011, Proceedings, Part II 24*, pp. 266–275, Springer, 2011.

[66] C. Sampedro, A. Rodriguez-Ramos, H. Bavle, A. Carrio, P. de la Puente, and P. Campoy, "A fully-autonomous aerial robot for search and rescue applications in indoor environments using learning-based techniques," *Journal of Intelligent & Robotic Systems*, vol. 95, pp. 601–627, 2019.