



HOKKAIDO UNIVERSITY

Title	VTOL-UAV Docking System for Heterogeneous Multi-Robot Team
Author(s)	Narváez Fuertes, Eduardo
Degree Grantor	北海道大学
Degree Name	博士(工学)
Dissertation Number	甲第14436号
Issue Date	2021-03-25
DOI	https://doi.org/10.14943/doctoral.k14436
Doc URL	https://hdl.handle.net/2115/81378
Type	doctoral thesis
File Information	Eduardo_Narvaez_Fuertes.pdf



VTOL-UAV Docking System for Heterogeneous Multi-Robot Team



Eduardo Narváez Fuertes

Graduate School of Engineering
Hokkaido University

This dissertation is submitted for the degree of
Doctor of Philosophy

February 2021

To my lovely wife . . .

Acknowledgements

This thesis work has been possible thanks to the assistance of many people. From the first stages of my career, the help and support of several persons have contributed to the learning process to culminate this stage.

I would like to express my sincere gratitude to Professor Yukinori Kobayashi, who kindly received me in his laboratory and helped me fulfill my dream of beginning and finishing the doctoral course. His caring and wisdom have taught me significantly to be a better professional and person.

I would especially like to thank Associate Professor Takanori Emaru, who has always been ready to give his assistance and make available all the means to complete my research.

I would like to thank Assistant Professor Ankit Ravankar for the several discussions and his help with the experiments and document preparation; this work would not have been possible without his continuous support even after hours. In the same way, I would like to thank Assistant Professor Abhijeet Ravankar, now in Kitami Institute of Technology, for his valuable academic advice and kind help, especially at the beginning of my doctoral course.

I would like to thank the Ministry of Education, Culture, Sports, Science, and Technology (MEXT), Japan, which provided me with the scholarship during my doctoral course.

I would like to express my heartfelt gratitude to my mother for her constant loving and protection and whose words I bring with me everywhere. To my father for teaching me the value of hard work. My sister, for making me believe in myself and inspire me to pursue great things, and my little brother, who is the reason why I always want to be better. Words are not enough to thank every member of my family for their loving words and sincere wishes that I carry with me always.

Lastly, I would like to thank the person who accompanies me on every adventure, my lovely wife, Marcela Calpa. For your committed assistance during the experimental part of this thesis and motivating me constantly. Absolutely nothing would have been possible without your hand guiding me; you always encouraged me to be fair and aim for a bigger dream. You are my best friend and my everlasting love.

Abstract

In the current trend for the field of robotic exploration, the heterogeneous robot teams have been acquiring great relevance because they offer significant advantages for exploration and navigation missions in unknown environments. Notably, the robot teams with aerial and ground platforms have superior efficiency due to their mixed abilities and shared sensor resources. Within this classification, the robot teams that use vertical take-off and landing (VTOL) aerial platforms and robust ground robots have significant benefits because they can see the same environment from different perspectives, extracting valuable information and adding great redundancy to the measurement process. This type of robot team is ideal for applications that require great precision and efficiency and to explore unknown and dangerous environments. However, the VTOL unmanned aerial vehicles (VTOL-UAV) have limitations due to their high energy consumption and reduced payload capacity, restricting the exploration abilities of the complete robot team. When the VTOL and ground robot platforms are deployed together as a team, at some point in the mission, the aircraft needs to return to the base for landing and battery replenishment to continue with the exploration. This task restricts the team to explore the environment in-depth and to work for longer durations. As a solution to this problem, it is presented in this work an autonomous system for the docking of a VTOL-UAV with the mobile manipulator of the robot team.

The proposed system contains a central measurement-actuation unit that consists of a robot manipulator mounted in a mobile platform with a visual sensor configured as an eye-in-hand device. The extracted visual information is used to calculate the necessary arm movements to execute a stable UAV tracking to achieve air-ground robot contact. It is presented with a complete control scheme that merges predictive and optimization approaches to calculate precise commands for both robot platforms. The controller workspace is designed to include environmental and structural constraints for safe air-ground contact. The docking strategy allows the multi-robot team to economize and potentially recover aircraft energy and also enables the robust ground platform to protect and store the aircraft facilitating to continue the mission for a longer duration. The on-site autonomous docking can be initiated

multiple times, allowing the team to explore larger areas and complete complex and long-term missions. The main focus of this thesis is to create a robust system capable of performing a safe and repeatable docking task within complex environments, enhancing the robot team and its usability.

The proposed method is presented with the corresponding theoretical development and complete hardware and control design. The validity of the system is tested with simulation and experiments in a real environment. The collected data for simulation and the real environment is represented in graphical form, and the results are presented with the proper discussion. In the end, some concluding remarks are presented along with recommendations for future research.

Table of contents

List of figures	ix
List of tables	xii
Nomenclature	xiii
1 Introduction	1
1.1 Research Background	1
1.2 Related Work	3
1.3 Main Contribution	7
1.3.1 Published work	8
1.4 Thesis Outline	10
2 Theory and Background	11
2.1 Robot Motion and Control	12
2.1.1 VTOL-UAV Motion	12
2.1.2 UGV Motion	15
2.1.3 Robot Arm Motion	17
2.1.3.1 Forward Kinematics	18
2.1.3.2 Inverse Kinematics	19
2.2 Mobile Manipulation	20
2.3 Robot Pose Estimation	21
2.3.1 Global Pose Estimation	22
2.3.2 Relative Pose Estimation	24
2.4 Summary	27

3	VTOL-UAV Docking System	28
3.1	USAR Missions with Heterogeneous Robot Teams	28
3.2	Air-Ground Robot Team	31
3.2.1	VTOL-UAV	31
3.2.2	Wheeled Mobile Manipulator	33
3.3	VTOL-UAV Autonomous Docking	34
3.3.1	UAV Motion Tracking	36
3.3.1.1	Switching Regression	40
3.3.1.2	Corded Local Path Planning	43
3.3.1.3	Selection and Optimization	49
3.3.2	Contact	53
3.3.3	Safety Functions	54
3.3.4	Docking Device	55
3.4	Summary	58
4	Simulation and Experiments	59
4.1	Simulation	60
4.1.1	First Simulation Approach	61
4.1.2	Results of the First Simulation Approach	63
4.1.3	Simulation of a Complete USAR Mission	64
4.1.4	Results of the Simulation of a Complete USAR Mission	68
4.2	Experiments in a Real Environment	71
4.2.1	Robotic platforms	71
4.2.2	Architecture and Network	74
4.2.3	Experiment for a single VTOL-UAV Docking	75
4.2.4	Controller Parameters	77
4.2.5	Experiment Design for a Complete USAR Mission	78
4.2.6	Results for Experiment in Real Environment	81
4.3	Summary	87
5	Discussion, Conclusion and Future Work	89
5.1	Discussion	89
5.2	General conclusion	91
5.3	Future Work	92

References	94
Appendix A Tools and Supporting Figures	104
A.1 UAV Deployment Maneuver	106
Appendix B Technical Drawings	108
B.0.1 Robot Arm	108
B.0.2 Docking Device	111

List of figures

2.1	Quad-rotor scheme and propellers positioning.	13
2.2	Main axes of the quad-copter.	13
2.3	Possible motion when the speed of the propellers are modified. The increment in speed is represented with a thick curved arrow, and the speed reduction with a thin arrow.	14
2.4	Possible movements for the differential drive with two actuated wheels and a caster wheel.	16
2.5	Coordinate system for the robot arm and location of main Denavit–Hartenberg parameters.	17
2.6	Mobile manipulator and its coordinate system.	21
2.7	Representation of the global map of a scene.	23
2.8	Relative coordinate system of the mobile manipulator.	25
3.1	Priority levels for tasks execution in the air-ground robot team.	30
3.2	Air-ground robot team.	32
3.3	General flow for the VTOL-UAV docking operation.	35
3.4	General control scheme.	37
3.5	Schematic of <i>Motion Engine</i> process.	39
3.6	Types of cords for the regression that guides the arm movements.	41
3.7	Vicinities with their respective <i>cords</i>	44
3.8	Generated environment map with obstacles and prohibited areas.	46
3.9	Vicinities and <i>cords</i> with obstacle shift.	48
3.10	Representation of the path calculation for the advance of the end effector position.	51
3.11	Main components of the docking device.	56

3.12	Representation of the UAV docking sequence with a close-up view of the docking device.	57
4.1	Robot paths for the exploratory simulation with reactive approach.	62
4.2	Robot paths in 2 dimensions for the exploratory simulation.	63
4.3	Position variations and altitude profiles for UAV and end effector for the first simulation with reactive approach.	65
4.4	Building screenshot and main operating console for the simulation of the experiment in Gazebo simulator and ROS.	66
4.5	Map and robot paths for simulation experiment.	69
4.6	Power used by the UAV motors during the simulation experiment.	70
4.7	UAV battery levels during the simulation.	71
4.8	End effector of the robot arm.	72
4.9	Bottom view of the UAV.	73
4.10	Communication network between robot team and operator.	74
4.11	Docking sequence in controlled environment from initial robot encounter to final storage; the UGV is in motion with low speed.	76
4.12	Photograph of the starting point (Remote base) for the experiment in real environment.	78
4.13	Screenshots of the available camera streamings for remote operation of the robot team.	79
4.14	Robot team paths for real robot experiment. The yellow area represents the initial operator knowledge about the location of the <i>target</i>	82
4.15	Positions of the UAV and end effector during the first docking process in real environment.	84
4.16	Positions of the UAV and end effector during the second docking process in real environment.	85
4.17	UAV battery level during the experiment.	86
A.1	Marker bundle.	105
A.2	Sequence of photographs for the deployment maneuver.	107
B.1	Detailed technical drawing of the robot manipulator with the Part A of the docking device.	109

B.2	Detailed technical drawing of the robot manipulator while fully extended. .	110
B.3	Detailed technical drawing of the docking device.	112
B.4	Detailed technical drawing of the Part A of the docking device with cross section.	113

List of tables

4.1	Video URLs for simulation and real experiments	60
4.2	Time information for simulation	68
4.3	Simulation Data	68
4.4	Controller Parameters	77
4.5	Time information for experiments in real environment	83
4.6	Experiment data	83

Nomenclature

Greek Symbols

τ Torque

Superscripts

$\mathcal{CA}$ Camera coordinates

$\mathcal{M}$ Manipulator coordinates

Subscripts

A Relative to the Part A of the docking adapter

B Relative to the Part B of the docking adapter

C Camera relative

Q Quad-rotor relative

VS Online Position-Based Visual Servoing

Other Symbols

$\mathcal{E}$ Edges of the planning graph

$\mathcal{F}$ Forward Kinematics equations

$\mathcal{G}$ Hierarchical graph for path planning

I Cord item

L_C Longitude of the cord, extension of the predicted movements

L_S	Separation between cord items
N	Number of items in a cord
n	Order of the regression for prediction of movements
O_C	Offset of the position of the camera with respect to the center of the end effector
OP	List of P_{VS} values
P	Pose, comprises the 3 Cartesian positions x , y and z and the 3 angular positions <i>yaw</i> , <i>roll</i> and <i>pitch</i>
S	Section of the vicinity
$\mathcal{V}$	Vertices of the planning graph

Acronyms / Abbreviations

DoF	Degree of Freedom
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
USAR	Urban Search and Rescue
VTOL	Vertical Take-off and Landing

Chapter 1

Introduction

1.1 Research Background

The robotic systems have gained significant improvements in recent years. Since the rapid evolution of the processors, hardware acceleration, graphical processing units, and their integration with machines and electro-mechanical components, the robots have become more capable of resolving the issues of several fields and applications. From heavy industry, military, education, and daily life, the robots are indispensable for the continuous development of society, and as such, they are in constant change and evolution.

As a consequence of recent technology improvement, and mostly because of advancements in artificial intelligence, machine learning, and sensor development, the cooperative schemes in robotic systems have gained significant interest because of their potential in solving complex problems. Naturally, sets of robots with similar or different features can be integrated into teams for complex problem solving, introducing new possibilities for the robotic systems.

Recently, the interest in heterogeneous multi-robot systems for precise monitoring tasks, Urban Search and Rescue (USAR) missions, industry applications, and space exploration has been increasing [1]. Features like fault tolerance, flexibility, and robust integration, make the robot teams truly attractive to solve distributed data collection and remote sensing missions [2]. By choosing a multi-agent approach for the design, redundancy is automatically included as a central component of the system. Moreover, with cooperative behavior and parallelism, the sensing efficiency is drastically improved. In general, the heterogeneous team of robot platforms has a higher probability of finding optimal solutions for the assigned task in a reliable, faster, and more efficient way than single robot systems [3].

The tasks for the deployed robot teams consist of detailed exploration, precise navigation, and interaction within unknown environments, which in most cases, are not solvable by a single agent [4, 5]. For monitoring unknown regions with diverse structures and obstacles, it is convenient to use heterogeneous teams that dynamically combine individual robot capabilities [6].

In this context, the heterogeneous robot teams that use aerial and ground platforms have a significant advantage as they can sense the environment from different perspectives offering an increment in redundancy. By using a distributed configuration, the sharing of resources potentially leads to an increase in general performance.

Specifically, to solve the multiple challenges of exploring unknown environments, Unmanned Aerial Vehicles (UAV) with Vertical Take-off and Landing (VTOL) capability are very suitable. They are great measurement platforms, as they provide a high point of view, hovering ability, and outstanding versatility in their trajectories. On the other hand, the Unmanned Ground Vehicles (UGV) provide a low point of view and robust platforms that can carry a large payload and can be equipped with different sensor suits, actuators, and energy storage cells.

Such capabilities are symbiotic to each other so that the heterogeneous team can share information and process power through a wireless channel. The information extracted from the UAV sensors can be merged with the information from the UGV sensors to generate a complete database that can be used to build a precision map of the environment for navigation and other purposes [7].

A big concern for using the VTOL-UAV is power efficiency, limited payload capacity, and weak impact protection. These factors can affect team behavior, and overall mission success [8]. For this reason, it is possible to rely on the UGV robustness to counteract the UAV drawbacks as it can navigate safely through critical areas like door frames or areas with low ceilings. It can also transport enough energy storage cells to supply for its tasks and potentially for many UAV overflights [9]. A specific interaction is essential for resource sharing, and evidently, physical contact between the robots is required. This contact requirement is the main problem addressed in this work. With a stable connection mechanism, it is possible to protect the UAV and enable the robot team to explore vast environments and perform detailed inspections without needing the UAV to return to the charging station. The problem complexity lies in finding a proper attitude of the robots and an optimal trajectory; while considering the features and limitations of the robot, along with the environmental and dynamic constraints.

1.2 Related Work

After a revision of the current state of the art of the field, it is possible to identify a clear differentiation on the the previous works related to the heterogeneous robot system. Depending on the approach, purpose and methods, the works can be divided into two main classifications.

1. Works focused on the creation of control architectures that seek the cooperation and synchronization of robots for navigation and exploration purposes.
2. Works aimed towards building systems capable of solving more complex and extended missions, including physical interactions between robots in cluttered unknown environments.

The main concern of the former group is to generate a suitable architecture where mixed data from the robot sensors can be merged, thereby creating a set of valuable information for exploration and navigation processes along with scheduled tasks to achieve a goal. For example, work such as [10], presents a bio-inspired pheromone deposition algorithm for a multi robot cooperative scheme that increases the exploration and mapping abilities while economizing computational power. In [11] similar pheromone trails are implemented this time including robot uncertainty to avoid the "blind leading the blind" scenario. In [12], a hitchhiking based approach is adopted where resources between multiple robots are shared based on leader-follower action to save computation cost when both robots are moving to a similar position in the map. In [13] it is described a use case for multi-robot team combining autonomous navigation with limited human operation for disaster scenarios. Work in [14], presents sharing of obstacle information in multi-robot scenario, including the uncertainty in measurements for calculation of improved path planning and navigation. Work in [15] presents the deployment of multiple robots that share sensor data for the observation and precise pose estimation of multiple moving targets.

Another interesting application of a cooperative robot team is the one to be deployed on Mars with the Perseverance rover that carries the Ingenuity autonomous helicopter. The helicopter will be released from the rover's belly and placed on the surface of Mars to take short flights for general exploration [16–18]. These type of missions demonstrate the advantageous of using mixed robot platforms in complex missions within unknown environments.

Furthermore, in cooperative robotics, the map generation task play an important role for the missions as the localization of the team and the data gathering from environment structures and obstacles is a need for collaborative robot navigation. Approaches like [19] proposed a collaborative mapping with multi-robot systems, focusing in the accumulation of enhanced 3D map information from heterogeneous agents. In [20], the mapping is designed with topological graph nodes for the navigation of multi robot systems in large environments. Works in [21–26] propose varied methods to build a 3D terrain map based on a precise mixture of information from UAV and UGV sensors, all of them advancing one step forward by calculating optimal paths to guide robots towards a designated target while considering the energy saving and resource sharing concepts. Similar works are presented in [27–30] particularly for GPS denied environments and using the type of sensor systems for indoor environments. In this last case, the robots make use of the different options for measurements focusing on merging as much sensor information as possible to reach high accuracy in the estimations.

Moreover, in [31], the emphasis is on creating a cooperative control scheme that uses the visual information of an UAV to calculate precisely the relative localization of multiple ground robots. Similarly, [32] uses a vision based approach to synchronize the relative position of a dynamic target, this time with a camera mounted on the tip of a robot arm designed for space environments. Also, a couple of methods for the coordination of a group of UAVs to provide continuous coverage to a moving ground convoy for mission development is presented in [33] and [34]. In [35] it is proposed to use the visual servoing technique to track a UGV using the information of UAV sensors, creating a closed loop within the robot team. Meanwhile, in [36], a switched cooperative control scheme to coordinate groups of ground and aerial robots for localization of a dynamic target is proposed.

The cited works that use a heterogeneous robot teams with mixed sensor capability, make use of a cooperative control scheme for mission solving. Their different sub tasks for exploration, global guidance, and obstacle detection are improved with the use of measurements from different points of view.

In the second group of research works, the primary intention is to provide the heterogeneous robot teams with the ability to interact with the environment and between members. A robot team with this kind of features can build mechanisms to share resources and consolidate a synergistic integration to provide better results in complex and long term missions.

In [37], authors have proposed a strategy for coordination between a refueling UGV and a quad-rotor for exploration of large areas. Although it only presents path calculations, it

improves the UAV-UGV cooperation and routing for road networks. In [38], a measurement system and algorithms were developed for vision based aerial refueling, several simulations were presented with techniques to solve the pose estimation of a tanker and a receiver UAV. A different approach is presented in [39], where the UGV transports the UAV to a set of target points to be deployed for precise exploration. In [40], the UAV is designed to transfer (pick and place) parcels into mobile robots for autonomous transportation in warehouses with impeded ground exploration. [41] presents a cooperative object transportation scheme for ground robots with minimal inter-vehicle communication. All of these works rely on visual information to calculate the relative pose of the robots and plan a safe rendezvous. They also coincide with the need to improve the physical linking process to enable resource sharing between members and to treat temporarily the team as a single entity.

Many of the previous works on air-ground robots are focused on the contact mechanism as a crucial task for aircraft recovery and to increase overall UAV autonomy and capabilities. Designing such contact mechanisms is a challenging task because the team is always exposed to many external variables, and demand the control strategies to be fast and robust enough to handle such variables.

A standard solution that has gained significant importance in the field is the visual assisted landing of VTOL-UAVs in static or mobile landing platforms. Research works in [42–44] developed UAV landing techniques with visual input for the calculation of the error between the UAV and the marker position fixed on the helipad. Moreover, when the landing zone is in motion, the control techniques and advanced strategies, as presented in [45–49], are required to implement a robust tracking and final approximation for a safe landing.

In [50], a robust contact mechanism system is proposed, where the energy consumption of the whole team in reaching the target is analyzed. It is used a UAV-UGV system that behaves with a strong interrelationship, and is capable of negotiate actions for obstacle evasion, with the UAV landing and UGV lifting capabilities within disaster scenarios.

In general, the direct UAV landing in UGV proposals present control laws designed to trust mainly in the under-actuated UAV controllers. This can cause difficulties for complex missions, as the UAV actuators are not in full control of the six Degrees of Freedom (DoF). This situation means a risk when facing aleatory external variables [51]. Also it is required a large landing platform that can cause interference for ground navigation.

To address the UAV landing problems, works such as [52–55] have considered the VTOL-UAV landing on surfaces subjected to 6 DoF variations, e.g., helipads in ships, UGVs in uneven terrain, or earthquake situations. For these cases, the systems are designed to track

cartesian movements and to compensate for the angular deviation with software solutions ([52, 53]) or with actuated platform mechanisms ([54, 55]).

Another option for UAV landing is to give the ability to the UAV of absorb the impact of the approach or grab a fixed structure to stabilize its touchdown. For example in [56] the UAV is provided with perching and resting abilities inspired in the mechanisms used by birds and bats, with a special actuated landing gear; in this proposal, the UAV can reach structures and acquire a resting position. Similarly, works in [57–59] present different perching mechanisms to make the UAVs reach stable structures and continue the work with a redeployment. In [60], the UAVs act as 5G+ access points, and the effects on the network if the UAVs were to be airborne or perched in buildings or fixed structures is analyzed.

There are some other works that explore the use of a grounded robotic arm to assist the UAV landing operation, in [61] a design of a linear state-space controller to handle an active UAV attached to the end effector with a universal hing is presented. This work is focused on solving the unstability of the system after the docking has been achieved and the motors are still active. The same author proposed an improved control system in [62]. Although these works present interesting designs to control the multi-robot system during an important step of the linking process, the first contact and linking is not addressed.

One exciting work summarized in [63] proposes the direct landing in a mobile robot manipulator. This is achieved by installing a landing pad in the end effector of the robot arm of a mobile manipulator, the arm and the whole ground robot moves based on the UAV trajectory and variations. It is proposed an assisted landing with significant redundancy in a moving robot, exploring the benefits of the use of a fully controlled 6DoF robotic arm for this type of interaction.

Although there are important approaches that address this particular touchdown or landing step, there are still limitations that restricts the stability of the task. When a direct landing in flat platforms (fixed or actuated) is used, the UAV is susceptible to fall without a safe locking mechanism in contact state, and the instabilities caused by external disturbances are not fully addressed. When the controller of the UAV is in charge of the complete task, there will be always the risk of loose the stability because of its under-actuated characteristic. It is evident the need of a better solution that facilitates the UAV-UGV contact task and that provides enough robustness and stability to ensures the UAV safety in the process.

As a new improvement of the techniques in the state of the art, it is presented in this thesis a novel VTOL-UAV docking with a mobile manipulator. In this approach, a single control loop manages the UAV and the robot arm position and orientation to implement a

safe strategy for UAV docking. The proposed method was tested on a simulated and real UAV-UGV team in different scenarios with further improvements such as robust tracking and locking mechanism to store the UAV. Also, the control algorithms addressing issues such as the robustness, stability and environmental constraints are presented.

1.3 Main Contribution

This thesis proposes a system that can exploit the hardware and software resources of a heterogeneous robot team. A measurement and actuation platform that allows an autonomous linking process (docking) with the aerial robot is developed. A control strategy is implemented to transform the visual measurements into precise robot movements for the linking task. With this system, the air-ground robot team is capable to make a safe and efficient contact for temporary storage and energy replenishment, and can overcome the limitations and enhance its ability to explore unknown areas for a longer time.

The main focus is in achieving great robustness and safety in the linking process, so that the cooperation between robots becomes natural and the team acquires superior features to complete USAR missions. In this context, it is not possible to rely only on the controller of the VTOL-UAV due to its under-actuated character, as it has only 4 actuators to move in a six DoF space. Then, the use of the ground robot manipulator becomes essential because it adds redundancy to the control scheme while increasing the range of action allowing to do a contact task in a larger workspace, and making it easy to compensate for variations caused by air and ground external disturbances. Furthermore, the use of the manipulator also allows to secure and handle the UAV chassis and store it in an appropriate position that does not obstruct the ground sensors, becoming part of the UGV but without interfering with the ground navigation.

This thesis work presents a completely autonomous system that manages the robot platforms from the UAV-UGV first rendezvous, UAV tracking task, to the final contact process. To support these steps, an algorithm that fuses the reactive, predictive, and optimization approaches is proposed. This algorithm calculates the end effector trajectory for UAV tracking and simultaneously calculates the UAV relative pose control for contact tasks. It incorporates robotic and environmental constraints in the control loop by considering the proper limitations of the robot platforms, e.g., robot structures, sensor placements, and motion properties while negotiating static and dynamic obstructions from the unknown environment.

The work presented in this thesis is the first working principle that provides a different perspective in the research of aerial and ground robots integration and the use of cooperative architectures for complex mission deployment. This perspective can open the door for new exploration strategies of heterogeneous robot teams in unknown environments.

As a result of the research work of this thesis, a conference paper and a journal paper were published. The information of each published work with the date of publication and abstract are summarized as follows.

1.3.1 Published work

The following are the articles that publish the research work of this thesis for easy access of the academic community.

1. Narváez, E., Ravankar, A. A., Ravankar, A., Kobayashi, Y., & Emaru, T. (2017, December). *Vision based autonomous docking of VTOL UAV using a mobile robot manipulator*¹. In 2017 IEEE/SICE International Symposium on System Integration (SII) (pp. 157-163). IEEE. DOI: 10.1109/SII.2017.8279205.

Abstract: This paper proposes a system that allows the docking of a hovering VTOL UAV (Vertical Take-off and Landing Unmanned Aerial Vehicle) by a mobile manipulator during an exploration or navigation mission for use in heterogeneous robotic team. A single camera configured as eye-in-hand device at the end of the mobile manipulator is used to detect and track the position and orientation of a fiducial marker which is fixed at the bottom of the UAV. By employing visual information in a specified workspace, the position of the UAV is controlled in a predefined 3D coordinate system that allows the UAV to hover in a stationary position with respect to the mobile robot base. At the same time, the end effector is programmed to continuously follow the movements of the UAV in order to prevent the loss of sight. The final contact between the UAV and manipulator is made possible with the approach of the end effector in a controlled manner, while maintaining visual contact with the UAV during the docking process. A special magnetic connection is designed that allows coupling of the two parts that are located at the end effector and at the bottom of the UAV, respectively. The effectiveness of the proposed system is tested through simulations and the results are verified.

¹The first published work is fully referenced in [64]

2. Narváez, E., Ravankar, A. A., Ravankar, A., Emaru, T., & Kobayashi, Y. (2020). *Autonomous VTOL-UAV Docking System for Heterogeneous Multi-Robot Team*². IEEE Transactions on Instrumentation and Measurement. DOI:10.1109/TIM.2020.3039649.

Abstract: Heterogeneous robot teams offer significant advantages for exploration and navigation missions in unknown environments. Notably, the air-ground robot teams have superior efficiency due to their mixed abilities and shared sensor resources useful when completing missions. However, the high energy consumption and reduced payload capability of the vertical take-off and landing (VTOL) aerial platforms limit the complete robot team's behavior. At some point in a mission, the aircraft needs to return to the base, land, and replenish its batteries. This task restricts the team to explore the environment in-depth and to work for longer durations. To solve this problem, we present an autonomous system for docking a VTOL-UAV with a mobile manipulator. The central measurement-actuation unit consists of a robot manipulator mounted on a mobile platform with a visual sensor configured as an eye-in-hand device. The visual information is used to execute a stable UAV tracking to achieve air-ground robot contact firmly. This strategy allows the multi-robot team to economize and potentially recover aircraft energy. It also enables the robust ground platform to protect and store the aircraft facilitating to continue the mission for a longer duration. The on-site autonomous docking can be initiated multiple times, allowing the team to explore larger areas and complete long-term missions. The proposed method has been tested in simulations and real environments with detailed experimental results.

²The second published work is fully referenced in [65]

1.4 Thesis Outline

The remainder of this thesis is structured as follows.

- Chapter 2 describes the theoretical background of the technologies used in the proposed system. It details the general features of the robot platforms and their locomotion, along with the framework of operation. It also includes the techniques for robot motion and control and pose estimation that allow the exploration with heterogeneous robot systems.
- Chapter 3 contains a brief description of the missions for heterogeneous robot team exploration, to contextualize the application of the system. It also describes the main theoretical approach to build the main control scheme and the necessary steps to perform a docking process. At the end of the chapter, the design of the docking device that allows the final UAV contact and storage is presented.
- Chapter 4 presents the experimental section of the thesis. It describes the robot team that allows the implementation of the VTOL-UAV docking system, along with the testing environments and mission design. It also summarizes the results of each experiment, and gives a pertinent discussion of the experimental tests.
- Chapter 5 concludes the thesis with the main improvements on the state of the art, and gives discussion and recommendations for possible further improvements on the VTOL-UAV docking system.

Chapter 2

Theory and Background

The VTOL-UAV docking system comprises several robot platforms that use their own coordinate systems and are handled within different domains. They have their own control techniques and use different types of sensor systems, making their integration into a single team a very challenging mission. To integrate them in a team to solve the proposed docking task, background knowledge of the relevant techniques and general concepts are necessary. In this chapter, the technology scenario used in the robot exploration field will be explained in detail. Specifically, the basic ideas used to achieve the precise control of robot platforms, their localization, and their integration in teams for deployment in unknown scenarios.

First, aerial and ground robot motion considerations will be detailed, along with a description of the framework and location coordinates. Once the basics of robot motion are described, the concept of manipulation is pictured as an essential part of interactive robotic exploration. As the main instrument for the docking system, the robot manipulator is described along with its coordinate system and design parameters. Also, the process for calculating arm position based on the joints angles and the inverse estimation are described. As the next step, the combination of the manipulator platform the robot navigation is described to introduce the concept of mobile manipulation and its importance in USAR missions.

Continuing with the theoretical background of this thesis, it is also described the Robot Pose Estimation process achieved with precise calculation techniques that merge sensor measurements with environmental uncertainties. The pose estimation is essential for the robot interaction in cooperative systems and the definition of a global and local approach to creating the appropriate setup in the development of the docking system. For this work, the local pose estimation for mobile manipulation is emphasized. At the end of this chapter, a brief summary of the theory and background contents is presented.

2.1 Robot Motion and Control

It is important in every navigation system to consider the locomotion type and the dynamics that involve the motion of a robot platform. Depending on the type of robot, the control scheme and the coordinate system varies. The corresponding algorithmic and path planning should consider the configuration space, features, and limitations of the robot platform to achieve a stable motion and optimal displacement in the robots.

In this section, the main robot platforms that compose a heterogeneous robot team are described, particularly the robots with an intimate interaction and those who make possible the implementation of the VTOL-UAV docking system.

2.1.1 VTOL-UAV Motion

The Vertical Take-off and Landing Unmanned Aerial Vehicles are robot platforms based initially on the classical helicopter aircraft composed of propellers that create propulsion by moving the air with rotating blades. Of great interest are the UAVs that combines a multi-rotor scheme to produce thrust in different directions [66].

The quad-copter or quad-rotor is a UAV with VTOL capability that combines four rotors around a lightweight cross-like chassis. The four rotors with their propellers are placed in the extreme of each arm, with two of them rotating clockwise and the other two counterclockwise. The arrangement of propellers and chassis give the quad-rotor a radial symmetry, which means that the center of gravity is precisely in the center of the cross. See Fig. 2.1.

The aircraft in flight is free to rotate in three main axes; the related rotations are called yaw, roll, and pitch. For roll and pitch rotations, the axes are aligned to the arms of the aircraft. In the particular case of the quad-copter, they are very similar and can be confused if no reference indicates the nose of the chassis. On the other hand, the yaw axis passes vertically through the center of the aircraft, producing variations in the heading angle when the aircraft is horizontal. The main axes and the direction of rotation are drawn in Fig. 2.2.

The aircraft controller uses the torque difference of the propellers to produce movements in pitch, roll, and yaw axes. When the four propellers have the same angular velocity (Fig. 2.3a), the net torque about the yaw axis is zero, then the aircraft is hovering or changing its altitude.

As shown in Fig. 2.3b, the angular velocity is increased for the two rotors rotating in the same direction, causing variation in the yaw angle. Moreover, if the angular velocity

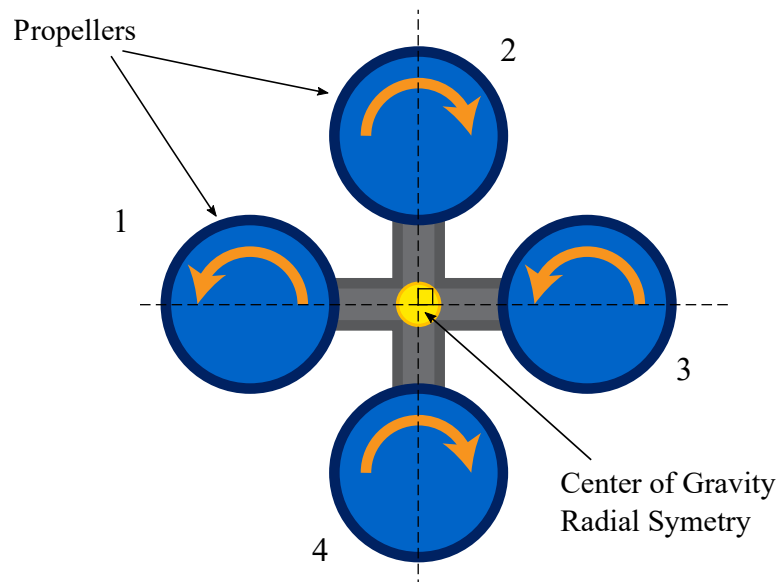


Fig. 2.1 Quad-rotor scheme and propellers positioning.

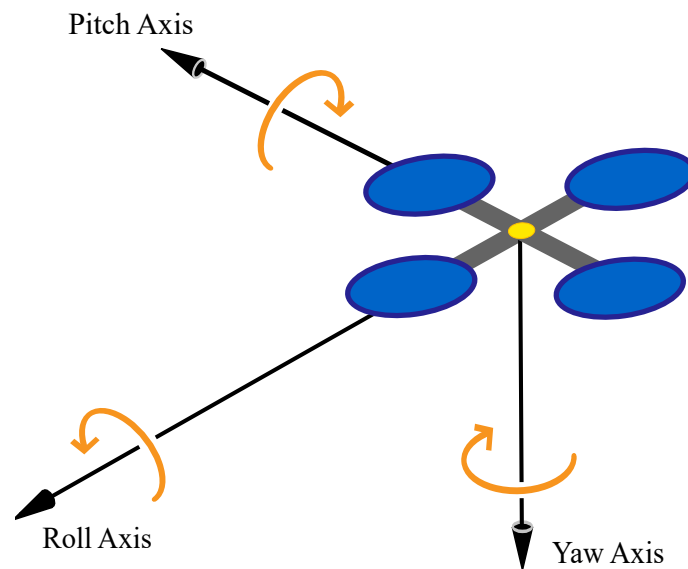


Fig. 2.2 Main axes of the quad-copter.

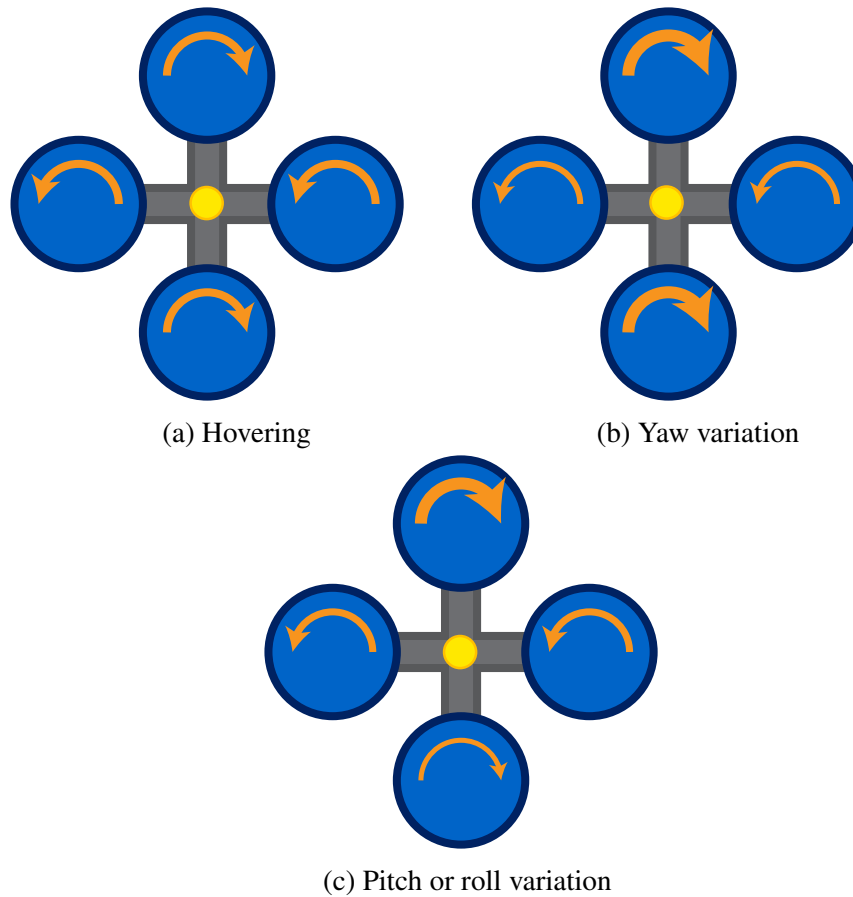


Fig. 2.3 Possible motion when the speed of the propellers are modified. The increment in speed is represented with a thick curved arrow, and the speed reduction with a thin arrow.

is increased for one rotor while decreased for the diametrical opposite rotor, a pitch or roll variation is achieved, see Fig. 2.3c.

The combination of the rotations in the yaw, roll, and pitch axes produces the required movement. This is because the lateral and frontal movement is coupled to the roll and pitch axis, respectively. Furthermore, the yaw rotation can manage the direction of the nose and the displacement. As mentioned in the previous chapter, the quad-rotor is an under-actuated device because it cannot control its yaw and roll orientation without producing displacement, and it only has four actuators to control an aircraft with six possible Degrees of Freedom [67].

2.1.2 UGV Motion

The Unmanned Ground Vehicles are platforms that move in contact with the ground; they are much simpler than the aircraft because the UGV altitude is fixed, and it is assumed as a zero value. Thus, it is only considered movements in a two-dimensional space, which simplifies the approaches used for navigation and mapping.

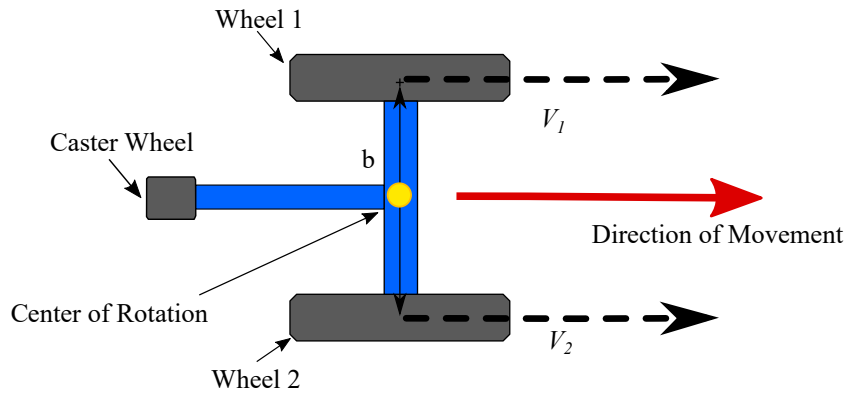
There are many configurations possible for a ground vehicle, but generally, they are wheeled robots with one or more actuated wheels. Among the wheeled type of robots, the most common types of locomotion are:

- Ackermann steering
- Skid steering
- Slip steering
- Differential drive
- Omni-directional
- Rolling body

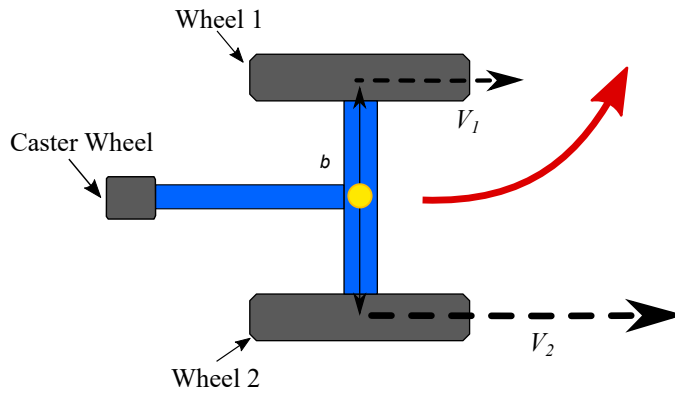
In this thesis, the use of ground robots with the differential drive is emphasized, which consists of a vehicle with a set of independently controlled wheels placed on either side of the robot body.

One of the most common UGV platforms with differential drive are the ones that combine two actuated wheels with a caster wheel. The actuated wheels are placed on each side of the chassis in front of the center of gravity, and the caster wheel is located in the back with the only purpose of weight stabilization.

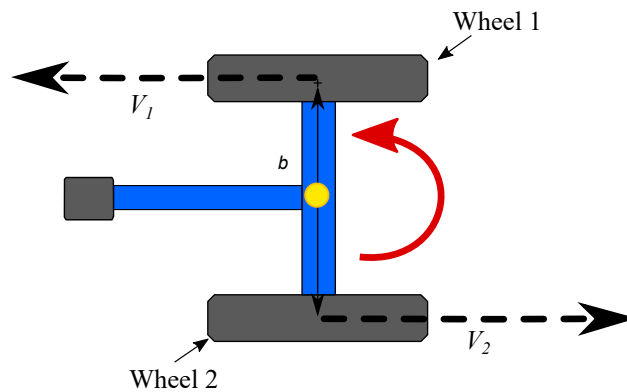
The variation in angle and displacement is achieved with the difference in rotation of the actuated wheels [68]. As shown in Fig. 2.4a if the wheels move with the same speed, the vehicle moves forward (or backward depending on the movement direction), when the wheels move with the same direction, but at a different speed, the vehicle produces a curved movement that turns in the direction of the slower wheel, Fig. 2.4b. Also, the UGV with differential drive can rotate along its own rotating axis if the wheels move in opposite directions as shown in Fig. 2.4c.



(a) Straight movement when V_1 is equal to V_2 .



(b) Rotated displacement when V_1 is smaller than V_2 .



(c) Rotation along the own body when V_1 is in opposite direction to V_2 .

Fig. 2.4 Possible movements for the differential drive with two actuated wheels and a caster wheel.

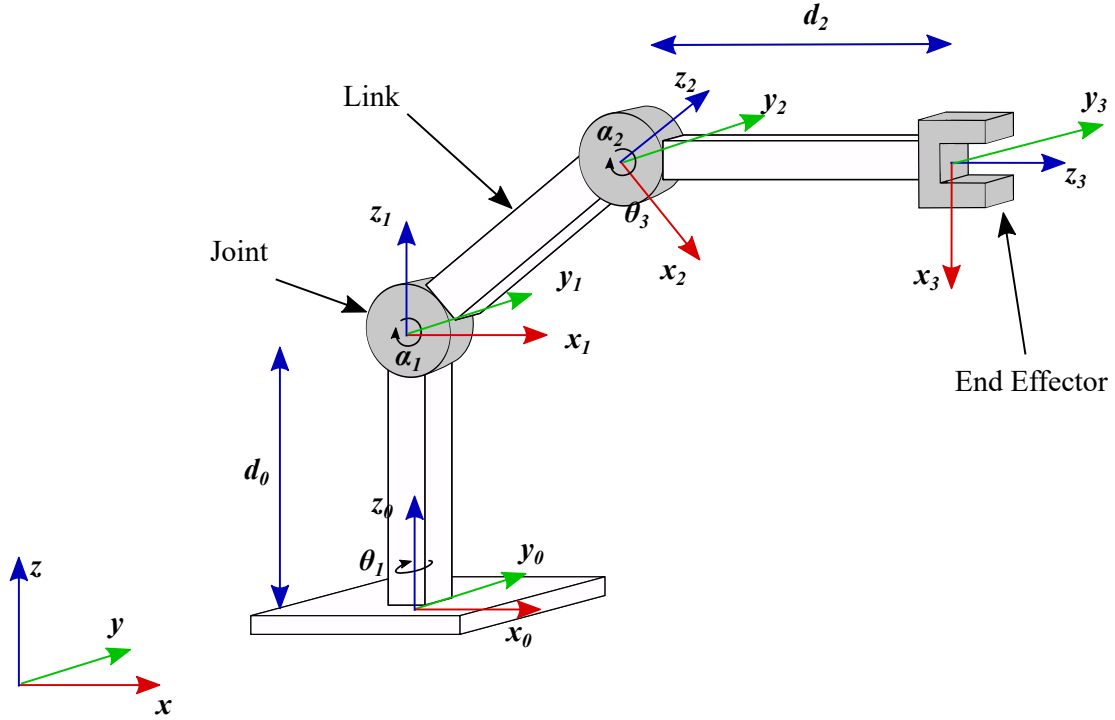


Fig. 2.5 Coordinate system for the robot arm and location of main Denavit–Hartenberg parameters. In this case, the robot arm has only 3 joints $n = 3$.

2.1.3 Robot Arm Motion

The robot arm is a mechanical device composed of actuated joints connected to form a moving chain. Usually, the first link of the robot is anchored to a stable structure, and the last link is provided with a manipulation device, like a robot hand, claw, gripper, or any kind of tool to interact with the environment. This device at the endpoint is often called *end effector*, as it is in charge of produce some effect on the environment.

Each actuator of the arm is controlled independently to position the end effector collectively. The angular position of each motor affects the final location and angle of the end effector, and depending on the number of links, it is possible to find more than one combination to achieve the same end effector state.

As shown in Fig. 2.5, the robot arm has a global coordinate system relative to the base where it is firmly attached. Moreover, every joint has its own coordinate system that is relative to the link where it is attached. Then the angle of each joint is measured in relation to their own coordinate system; the same concept is extended to arms with several joints.

2.1.3.1 Forward Kinematics

In order to calculate the position and rotation of the end effector with the angles of the joints, it is necessary to use the forward kinematics equation of the robot. Each robot has a different forward kinematic equation, because it depends on the length of the links and the angular and displacement offsets; its formulation is done with the help of the Denavit–Hartenberg parameters of the n joints of the arm ¹

- d_n : displacement along z_{n-1} .
- θ_n : rotation around z_{n-1} axis.
- r_n : displacement along the direction of y_n axis.
- α_n : rotation around y_n axis. From z_{n-1} to z_n .

The parameters are determined in base on the axes coordinates drawn on every joint of the arm numbered with sub indexes from 0 to n . The parameters are used for the calculation of a transformation matrix H_i for every joint. The equation to reach the transformation matrices is as follows.

$${}^i_{i-1}H_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_{i,i+1} & \sin \theta_i \sin \alpha_{i,i+1} & \alpha_{i,i+1} \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_{i,i+1} & -\cos \theta_i \sin \alpha_{i,i+1} & \alpha_{i,i+1} \sin \theta_i \\ 0 & \sin \alpha_{i,i+1} & \cos \alpha_{i,i+1} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.1)$$

with $i = 1, \dots, n$. The obtained matrices are the homogeneous transformation from the previous joint with index $i - 1$ to the current joint i . The resulting transformations are multiplied to obtain the final homogeneous transformation matrix that transforms from joint 0 to joint n .

$${}^n_0H = \prod_{i=1}^n {}^i_{i-1}H_i \quad (2.2)$$

The n_0H matrix contains the Rotation and Displacement of the end effector based on the angles and offsets of the robot arm. If we consider that the robot arm usually contains only rotational joints (without prismatic joints), and the offsets are usually zero (as is the case of

¹Notation based on the original Denavit Hartenberg parameters in [69] and the examples in [70–72]

the robot arm used in this thesis), then we can define a Forward Kinematics equation with respect to the joint angles q based on the Denavit–Hartenberg parameters.

$$F(q) = {}^0_n H(q) \quad (2.3)$$

Note that the joint angles q can be placed as α or θ angles in the Denavit–Hartenberg formulation.

2.1.3.2 Inverse Kinematics

Given the final end effector location, the Inverse Kinematics calculation produces the joint angles of each joint of the arm. There are many methods for this calculation, in this section it is explained the details of the algebraic solution approach.

By using the homogeneous transformations of each joint of the robot ${}^i_{i-1}H_i$ with $i = 1, \dots, n$, it is possible to find a set of equations that solve the problem of finding the joint angles. Note that the transformation from link 0 to link n n_0H contains the desired rotation (r_{ij} with $i, j = 1, 2, 3$) and translation (p_x, p_y, p_z) values for the end effector.

$${}^n_0H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.4)$$

With the calculation of the inverse matrix $M^{-1}M = I$, being I the identity matrix, it is possible to rewrite the Equation (2.2) in the following way.

$${}^1_0H^{-1} {}^n_0H = {}^1_0H^{-1} {}^1_0H {}^2_1H {}^3_2H \dots {}^n_{n-1}H_n, \quad (2.5)$$

that is simplified as,

$${}^1_0H^{-1} {}^n_0H = {}^2_1H {}^3_2H \dots {}^n_{n-1}H_n \quad (2.6)$$

In similar way, the rest of the equations are obtained.

$$[{}^1_0H^{-1} {}^2_1H^{-1}] {}^n_0H = {}^3_2H \dots {}^n_{n-1}H_n, \quad (2.7)$$

repeating the process for all joint indexes until forming the equation for the n index.

$$[{}^1_0H_1^{-1} {}^2_1H_2^{-1} \dots {}^{n-1}_{n-2}H_{n-1}] {}^n_0H = {}^n_{n-1}H_n \quad (2.8)$$

All the resulting equations are non-linear equations expressed with the inverse homogeneous transformations. These can be used to find arithmetically each values of the Rotation and Translation values of n (n_0H).

The multiplication of the matrices and calculation of their inverses are of great complexity, and it is convenient the use of computational tools to speed up the process.

A more complete explanation and theoretical development of the forward and inverse kinematics can be found in the works in [72–74]

2.2 Mobile Manipulation

The previous sections have mentioned the main features and the motion capabilities of the UGV and the robot arm. This section introduces a concept born with the combination of both robot platforms in a single robot platform.

The *mobile manipulation* consists of the ability of a robot to interact with the environment on-site, and it is achieved by designing a complete robot system comprised of a robot actuator with a mobile platform [75].

In the robot team created in this thesis, the mobile manipulator is designed with a robot arm attached to a robust mobile robot base that can be self-transported to the place of interest bringing robust and versatile 6DoF manipulation to the scene. See Fig. 2.6.

This type of robot arrangement is significant for different applications. On most occasions, it is the only tool to perform essential tasks within environments where a human agent cannot enter, for example, collapsed buildings, caves, disaster scenes, or small passages. The use of mobile manipulation can replace direct human intervention and bring other vital features like robustness, heavy-duty manipulation, precise positioning, and quick displacements.

The mobile manipulator provides the user with the advantages of both robot platforms and more actuation choices. Extra degrees of freedom are added to the manipulator, which means the workspace is expanded unlimitedly. Also, the manipulation tools deployed in the scene can be useful for a better scene assessment and obstacle detection and removal, which means navigation and exploration become more reliable. As a disadvantage of the use of the mobile manipulator, the addition of wheels to the manipulator creates a complex platform with increased difficulty for control and operation; it is required to build an excellent

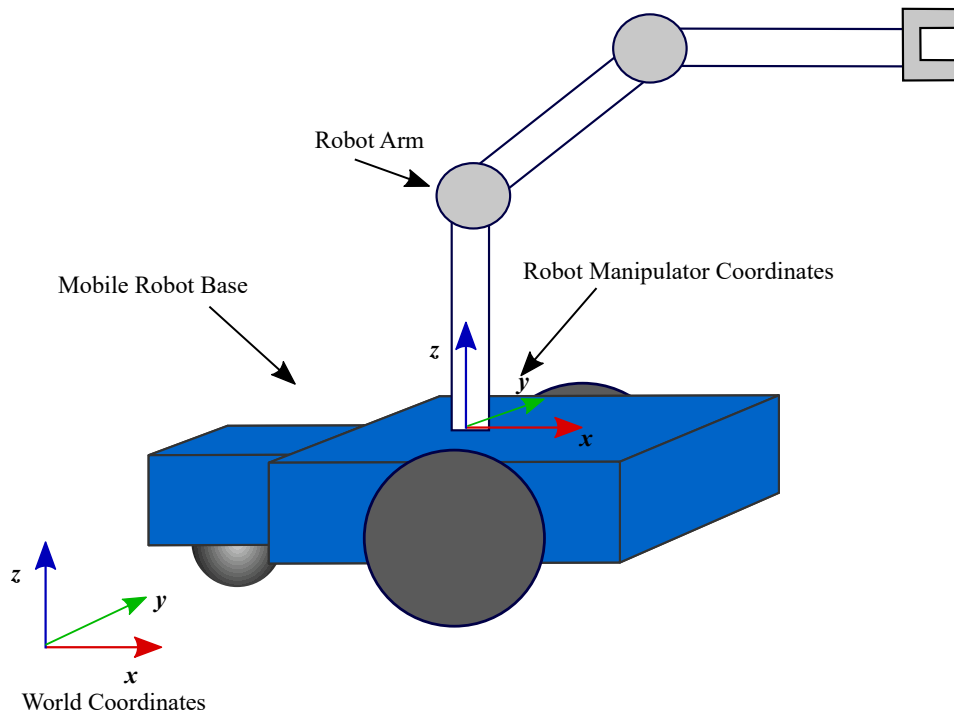


Fig. 2.6 Mobile manipulator and its coordinate system.

coordinate system to represent the large workspace and a good control strategy that handles the integrated robot.

2.3 Robot Pose Estimation

The robot pose estimation is the ability of the robot platform to determine its own position and orientation in a defined coordinate system. This process is essential for robot navigation because it allows to determine the position of the robot in the global or local map and to plan a safe route towards a goal. With a precise pose estimation, it is possible to build a robust navigation system and perform an optimal scene interpretation to reach the goal while avoiding dangerous situations.

The robot pose estimation is also crucial for robot teams because it provides with the estimation of the position of all robots in a common coordinate system. If the state of all robots is known, the planning tasks can be performed optimally, and the common goal can be reached in less time.

Two types of robot pose estimation concern the designers of heterogeneous robot teams, they are the global and relative pose estimation. These two concepts are essential for the UAV docking system designed in this thesis; they are explained in detail below.

2.3.1 Global Pose Estimation

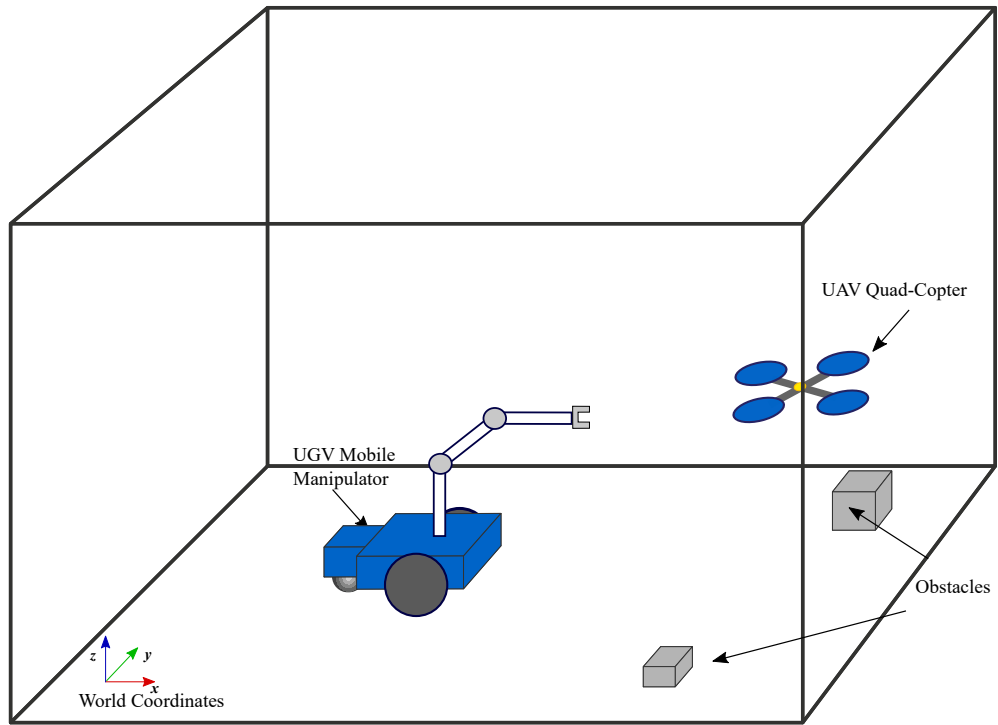
When the team is working within an environment, it is necessary to define a common coordinate system that allows to reference all involved objects in the scene with a unique origin point. This coordinate system is as large as possible, but it will be limited to the needs of the robotic system; thus, it is not needed to reference objects or areas that are not reachable by the robot team. The defined map should have enough resolution to represent precise positions and motions of all robot platforms and their relationship with obstacles and structures.

The construction of a global map with its own coordinate system requires the use of sensors that provide absolute measurements or provide data of fixed structures in the scene. For example, the use of GPS (Global Positioning System) signals allows calculating the position in a planetary framework, which is an immediate solution that is the first option in a robotics application. However, the GPS signals are not available in enclosed scenarios or on the sides of tall buildings.

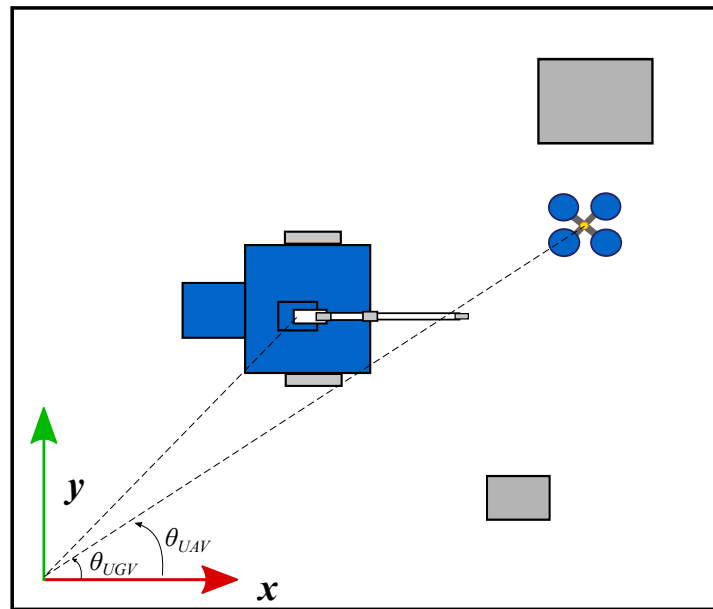
Considering that the global map only needs to represent the concerning areas, it is possible to use sensors for measurements of fixed structures in the scene to build a map. For example, the laser, visual, or sonar sensors, can measure the distance of the robot to fixed structures as explained in [76, 77]. This allows the construction of the global map, by tracing the fixed structures and determine the robot position and orientation relative to them.

The global map is built with the measurement of static elements in the scene, by analyzing the sensor data, the structures that persist in the scene are represented in a grid. By knowing the 3D distance of the robot and these elements, it is possible to define the reachable space and the prohibited space of the environment; these spaces define the global map for the robotic system.

As shown in Fig. 2.7a, the detected objects and the robots have associated an angle and a vector that represent the position and orientation in the map. With this information, the system is aware of the state of the elements, and the robots recognize themselves and the rest of the objects and walls in a common framework.



(a) Three dimensional global map of the scene. The enclosing area represents the limits of the global map and the walls that are a fixed reference.



(b) Two dimensional global map of the scene.

Fig. 2.7 Representation of the global map of a scene.

2.3.2 Relative Pose Estimation

When an interaction between robots is required, it is useful to count with a relative pose estimation mechanism to set a smaller framework for the concerning robots. In the relative pose estimation, the coordinate system is fixed to a point within a robot platform. That means that the coordinate system can be in movement and that the precise position and orientation of other robots or objects can be calculated in relation to this local coordinate system [78–80].

With the inclusion of a local coordinate system, the translation and rotation of the self origin of coordinates are ignored, reducing the complexity greatly in calculating the positions and orientations. The local coordinate system allows to section the frameworks and simplify the expression of the state of the robots.

For the local coordinate system construction, the origin of the sensor is previously defined in a fixed point on the robot chassis. It is more convenient to make this origin point coincide with the center of the robot without any offset, then the robot position point in global coordinates will be the origin of the local coordinate system, which will simplify calculations later.

The sensors for the local coordinate system measure the features of interest for object detection and robot localization. They can also measure the state of other robots and reference the data in relation to its origin. This feature is of great interest for the cooperative robot teams because it can complement the global system with a local estimation that can be more robust and efficient.

If the different tasks make use only of the local coordinate system, the uncertainty contained by the global map is avoided. Thus, the local approach is convenient for implementing complex tasks like the VTOL-UAV docking system that require a high level of precision.

For the VTOL-UAV docking system, a local coordinate framework is essential to represent the pose of the UAV relative to the mobile manipulator. With the information of the position relative to the UGV, there is no need to consider the global estimation for the whole docking system. The necessary information to measure the UAV pose is extracted by the UGV sensors that are particularly looking for the UAV.

If the sensors are not placed in the center of the mobile manipulator, it is required to remove the offset with a simple algebraic operation. Moreover, if the sensors are attached to a link of the robot arm, it is required to add the position and orientation of the link to transform the measurement to the local UGV coordinates. This process needs to be executed every time the measurement is updated and the robot arm moves, hence, the importance of

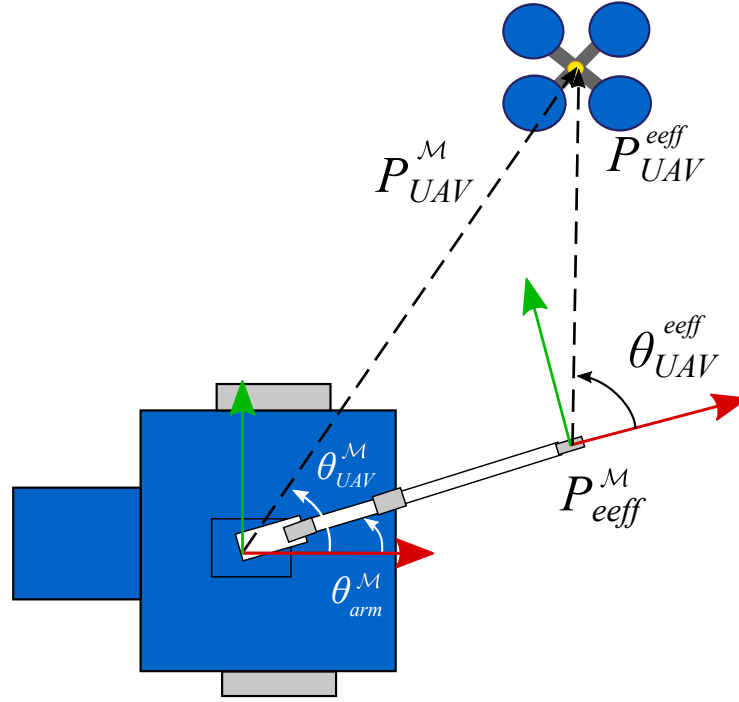


Fig. 2.8 Relative coordinate system of the mobile manipulator. The $\mathcal{M}$ superscript represents the mobile manipulator framework, while the eff superscript represents the end effector relative framework.

well defined parameters and correct implementation of the kinematic equations of the robot arm for precise transformations.

The local coordinates are used for specific tasks, allowing to focus on the current assignment and avoiding certain complexities and uncertainties. Nevertheless, the local and global coordinates are tightly related, and a transformation matrix between the two of them is always available. This transformation is formed by the translation and rotation components of the robot.

To convert the local measurements into global coordinates, it is only necessary a simple multiplication by the transformation matrix. Similarly, if the pose of another robot is calculated in local coordinates, it can be transformed to global coordinates, always considering and adding the uncertainties involved in the measurement process.

For a detailed explanation of the transformation process, a common situation for the heterogeneous air-ground robot team is shown in Fig. 2.8. The Mobile Manipulator with a 3 DoF robot arm and a flying UAV is represented in the figure. The sensor in the end effector of the robot arm is used to measure the pose of the UAV (P_{UAV}^{eff}); this measurement will be within end effector coordinates eff . This coordinate system has its origin point in the center

of the end effector sensor. To transform this measurement P_{UAV}^{eff} into Mobile Manipulator coordinates $\mathcal{M}$, it is necessary to find the transformation matrix between the *end effector* and the Mobile Manipulator ${}^{\mathcal{M}}_{eff}H$.

The matrix ${}^{\mathcal{M}}_{eff}H$ is formed by the matrices that transform measurements from the end effector to arm coordinates ${}^{arm}_{eff}H$, and from arm to mobile manipulator coordinates ${}^{\mathcal{M}}_{arm}H$.

$${}^{\mathcal{M}}_{eff}H = {}^{arm}_{eff}H {}^{\mathcal{M}}_{arm}H \quad (2.9)$$

Note that the origin of mobile manipulator coordinates is the same of the origin of arm coordinates and the orientation is conveniently arranged to coincide. Thus, both coordinates are equivalent which means that the transformation ${}^{\mathcal{M}}_{arm}H$ becomes a unitary matrix and the transformation ${}^{\mathcal{M}}_{eff}H$ simplifies as follows.

$${}^{\mathcal{M}}_{eff}H = {}^{arm}_{eff}H, \quad (2.10)$$

which means that it is only needed to find a transformation from the end effector to the origin of the arm coordinates. This transformation was introduced in Section 2.1.3.2 as the Inverse Kinematics relation of the arm. The position and orientation of the end effector are set in the corresponding matrices², and the calculations are carried out to reach the final expression for the transformation ${}^{\mathcal{M}}_{eff}H$.

Finally, the pose of the UAV in local $\mathcal{M}$ coordinates $P_{UAV}^{\mathcal{M}}$ is calculated with the multiplication of P_{UAV}^{eff} by the transformation matrix ${}^{\mathcal{M}}_{eff}H$.

$$P_{UAV}^{\mathcal{M}} = {}^{\mathcal{M}}_{eff}H P_{UAV}^{eff} \quad (2.11)$$

For the proposed docking task, the UAV and the end effector close interaction require a reliable and simple measuring framework. For this reason, their orientation and translation are estimated only in relation to the local coordinates of the Mobile Manipulator $\mathcal{M}$. As previously explained, the preparation of UAV location in local coordinates can be achieved with simple calculations avoiding global noise and unnecessary estimations. For this reason a local measurement for UAV pose will be the main input for control purposes in the design presented in this thesis.

²The rotation and translation values are placed in the transformation matrix in the same way as in Equation (2.4)

2.4 Summary

This chapter detailed the robot motion control concepts of each robot platform involved in the UAV docking system. The ground robots are platforms of relatively easy control and outstanding robustness, while the UAVs are of more difficulties but far more potential. The robot arm design requires the consideration of more parameters, pieces, and actuators, and its control is of increased complexity. However, with the correct mathematical approach and good computational tools, the robot arm offers excellent precision and redundancy.

The mobile manipulator platform was also described as an integration of the robot arm and UGV, which offers significant capabilities to the team and enables the on-site docking process. The pose estimation details with the global and local approach are presented, focusing on the local setup as a vital technique for the docking system.

In the state of the art of the field, it is possible to find many options for the management of the cooperative scheme with air-ground robots, and they can be merged to perform robot interaction tasks. The technologies available have great potential to create systems with multiple robot integration for solving complex robot missions and are essential for constructing the VTOL-UAV docking system.

Chapter 3

VTOL-UAV Docking System

In this chapter, the details of the VTOL-UAV docking system applied to air-ground robot teams are described. A brief introduction to the USAR missions with heterogeneous robot teams is presented as a context to understand the different priorities in the handling of a robot team in such complex environments, and to explain the need of the robots contact process. Also, the air-ground robot team, its main features, and the theoretical approach to build the VTOL-UAV docking system are described. To give a detailed and easy to understand explanation, the algorithms and mathematical methods along with a general flowchart of the control scheme that combines predictive and optimization approaches to obtain a robust, safe and repeatable docking are presented.

3.1 USAR Missions with Heterogeneous Robot Teams

The USAR robotic missions are distinguished by being developed within enclosed structures like buildings or rooms, often semi-collapsed, unstable, with obstacles and with the presence of debris, caused by natural disasters or by accidental or deliberated reasons. The environment of a USAR mission is completely unknown and can present situations of great danger for human exploration.

A heterogeneous robot team deployed in such unknown environment is required to perform a series of navigation, exploration, and interaction tasks that are executed with different priority levels defined by the mission design. The list of autonomous tasks given to the robot team needs to have a clearly defined hierarchical order, then all the tasks have a clear priority for normal operation and more importantly, this helps to define the course of action for emergency situations. In the case of urban search and rescue robotic operations

for indoor environments, the following is an example of a list of prioritized tasks to be accomplished by the robots:

1. Reach and interact with the target.
2. Maintain safety for all robot platforms.
3. Economize energy.
4. Extract as much information as possible from the scene.

In the field of search and rescue operations, the process of reaching the main target could belong to a higher level of priority, while the protection of the robots can be placed at a slightly lower level. It is because of the critical nature of the USAR applications, as a clear example, the robots in [81] were designed to provide assistance and explore the unstable environments after the World Trade Center disaster, in this case, the safety of the robots is not prioritized because the main concern is to guard the stability of the scene to avoid any extra risk to the victims; the robots performed navigation and exploration in search for victims and structural information that was valuable for the rescue team, some robots were damaged or lost, but the situation worth the risk.

On the other hand, in emergency response cases described in the Fukushima disaster [82], the robots needed to quantify water motion, recover samples, and rapidly install a water gauge in a radioactive environment. In this case, the building was completely abandoned, then, the robot safety was the top priority as it needed to return to a safe area for sample retrieving, if the robot had been lost at the scene, the entire mission would have failed.

Furthermore, in cases where the supervision of a human operator is involved, the human agent in the loop can provide an expert opinion of the state of the whole mission and, when necessary, perform goal modification or cancellation with top priority at any time.

In other cases, the primary purpose of the robot team is to explore the environment and provide sensor information for obstacle detection [7]. For that, all the resources of the team are used to follow trajectories to survey the scene and to merge heterogeneous sensor information. In this case, the mission is not firm, and it is possible to prioritize with less rigor giving more importance to data extraction and robot protection.

The system developed in this thesis can help to accomplish a mission where the main goal is to reach and interact with a *target* in an unknown environment. It is also important to extract as much information as possible from the scene with a detailed inspection approach

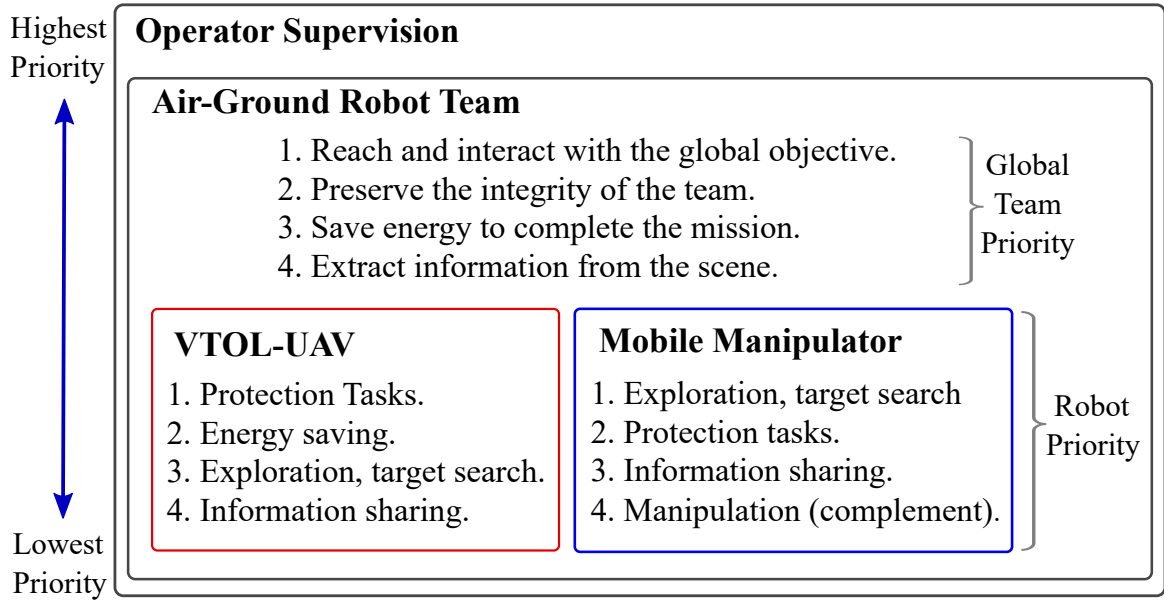


Fig. 3.1 Priority levels for tasks execution in the air-ground robot team.

to facilitate navigation and share essential data with future missions. If the building is large and is composed of several rooms and aisles, the mission needs a long time to be finished. And if the scene is cluttered and contains aerial or ground navigation difficulties, the robots need to execute mechanisms to solve local problems and plan for an optimal way to reach the final goal.

The air-ground team here proposed is composed of a lightweight VTOL-UAV and a Mobile Manipulator platform (mobile robot + manipulator) in a centralized configuration. This team and the controller architecture are described in detail in Section 3.2 and 3.3. For this kind of air-ground robot team, it is essential to consider the features of each robot platform to establish the priority for the tasks. Each platform has to prioritize their tasks, but as part of a team, they also have a common goal and general priorities (please refer to Fig. 3.1).

For the mobile manipulator, the significant priority should be in following the shortest possible trajectory to reach the global *target*, and at the same time, it should be concerned in performing basic sensing of the scene for information and navigation purposes. For example, the *target* can be to reach a person, open a valve, recognize or pick up an object. Then, the on-board sensors and actuators are used for obstacle detection and to interact with the environment for a safe navigation.

The aerial robot platform is designed solely for exploration purposes. It can cover large areas and can potentially provide sensor information with six DoF. Such information is

supplied as complementary information, which means the team, although not optimal, can continue pursuing the goal even if the aircraft is not providing any data. Moreover, because of its high risk of damage, and limited autonomy, the UAV is forced to execute a high priority self-protection task. It continually checks the battery level, distance with the charging station, and possible obstacle collisions in the vicinity. The UAVs' lower priority exploration tasks can be immediately interrupted or delayed if the protection process requires to perform any operation.

Both of the robot platforms are in constant communication with each other performing full or semi-autonomous tasks, but the team acts under the orders from a human supervisor with topmost priority. As shown in Fig. 3.1, the global priorities are in charge of the management of the team as a whole, and local priorities to control the behavior of each platform. As stated earlier, the higher levels can outrank the lower levels, for example, when the UAV has some protection issue (e.g., energy deficiency or presence of obstacle that impedes exploration), the complete team does not stop trying to reach the main *target*. The aircraft should find a way to protect itself, e.g., by reaching a charging station or calculate trajectories to avoid obstacles.

On the other hand, if the exploration and *target* pursuit are being developed correctly, the rest of the computational and hardware resources (including mobile manipulation) can be used to ensure the safety of the robots specially of the UAV.

3.2 Air-Ground Robot Team

It is possible to use several types of multi-robot systems to solve industrial, USAR, or even spacial missions. In this thesis, we propose the use of an air - ground robot team composed by a VTOL-UAV and a mobile manipulator with centralized architecture (see Fig. 3.2). It is described here the main features of each of the robot platforms, and the general attributes that make possible the VTOL-UAV Docking system. Also, in Chapter 3 it is described the robot platforms with more technical details, focusing in the the hardware and software used for simulation and real experiments.

3.2.1 VTOL-UAV

This platform is a lightweight multi-rotor aircraft with excellent stability and maneuverability. The robot comes with a visual sensor, barometer, sonar, gyroscope, inertial measurement

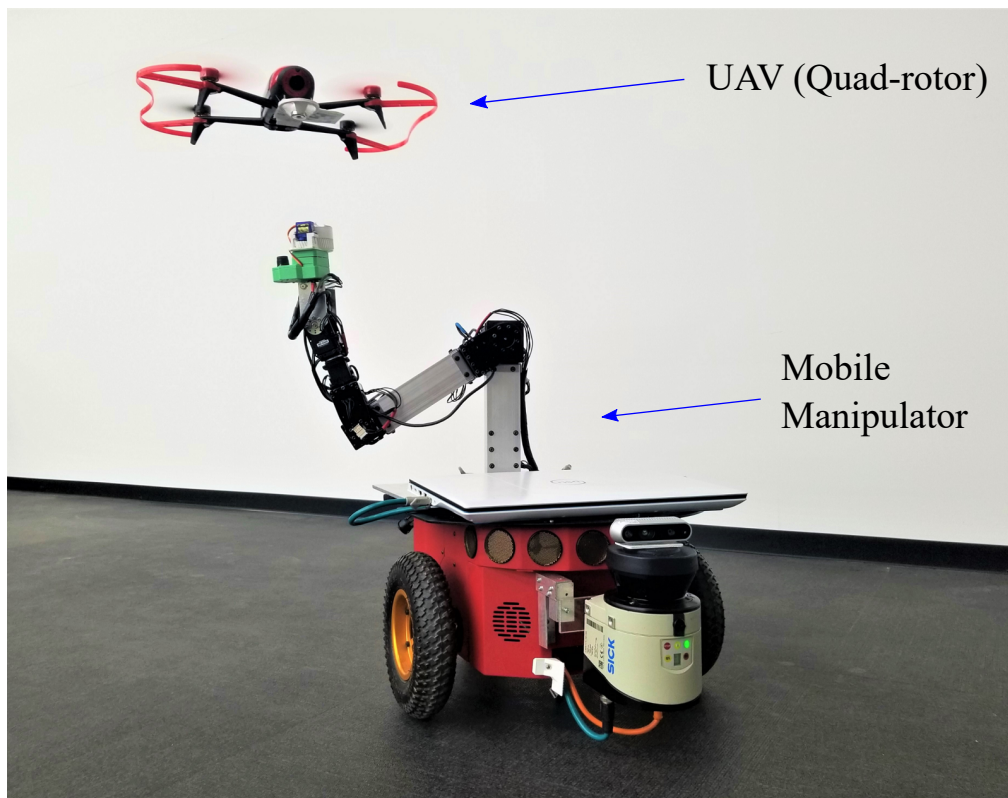


Fig. 3.2 Air-ground robot team. Photography of the robot team with the VTOL-UAV in flight and the mobile manipulator with its arm pointed towards the UAV.

unit (IMU), and battery level sensors. Four actuators with rotating propellers are placed in symmetrical arms. The controller is in charge of the velocity variations of each propeller to obtain the exact combinations to achieve linear and angular movements, more specifically, the *altitude*, *yaw*, *roll* and *pitch* variations. It is equipped with a docking adapter and a fiducial marker (front - bottom) to be detected with visual sensors from the ground for contact purposes.

This robot platform has the ability to hover in a static place and to move with great agility. However, the UAV has six DoF; this number is larger than the number of independent control inputs, which means that it is an under-actuated platform. The under-actuation characteristic limits the UAV on the number of configurations that can be directly controlled [51]. This means that the UAV is capable of a great variety of movements, but its control is difficult, and it cannot respond well to large external disturbances. This type of aircraft is not robust enough for applications that require great stability and reproducibility.

3.2.2 Wheeled Mobile Manipulator

It is a particular type of robot with a wheeled base and a robot manipulator attached to the top deck. The base has excellent robustness to carry a heavy payload and enough agility to move in ground terrain. However, it moves in a 2-dimensional space and, unless it uses holonomic wheels, it can have restricted movements. Also, depending on the wheels-suspension set, it can overcome most debris on the ground, but it can have some troubles when dealing with large obstacles.

In the robot team designed for this work, a mobile manipulator with a two-wheeled differential drive and caster wheel and an especially designed 6 DoF robot arm is used.

The mobile manipulator is designed to bring manipulation to the environment and interact with the scene with more versatility. There are many possible tool configurations for the end effector of the robot manipulator, and it can be a gripper, a variety of actuators, or a set of sensors. For our purposes, the end effector holds a camera that provides visual information with an eye-in-hand configuration useful for a detailed exploration when a close view of an area is required. In our case, it is used as the main measurement unit for UAV detection. Additionally, the end effector is equipped with a docking device designed specifically for the system proposed in this thesis. This system is added to secure the UAV by locking its adapter and ensuring a safe UAV docking and storage.

The Wheeled Mobile Manipulator uses its sensors to measure its position in reference to static structures and build a global map. The measurements are also used to create a local map of close objects and obstacles that represent an immediate threat. Similarly, the Mobile Manipulator measurements (primarily visual measurements) are used to detect the VTOL-UAV platform and determine its exact position and orientation in local coordinates; this data is the fundamental information to perform the docking process.

3.3 VTOL-UAV Autonomous Docking

The docking between the air-ground robot team is a crucial task for long-term mission planning and exploration. With a robust performance of this task, the robot team can achieve a safe contact between the air and the ground robot, and the capabilities of the complete heterogeneous team are extended [65]. In this subsection, the docking task is described in detail. As shown in Fig. 3.3 , the proposed docking operation is divided into four sub-tasks:

- Initial encounter
- Tracking
- Contact
- Storage

The initial encounter is the process when the two robot platforms reach a meeting point; the end effector searches for the UAV with the camera pointed upwards. When the UAV marker enters the vision field, the UAV relative position is extracted, and the tracking task starts.

The tracking task is initiated after the first encounter, which is when the UAV marker enters in the vision field of the end effector camera, and continues until the final docking when the aircraft is secured at the end effector. The contact task is when the end effector of the arm approaches to the UAV until it locks the UAV with a docking device. Finally, after docking the aircraft, it is stored in a safe position withing the mobile manipulator. The complete VTOL-UAV docking process is shown in Fig. 3.3, and the different sub-tasks are described in detail below.

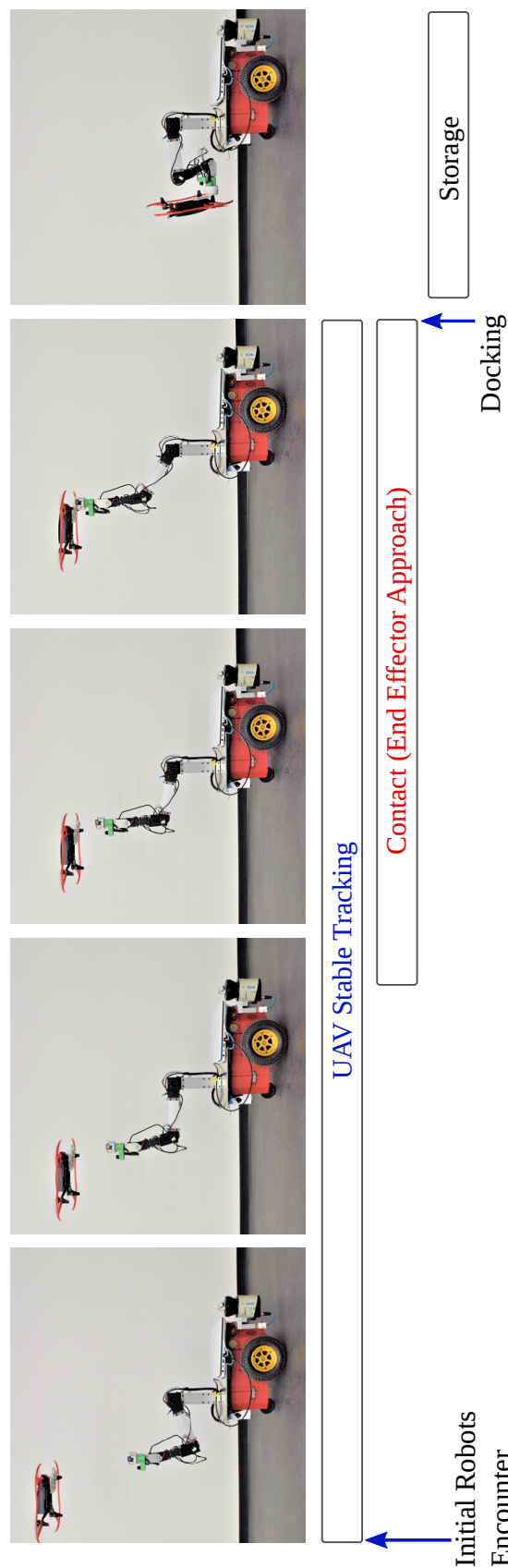


Fig. 3.3 General flow for the VTOL-UAV docking operation.

3.3.1 UAV Motion Tracking

For the docking process, only the information extracted with the camera is used. Both the UAV and the robot manipulator are being controlled simultaneously. The manipulator takes care to pair (align) its end effector tip with the UAV adapter, meanwhile the UAV tries to follow the movements of the mobile manipulator and hover in a reachable position.

During the final approach and the actual contact, the position and orientation of the end effector relative to the aircraft need to be stable and constant. For this steady-state, the control system design needs to take the following robot considerations:

- *UAV lateral and longitudinal movements*: The camera attached at the end effector is pointed to the UAV marker extracting its relative pose. While the UAV is in motion, the end effector follows and plans to keep the marker close to the center of the image. The end effector will move in a single 2D layer of its workspace if there is no obstacle or constraint.
- *Self prohibited positions*: The end effector should move within the allowed workspace and avoid prohibited positions. The arm joints need to be controlled within its range.
- *UAV features*: If the UAV has any large displacements, for example, if it moves faster laterally than longitudinally, then the end effector should follow such motions accordingly. Another important aspect to consider is the UAV sensor placements. For the VTOL aircraft, the altitude sensors are pointing in downwards direction, if they detect the mobile manipulator structure below them, the altitude measurement can be relative to the UGV instead of the ground, this situation can cause loss of sight and interrupt the *tracking process*.

A control system that considers all the previous factors to perform an optimal tracking task is designed. It controls the UAV pose to locate in a reachable position within the manipulator workspace and uses a series of optimization, prediction, and reactive algorithms to build a *Motion Engine* that calculates proper trajectory points for the end effector.

As shown in the complete controller scheme in Fig. 3.4, the UAV pose in manipulator frame $P_Q^{\mathcal{M}}$ is used to feed the manipulator *Motion Engine* (that controls the end effector motion), and at the same time is used for UAV pose control with its respective transformations.

The UAV pose is being controlled within $\mathcal{M}$ coordinates, for that it is selected a fixed UAV pose in the same framework as reference for the UAV control loop. This control in $\mathcal{M}$ framework allows the UAV to be in a constant pose with respect to the mobile manipulator.

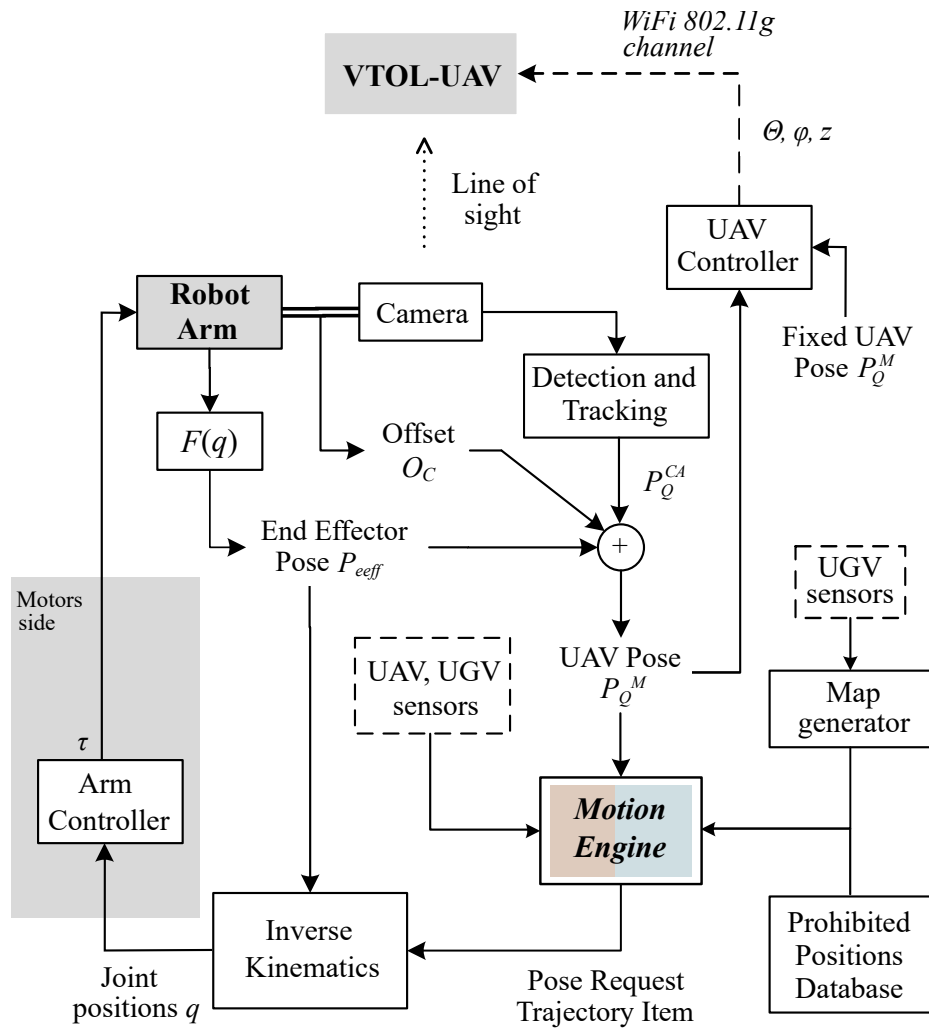


Fig. 3.4 General control scheme.

The current UAV pose P_Q^M is extracted from camera measurements, and transformed to manipulator coordinates. This transformation is achieved by merging the end effector position, given by the forward kinematic relation of the manipulator $\mathcal{F}(q)$, with the UAV pose in camera coordinates $P_Q^{C_A}$. To these values it is also added the offset O_C caused by the position of the camera that is not located exactly in the center of the end effector. The Equation (3.1) summarizes this transformation.

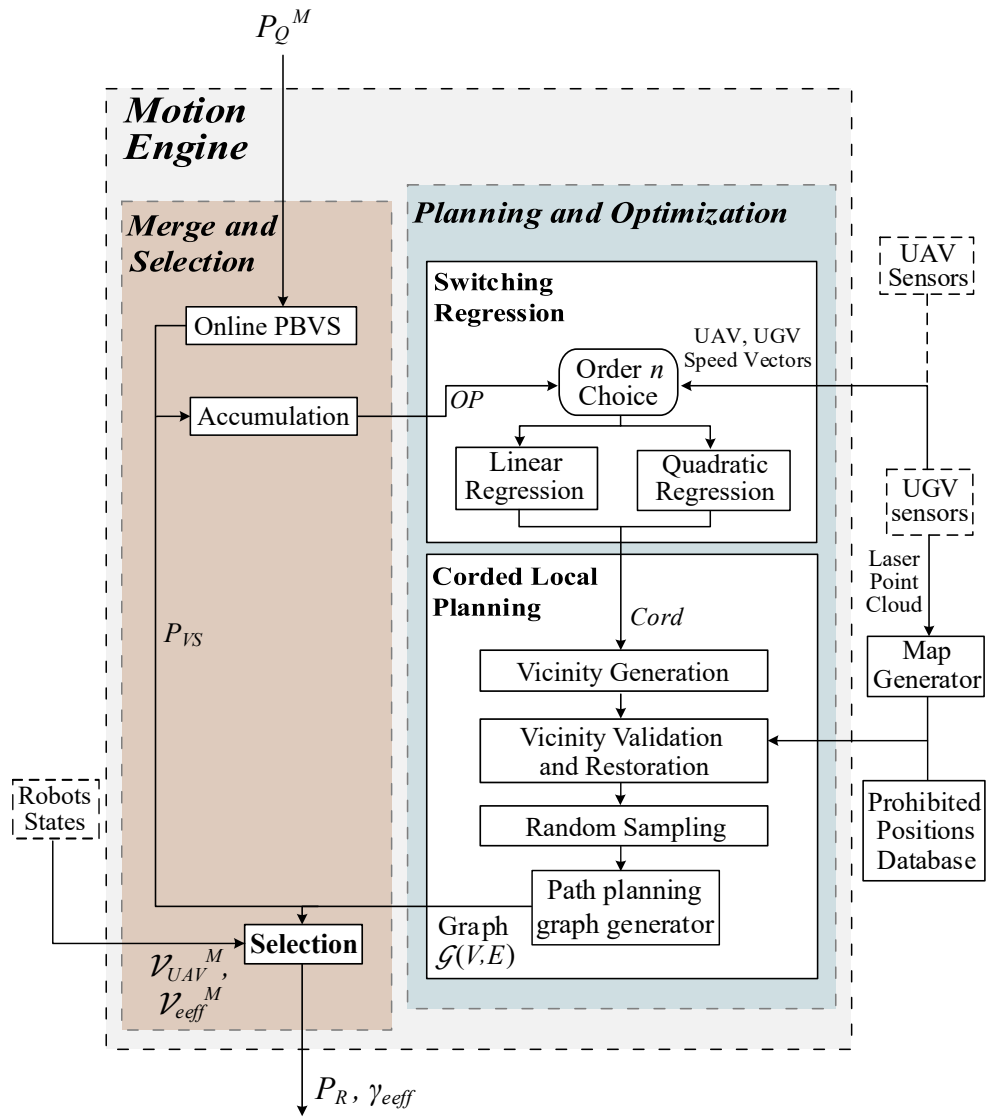
$$P_Q^M = \mathcal{F}(q) + P_Q^{C_A} + O_C. \quad (3.1)$$

The resulting pose P_Q^M becomes the input for the *Motion Engine* block. This block is one of the main contributions of this thesis, it consists of a trajectory generator that incorporates the UAV and UGV states for optimization and motion planning.

An expansion of the *Motion Engine* block is detailed in Fig. 3.5. In this figure it is possible to see the general flow of the block; the blue shading indicates the Planning and Optimization algorithms, while the orange part represents the algorithm for reactive control and high-speed distribution of results called Merge and Selection. During the execution, the blue part runs nearly at $f_B = 4$ Hz, while the orange part approximately twice as fast $f_O = 2f_B$. This frequency separation design is to isolate the slow performance algorithm of the blue part, while using a fast process in the orange part to afford real-time control of the robot arm, ensuring to bring instructions fast enough to follow the UAV movements and not lose the line of sight.

The Planning and Optimization process (blue shading in Fig. 3.5), is a sequential chain that starts with the calculation of a basic switching regression and a local path planning to provide a dense graph, while the Merge and Selection process (orange shading in Fig. 3.5) is in charge of the calculation of an *Online Position-Based Visual Servoing* (Online-PBVS) P_{VS} value calculated based on the work in [83]. The same Merge and selection process is in charge of storing past P_{VS} values in a list (*OP*), and merging its results with the Plannification tree that comes from the blue shading process and the UAV+UGV states to finally provide the next optimal end effector trajectory.

The sequential processes that happens inside the mentioned Planning and Optimization part (blue shading in Fig. 3.5), as well the last Selection task require a more detailed explanation, as they are a vital part of the UAV docking process. A complete description for the Switching Regression, Corded Local Planning and Selection tasks is presented below.

Fig. 3.5 Schematic of *Motion Engine* process.

3.3.1.1 Switching Regression

This first regression is to estimate the future UAV relative movements and provide a basic guide for UAV tracking. The rest of the Planning and Optimization algorithms will be built on top of this basic estimation, hence the importance of a well formed regression.

The switching regression considers the velocity differences between the mobile manipulator and the UAV to create a series of possible end effector movements connected by a cord. The objective is to derive the UAV relative motion intention and project the end effector towards the possible motion for tracking preservation.

The predicted movements are calculated for a short and constant range L_C (around $L_C = 15$ cm), and it is assumed that the robots will not perform complex motions for these small distances. Thus, only two important cases are necessary to implement, the linear and the quadratic regression (order $n = 1$ or $n = 2$ respectively).

The algorithm will analyze the state of the robots for switching (mainly considering their speed and acceleration). If the UGV is moving faster than the UAV, a linear mode will be used to replicate the UAV moving forward to catch up the UGV. On the other hand, if the UAV is having more activity and moving with higher speed than the UGV, a quadratic mode is selected to predict the natural movement of the UAV, that can present curves when trying to stabilize its position at low speed.

With any of these two cord types (linear and quadratic), the result will be a connected succession of end effector poses (*Items*), that build the *cord* and guides the motion planning, as shown in Fig. 3.6. The resultant *cord* will have N items (I_i with $i = 1, 2, \dots, N$) separated between each other with a distance equal to L_S and a total length of L_C ($L_C = NL_S$).

The first item I_1 is filled with P_{VS} calculated in the *Select and Merge* block. The rest of the points I_i for $i = 2, 3, \dots, N$ are calculated depending on the selected order n ,

$$I_i = [x_i \ y_i \ z_i] \quad (3.2)$$

$$y_i = f(x_i) = \alpha_0 + \sum_{k=1}^n \alpha_k x_i^k. \quad (3.3)$$

By using the curve fitting techniques explained in [84], it is possible to use least squares calculation and Gaussian elimination to find the coefficients α_n necessary to build the $f(x_i)$ function and finally find the rest of I_i elements. The input for this curve fitting calculation is a set of old P_{VS} values $OP = [x_j, y_j]$ for $j = 1, 2, \dots, N_{OP}$ provided by the Merge and selection block.

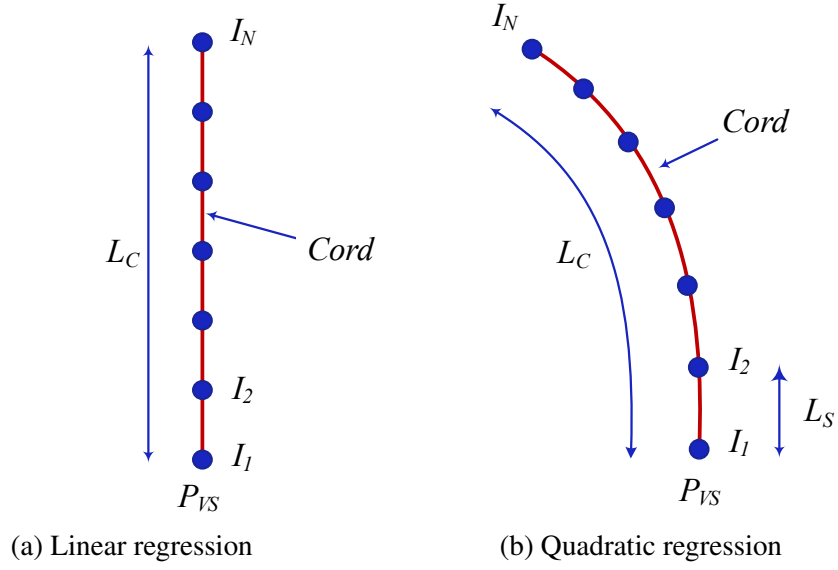


Fig. 3.6 Types of cords for the regression that guides the arm movements.

The following matrices¹ are used to define the equation system for the calculation of the α_n coefficients [84].

$$\begin{aligned}
 X &= \begin{bmatrix} N_{OP} & \sum x_j & \sum x_j^1 & \dots & \sum x_j^n \\ \sum x_j & \sum x_j^1 & \sum x_j^2 & \dots & \sum x_j^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \sum x_j^n & \sum x_j^{n+1} & \sum x_j^{n+2} & \dots & \sum x_j^{n+n} \end{bmatrix}, \\
 A &= \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}, \quad B = \begin{bmatrix} \sum y_j \\ \sum x_j y_j \\ \sum x_j^2 y_j \\ \vdots \\ \sum x_j^n y_j \end{bmatrix}, \tag{3.4}
 \end{aligned}$$

then the following system is defined,

$$XA = B, \tag{3.5}$$

¹The summations of Equation (3.4) are for $j = 1$ to $j = N_{OP}$, note that n is the selected order of the regression.

which is used to calculate carefully a solution for the coefficient matrix A that contains the α_n values,

$$A = X^{-1}B. \quad (3.6)$$

Once the coefficients α_n are available, the values for x_i and y_i are derived by using the Equation (3.3) with the considerations detailed in the Algorithm 1. The values for x_i and y_i are from the addition of a small value δ iteratively, or from the result of the Equation (3.3). This selection is done as indicated in the Line 5 of Algorithm 1 in base of the velocity components in each axis.

Algorithm 1 *Cord Calculation*

Input $P_{VS} = [x_{VS} \ y_{VS} \ z_{VS}]$, OP
Output $I_i = [x_i \ y_i \ z_i]$

```

1:  $I_1 \leftarrow P_{VS}$ 
2:  $\Delta x, \Delta y \leftarrow \text{CalculateAxesVariation}(OP)$ 
3: for  $i \leftarrow 2$  to  $N$  do
4:    $z_i \leftarrow z_{VS}$ 
5:   if  $\Delta x > \Delta y$  then
6:     for  $\delta \leftarrow 0$  to  $L_S$ , with small increments do
7:        $x \leftarrow x_{i-1} + \delta$ 
8:        $y \leftarrow \alpha_0 + \sum_{k=1}^n \alpha_k x^k$ 
9:       if  $(\sqrt{y^2 + x^2}) \geq L_S$  then
10:         $x_i \leftarrow x$ 
11:         $y_i \leftarrow y$ 
12:   else
13:     for  $\delta \leftarrow 0$  to  $L_S$ , with small increments do
14:        $y \leftarrow y_{i-1} + \delta$ 
15:        $x \leftarrow \alpha_0 + \sum_{k=1}^n \alpha_k y^k$ 
16:       if  $(\sqrt{y^2 + x^2}) \geq L_S$  then
17:         $x_i \leftarrow x$ 
18:         $y_i \leftarrow y$ 
19: return  $I_i = [x_i \ y_i \ z_i]$ 

```

Note that for the whole curve fitting calculation, the values for altitude z_i are all the same and equal to the z value of the P_{VS} pose. This means that the complete *cord* is maintained in the same height, only being modified by later computations to avoid constraints.

On the other hand, if the UAV relative pose is more or less static in $\mathcal{M}$ frame and the UGV is not moving in any direction, the information to perform prediction is not enough and the regression is canceled. In this case, the cord contains a single item $N=1$ filled with P_{VS} , and the planning will continue without main corded guidance (*cordless*).

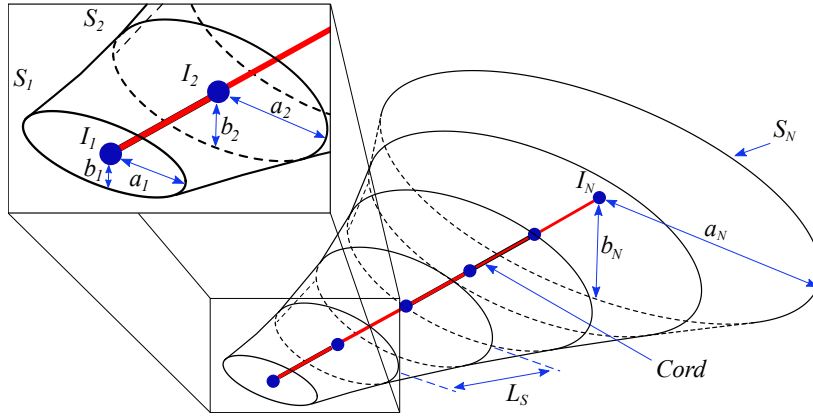
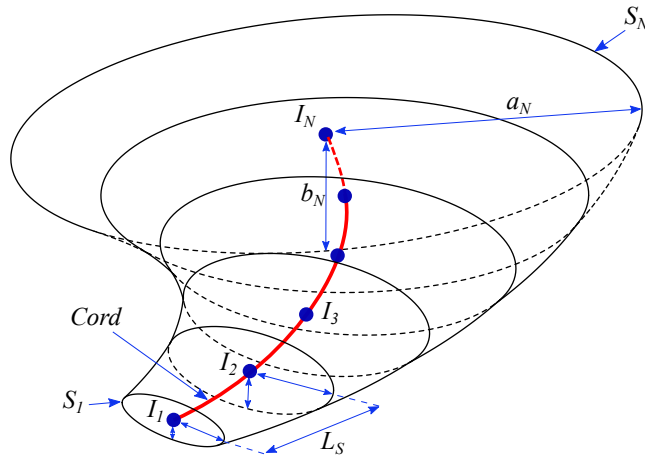
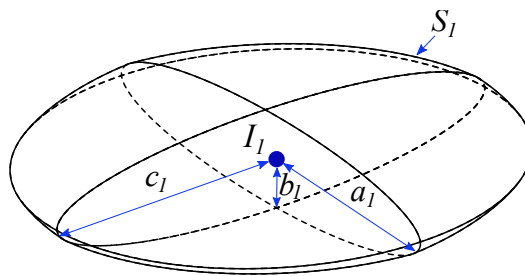
If the relative movement intention changes abruptly, a cancellation flag is created to impede the use of the cord for motion planning, the cord is replaced with a single item $N = 1$ to give time for a fresh pose accumulation, and a new cord is drawn in the next iteration to resume planning without delay.

3.3.1.2 Corded Local Path Planning

The behavior of the aircraft is difficult to predict. The computation of the *switching regression* provides a basic notion for the future relative UAV movements, but there is an exponentially growing uncertainty as the predicted distance increases. To include this factor in manipulator calculations, it is proposed to manage the end effector movements along a motion planning tree expanded in a local guided region. This region or vicinity will be calculated in the spatial surroundings of the mentioned *cord*, and its volume encloses all the possible 3D locations that have a great probability to place the end effector in an optimal tracking position for near-future iterations.

The vicinity is created with sections (S_i with $i = 1, 2, \dots, N$) delimited by ellipses centered in the items I_i . The ellipses are drawn perpendicularly to the main direction of the *cord* (see Fig. 3.7) with Euler angles φ_i , θ_i and ψ_i respectively, and their axes (semi-major and semi-minor axes a_i and b_i respectively) depending on the distance of the center (*Item*) to the cord origin as follows.

$$\begin{aligned} L_{os} &= 1 + \sum_{k=1}^i L_S \\ a_i &= K_a L_{os}^2 \\ b_i &= \frac{K_b L_{os}}{2}, \quad \text{with } i = 1, 2, \dots, N, \end{aligned} \tag{3.7}$$

(a) Linear *cord*(b) Quadratic *cord*(c) No *cord*Fig. 3.7 Vicinities with their respective *cords*.

where K_a and K_b are gains that determine the size of the ellipses. The semi-major axes a_i depends exponentially on the progression of the items (L_{os}) and determines the width of the ellipses perpendicular to the main direction. The semi-minor axes values b_i determine the size of the ellipse in z -direction and they grow linearly. This is because the vertical position of the end effector does not affect the tracking. If only the altitude changes in a small band, the marker position in the image is unaltered and the only uncertainty source is the scale variation.

In the case the cord has a single item ($N=1$), the ellipse gains are doubled, $K_{sa} = 2 K_a$ and $K_{sb} = 2 K_b$, and the vicinity becomes an ellipsoid, see Fig. 3.7(c).

$$\begin{aligned} a_1 &= K_{sa} L_{os}^2 \\ b_1 &= \frac{K_{sb} L_{os}}{2} \\ c_1 &= L_s \end{aligned} \tag{3.8}$$

The next step is to confirm whether the entire volume of the vicinity is confined within the allowed manipulator workspace by checking contacts with obstacles, prohibited locations or with singularity areas. For that purpose, it is convenient to use a 3D local map generated with UGV sensor information that is previously loaded with prohibited zones defined by the *mobile manipulator* dynamic and structural features, e.g. location of the mobile base, or areas where the end effector will fall in cases of singularities (See Fig. 3.8).

In Fig. 3.8, the green region represents the detected obstacles and will be drawn in the map as columns perpendicular to the ground expanded in polar coordinates thereby prohibiting the end effector to enter in the volume of the obstacle plus everything above, below and the space not seen by the sensors (gray area of Fig. 3.8). This consideration is taken because the end effector is tracking the UAV with the eye-in-hand camera, and the space between the end effector and the aircraft cannot be obstructed, it is important to keep the camera field of view clear from obstacles so the line of sight is maintained. Additionally, as the mobile robot platform only detects the faces that are facing the sensor, the obstacles are thickened for all points behind their visible faces. Thus, the obstacles are represented by forming rays drawn from the visible faces and moving away from the center of the UGV, generating obstacles expanded radially that try to compensate for the blind areas of the sensor.

The prohibited positions, e.g. singularity area or boundaries, are represented with their exact position in the map without column and radial expansion (orange and blue areas in

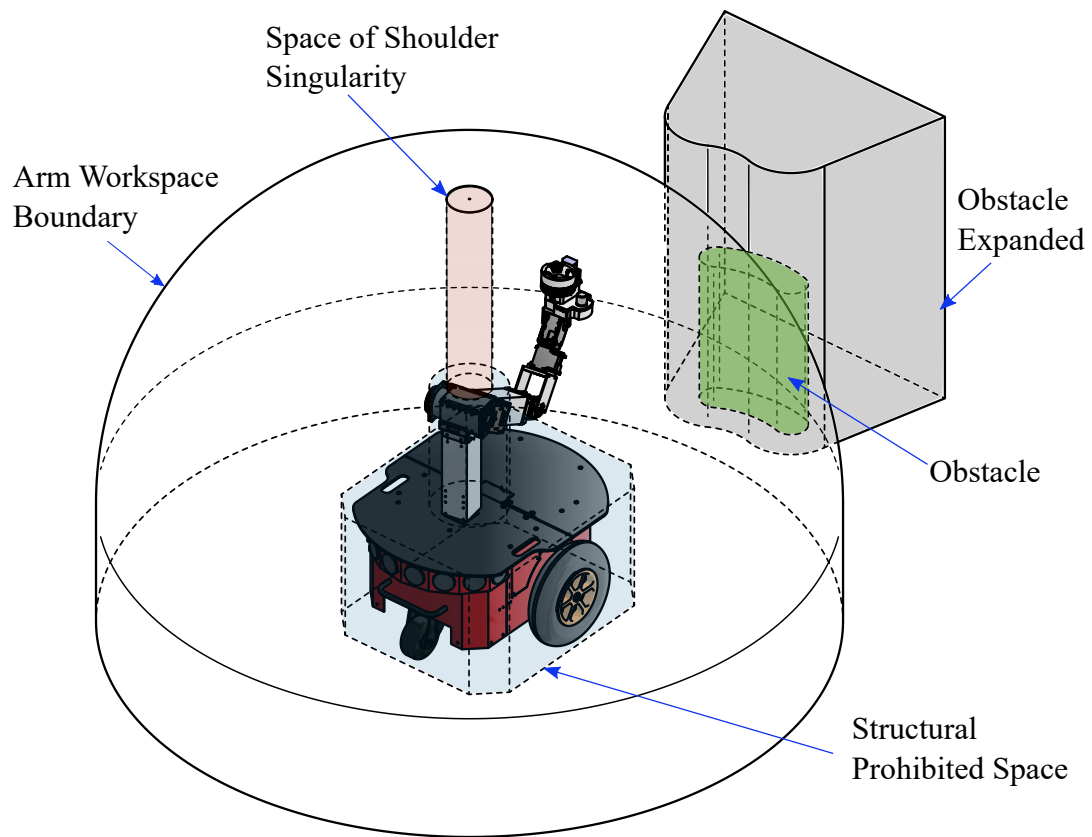


Fig. 3.8 Generated environment map with obstacles and prohibited areas.

Fig. 3.8). The complete group of prohibited locations in the map are thickened with a value of $a_N + T_s$ to compensate the major ellipse semi-major axis a_N and with an extra thickness T_s (see Fig. 3.9) so the obstacle avoidance has a safety clearance to ensure the vicinity positions will never reach the obstacle.

If any part of the vicinity touches a prohibited location in the map, the obstacle avoidance algorithm iteratively shifts the *cord* to the closest free position, and smooth its curvature until the *cord* is valid, and optimized as shown in Fig. 3.9.

After the vicinity is repaired, the *cord* does not have any contact with the obstacle, but the vicinity may intersect with the map obstacle. However, because the map contains expanded obstacles, the vicinity never intersects with the real obstacle leaving T_s length as a safety zone; all the points inside the vicinity are reachable by the end effector.

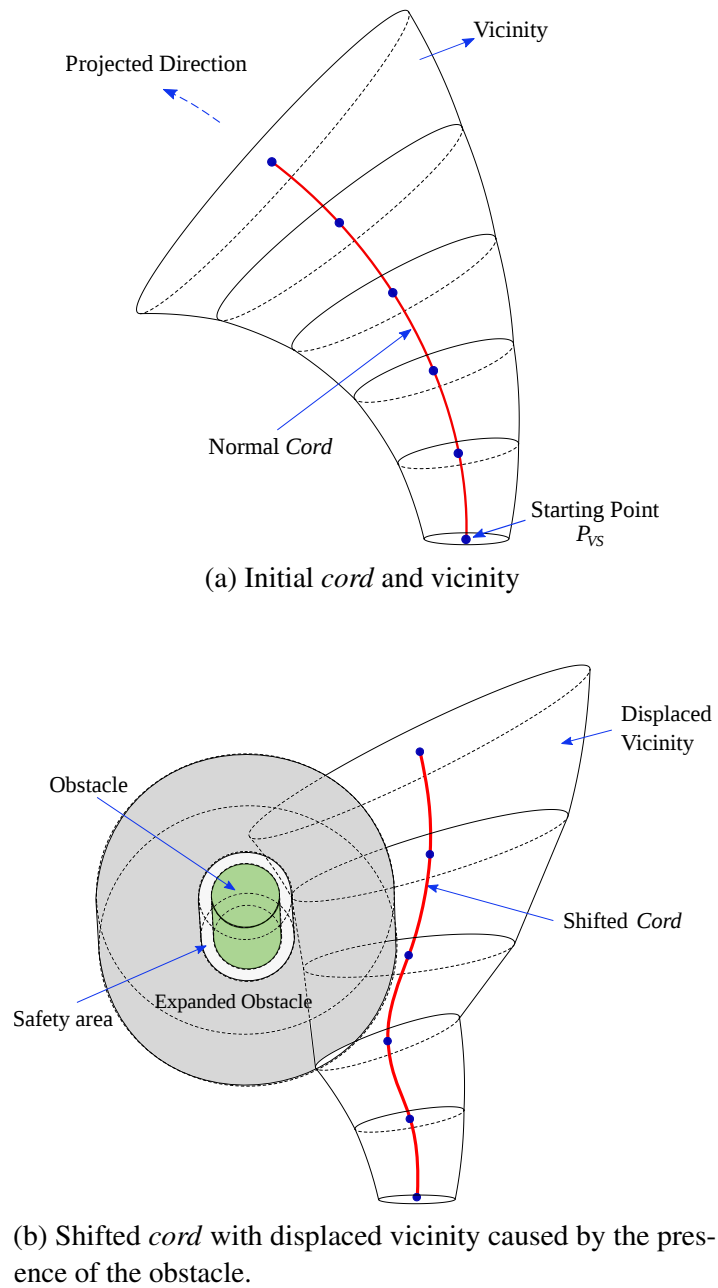
To add an extra level of safety, the vicinity is subjected to a quick validation process which is performed by checking scattered random poses within the vicinity. If these poses are not prohibited and they are achieved with allowed joint angles, the *vicinity* has a great probability to be valid, and can be used for the planning process.

The last step consists in the generation of a connected graph for end effector movements. There are many probabilistic approaches for path generation available in the literature, but in this research work, the Rapidly Exploring Random Tree Star (RRT*) algorithm proposed by [85] was chosen, as it is easy to implement and can quickly calculate a proper solution for small volumes.

The Graph Generation algorithm will connect valid nodes or vertices V with edges E in a hierarchical graph $\mathcal{G}(V, E)$. As shown in Algorithm 2, a set of random poses are filled with recycled vertices from the previous graph and new random poses, to be later evaluated for possible addition in the graph for each vicinity section S_i .

The random poses are calculated by generating quasi-random values around the origin within an ellipse of semi-axes a_i, b_i and L_s , and are rotated and translated with their angles $yaw_i, roll_i$ and $pitch_i$ and center point I_i of their respective ellipses. Thus, the complete set of random poses is within the vicinity, which means that they are generated in a space free of obstacles and constraints and there is no need of collision evaluation for each new pose added to the graph.

The next step is to confirm that each pose in the list of random poses are reachable by the end effector. As the RRT* graph should contain a dense sampling of poses for the vicinity, it is necessary to add several hundreds of valid vertices in every iteration. To speed up the whole process, the arm model and the Jacobian matrix are pre-loaded for parallel

Fig. 3.9 Vicinities and *cords* with obstacle shift.

Algorithm 2 *Path Planning for End Effector Movements*

Input $Vicinity(S_i, N), \mathcal{G}_0(V_0)$
Output $\mathcal{G}(V, E)$

```

1: for  $i \leftarrow 1$  to  $N$  do
2:    $Rand\_Poses\_List \leftarrow InitializeSectionRandomPosesList()$ 
3:    $Rand\_Poses\_List \leftarrow RecycleVertices(V_0)$ 
4:   while  $Rand\_Poses\_List.size() < MaxSectionSamples$  do
5:      $RndPose \leftarrow RandomPoseGenerate(a_i, b_i, L_S)$ 
6:      $R\_RndPose \leftarrow Rotate(RndPose, S_i.angles)$ 
7:      $RT\_RndPose \leftarrow Traslate(RotatedPose, S_i.center)$ 
8:      $Rand\_Poses\_List \leftarrow append(RT\_RndPose)$ 
9:    $Valid\_Section\_Poses \leftarrow IK\ Engine(Rand\_Poses\_List)$ 
10:  for  $j \leftarrow 1$  to  $Valid\_Section\_Poses.size()$  do
11:     $\mathcal{G}(V, E) \leftarrow append(RRT^*(Valid\_Section\_Poses[j]))$ 
12: return  $\mathcal{G}(V, E)$ 

```

calculation in each of the M computational threads of the *IK Engine* of Algorithm 3, each thread computes the Inverse Kinematics and verify the validity of the joint angles for random poses, and together they can retrieve results in small time and comply with the required path planning and optimization frequency f_B .

Finally, the RRT* algorithm as explained in [85] inserts each of the valid poses to the graph $\mathcal{G}(V, E)$, and connects its parent and cost with edges. This connected graph with dense sampling is used as a road map for end effector motions and with the help of the *selection* function, the best position to locate the end effector for tracking purpose is calculated.

3.3.1.3 Selection and Optimization

This process is designed to work at the high frequency f_B , providing real-time performance with an optimal end effector location. As the UAV tracking should be preserved in an environment with static and dynamic obstacles, the movements of the end effector should be fast and, if possible, acquire an anticipatory approach to be prepared in case of the risk of losing the line of sight with the aircraft. To do this, our approach forces the end effector to navigate in the planning graph $\mathcal{G}(V, E)$ contained in the vicinity, which is extended towards

Algorithm 3 *IK Engine***Input** *Rand_Poses_List***Output** *ValidPoses*

```

1: ValidPoses ← InitializeValidPosesList()
2: for M threads do in parallel
3:   Rand_Pose ← ExtractFirstElement(Rand_Poses_list)
4:   Joints ← InverseKinematics(Rand_Pose)
5:   isValid ← Check_Joint_Ranges(Joints)
6:   if isValid then
7:     ValidPoses ← Insert(Rand_Pose)
8: return ValidPoses

```

the main motion. Thus, the end effector has a higher probability to predict correctly the UAV relative pose in the short term.

The selection algorithm first uses the $P_Q^{\mathcal{M}}$ pose to calculate the end effector reactive target pose P_{VS} that locates the UAV marker in the center of the image. This calculation completes the process of Online-PBVS control. In order to determine how far the end effector should move for the next step, P_{VS} is used to guarantee a reactive tracking.

However, for the anticipatory approach, the velocity and acceleration vectors of the robot platforms are used to determine the general state of the tracking, which is a factor that determines how far the end effector will travel in the graph to predict the UAV movements. This factor is calculated by using the velocity vector of the UAV

$$\mathcal{V}_{UAV}^{\mathcal{M}} = [\Delta x_{UAV} \ \Delta y_{UAV} \ \Delta z_{UAV} \ \Delta \phi_{UAV} \ \Delta \theta_{UAV} \ \Delta \psi_{UAV}]^T \in \mathbb{R}^6, \quad (3.9)$$

which are compared to the velocities of the end effector

$$\mathcal{V}_{eff}^{\mathcal{M}} = [\Delta x_{eff} \ \Delta y_{eff} \ \Delta z_{eff} \ \Delta \phi_{eff} \ \Delta \theta_{eff} \ \Delta \psi_{eff}]^T \in \mathbb{R}^6, \quad (3.10)$$

both in $\mathcal{M}$ frame. These values are averaged to procure general indicators of the quantity of the movements of each platform. Then, the velocity value $\mathcal{V}_{tr} \in \mathbb{R}^6$ of Equation (3.11) is calculated as a single value that represents the quantity of movement expected in the image of the end effector camera.

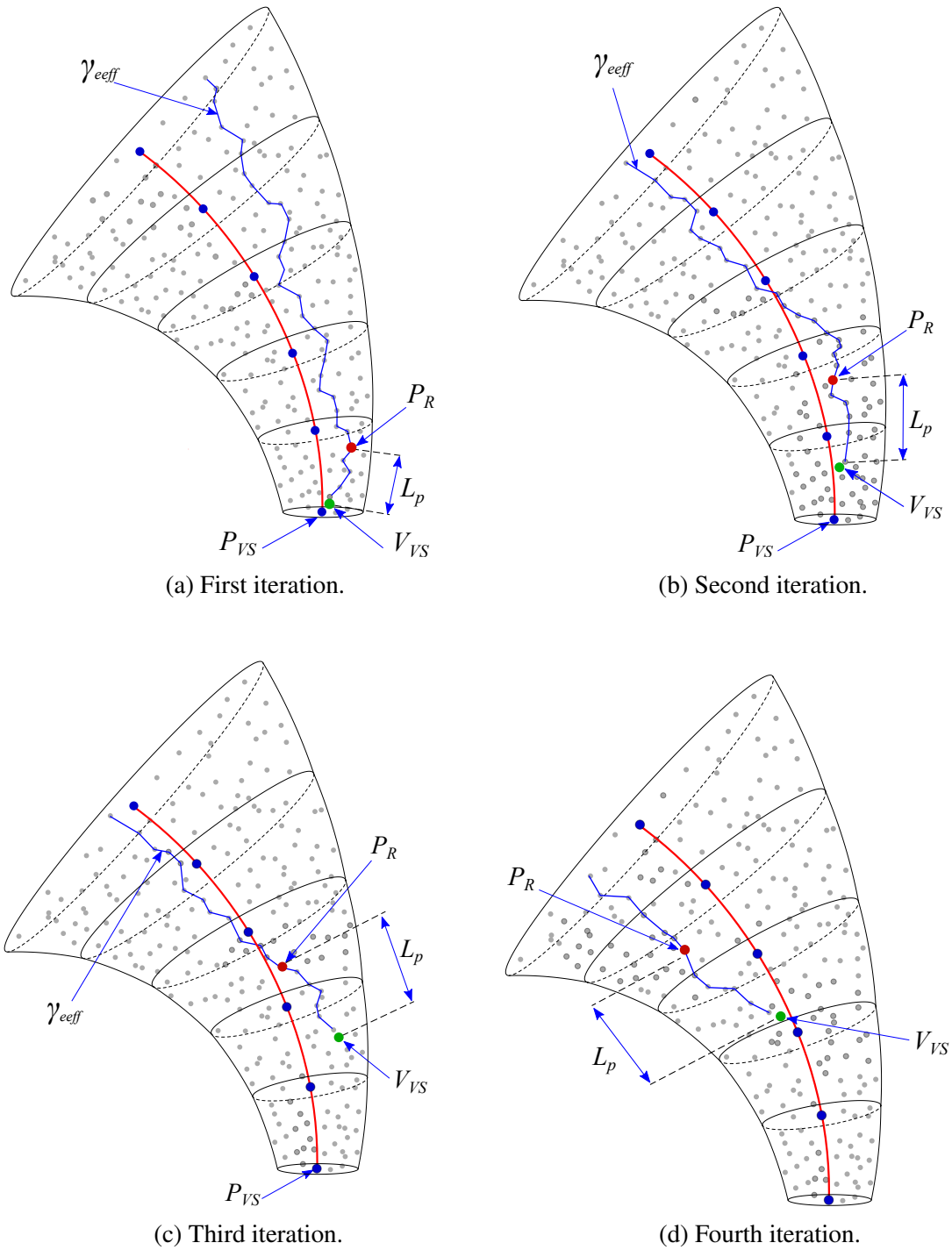


Fig. 3.10 Representation of the path calculation for the advance of the end effector position. The vicinities for quadratic *cords* are drawn along with their respective RRT* path γ_{eff} (blue lines), *items* (blue dots), connected vertices of the RRT* graph (gray dots) and the interest points for end effector movement.

$$\mathcal{V}_{tr} = \overline{\mathcal{V}_{eff}} + \overline{\mathcal{V}_{UAV}} \in \mathbb{R}^6, \quad (3.11)$$

with,

$$\overline{\mathcal{V}_{eff}} = \frac{K_{pos}[\Delta x + \Delta y + \Delta z] + K_{ang}[\Delta \phi + \Delta \theta + \Delta \psi]}{6},$$

$x, y, z, \phi, \theta \text{ and } \psi \text{ with } eff \text{ subindex},$

(3.12)

$$\overline{\mathcal{V}_{UAV}} = \frac{K_{pos}[\Delta x + \Delta y + \Delta z] + K_{ang}[\Delta \phi + \Delta \theta + \Delta \psi]}{6},$$

$x, y, z, \phi, \theta \text{ and } \psi \text{ with } UAV \text{ subindex},$

(3.13)

where the gain for position velocities is double the gain for orientation variances ($K_{pos} = 2 * K_{ang}$) giving more weight to the positional differences in the calculation of the tracking state $\mathcal{V}_{tr}$.

Note that the goal of the tracking process is to maintain the UAV hovering in a fixed position relative to the UAV, and to make the end effector camera to follow it keeping a minimal distance, this means that, in an ideal state both robots (UAV and end effector) should remain in this position and any variation should be corrected immediately avoiding deviations. With this in mind, a large value of $\mathcal{V}_{tr}$ means that it is being difficult for the tracking system to keep both robots in a steady position. Thus, the value $\mathcal{V}_{tr}$ is used as a general indicator of the tracking quality along the docking process.

Once a factor for the tracking state is ready, it is possible to derive the length of the path L_p through which the end effector will traverse. The L_p value is equal or less than L_C and inversely proportional to $\mathcal{V}_{tr}$.

$$L_p = K_{Lp} \frac{L_C}{\mathcal{V}_{tr}}, \quad (3.14)$$

where K_{Lp} is a constant that transform units and adjust the extension of movement.

The final step is to calculate a connected path γ_{eff} in the RRT* graph $\mathcal{G}(V, E)$ which connects a vertex V_{vs} with a point located at L_p distance as explained in Fig. 3.10(a). The value of γ_{eff} is updated at f_B frequency, with origin in V_{vs} which is the closest connected vertex to the visual servoing point P_{vs} . Note that V_{vs} is also updated at f_B frequency by

the *Merge and Selection* process, which increases the probability of providing a different path root every time. However, as mentioned before, if the direction of main movement changes suddenly, the *Planning and Optimization* process will work in a vicinity with unique ellipsoidal section to resume in the next iteration with a complete fresh vicinity.

The final end effector desired pose for each iteration P_R is the vertex in the path γ_{eff} located at a distance L_p from V_{VS} . This pose is optimized for tracking tasks, anticipating the direction of the main movement and taking into consideration the prohibited areas in the workspace and the state of the robot platforms.

Finally, P_R is transformed to joint goal angles q by calculating the inverse kinematic equations of the arm, and motor control signals are derived by using PID controllers for each of the six joints. The motors' commands are given as angle correction values that are later transformed into motor torques.

3.3.2 Contact

As indicated previously in Fig. 3.3, the *Contact Phase* starts with the authorization of the human operator and is executed simultaneously with the *Tracking Phase*. This process supervises the end effector motion for a slow vertical approach towards the UAV. The UAV should be hovering above the mobile manipulator at a reachable distance considering the maximum extension of the robotic arm, and it is waiting for the end effector approach. As this phase is activated while the *Tracking Phase* is in execution, the UAV motion is tracked even if the complete team is in movement.

Once the end effector is located at a smaller distance D_c from the UAV, the *docking device* starts securing sequence to attract and anchor the aircraft (refer to section 3.3.4). Once the aircraft is properly docked, the *Contact Phase* is finalized.

In this phase, the end effector is controlled in P_{RC} pose which is the same location of P_R but with the z value slightly changed at every iteration.

$$P_R = [x_{eff} \ y_{eff} \ z_{eff} \ \phi_{eff} \ \theta_{eff}] \quad (3.15)$$

$$P_{RC} = [x_{eff} \ y_{eff} \ z_{ceff}(t_c) \ \phi_{eff} \ \theta_{eff}], \quad (3.16)$$

where

$$z_{ceff} = z_{eff} + k_c \ln(t_c + 1.0) \quad (3.17)$$

Note that z_{ceff} depends on the duration of the *Contact Phase* t_c , using only the positive values of the logarithm ($t_c + 1.0 \geq 1$). Along with the gain k_c it generates a fast approach at the beginning but slower at the end of the *Contact Phase*.

Additionally, due to the UAV features, a new sensory constraint is added to the obstacle map. This constraint is caused by the sensor system mounted in the UAV that is pointing downwards, e.g. the sonar. This sensor measures the distance to the ground and produces a cone expanded from the UAV to the ground. If this cone is included in the map, then the end effector restricts its movements and produces safe docking without entering in this area during the complete process.

After the docking process is finalized, the UAV is fixed to the tip of the robot manipulator. For robustness and safety, the manipulator locates the aircraft chassis in a secure position for storage and transporting (see the rightmost image of Fig. 3.3).

3.3.3 Safety Functions

Due to the variability in the position of the UAV, the inference of external forces and the possible proximity to human agents, it is necessary to implement some safety measures to prevent docking failure or accidents. The human agent is acting as a supervisor for the UAV docking process, and as main operator for navigation; for all cases, emergency interruption is in operator level (higher level), and task interruption is in system level (robot team). The following are safety functions of the system, implemented with special care for execution in the real scenario.

1. If the UAV tracking is abruptly interrupted by external forces, or if the UAV is temporally not visible by the camera or any step of the motion planning is canceled, the value for z_{ceff} will decrease quickly to a low position to enlarge the camera field of view, increasing the probability to recover line of sight with the UAV marker and resume the normal tracking. For safety, the contact authorization is requested after any cancellation.
2. The end effector has a great strength and inertia, which can become dangerous for the operator, the robots and the environment. For this reason, it is necessary to implement levels of alert to handle failures. In the case of the UAV docking, two levels of alert were designed.

The first one is when the end effector is not in tracking state or is in resting, in this case it should remain in a low position and static, no other position should be allowed.

The second level is when a failure has happened, for example the motors are stalled, the UAV has crashed near the UGV, or a human operator is in imminent danger. In this case, the power of all motors of the arm should be cut, and the UAV should move away from the location or force a landing. This last level should have a special button in the operator console or a type of dead man's switch.

3.3.4 Docking Device

The docking device is a set of two complementary parts designed to lock the aircraft safely to the end effector. As shown in Fig. 3.11, *Part A* is attached to the tip of the end effector and contains an electromagnet and a set of levers. These two components are designed for attraction of the UAV and safe locking respectively.

The set of levers are actuated by the torque of a micro-servomotor that is previously transformed using geared arrangement. The levers are designed to hold the weight of the UAV chassis in any position and orientation².

The *Part B* is a lightweight adapter attached at the front-bottom of the UAV without interfering with its sensors. When *Part B* is aligned and with a distance less than D_c with the *Part A*, the electromagnet is activated to force an initial contact ($D_c = 0\text{mm}$). At this point, although the end effector is holding the UAV chassis, the UAV motors are still active, as the electromagnetic force is not enough to fix the aircraft especially if roll and pitch movements arise. To ensure that the aircraft is secured, the micro servo-motor is activated to move the gears and the levers to finally hold the external ring of *Part B* tightly.

After the mechanical locking is achieved, the UAV motors can be turned off, and the robot manipulator is in complete control of the aircraft chassis. Please refer to the sequence in Fig. 3.12 for a visual explanation.

In contrast, when the UAV is taking off from the end effector after being in the contact state, the docking device quickly releases the *Part B* in the adequate moment when the UAV motors have enough thrust to hold the flight.

The design takes into consideration the possible UAV movements and speed; the conical structure minimizes the effects of roll and pitch movements and allows small yaw variations. Moreover, the installed motor actuates the levers with considerable speed to quickly lock and unlock the UAV for a safe docking process.

²The levers hold the UAV initially in horizontal position with active motors, but for orientation change, the motors should be inactive to avoid strong torque forces in the docking device.

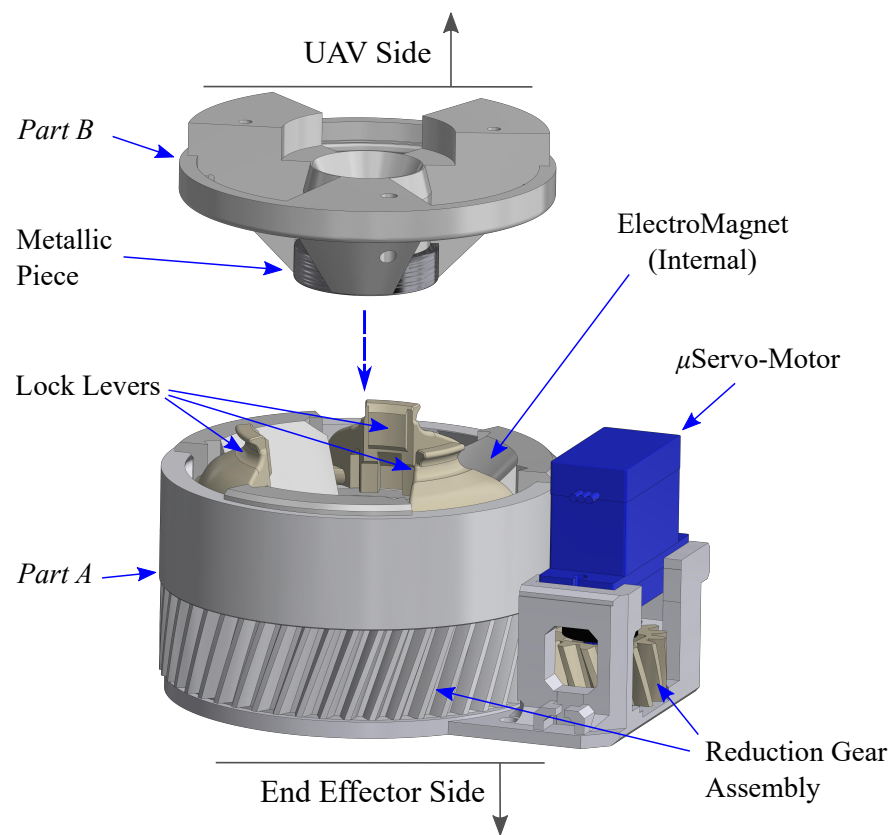
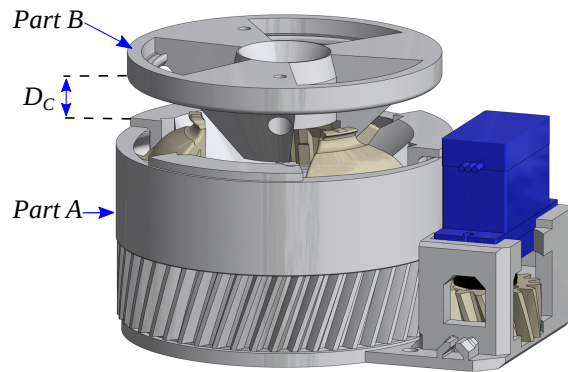
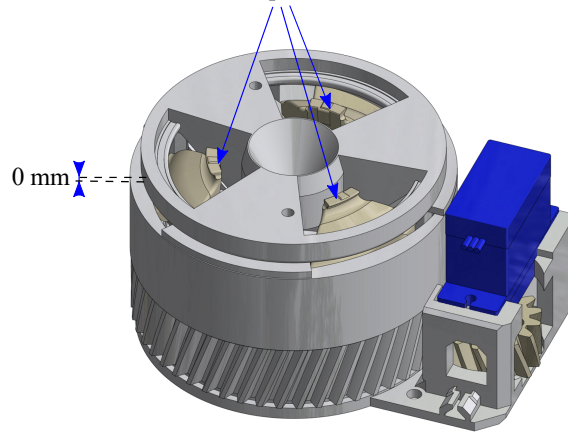


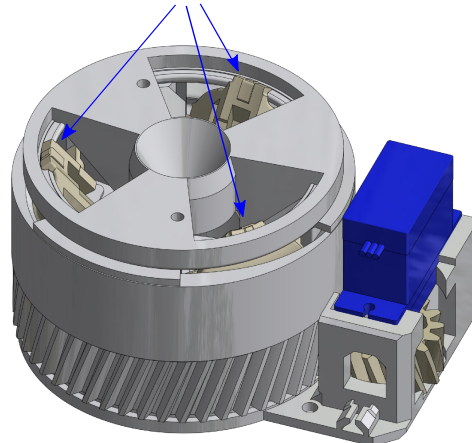
Fig. 3.11 Main components of the docking device.



(a) The *Part A* approaches *Part B* until D_C separation.
Levers in Open Position



(b) The electro-magnet is activated generating contact.
Levers in Lock Position



(c) The μ servo-motor is activated and the gear assembly moves the levers to lock the *Part B*.

Fig. 3.12 Representation of the UAV docking sequence with a close-up view of the docking device.

3.4 Summary

This chapter has described in detail the VTOL-UAV docking system for to the air-ground robot team. The environment and framework where the system can be applied was described along with the considerations that must be taken for execution of a complex mission with robots.

The robots to be used by the system were portrayed, and the features that enable their integration in a single robot team with heterogeneous capabilities were mentioned. Moreover, the attributes of each robot platform that facilitated the design process of the VTOL-UAV docking were described.

The VTOL-UAV docking system was described with all its sub-tasks, theoretic approaches, algorithms and hardware designs. Along the description, figures and visual explanations were supplied for a better understanding of the design. At the end, the complete system design was presented with all the safety considerations for a robust and repeatable docking process that can be applied to a heterogeneous robot team for mission solving. On the next two chapters of this thesis, the system is tested in simulated and real environment to prove the efficiency and robustness of the design.

Chapter 4

Simulation and Experiments

In this chapter, the heterogeneous robot team with autonomous docking capability is tested in simulated and real robots. Exploration strategies along with the details of the system that allows achieving the general goal are described. The results for both experiences are summarized.

The experiments are designed to prove the efficiency of the system during a complete USAR mission. Several measurements are taken from the sensors, including the distance traveled by each robot platform and, the relative position of the end effector and UAV marker during tracking and contact phases. To also elucidate a big picture of the efficiency and behavior of the VTOL-UAV during the mission, the energy consumption and the economy that represents the docking process in the USAR mission is examined.

For USAR missions, the indoor environment details are usually unknown by the human operator, there is no guarantee of the availability of a path that leads to the target, and there is no information about what kind of objects and obstacles are present in the scene. However, the main *target* features are well-known, and in some cases, a building map is available, and basic observation from the exterior of the building is possible. Thanks to the operator's expertise, it can be provided an initial notion of the location of the *target* and a preliminary route, as well as the general strategy for exploration.

The trajectory guidance for the UAV and UGV is assisted by the human operator for general exploration and rendezvous, while the *docking* process is executed as a fully autonomous task, taking control of the UAV and robot arm to achieve a robust contact.

During the mission, the UAV docking processes are required to respond to the needs of the complete team and the operator, then it is imperative to create a set of rules to follow

Table 4.1 Video URLs for simulation and real experiments

Video	Experiment	URL
1	Simulated environment	https://bit.ly/Docking_Simulation
2	Real environment	https://bit.ly/Docking_Real_Env

strictly. The following are the conditions defined to start a docking process with the designed air-ground robot team.

- When the UAV battery level falls below 20% or the distance with the mobile manipulator is becoming so large that is possible that can not be covered with the current battery energy.
- When the area is explored, and the current area has been surveyed with enough density in measurements.
- When the operator requests the docking.
- When the robot team is about to pass through a difficult area for aerial navigation.

If the conditions are suitable for UAV deployment, the human operator authorizes the UAV flight or deployment for aerial exploration to complement the ground assessment, after authorization the heterogeneous exploration begins.

The experiments for simulation and real environment described below are also explained with video resources. Readers are strongly encouraged to refer to the videos provided with this manuscript that can be found on the Internet for better understanding of the experiments. Table 4.1 contains the direct link to each video.

4.1 Simulation

The simulation of the system was used from the early stages of the design. Two main experiments were created along the design process to verify the feasibility of the system and prove the efficiency and robustness of the different features.

First, a simple simulation with generic robots within a simple scenario was created, then, as a next step, a complex environment with exact replicas of the available real robots and full docking functionality was simulated.

For both cases, the simulation platform used was the Robot Operating System (ROS, Kinetic Kame) and Gazebo simulator (v7.0.0) on a Linux distribution (Ubuntu 16.04.06

Xenial Xerus). The environment was installed in a desktop computer with 16 GB of RAM and a CPU with a clock speed of 3.6 GHz that reaches a maximum 4.2 GHz.

4.1.1 First Simulation Approach

The first advance towards UAV docking design was made with a simple simulation with generic robots and a basic environment. In this simulation, the robot team was transported in a square-like path to prove the capabilities of a simple UAV tracking and contact with only a reactive approach. This approach is the one described on the left side of the scheme in Fig. 3.5 and it does not include the predictive and optimization methods. It is used as a first stage of the complete docking system design to fine-tune the controllers, fix the coordinate systems, and analyze the limitations of the robot team.

The simulation presented here was executed without any complex environment features, like walls, obstacles, and friction with the floor or air strength variations. Only the weight of the robots with their collisions, and the UAV interaction with static air was considered. The UGV was teleoperated while the UAV was commanded to hover in a static point at a specific altitude [64].

When the UGV passed below the UAV with the end effector pointed upwards, the camera detected the UAV fiducial marker, and the tracking phase was initiated. After the first visual contact was achieved, the system controller started to handle the UAV position to maintain the marker in the center of the image, and at the same time, keeping the UAV position constant in relation to the mobile manipulator.

After the UAV tracking task started, the operator commanded the UGV to perform a series of curved movements and strong speed variations to test the limitations of the simple tracking algorithm. In this case, there was not used the predictive and optimization approach mentioned with the *Corded Local Path Planning* of Section 3.3.1.2; instead, it was used a simple approach with PID controllers without considering uncertainties and environmental variations.

After a relatively long path, the UAV docking was requested by the operator. The end effector began the slow approach towards the UAV maintaining visual contact and correcting the variations caused by the UAV movements. After approximately 16 seconds, the end effector reached the UAV position assigned for docking, at this point it was considered a docking was achieved.

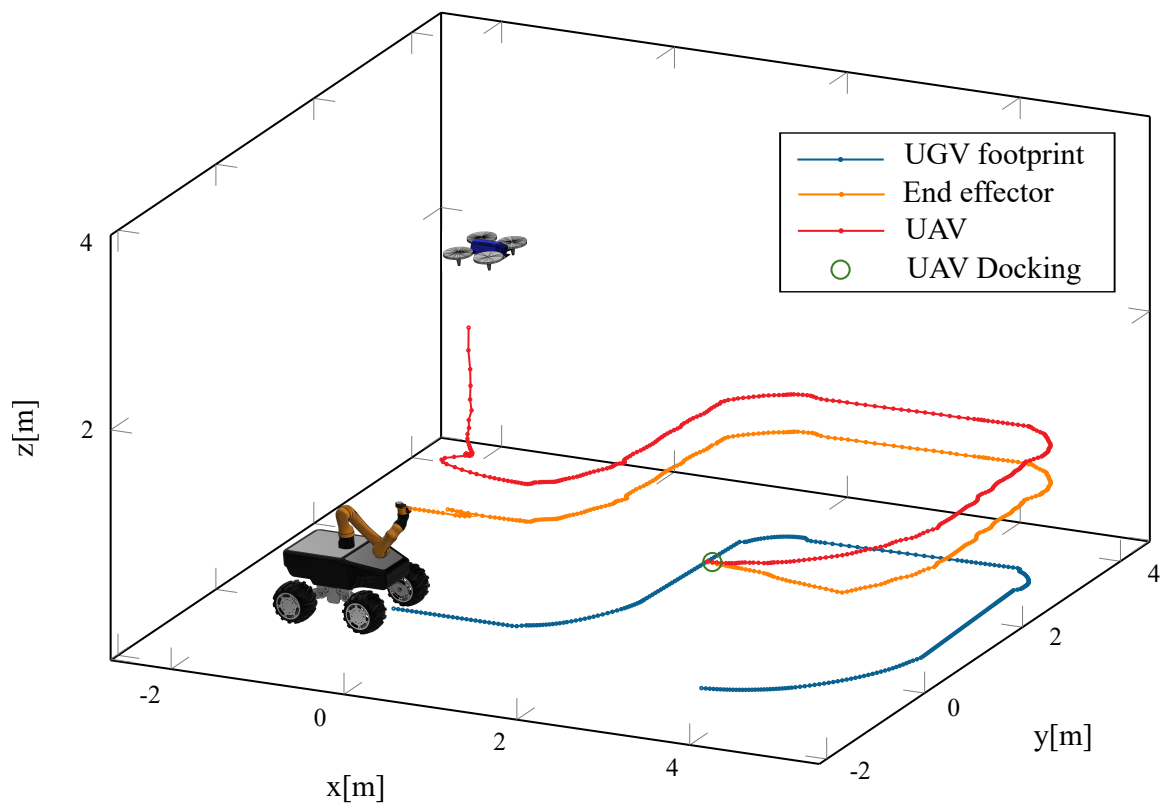


Fig. 4.1 Robot 3D paths for the exploratory simulation with reactive approach.

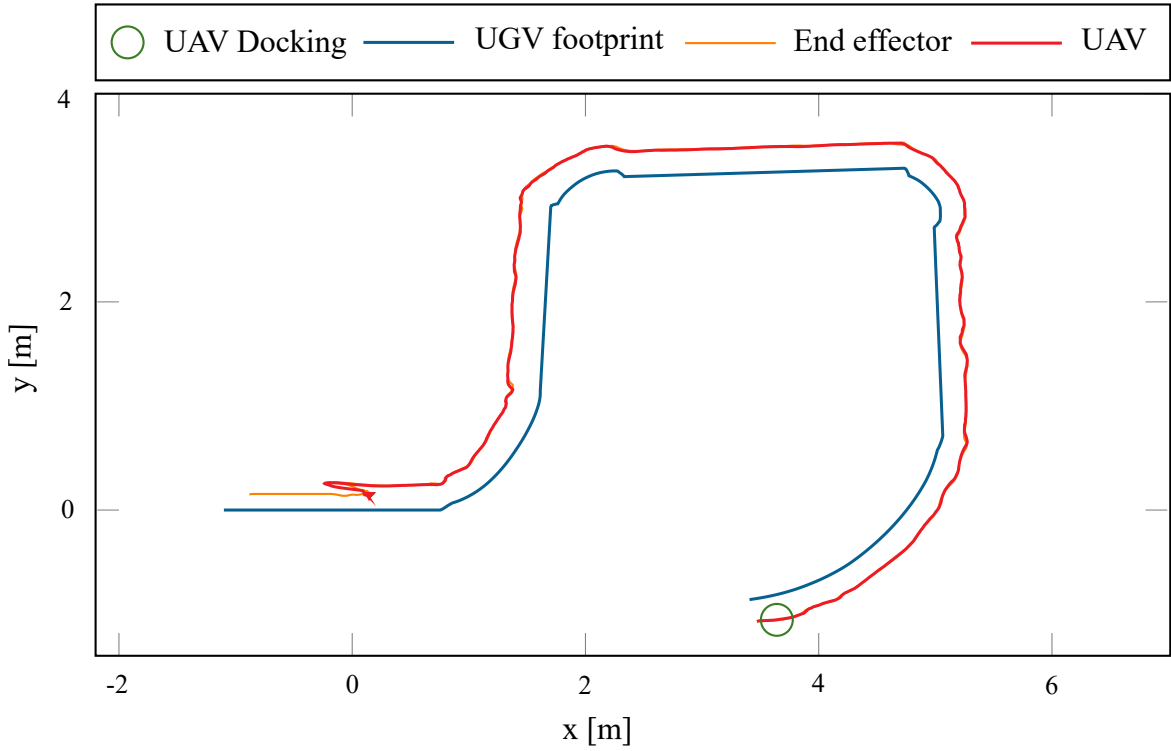


Fig. 4.2 Robot paths in 2 dimensions for the exploratory simulation.

In simulation, the contact and storage tasks were not applied because of simulator limitations and high complexity of implementation, but it was defined that the docking was achieved when the tip of the end effector reached a position very close to the UAV where the Part A and Part B of the docking device can make contact.

4.1.2 Results of the First Simulation Approach

The precise paths of each robot platform are displayed in Fig. 4.1 for 3 dimensional space and in Fig. 4.2 for a 2 dimensional view. The different variations in speed (separation of graph tips for 3D graph) and the curved paths for the UAV, UGV and the end effector of the arm are shown in the figures.

During the experiment, the contact phase was requested while the UGV was performing a turn. Such movements increased considerably the difficulty of the task, however, at the end the controller managed the situation and the docking was performed successfully. The final contact point is marked with green circle in the resulting figures.

Fig. 4.3a shows the position difference in x and y axes between the center of the UAV and the tip of the end effector. This measurement was taken from the variations of the position of the marker in the camera image. It is possible to see that there are large variations in x mostly immediately after the first visual contact with the UAV, but the differences remain small until final docking.

Fig. 4.3b represents the altitudes of the UAV and end effector during the simulated mission. At the beginning the UAV was in a high altitude but it descends to a constant right after the first encounter. This constant altitude is maintained until the end with small variations. Meanwhile the end effector tracks the UAV from a constant altitude to finally make a slow approach until reaching the UAV altitude.

The measurements in Fig. 4.3a are taken relative to the camera coordinates, while the measurements of Fig. 4.3b are with manipulator coordinates.

This simulation was useful to demonstrate the feasibility of the system in the early stages of the design. The controllers for the UAV, UGV and robot arm were fine-tuned to reach a correct UAV tracking and smooth and robust final approach.

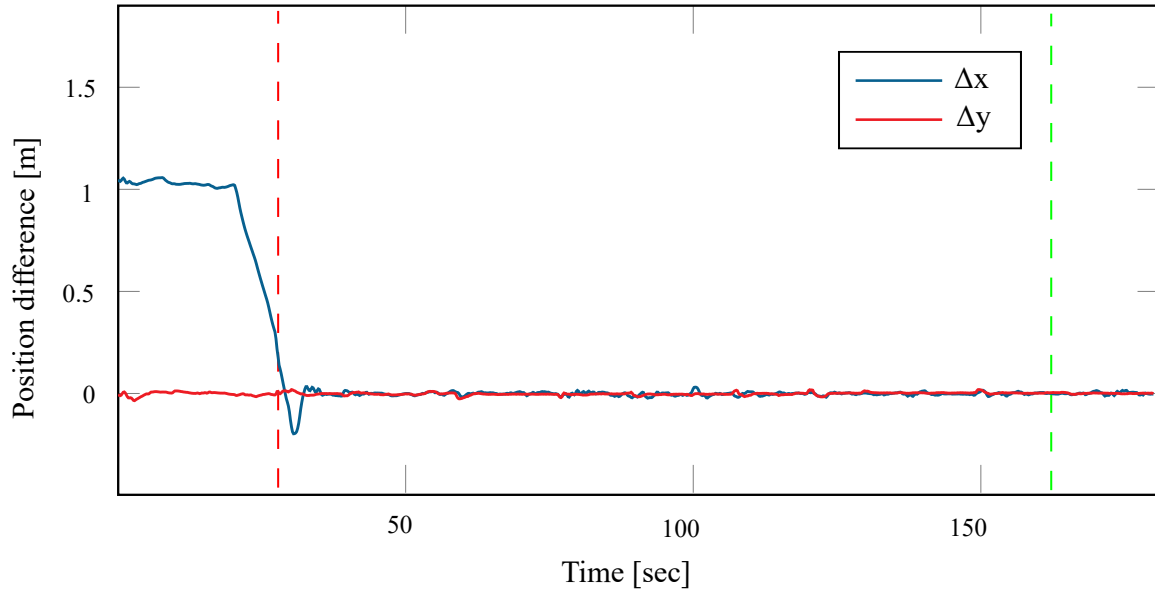
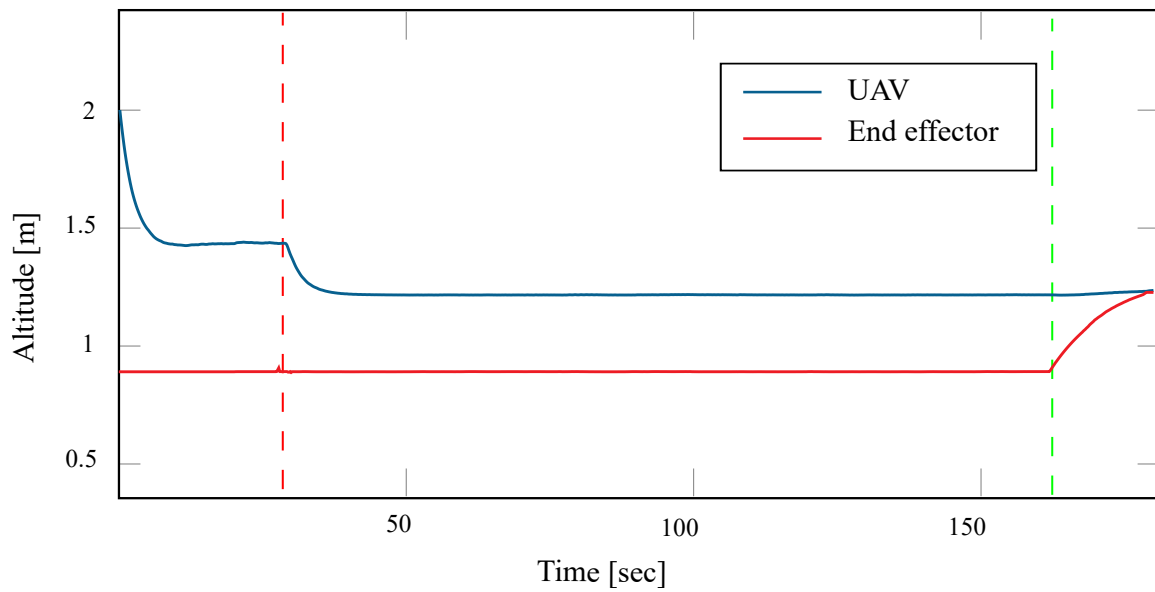
This initial experiment was applied in a controlled environment without the inference of environmental variables like wind, uneven terrain, robot imperfections, and temperature changes. It was concluded that the system has the potential to be implemented in a real environment, but the controller algorithms should be improved, and to achieve even better robustness, the inclusion of uncertainties and environment variables in the control loop is required.

4.1.3 Simulation of a Complete USAR Mission

The next step for the simulation experiments, is the application of the complete UAV docking process with a robust tracking described in Chapter 3. Also, it is convenient to design a simulated experiment with more complexity and a large environment that include obstacles and a building similar to a real USAR zone to test the system more thoroughly.

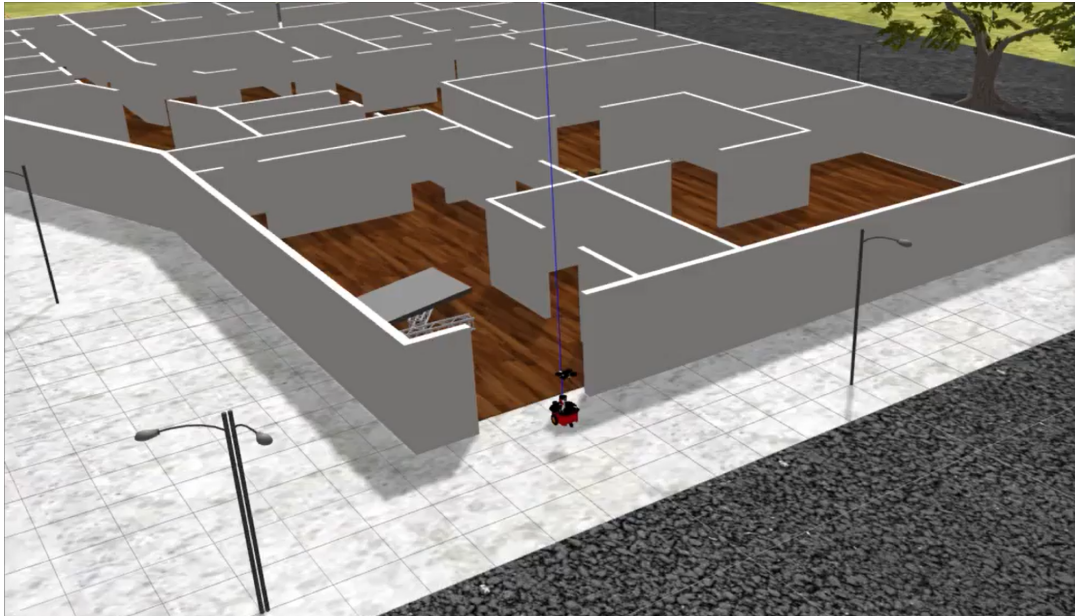
A screenshot of the simulated environment in Gazebo simulator with the building and robots is shown in Fig. 4.4a.

The main console for the simulation is shown Fig. 4.4b, in this figure, three windows with the UAV and end effector cameras with the basic drawing of the path planning and vicinities are at the left; on the right side of the same figure the simulated environment with the robot team during a docking process is displayed.

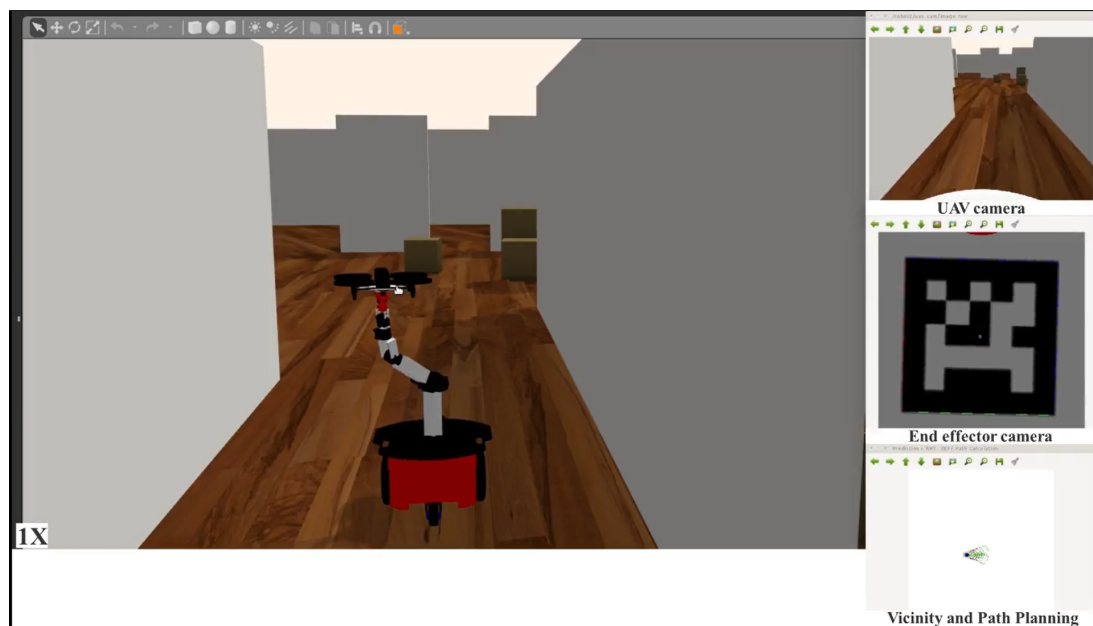
(a) Error graphs between UAV and end effector in x and y coordinates.

(b) Altitude of UAV and end effector.

Fig. 4.3 Position variations and altitude profiles for UAV and end effector for the first simulation with reactive approach. The red dashed line represents the beginning of the UAV tracking, and the green dashed line represents the authorization time where the UAV docking started.



(a) Simulated building and environment with the robots at the entrance.



(b) Main operating console with supporting camera and graphs visors.

Fig. 4.4 Building screenshot and main operating console for the simulation of the experiment in Gazebo simulator and ROS.

The robot team is deployed in an unknown environment to search for a target. The environment is a building with walls, obstacles, and floor textures. The map and the paths of the robots are shown in Fig. 4.5.

Initially, it is only known the approximate location of the *target* marked with the yellow circle in the map, and thanks to the human operator knowledge, a basic path γ_{mm} is initially given as a guide for ground navigation (dotted blue line).

With this initial knowledge, the mobile manipulator started the exploration following the initial path γ_{mm} in direction to the possible *target* location. Throughout the mission, γ_{mm} is modified because of the presence of obstacles and structures detected by the ground and aerial sensors to finally make the UGV robot to travel across the modified γ'_{mm} path (solid blue line). Along with the complete exploration of the environment, path smoothing is done to smooth out sharp corners of the UGV path, so the robot follows optimal trajectories for energy-saving and avoids structures that can cause interruption to the navigation [86].

Meanwhile, the aerial robot performs a series of movements to explore different building areas. It tries to follow a path γ_{UAV} that allows covering the available zones (solid red line). If possible, every D_e distance, the UAV stops and perform yaw variations to make detailed measurements of the scene in a static position. If any of the two robots find the target with its own sensors, the exact location of it is marked in the map, and the UGV path is immediately redirected to follow the new path γ'_{mm} that leads to the new goal.

As a necessary consideration, the UAV is restricted to fly to distant locations; it is defined as a maximum radius distance D_{team} measured from the ground robot, which is a safety measure. The radius is selected by calculating the approximate energy required to travel this distance and perform a docking process; the UAV should be able to return to the base and perform a docking with less than 20% of energy.

The UAV exploration continues until the target is discovered, or until the system or the operator requests a *docking*. In either situation, the UAV exploration is canceled and, it proceeds to move to a meeting point where it is detected by the end effector camera to initiate the *docking* process.

The complete simulation was recorded and made available in Video 1 of Table 4.1. The results of this experiment are summarized below with the corresponding figures and tables, and also with a proper discussion.

Table 4.2 Time information for simulation

Process	Time [s]
Docking 1	22.2236
Storage 1	62.254
Docking 2	26.2709
Storage 2	88.7853
Total mission time	723.25

Table 4.3 Simulation Data

	Mobile Manipulator	UAV
Distance traveled [m]	68.0523	112.8383
Operation time [s]	723.25	572.1461
Energy consumed [mAh]	-	986.2606
Energy recovered [mAh]	-	185.6458

4.1.4 Results of the Simulation of a Complete USAR Mission

In the simulation experiment, two *docking* processes were requested by the operator to protect the aircraft from passing through a corridor and a door frame. The locations for docking and take-off and also the interest points of the experiment are indicated in Fig. 4.5. The total time for contact and distance traveled by the robots are summarized in Table 4.2 and Table 4.3 respectively.

A very interesting information, is the UAV energy consumption during the mission. For its analysis, the power information of each of the UAV motors during the simulation is presented in Fig. 4.6. With this data, the accumulated energy consumption during the mission was calculated to draw the battery drainage process as is shown in Fig. 4.7.

Note that, during the time the UAV is attached to the end effector, the energy consumed by the aircraft is zero, and the battery level remains constant. However, if a charging system is installed to charge the UAV battery during contact, some of the energy can be restored which means the aircraft will have more autonomy to contribute to the mission.

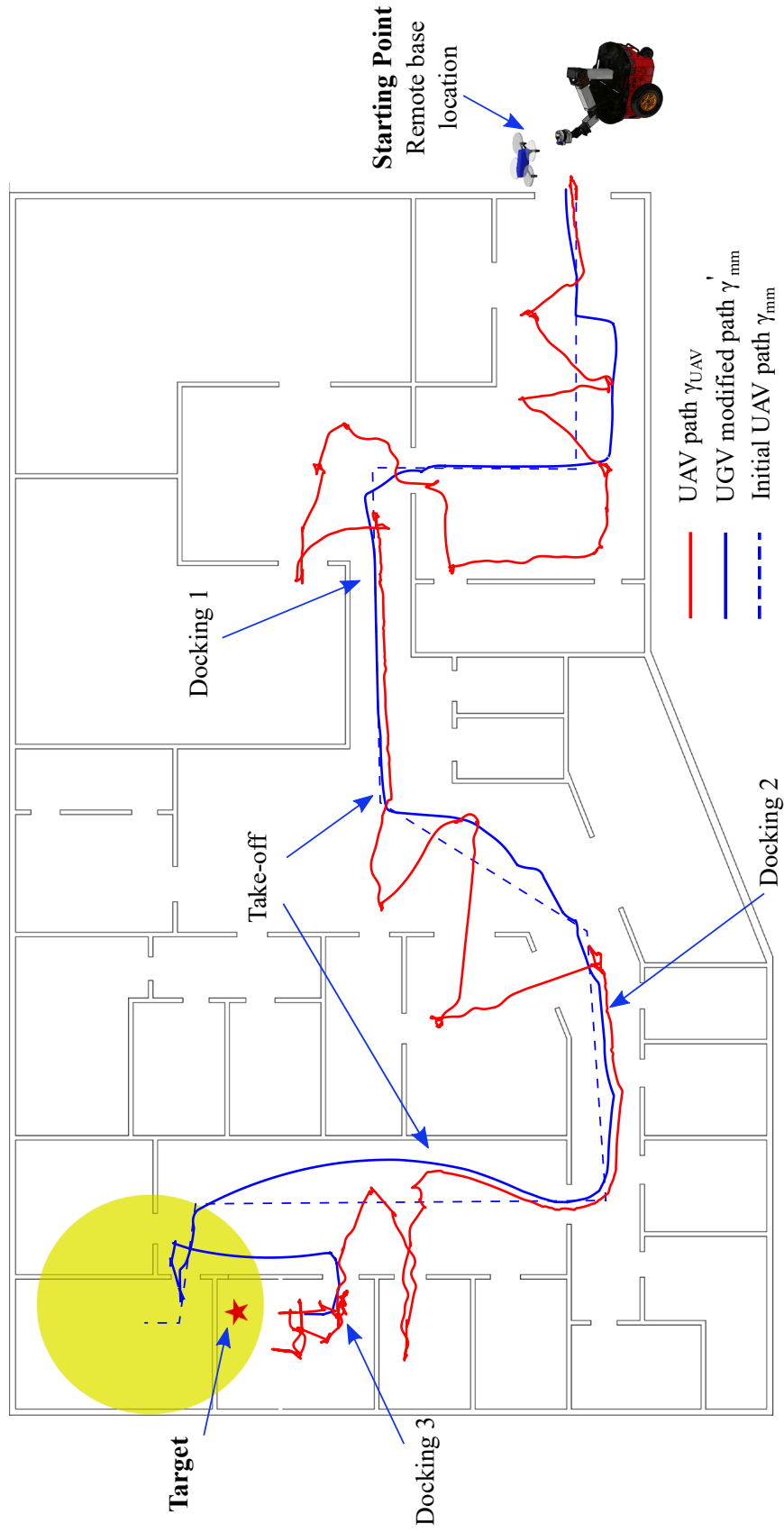


Fig. 4.5 Map and robot paths for simulation experiment. The yellow circle marks the initial knowledge of the possible location of the *target* and the gray areas are portions of UGV trajectory where UAV was in storage state.

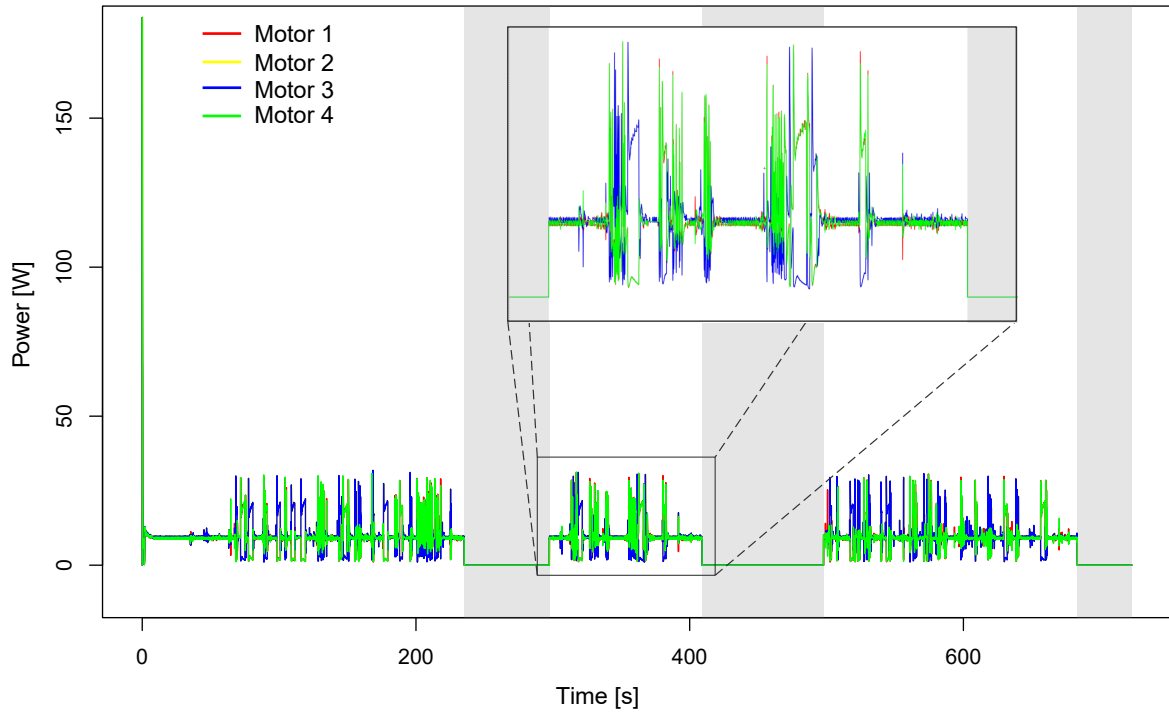


Fig. 4.6 Power used by the UAV motors during the simulation experiment. The gray areas represent storage processes.

The decrease in the percentage of energy along with the mission is shown in Fig. 4.7. The blue line represents the process without charging capacity, and the red line includes an energy recovery process installed with a standard charger of 12.6 V and 3.5 A. A 2700 mAh battery for the aircraft is considered, which sees its energy reduction to 63.47% at the end of the mission without charging process. Similarly, if it is included a charging process in the system, the energy would decrease to 70.34% at the end of the mission.

With these results for energy measurements, it is possible to say that by using the proposed docking system of this thesis, the UAV can recover near to 185.64 mAh of energy, which represents 6.87% of the battery capacity. This recovered energy represents a considerable amount of aircraft flight time that can be utilized for detailed inspection and other critical tasks providing crucial measurements to increase the usefulness of the robot team.

During the simulation experiments, the controllers were analyzed and tuned, and it was determined that the software designed to manage the robots for the UAV docking task was ready to start tests in the real environment.

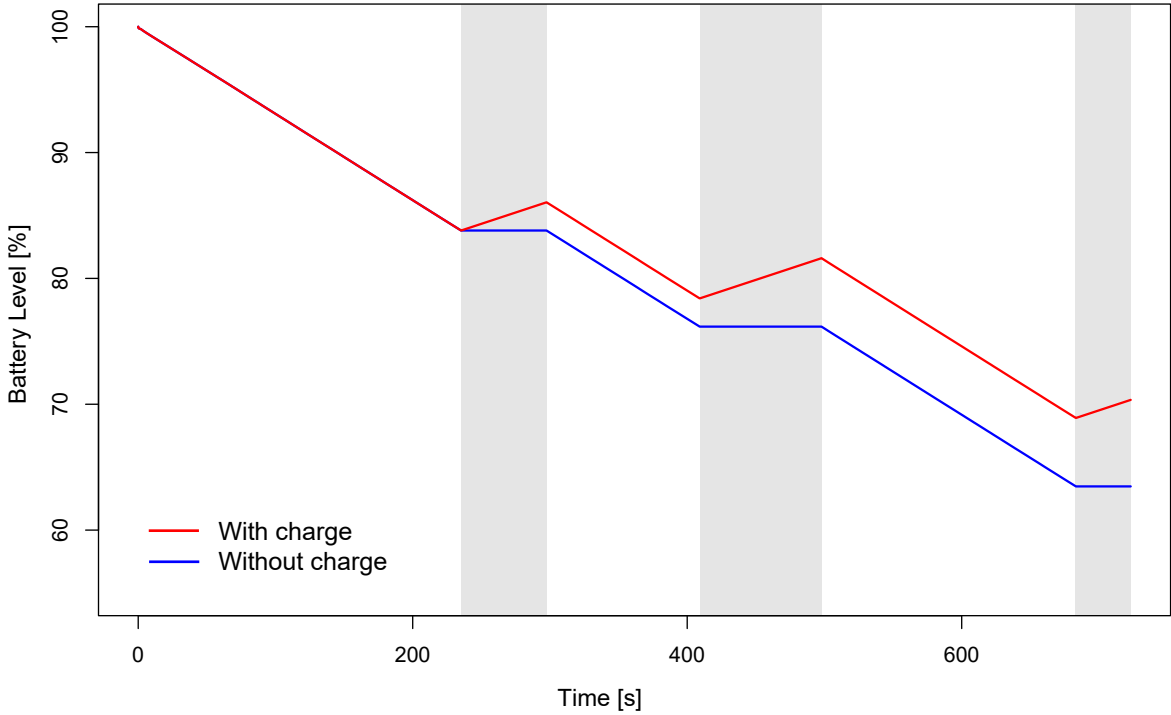


Fig. 4.7 UAV battery levels during the simulation. The gray areas represent storage processes.

4.2 Experiments in a Real Environment

In this section, a description of the complete experiment for USAR environment along with the details of the robot platforms, sensors and selected architecture are presented.

The executed mission is similar to the one in the simulation case, with similar assumptions and mission design, however, with a real environment that exhibits the challenges of external variables with some debris and smaller obstacles. The complete experiment was recorded and summarized in a video that can be found on the internet. Please refer to link number 2 of Table 4.1).

At the end of this section, the results for the complete mission with the corresponding analysis are summarized.

4.2.1 Robotic platforms

The UGV used in the experiment is based on the Pioneer 3-DX robotic platform by Adept MobileroBots. It implements differential drive motion with 10 KG payload capacity and 7 hours of runtime. It is equipped with a 2D range LiDAR (Sick LMS111-10100) and an RGB-D camera (Realsense D435) facing the front of the platform.

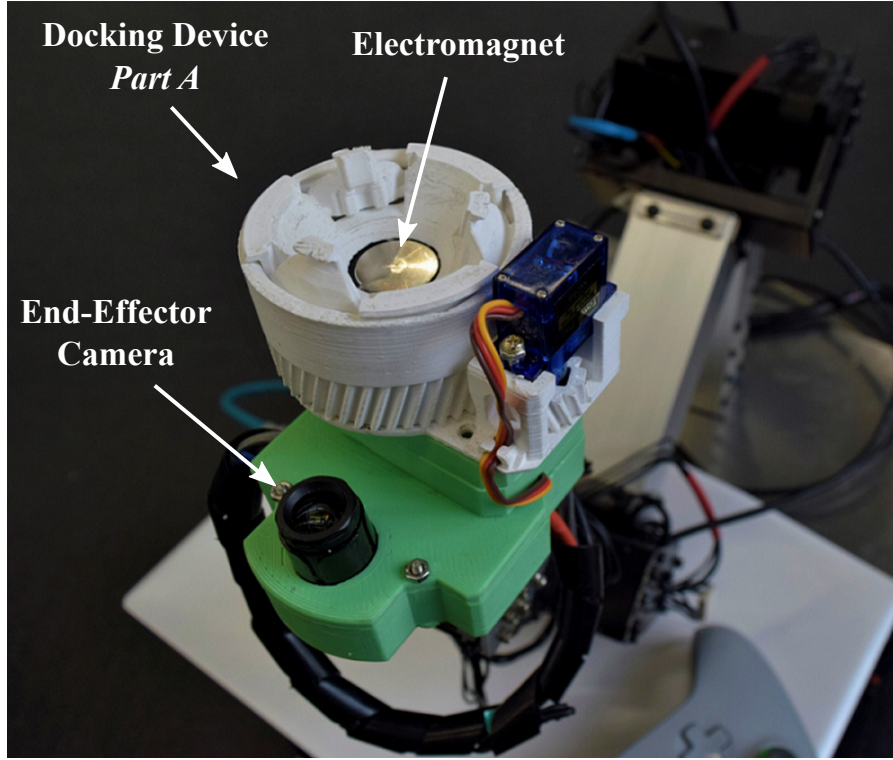


Fig. 4.8 End effector of the robot arm. The green part corresponds to the end effector base and camera arrange, the white portion is the Part A of the *Docking Device*.

To complete the configuration for the mobile manipulator, a 6 DoF robot arm was designed from scratch with high precision Dynamixel motors (Pro and MX series) as rotational actuators for each joint. The first links of the robot arm were designed with extended longitude; this conception provides a broader vertical range which is essential for the end effector approach in *Contact Phase*. Please refer to the Fig. B.1 and Fig. B.2 of the Appendix B for technical drawings of the robot arm, where the precise dimensions, motor references and other details are shown.

The *Docking Device* prototype described in Section 3.3.4 is attached at the 6th joint of the manipulator (see Fig. 4.8). It is fabricated with 3D printed parts of Acrylonitrile Butadiene Styrene (ABS) material and assembled along with a standard micro servo-motor (positional rotation), geared arrange, three levers, and a small size electromagnet (12V - 250N). Also, the end effector camera (PS Eye 640×480 pixels at 60 Hz) is mounted in the arrangement of the 6th joint with an offset of $O_C = [0.045, 0.01, 0.03]$ m. See Fig. B.3 and Fig. B.4 of Appendix B.

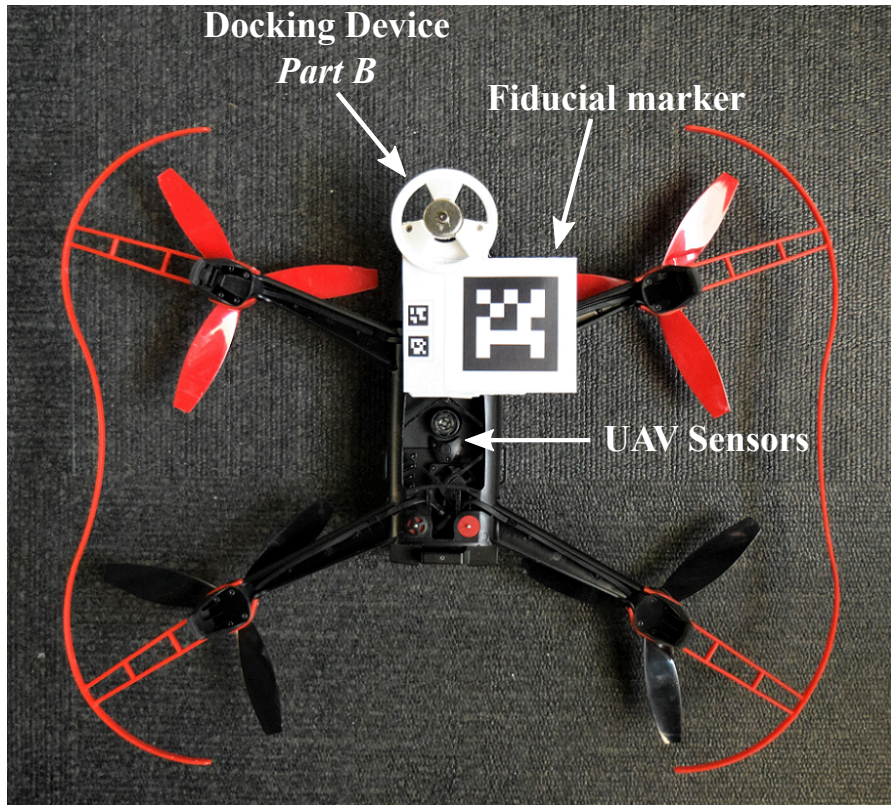


Fig. 4.9 Bottom view of the UAV.

The aerial robot platform is based on a commercial quad-rotor, Bebop 2, by Parrot Company. It has an autonomy of 20 minutes (battery of 2700 mAh). It is equipped with a stabilized full HD camera for detailed exploration and with a set of sensors for its stabilization system (IMU, barometer, accelerometer, gyroscope, vertical camera for optical flow, and ultrasound sensor). As shown in Fig. 4.9, the Part B of the *Docking Device* was installed in the UAV front bottom without interfering with its sensors, it was also 3D printed with ABS material and it contained metallic material in the center. The UAV fiducial marker was also installed at the bottom of the UAV, designed as a bundle arrange of AprilTags [87]. It was decided to use this type of arrange, because it allows to combine markers with different sizes for detection in medium distances (large markers) and small distances (smaller markers). The Appendix A contains more information about the detection library and the used bundle for the real experiments (Fig. A.1).

For safety and stability, the speed of the aircraft was limited to 4 m/s in lateral displacements and 1 m/s for vertical variations. By its side, the UGV longitudinal speed was limited to 2 m/s, and each of the robot arm joints to a maximum of 15 rpm. During the docking

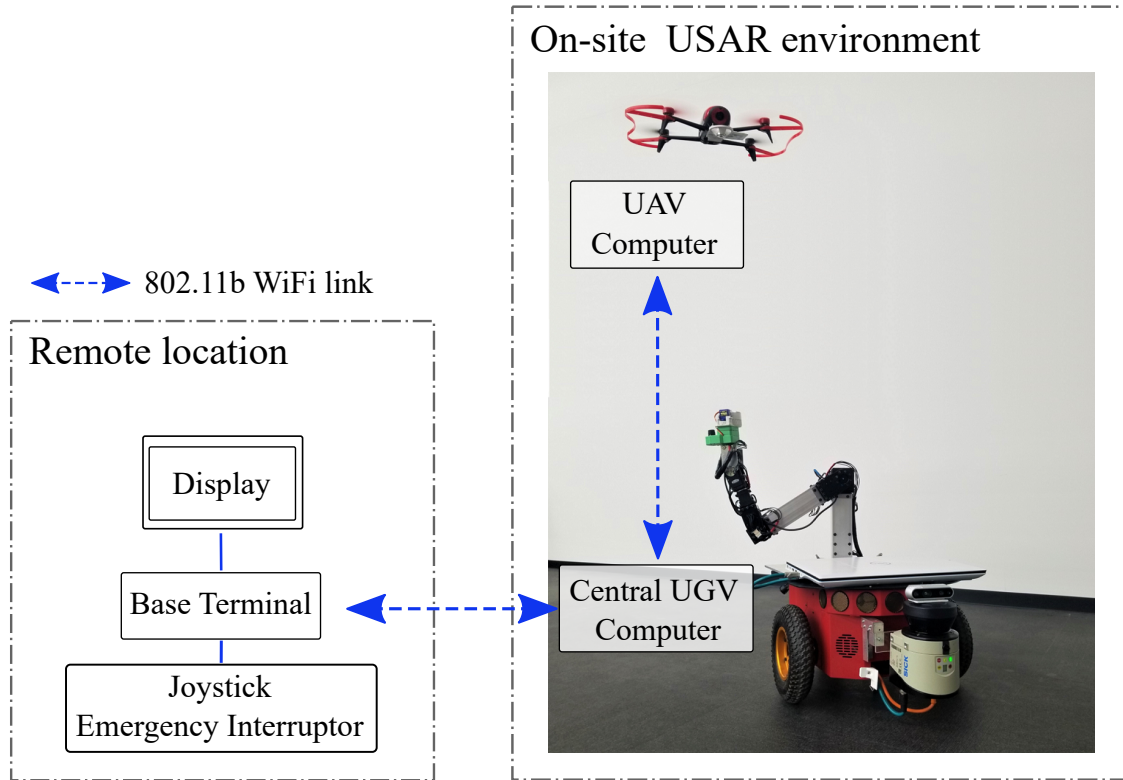


Fig. 4.10 Communication network between robot team and operator.

process the UGV speed was limited even more to a maximum of 0.5m/s to avoid disturbances and to perform the docking process along a shorter distance.

The complete robot team platforms used in the experiment are photographed in Fig. 3.2. Also, as early mentioned, the detailed technical drawing of the designed robot arm and the docking device can be found in the Appendix B.

4.2.2 Architecture and Network

The architecture of the robot team is centralized around the mobile robot manipulator. This means that the significant part of the computation process is realized by the onboard UGV computer, while the UAV computer systems are in charge only of the activities that require immediate action, like the stabilization for its movements and self emergency monitoring. The rest of the tasks, like navigation, mapping and task allocation are performed in the UGV central computing unit.

This architecture brings a great advantage, because the information of the onboard sensors are processed immediately in the UGV without communication delay. For example the image

of the end effector camera is immediately analyzed in the main processing unit and the commands for the arm joints and UAV motors are quickly available.

On the remote operator side, there are only supervision and tele-operation tasks available, and there is no sharing of processing power between the team and the base. The camera visual information comes from the robot team to maintain the operator informed of the state of the robots, but only commands for navigation and emergency indication are sent back from the operator to the robot team.

All the information is communicated through Wi-Fi 802.11g channels between the robot team and the operator base and also between the air and ground robot platforms. A breakdown of the communication network and basic architecture design is portrayed in the scheme of Fig. 4.10.

4.2.3 Experiment for a single VTOL-UAV Docking

As a first step of experimentation in a real environment, a series of short tests were performed only for the UAV docking task. The purpose of this phase was to check the behavior of the real robots under control of the designed system.

The robot team was prepared to execute a single docking in a controlled environment. First, the test was for robots in static position with manual UAV approach¹, next, the UAV was put in flight for a slow docking with several speed restrictions that were slowly relaxed. After the first tests were done correctly, the mobile manipulator was set in motion with various velocities combinations to test the tracking under difficult conditions.

With each test, all the controllers were fine-tuned, and the positions of the end effector along the process were improved. More importantly, the limitations of the robots with the docking system implementation were exposed which gave a unique opportunity to improve the system.

A complete docking process from the first initial encounter to the final contact and storing was photographed in Fig. 4.11. In this sequence the process was executed quickly, the tracking and contact were done correctly while the UGV was moving following a straight path with low speed. Additionally, a sequence for UAV deployment is presented in Fig. A.2.

After the improvement of this phase, the docking system was robust and repeatable facing different situations. It was determined that the system was ready for deployment for real USAR mission solving.

¹The UAV was deactivated, and the hovering was simulated by holding the chassis with a string.



(a) Initial Robots Encounter



(b) UAV stable tracking



(c) UAV stable tracking



(d) Starting of contact phase



(e) Contact phase



(f) Docking



(g) Transition to storage position



(h) Storage

Fig. 4.11 Docking sequence in controlled environment from initial robot encounter to final storage; the UGV is in motion with low speed.

Table 4.4 Controller Parameters

Parameter	Value	Units	Description
N_{OP}	4	-	Number of old UAV poses P_{VS} stored in the OP vector.
L_C	15	cm	Approximate total length of the cord.
N	13	cm	Number of <i>items</i> in the cord.
L_S	1.2	cm	Separation between <i>items</i> in the cord.
ΔP	6	mm	Maximum distance of movement per iteration allowed for the end effector.
δ	0.1	mm	Small increment for cord calculation.
RRT* Poses	500	-	Approximate number of random poses calculated on every iteration of the path planning.
f_B	4	Hz	Frequency of operation for the Planning and Optimization block (blue part).
f_O	8	Hz	Approximate frequency of operation for the Merge and Selection block (orange part).
D_C	7	mm	Approximate distance of separation between <i>Part A</i> and <i>Part B</i> for electromagnet activation.

4.2.4 Controller Parameters

As a result of the fine-tuning process done during the simulation and the initial experiments, the controller parameters for the real scenario were defined. The main parameters are summarized in Table 4.4.

Note that according to the frequency of iteration f_O of the *Merge and Selection* process and the number N_{OP} of past measurements P_{VS} , the previous data used for the regression is extended around 0.4 to 0.8 seconds before the current UAV position.

Furthermore, it is also included the ΔP value that is used internally by the RRT* algorithm to define a maximum motion allowed for the end effector in each iteration. Additionally, in the same table are included the average poses produced every iteration by the *IK Engine* for graph update and the different values for cord calculation.

With these values defined and checked, the controller behaves robustly enough for a stable tracking and docking operation with real robots. Thus, it is possible to proceed with the experiments for a complete USAR mission in real scenario.

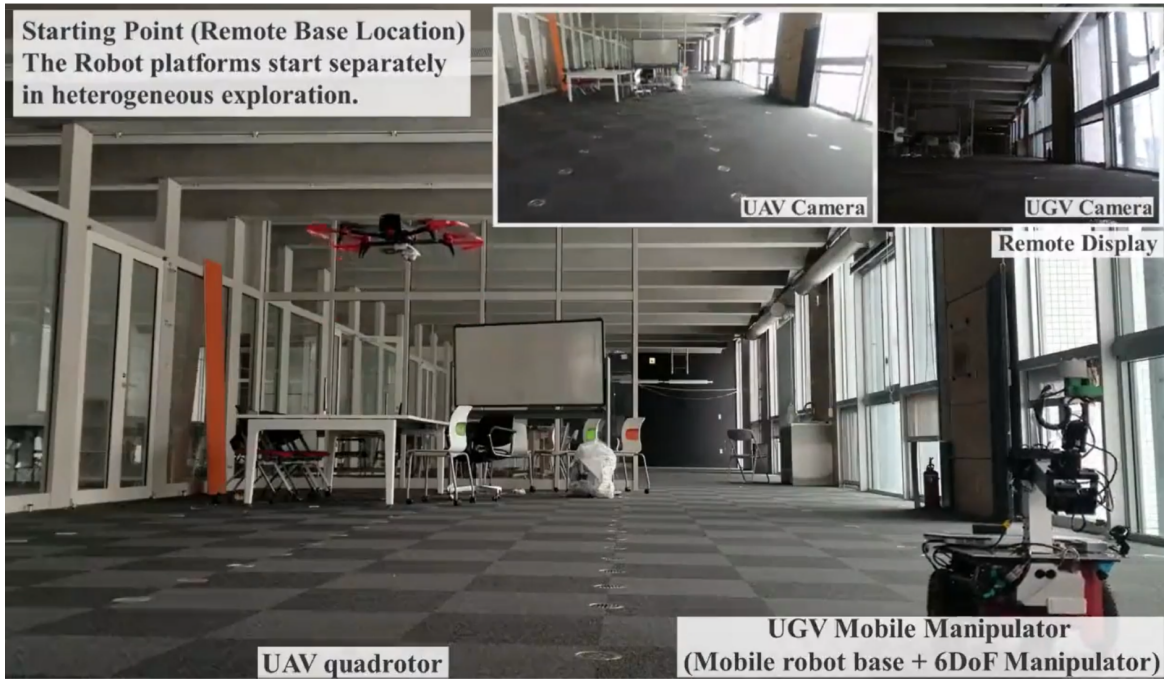


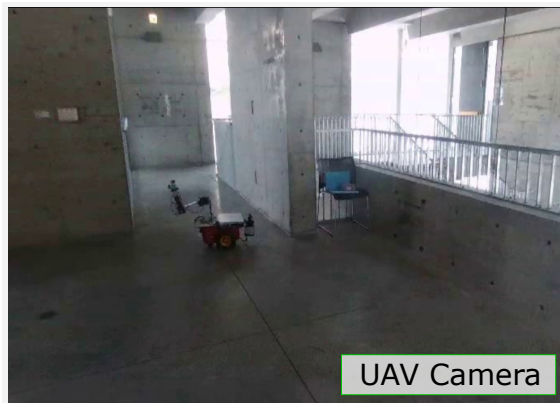
Fig. 4.12 Photograph of the starting point (Remote base) for the experiment in real environment. It is also shown the UAV and UGV camera image that is the information seen by the operator.

4.2.5 Experiment Design for a Complete USAR Mission

The robot team was deployed at the university architecture building entrance as shown in Fig. 4.12. The building contains rooms, long corridors, and an empty hall with some obstacles. A target is located in the depth of the building.

The robot platforms start exploration separately, and similar to the simulation case, the goal of the robot team is to find a *target* in the less time possible. It is assumed that the operator only knows roughly the possible location of the *target* in the building. However, it is not known if there is an available path to reach it.

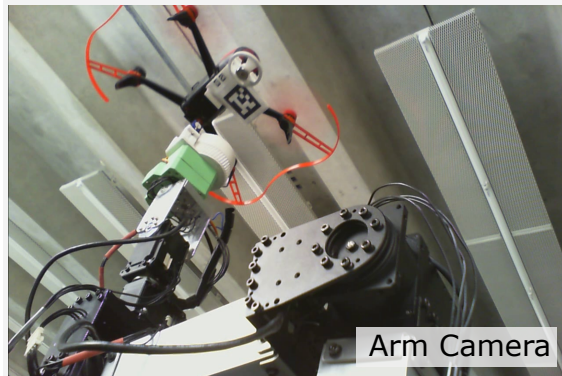
It is also assumed that the building has particular risk or hazardous elements. Therefore the human operator was required to supervise the robot platforms from a remote location. The available camera streamings for the operator are shown in the screenshots of the Fig. 4.13. The aerial and ground robots were moving autonomously along a series of set-points given continuously by the operator, and if necessary, it was possible to take total control of the movements of both the platforms to preserve the safety of the complete team.



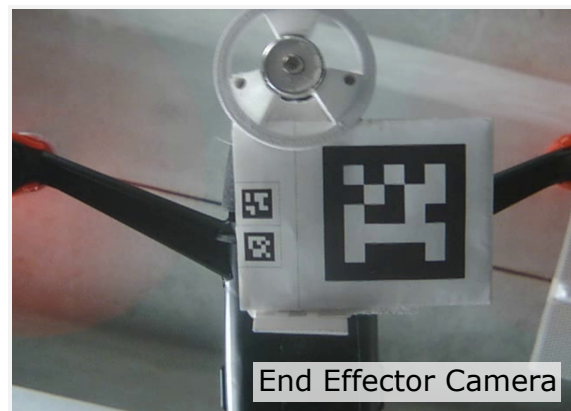
(a) UAV exploration camera.



(b) UGV exploration camera.



(c) Arm monitoring camera.



(d) End effector camera.

Fig. 4.13 Screenshots of the available camera streamings for remote operation of the robot team.

During the experiment, the robots followed a trajectory in direction to the given set-point. The human operator selected each set-point continuously depending on the direction of the target and the zones to explore. In the case of the UAV, the human operator often needed to take control because it was necessary to pass over frames, avoid walls, direct the camera to the point of interest, and correct trajectories due to bad odometry. Similar attention was required for the UGV movements, especially for turning points, complicated obstacles, and path corrections.

The robot team began the mission separately and during the exploration executed 2 autonomous docking processes and one UAV deployment. The ordered outline of the experiment is given below.

1. The robot platforms started exploration separately. The UAV tried to register as many number of areas as possible while the UGV went directly towards the potential target location γ_{mm} .
2. During UAV exploration, a door frame with some obstacles that represent danger was detected. Also, the human operator determined that the risk was high because the door led to a narrow corridor. Consequently, the first docking was requested by the operator.
3. The UAV moved to a meeting point to wait the UGV to pass beneath. The autonomous docking process was executed successfully and the ground exploration continued with the UAV in storage position.
4. After some time with ground exploration, the human operator found an large open space visible from the UGV camera, and requested UAV deployment.
5. The UAV was deployed and the heterogeneous exploration started again.
6. After some time, the UAV found the wanted *target* and alerted the system and the human operator.
7. The UGV changed trajectory γ'_{mm} in direction to the target found by the UAV.
8. The human operator decided that the UAV exploration is no longer necessary and requested a second docking process.
9. The UAV was directed to a second meeting point to wait the UGV. A second autonomous docking process was executed and the UAV was stored again.

10. The UGV recognized in detail the *target* and completed the goal of the mission. The robot team returned to the base at the entrance of the building with the UAV in storage position.

Once the experiment was finished, the data was compiled to analyze and quantify the behavior of the robot team in the mission. In the remainder of this chapter, the resulting data is presented, and the advantages of the use of the UAV docking processes in the real scenario are discussed.

4.2.6 Results for Experiment in Real Environment

During the mission, the data of the sensors was used to compile a 2D map of the reachable parts of the building. Fig. 4.14 shows the grid map of the environment built specifically with the front Lidar and RGB-D sensors of the UGV. The same figure also displays the trajectories of the robot platforms during the experiment with solid lines for the paths during *target* search and dashed line for the returning path of the complete team.

For this experiment, the yellow area painted in Fig. 4.14 represents the possible location of the *target* where the UAV tried to explore thoroughly, this was the only information known by the operator. In the same figure, the photographs of the relevant locations for UAV docking and deployment are shown. Also, the starting point and target location with the corresponding location in the map are indicated.

Fig. 4.14 and the results of this experiment might be more understandable if they are seen together with the video resource provided (Video 2 in Table 4.1), as mentioned before, the readers of this thesis are encouraged to refer to the supplied video resources.

After the *target* was recognized, the robot team returned to the base with the UAV stored in a safe position. For the returning process, the aerial exploration was not necessary, as the team already known that a safe path is available, and the ground sensors with the human supervision were enough to get the team back to the initial point. The complete team reached the initial base safely without any energy concern.

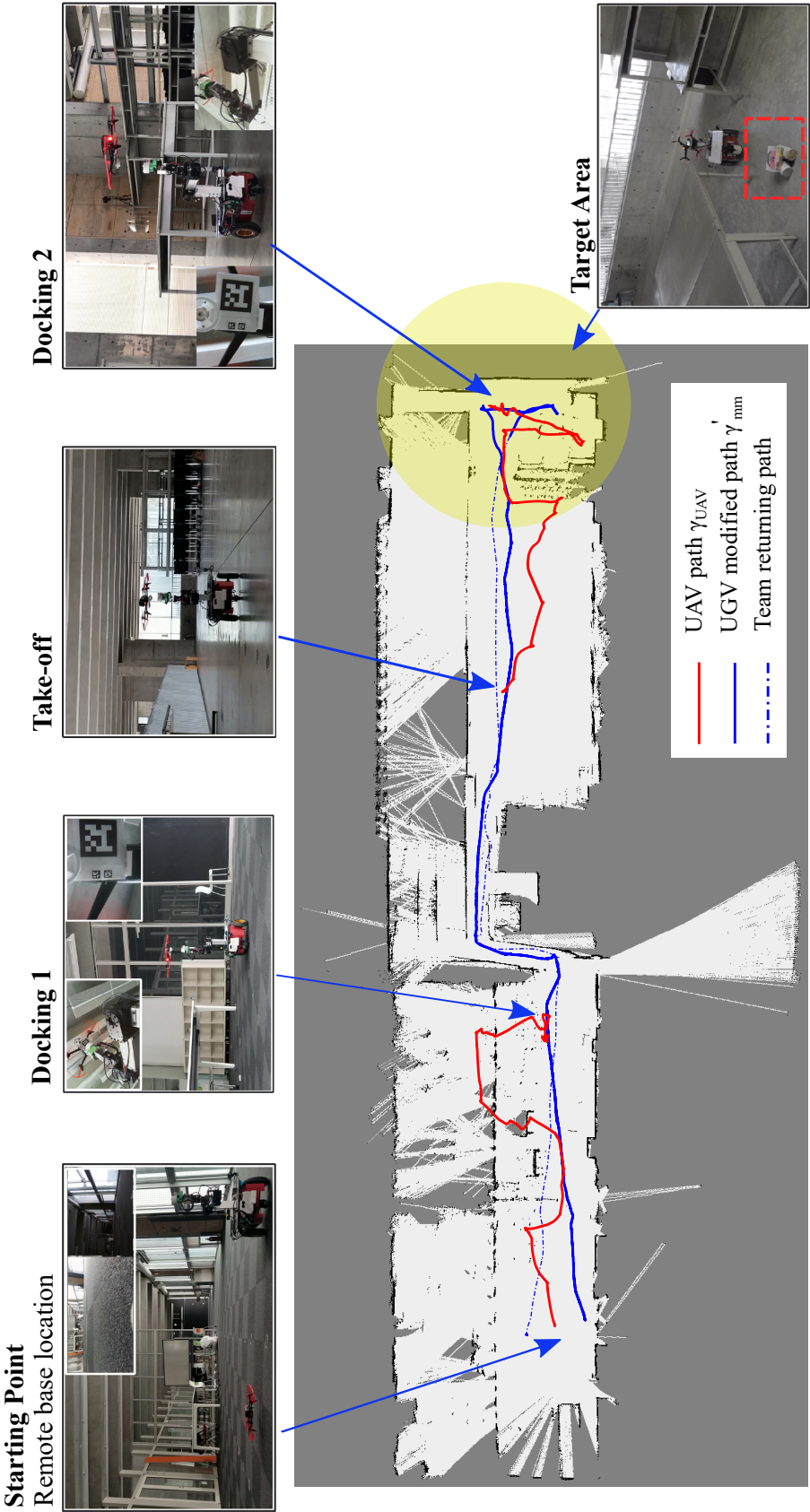


Fig. 4.14 Robot team paths for real robot experiment. The yellow area represents the initial operator knowledge about the location of the *target*.

Table 4.5 Time information for experiments in real environment

Task	Time [s]
Docking 1	24.76
Storage 1	141.6
Docking 2	21.72
Storage 2	369.9
Complete mission	1110

Table 4.6 Experiment data

	Mobile Manipulator	UAV
Distance travelled [m]	170.076	98.53
Operation time [s]	1110.0	599.25
Energy consumed [mAh]	-	1840
Energy saved [mAh]	-	1368

Table 4.5 summarizes the time required for the complete experiment and for each docking and storage processes, and Table 4.6 shows the total distance traveled by the platforms and the energy used by the UAV with the economized energy achieved thanks to the use of the UAV docking process.

Fig. 4.15 shows the position of the UAV and the end effector for each axes x , y and z , during the first docking process in real environment, while Fig. 4.16 draws the same for the second docking process. In these graphs it is possible to see the exact position of the robots and get more details of the motion that are not visible directly from the naked eye. Also it allows to elucidate the behavior of the controllers and the delay of the control loop during the task.

Fig. 4.15 and Fig. 4.16 show the graph of position of the marker and end effector during the two docking processes; all the data printed in the graphs are in manipulator coordinates.

In Fig. 4.15, both the docking processes were performed correctly, and the end effector followed the UAV marker. Although the line of sight was lost in some instant of the first docking, the tracking was quickly recovered and stabilized, and stable contact processes were achieved, as can be seen from the plots of the UAV and end effector trajectories.

During the autonomous UAV docking processes, the system calculated viable end effector locations to preserve the line of sight with the UAV marker and performed a smooth motion tracking. In some instants, due to structural constraints, the system modified the local area shape to supply viable locations (see z -direction of Fig. 4.15).

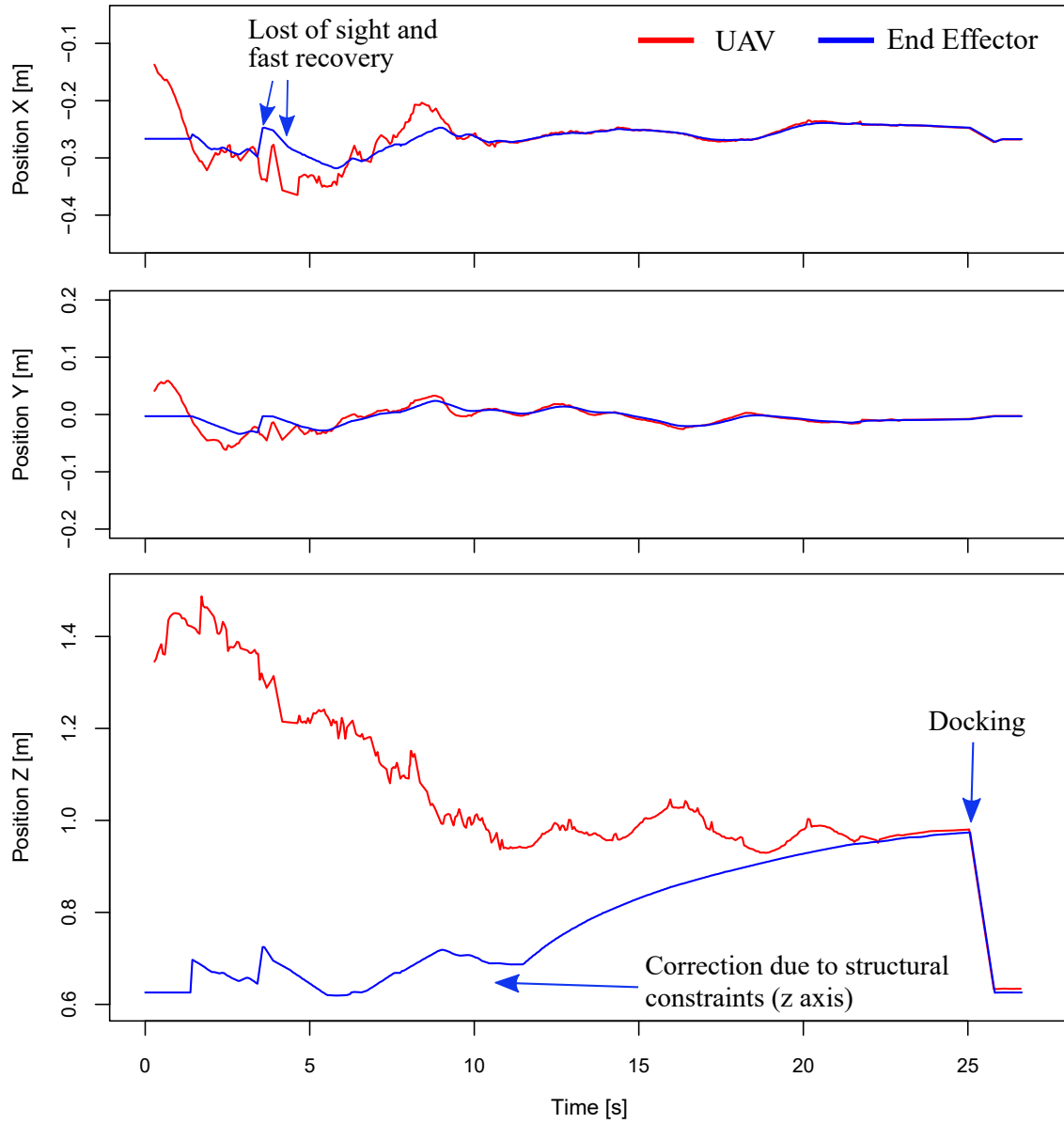


Fig. 4.15 Positions of the UAV and end effector during the first docking process in real environment.

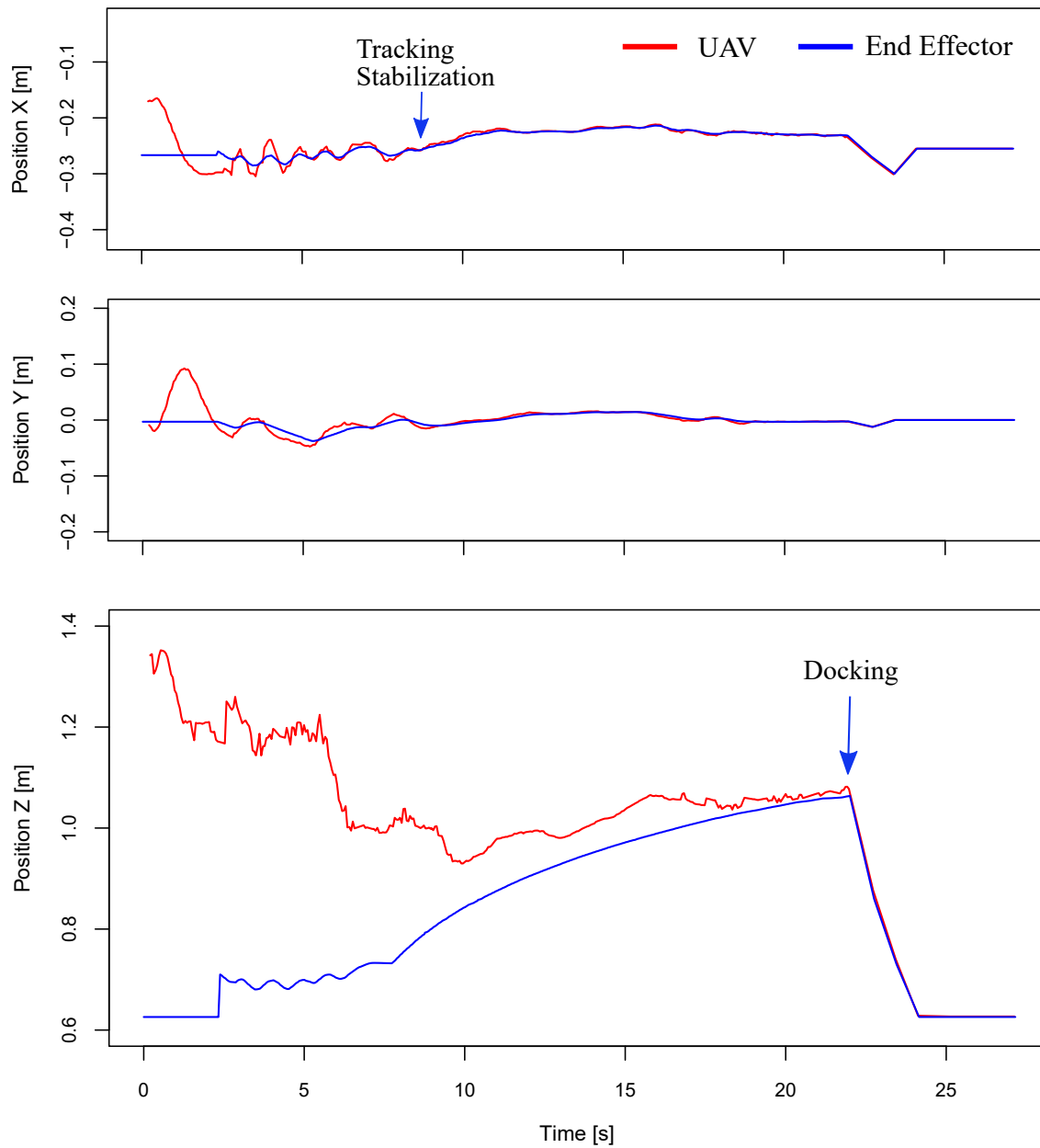


Fig. 4.16 Positions of the UAV and end effector during the second docking process in real environment.

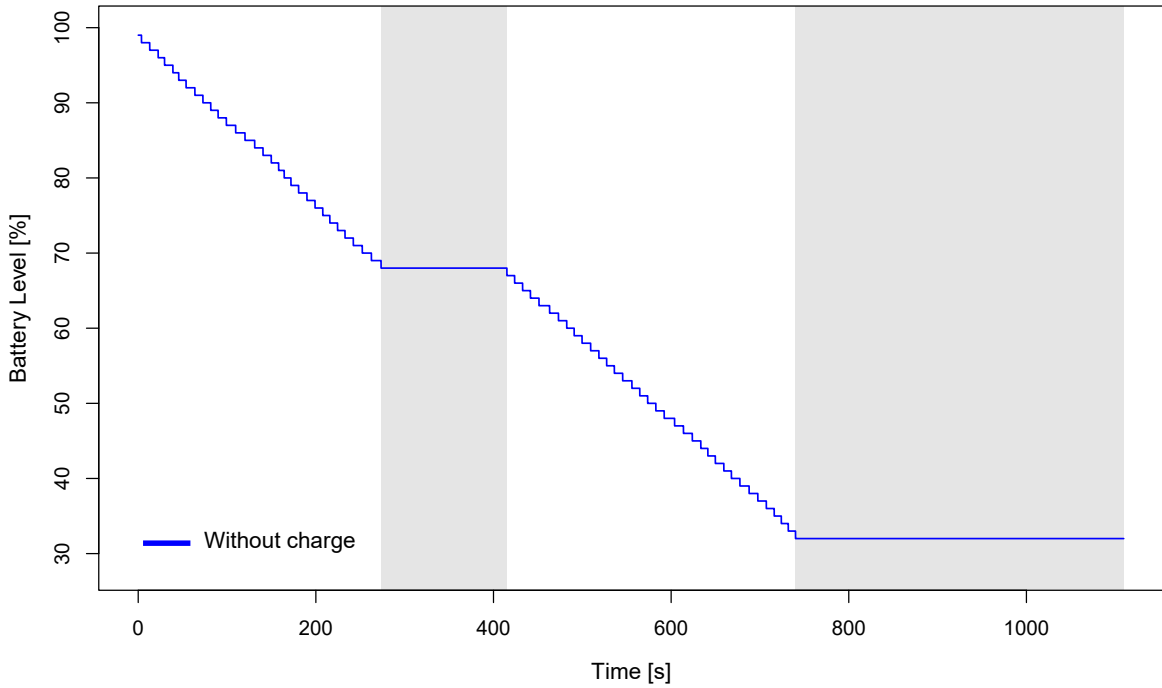


Fig. 4.17 UAV battery level during the experiment. The gray areas represent storage processes.

The UAV battery level during the experiment is plotted in Fig. 4.17, where it is possible to see the drain of the battery power during UAV operation and energy saving during storage. Note that, in the figure, the battery level is drawn with values rounded to integers, because this is the way the sensors of the used robot gives the battery information.

The energy consumed during the experiment drained the battery of the UAV to 34%, but as expected, the energy consumed by the aircraft was almost none during the storage state as indicated by the gray zones in the plot (see Fig. 4.17). If we extrapolate the energy consumption, assuming no docking available, the energy consumed would be 3200 mAh approximately, which exceeds the capacity of the used quad-rotor.

During the experiment in the real scenario, the docking processes were performed with high accuracy and robustness. The UAV motion tracking helped to maintain visual contact between the end effector camera and the UAV marker, predicting the movements and staying one step forward while considering structural constraints. As shown in the comparisons of Fig. 4.15 and Fig. 4.16, the system calculated the end effector locations required to follow and anticipate the movements while counteracting the delays. When the pose requests were close to structural constraints, variations in the z -axis were made to elude collision with the self-structure or avoid falling into singular positions.

The experiment proved the advantages of our system for robustness and energy management. The robot arm followed the UAV movements fast and smooth enough during tracking and final approach, compensating for the limitations of the UAV. The range for available docking was also increased because the arm workspace was larger than the UGV top surface. During the *Contact Phase*, the docking device locked the aircraft with enough strength for vertical storage, allowing protection and energy-saving without affecting the ground navigation.

After the second docking process and *target* recognition, the team returned safely to the initial base, and it was possible to save 1368 mAh of UAV energy, allowing to do a heterogeneous exploration in the depth of the building that would be impossible without a docking approach. In addition, if the docking process is combined with a fast-charging device, the UAV can recover energy during storage and further increase the utility of the robot team.

4.3 Summary

In this chapter, experiments in a simulated and real environment were presented. The first simulation approach demonstrated the feasibility of the system and helped to improve the design in an early stage. With this first simulation, it was possible to fine-tune the controllers and arrange the coordinate systems while analyzing the behavior of the UAV and end effector along with the task. The possible limitations and advantages of the team were clarified, along with the needs for the next steps.

For the continuation of the design process, it was delineated the simulation for the complete USAR mission (see Video 1 of Table 4.1) that demonstrated the ability of the robot team to handle environmental variables and difficult paths for the exploration of a complete building. In this simulation, the team was able to follow an original path and perform modifications according to the situation of the environment. Once the docking processes were requested, the UAV reached the rendezvous points before the arrival of the mobile manipulator, and the autonomous docking was executed with robustness. By repeating the docking process many times during the experiment, it was demonstrated the robustness and repeatability of the approach and its potential to facilitate the exploration of large unknown environments and enable middle and long term missions.

As the next step in the design process, the VTOL-UAV docking system was installed for real robots. For that purpose, the UAV and UGV robots were prepared; the details for

the different hardware parts, architecture, and control system were presented. Specifically, the materials, components, and drawings of the docking device and robotic arm design were presented. Also, the parameters achieved from the fine-tuning of the controller were summarized.

A first controlled approach was executed in the real environment as a preliminary test for robot improvement and control fine-tuning. In the next step, a complete USAR mission was prepared to be resolved by the heterogeneous robot team (recorded in Video 2 of the Table 4.1). Overall, the complete mission was performed successfully without any damage to the robots, and the team returned to the base with the UAV in the storage position and with enough battery available. Furthermore, heterogeneous exploration was performed in the depth of the building, and difficult regions in the map that were not feasible for UGV sensings, such as railings and narrow shafts, were explored.

Finally, it is safe to say that both simulation processes were very useful for the design process and the analysis of all the requirements and limitations of the system. The experience gained during the first simulation phase helped to improve the docking system and ease the construction and execution in a real scenario. During the last experimental phase, the system was successfully applied in a real environment, the docking task was proven, and the complete USAR mission was completed. The system proposed can be applied in different situations, and it allows to deploy a heterogeneous robot team for long term missions in this type of environments.

Chapter 5

Discussion, Conclusion and Future Work

5.1 Discussion

The USAR missions in the real-world have great complexity and present many technical difficulties for the robotic systems. The unknown environment and the tasks to resolve are of great challenge for the robotic systems. Nevertheless, teams of robots with heterogeneous capabilities have a large number of potential benefits for solving this particular type of mission. The strong integration of robots with aerial and terrestrial capabilities can help to create a complete robotic team that can tackle the missions with different perspectives and outstanding efficiency and robustness.

Every robot team with a VTOL-UAV robot platform as a member has a severe time limitation due to the UAV's high energy consumption and limited payload capacity. The UAV platform can load only small energy storage cells, which reduces the operating time to a very short time, limiting its usage and adding a considerable restriction for the complete team. After its energy has been depleted, this robot needs to replace or recharge batteries by returning to a safe place. That means the heterogeneous abilities are limited in time, they cannot be used for large environments, and there is the probability that they are not available when most needed.

The VTOL-UAV docking system presented in this thesis helps the robot team to overcome the VTOL-UAV limitation by performing physical contact between an aerial and a ground robot on-site. With this ability, the robot team is able to store the UAV and re-deploy it according to the needs of the mission, saving UAV energy and providing temporary protection and transportation to complete long term USAR missions.

Compared to the works that execute the landing of VTOL-UAVs on a flat surface on top of the UGV ([42–49]), the presented docking system uses the robot arm of the mobile manipulator to perform the final approach for contact. In this case, the robotic arm offers stable and smooth movements that can be controlled precisely. The robot arm can follow and anticipate accurately the UAV movements and perform corrections and in case the external disturbances affect the aircraft stability. In the end, the robot arm can reach the contact state with more stability and robustness.

By trusting more in the robot arm attached to a ground platform, the controller system can execute a safe UAV tracking and a repeatable contact process. This is an excellent advantage to the landing proposals in the research field because it avoids relying on the controller of an under-actuated platform that is strongly affected by external disturbances.

Moreover, with the robust robotic manipulator that can be controlled precisely in real-time, the optimization and predictive approaches can be easily included for improved behavior in the tracking and contact approach. At the same time, the environmental and structural constraints are considered to ensure a UAV docking without interruptions of any type.

The work in [63] has proposed a basic approach on the use of a robotic arm of the mobile manipulator to stabilize a landing platform for the UAV. However, this approach does not provide a safe locking mechanism to protect the UAV after contact and prevent accidents. In comparison, the system proposed in this thesis also includes a docking device to facilitate the final contact and immediately lock the UAV to the tip of the end effector of the arm. The docking device uses an electromagnet for initial pulling and a set of actuated levers that secure the UAV chassis to allow placing it in a storage position. In case the re-deployment of the UAV is required, the docking device allows a fast release to comply with the requirements of the mission.

With the software and hardware mechanisms included in this work, the docking system helps the heterogeneous multi-robot team to build a strong interaction between robot members that allows reliable cooperation and resource sharing. Also, the mixed abilities of the team can be used in the depth of a large USAR environment and the UAV energy can be saved and restored. A robot team with such capabilities is more useful and valuable, opening the door to expand its usability for many types of applications and situations.

5.2 General conclusion

This thesis developed the novel idea of VTOL-UAV autonomous docking with a mobile manipulator for air-ground multi-robot systems. The presented approach capacitates the robot team with a mechanism to detect and measure precisely the UAV movements for tracking and approximation and to safely link the aircraft chassis to the arm end effector for temporary storage.

The system allowed the air-ground robot team to cooperate and interact thoroughly, increasing effectiveness and benefiting all platforms. During the USAR mission, the aerial robot can be deployed at the right moment to provide information from its useful point of view to the ground platform for better exploration. When the UAV needs protection or to save (and potentially resupply) energy, the UGV can store and transport the aircraft through unsafe areas.

The autonomous VTOL-UAV docking process with a mobile manipulator was explained in two main steps: the stable UAV motion tracking and the contact process. Also, the conditions for the first detection were detailed, along with the benefits of implementing temporary storage for this kind of robot teams.

The controller that governs the aircraft and the robot arm was also described, along with the reactive, predictive, and optimization methods included to enhance its behavior for the task. The controller uses mainly visual information from a monocular camera and integrates the robotic and environmental constraints in the control loop for a robust UAV tracking and posterior approach. Additionally, the physical contact between robots was analyzed, and a solution for the robust design of a docking device that facilitated the final contact and safe storage was presented. In case of emergency or failure, safety tasks were included to protect the robots and possibly the human agents in the field, also to perform an attempt of track recovery and continuance of the docking process.

With the inclusion of the robot arm for the docking process, it was added enough redundancy and range to prevent failure due to environmental variations and UAV limitations. Therefore, providing a repeatable UAV docking process that increased the advantages of the air-ground robot team for USAR missions.

The system was tested with simulations and real scenario experiments, proving that the docking process can be executed safely one or more times during the mission. In the tests, the robots executed exploration to find a target in the depth of a large building, performing heterogeneous surveys in interest points, storing and deploying the aircraft when convenient

to finally make a return of both platforms to a safe location. Overall the proposed missions were completed successfully, and the advantages of the inclusion of the docking system in the heterogeneous robot team were discussed.

5.3 Future Work

For the continuation of this research work, it is convenient to test the *UAV Motion Tracking* approach for dynamic obstacles and external perturbations and for more extended missions and with multiple agents. The docking system can be effectively incorporated into different types of air-ground robot teams, and it can complement varied exploration strategies and cooperative technologies.

The following are the limitations of the current work that are important to address in future works in the field.

- The robot manipulator used in this thesis does not use an Inertial Measurement Unit (IMU) to sense the mobile base attitude and vibration. If an IMU is included, it is possible to include the measurements of the UGV movements into the controller to stabilize the end effector position optimally. The IMU can make it possible to use the docking system safely in environments with uneven terrains.
- The robot manipulator can perform tracking of the UAV movements at relatively low speeds. With better hardware equipment, like a better computing processor and better motors for the robot arm, it is possible to achieve more speed in the tracking process to keep the UAV in sight at every time. Also, it is worthy to implement algorithms to be executed by Graphics Processing Units (GPUs) to give more power to the calculations of the end effector positions to build a larger planning graph.
- It is necessary to include a detachable docking device to free the robot arm of the dedicated docking tool. This will allow the robot arm to store the UAV within the UGV chassis and continue with other manipulation tasks using different tools.
- It is necessary to include a charging port to replenish the UAV batteries when it is in the storage position. This needs to be designed for inclusion within the docking device to allow fast charging. It is also essential to explore the possibility of a wireless charging device to avoid direct electric contact and simplify the mechanism.

-
- It is important to test different path planning algorithms for the controller presented in this work. There are new planning algorithms in the state of the art that can enhance UAV motion tracking to become faster and include more redundancy. It is also essential to find other methods to represent the obstacles and constraints in the obstacle map in a better way.
 - It is possible to create a smart energy monitor to control and handle the energy resources of the robot team. When necessary, the monitor will require a docking task to save UAV energy, and a deployment when the UAV has enough energy to explore; always considering the UGV energy along the mission. This monitor, can be included as a software module to generate requests for the main docking system, improving the exploration strategy and reducing the attention required by the human supervisor.

References

- [1] Benjamin B Reed, Robert C Smith, Bo J Naasz, Joseph F Pellegrino, and Charles E Bacon. The restore-1 servicing mission. In *AIAA SPACE 2016*, page 5478. 2016.
- [2] Maurizio Rossi and Davide Brunelli. Autonomous gas detection and mapping with unmanned aerial vehicles. *IEEE Transactions on Instrumentation and measurement*, 65(4):765–775, 2015.
- [3] Tamio Arai, Enrico Pagello, Lynne E Parker, et al. Advances in multi-robot systems. *IEEE Transactions on robotics and automation*, 18(5):655–661, 2002.
- [4] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, and Takanori Emaru. Symbiotic navigation in multi-robot systems with remote obstacle knowledge sharing. *Sensors*, 17(7):1581, 2017.
- [5] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, and Takanori Emaru. Hitchhiking robots: A collaborative approach for efficient multi-robot navigation in indoor environments. *Sensors*, 17(8):1878, 2017.
- [6] Stephen Balakirsky, Stefano Carpin, Alexander Kleiner, Michael Lewis, Arnoud Visser, Jijun Wang, and Vittorio Amos Ziparo. Towards heterogeneous robot teams for disaster mitigation: Results and performance metrics from robocup rescue. *Journal of Field Robotics*, 24(11-12):943–967, 2007.
- [7] Mario Garzón, Joao Valente, David Zapata, and Antonio Barrientos. An aerial-ground robotic system for navigation and obstacle mapping in large outdoor areas. *Sensors*, 13(1):1247–1267, 2013.
- [8] Jeffrey Delmerico, Stefano Mintchev, Alessandro Giusti, Boris Gromov, Kamilo Melo, Tomislav Horvat, Cesar Cadena, Marco Hutter, Auke Ijspeert, Dario Floreano, et al.

- The current state and future outlook of rescue robotics. *Journal of Field Robotics*, 36(7):1171–1191, 2019.
- [9] Davide Falanga, Alessio Zanchettin, Alessandro Simovic, Jeffrey Delmerico, and Davide Scaramuzza. Vision-based autonomous quadrotor landing on a moving platform. In *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pages 200–207. IEEE, 2017.
- [10] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, and Takanori Emaru. On a bio-inspired hybrid pheromone signalling for efficient map exploration of multiple mobile service robots. *Artificial life and robotics*, 21(2):221–231, 2016.
- [11] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, and Takanori Emaru. Avoiding blind leading the blind: Uncertainty integration in virtual pheromone deposition by robots. *International journal of advanced robotic systems*, 13(6):17298814166666088, 2016.
- [12] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, Yohei Hoshino, Chao-Chung Peng, and Michiko Watanabe. Hitchhiking based symbiotic multi-robot navigation in sensor networks. *Robotics*, 7(3):37, 2018.
- [13] Jorge de León, Mario Garzón, David Garzón, Eduardo Narváez, Jaime del Cerro, and Antonio Barrientos. From video games multiple cameras to multi-robot teleoperation in disaster scenarios. In *2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 323–328. IEEE, 2016.
- [14] Abhijeet Ravankar, Ankit A Ravankar, Yohei Hoshino, and Yukinori Kobayashi. On sharing spatial data with uncertainty integration amongst multiple robots having different maps. *Applied Sciences*, 9(13):2753, 2019.
- [15] Asif Khan, Bernhard Rinner, and Andrea Cavallaro. Cooperative robots to observe moving targets. *IEEE transactions on cybernetics*, 48(1):187–198, 2016.
- [16] Ned Potter. A Mars helicopter preps for launch: The first drone to fly on another planet will hitch a ride on NASA’s perseverance rover-[news]. *IEEE Spectrum*, 57(7):06–07, 2020.

- [17] NASA. Mars helicopter. <https://mars.nasa.gov/technology/helicopter>, 2020. Accessed: 2020-10-15.
- [18] Jia-Rui Cook, Grey Hautaluoma, and Alana Johnson. 6 things to know about NASA's ingenuity Mars helicopter. <https://www.nasa.gov/feature/jpl/6-things-to-know-about-nasas-ingenuity-mars-helicopter>, 2020. Accessed: 2020-11-10.
- [19] Yufeng Yue, Chule Yang, Yuanzhe Wang, PG Chaminda Namal Senarathne, Jun Zhang, Mingxing Wen, and Danwei Wang. A multilevel fusion system for multirobot 3-D mapping using heterogeneous sensors. *IEEE Systems Journal*, 14(1):1341–1352, 2019.
- [20] Ankit A Ravankar, Abhijeet Ravankar, Takanori Emaru, and Yukinori Kobayashi. A hybrid topological mapping and navigation method for large area robot mapping. In *2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pages 1104–1107. IEEE, 2017.
- [21] Pileun Kim, Leon C Price, Jisoo Park, and Yong K Cho. UAV-UGV cooperative 3D environmental mapping. In *Computing in Civil Engineering 2019: Data, Sensing, and Analytics*, pages 384–392. American Society of Civil Engineers Reston, VA, 2019.
- [22] Luqi Wang, Fei Gao, Fengyu Cai, and Shaojie Shen. CRASH: A collaborative aerial-ground exploration system using hybrid-frontier method. In *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2259–2266. IEEE, 2018.
- [23] Jianqiang Li, Genqiang Deng, Chengwen Luo, Qiuzhen Lin, Qiao Yan, and Zhong Ming. A hybrid path planning method in unmanned air/ground vehicle (UAV/UGV) cooperative systems. *IEEE Transactions on Vehicular Technology*, 65(12):9585–9596, 2016.
- [24] Jeffrey Delmerico, Elias Mueggler, Julia Nitsch, and Davide Scaramuzza. Active autonomous aerial exploration for ground robot path planning. *IEEE Robotics and Automation Letters*, 2(2):664–671, 2017.
- [25] Hartmut Surmann, Nils Berninger, and Rainer Worst. 3D mapping for multi hybrid robot cooperation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 626–633. IEEE, 2017.

- [26] Ankit A Ravankar, Abhijeet Ravankar, Yukinori Kobayashi, and Takanori Emaru. Autonomous mapping and exploration with unmanned aerial vehicles using low cost sensors. In *Multidisciplinary Digital Publishing Institute Proceedings*, volume 4, page 44, 2018.
- [27] Jin Hyo Kim, Ji-Wook Kwon, and Jiwon Seo. Multi-UAV-based stereo vision system without GPS for ground obstacle mapping to assist path planning of UGV. *Electronics Letters*, 50(20):1431–1432, 2014.
- [28] Hailong Qin, Zehui Meng, Wei Meng, Xudong Chen, Hao Sun, Feng Lin, and Marcelo H Ang. Autonomous exploration and mapping system using heterogeneous UAVs and UGVs in GPS-denied environments. *IEEE Transactions on Vehicular Technology*, 68(2):1339–1350, 2019.
- [29] Juan-Carlos Trujillo, Rodrigo Munguia, Edmundo Guerra, and Antoni Grau. Cooperative monocular-based SLAM for multi-UAV systems in GPS-denied environments. *Sensors*, 18(5):1351, 2018.
- [30] Rajnikant Sharma and Clark Taylor. Cooperative navigation of MAVs in GPS denied areas. In *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 481–486. IEEE, 2008.
- [31] Lorenzo Rosa, Marco Cagnetti, Andrea Nicastrò, Pol Alvarez, and Giuseppe Oriolo. Multi-task cooperative control in a heterogeneous ground-air robot team. *IFAC-PapersOnLine*, 48(5):53–58, 2015.
- [32] Michael E Stieber, Michael McKay, George Vukovich, and Emil Petriu. Vision-based sensing and control for space robotics applications. *IEEE Transactions on Instrumentation and Measurement*, 48(4):807–812, 1999.
- [33] Xu Chu Ding, Amir R Rahmani, and Magnus Egerstedt. Multi-UAV convoy protection: An optimal approach to path planning and coordination. *IEEE Transactions on Robotics*, 26(2):256–268, 2010.
- [34] Herbert G Tanner and Dimitrios K Christodoulakis. Decentralized cooperative control of heterogeneous vehicle groups. *Robotics and Autonomous Systems*, 55(11):811–823, 2007.

- [35] Javier Gomez-Avila, Carlos Lopez-Franco, Alma Y Alanis, Nancy Arana-Daniel, and Michel Lopez-Franco. Ground vehicle tracking with a quadrotor using image based visual servoing. *IFAC-PapersOnLine*, 51(13):344–349, 2018.
- [36] Herbert G Tanner. Switched UAV-UGV cooperation scheme for target detection. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3457–3462. IEEE, 2007.
- [37] Parikshit Maini and PB Sujit. On cooperation between a fuel constrained UAV and a refueling ugv for large scale mapping applications. In *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1370–1377. IEEE, 2015.
- [38] Haibin Duan and Qifu Zhang. Visual measurement in simulation environment for vision-based UAV autonomous aerial refueling. *IEEE Transactions on Instrumentation and Measurement*, 64(9):2468–2480, 2015.
- [39] Fernando Ropero, Pablo Muñoz, and María D R-Moreno. Terra: A path planning algorithm for cooperative UGV-UAV exploration. *Engineering Applications of Artificial Intelligence*, 78:260–272, 2019.
- [40] Barbara Arbanas, Antun Ivanovic, Marko Car, Tomislav Haus, Matko Orsag, Tamara Petrovic, and Stjepan Bogdan. Aerial-ground robotic system for autonomous delivery tasks. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 5463–5468. IEEE, 2016.
- [41] J Evan Inglett and Erick J Rodríguez-Seda. Object transportation by cooperative robots. In *SoutheastCon 2017*, pages 1–6. IEEE, 2017.
- [42] Torsten Merz, Simone Duranti, and Gianpaolo Conte. Autonomous landing of an unmanned helicopter based on vision and inertial sensing. In *Experimental Robotics IX*, pages 343–352. Springer, 2006.
- [43] Alexander Miller, Boris Miller, Alexey Popov, and Karen Stepanyan. UAV landing based on the optical flow videonavigation. *Sensors*, 19(6):1351, 2019.
- [44] Kirk YW Scheper and Guido CHE de Croon. Evolution of robust high speed optical-flow-based landing for autonomous MAVs. *Robotics and Autonomous Systems*, 124:103380, 2020.

- [45] Yingcai Bi and Haibin Duan. Implementation of autonomous visual tracking and landing for a low-cost quadrotor. *Optik-International Journal for Light and Electron Optics*, 124(18):3296–3300, 2013.
- [46] Karl Engelbert Wenzel, Andreas Masselli, and Andreas Zell. Automatic take off, tracking and landing of a miniature UAV on a moving carrier vehicle. *Journal of intelligent & robotic systems*, 61(1-4):221–238, 2011.
- [47] Juan Roldán, Pablo Garcia-Aunon, Mario Garzón, Jorge de León, Jaime del Cerro, and Antonio Barrientos. Heterogeneous multi-robot system for mapping environmental variables of greenhouses. *Sensors*, 16(7):1018, 2016.
- [48] Pablo R Palafox, Mario Garzón, João Valente, Juan Jesús Roldán, and Antonio Barrientos. Robust visual-aided autonomous takeoff, tracking, and landing of a small UAV on a moving landing platform for life-long operation. *Applied Sciences*, 9(13):2661, 2019.
- [49] Edward A Macdonald. *Multi-robot assignment and formation control*. PhD thesis, Georgia Institute of Technology, 2011.
- [50] Tamara Petrovic, Tomislav Haus, Barbara Arbanas, Matko Orsag, and Stjepan Bogdan. Can UAV and UGV be best buddies? towards heterogeneous aerial-ground cooperative robot system for complex aerial manipulation tasks. In *2015 12th international conference on informatics in control, automation and robotics (ICINCO)*, volume 1, pages 238–245. IEEE, 2015.
- [51] Bara J Emran and Homayoun Najjaran. A review of quadrotor: An underactuated mechanical system. *Annual Reviews in Control*, 46:165–180, 2018.
- [52] Panagiotis Vlantis, Panos Marantos, Charalampos P Bechlioulis, and Kostas J Kyriakopoulos. Quadrotor landing on an inclined platform of a moving ground vehicle. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2202–2207. IEEE, 2015.
- [53] Stephen A Conyers, Nikolaos I Vitzilaios, Matthew J Rutherford, and Kimon P Valavanis. A mobile self-leveling landing platform for VTOL-UAVs. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 815–822. IEEE, 2015.

- [54] Alexandre Campos, Jacqueline Quintero, Roque Saltarén, Manuel Ferre, and Rafael Aracil. An active helideck testbed for floating structures based on a stewart-gough platform. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3705–3710. IEEE, 2008.
- [55] Yanting Huang, Zewei Zheng, Liang Sun, and Bing Zhu. Disturbance observer-based saturated control for a quadrotor landing on a vessel. In *2017 13th IEEE International Conference on Control & Automation (ICCA)*, pages 419–424. IEEE, 2017.
- [56] Kaiyu Hang, Ximin Lyu, Haoran Song, Johannes A Stork, Aaron M Dollar, Danica Kragic, and Fu Zhang. Perching and resting. a paradigm for UAV maneuvering with modularized landing gears. *Science Robotics*, 4(28), 2019.
- [57] Paul M Nadan, Tatiana M Anthony, Duncan M Michael, Jeffrey B Pflueger, Manik S Sethi, Kelli N Shimazu, Mindy Tieu, and Christopher L Lee. A bird-inspired perching landing gear system. *Journal of Mechanisms and Robotics*, 11(6), 2019.
- [58] Han W Wopereis, TD Van Der Molen, TH Post, Stefano Stramigioli, and Matteo Fumagalli. Mechanism for perching on smooth surfaces using aerial impacts. In *2016 IEEE international symposium on safety, security, and rescue robotics (SSRR)*, pages 154–159. IEEE, 2016.
- [59] FJ Garcia-Rubiales, P Ramon-Soria, BC Arrue, and A Ollero. Magnetic detaching system for modular UAVs with perching capabilities in industrial environments. In *2019 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED UAS)*, pages 172–176. IEEE, 2019.
- [60] Vitaly Petrov, Margarita Gapeyenko, Dmitri Moltchanov, Sergey Andreev, and Robert W Heath. Hover or perch: Comparing capacity of airborne and landed millimeter-wave UAV cells. *IEEE Wireless Communications Letters*, 9(12):2059–2063, 2020.
- [61] Moritz Maier, André Oeschger, and Konstantin Kondak. Robot-assisted landing of VTOL UAVs: Design and comparison of coupled and decoupling linear state-space control approaches. *IEEE Robotics and Automation Letters*, 1(1):114–121, 2015.

- [62] Moritz Maier and Konstantin Kondak. Landing of VTOL UAVs using a stationary robot manipulator: A new approach for coordinated control. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 1497–1502. IEEE, 2015.
- [63] Vincenzo Lippiello, Rafik Mebarki, and Fabio Ruggiero. Visual coordinated landing of a UAV on a mobile robot manipulator. In *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–7. IEEE, 2013.
- [64] Eduardo Narváez, Ankit A Ravankar, Abhijeet Ravankar, Yukinori Kobayashi, and Takanori Emaru. Vision based autonomous docking of VTOL UAV using a mobile robot manipulator. In *2017 IEEE/SICE International Symposium on System Integration (SII)*, pages 157–163. IEEE, 2017.
- [65] E. Narváez, A. A. Ravankar, A. Ravankar, T. Emaru, and Y. Kobayashi. Autonomous VTOL-UAV docking system for heterogeneous multirobot team. *IEEE Transactions on Instrumentation and Measurement*, 70:1–18, 2021.
- [66] Christoforos Kanellakis and George Nikolakopoulos. Survey on computer vision for UAVs: Current developments and trends. *Journal of Intelligent & Robotic Systems*, 87(1):141–168, 2017.
- [67] Tiago P Nascimento and Martin Saska. Position and attitude control of multi-rotor aerial vehicles: A survey. *Annual Reviews in Control*, 48:129–146, 2019.
- [68] Krzysztof Kozłowski and Dariusz Pazderski. Modeling and control of a 4-wheel skid-steering mobile robot. *International journal of applied mathematics and computer science*, 14:477–496, 2004.
- [69] Richard S Hartenberg and Jacques Denavit. A kinematic notation for lower pair mechanisms based on matrices. *Journal of applied mechanics*, 77(2):215–221, 1955.
- [70] Abdullah Aamir Hayat, Rajeevlochana G Chittawadigi, Arun Dayal Udai, and Subir K Saha. Identification of Denavit-Hartenberg parameters of an industrial robot. In *Proceedings of Conference on Advances In Robotics*, pages 1–6, 2013.
- [71] CR Rocha, CP Tonetto, and Altamir Dias. A comparison between the Denavit–Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators. *Robotics and Computer-Integrated Manufacturing*, 27(4):723–728, 2011.

- [72] Eric M Rosales and Qiang Gan. Forward and inverse kinematics models for a 5-DoF pioneer 2 robot arm. Technical report, Department of Computer Science, University of Essex, 2002.
- [73] Andrew Goldenberg, Beno Benhabib, and Robert Fenton. A complete generalized solution to the inverse kinematics of robots. *IEEE Journal on Robotics and Automation*, 1(1):14–20, 1985.
- [74] Serdar KuCuk and Zafer Bingul. The inverse kinematics solutions of industrial robot manipulators. In *Proceedings of the IEEE International Conference on Mechatronics, 2004. ICM'04.*, pages 274–279. IEEE, 2004.
- [75] Oussama Khatib. Mobile manipulation: The robotic assistant. *Robotics and Autonomous Systems*, 26(2-3):175–183, 1999.
- [76] Stephen Se, David Lowe, and Jim Little. Global localization using distinctive visual features. In *IEEE/RSJ international conference on intelligent robots and systems*, volume 1, pages 226–231. IEEE, 2002.
- [77] Patric Jensfelt and Steen Kristensen. Active global localization for a mobile robot using multiple hypothesis tracking. *IEEE Transactions on Robotics and Automation*, 17(5):748–760, 2001.
- [78] Agostino Martinelli, Frederic Pont, and Roland Siegwart. Multi-robot localization using relative observations. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, pages 2797–2802. IEEE, 2005.
- [79] Agostino Martinelli and Roland Siegwart. Observability analysis for mobile robot localization. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1471–1476. IEEE, 2005.
- [80] Xun S Zhou and Stergios I Roumeliotis. Robot-to-robot relative pose estimation from range measurements. *IEEE Transactions on Robotics*, 24(6):1379–1393, 2008.
- [81] Jennifer Casper and Robin R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 33(3):367–385, 2003.

- [82] Keiji Nagatani, Seiga Kiribayashi, Yoshito Okada, Kazuki Otake, Kazuya Yoshida, Satoshi Tadokoro, Takeshi Nishimura, Tomoaki Yoshida, Eiji Koyanagi, Mineo Fukushima, et al. Emergency response to the nuclear accident at the Fukushima Daiichi nuclear power plants using mobile rescue robots. *Journal of Field Robotics*, 30(1):44–63, 2013.
- [83] Arthur C Sanderson and Lee E Weiss. Adaptive visual servo control of robots. In *Robot vision*, pages 107–116. Springer, 1983.
- [84] Steven C Chapra, Raymond P Canale, et al. *Numerical methods for engineers*. Boston: McGraw-Hill Higher Education, 2010.
- [85] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *The international journal of robotics research*, 30(7):846–894, 2011.
- [86] Abhijeet Ravankar, Ankit A Ravankar, Yukinori Kobayashi, Yohei Hoshino, and Chao-Chung Peng. Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. *Sensors*, 18(9):3170, 2018.
- [87] John Wang and Edwin Olson. AprilTag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4193–4198. IEEE, oct 2016.

Appendix A

Tools and Supporting Figures

Fiducial Marker for Visual Detection

The fiducial markers used in this work are generated with the AprilTag 3 fiducial detection library. This convenient tool was used within the Robot Operating System (ROS) and Ubuntu.

The tool for fiducial marker generation and for pre-generated markers used can be found in the following link.

http://wiki.ros.org/apriltag_ros

Also, the ROS wrapper of the tool can be found in the following link.

https://github.com/RIVeR-Lab/apriltags_ros

The specific marker used for the visual detection in this work is the one in Fig. A.1. This bundle of markers was attached to the bottom-front on the UAV to be detected by the end effector camera of the mobile manipulator.

The bundle is designed and calibrated for detection for mid distances and also for smaller distances. It is composed by three different markers with different sizes. When the camera is far from the bundle, it will detect easily the large marker in the bundle (right side in Fig. A.1), and when the camera is very close to the bundle, i.e. near docking, it will detect the smaller markers (left side of Fig. A.1). Moreover, when the camera can see clearly more than one marker, they are detected and their location is used to calculate the global marker position, which gives redundancy to the measurement. This is the reason on using two small markers in the left side of the Fig. A.1, because with these, it is possible to get a good redundancy in the measurements for the docking process when the robot arm is very close to the UAV, which is the most critical portion of the process.

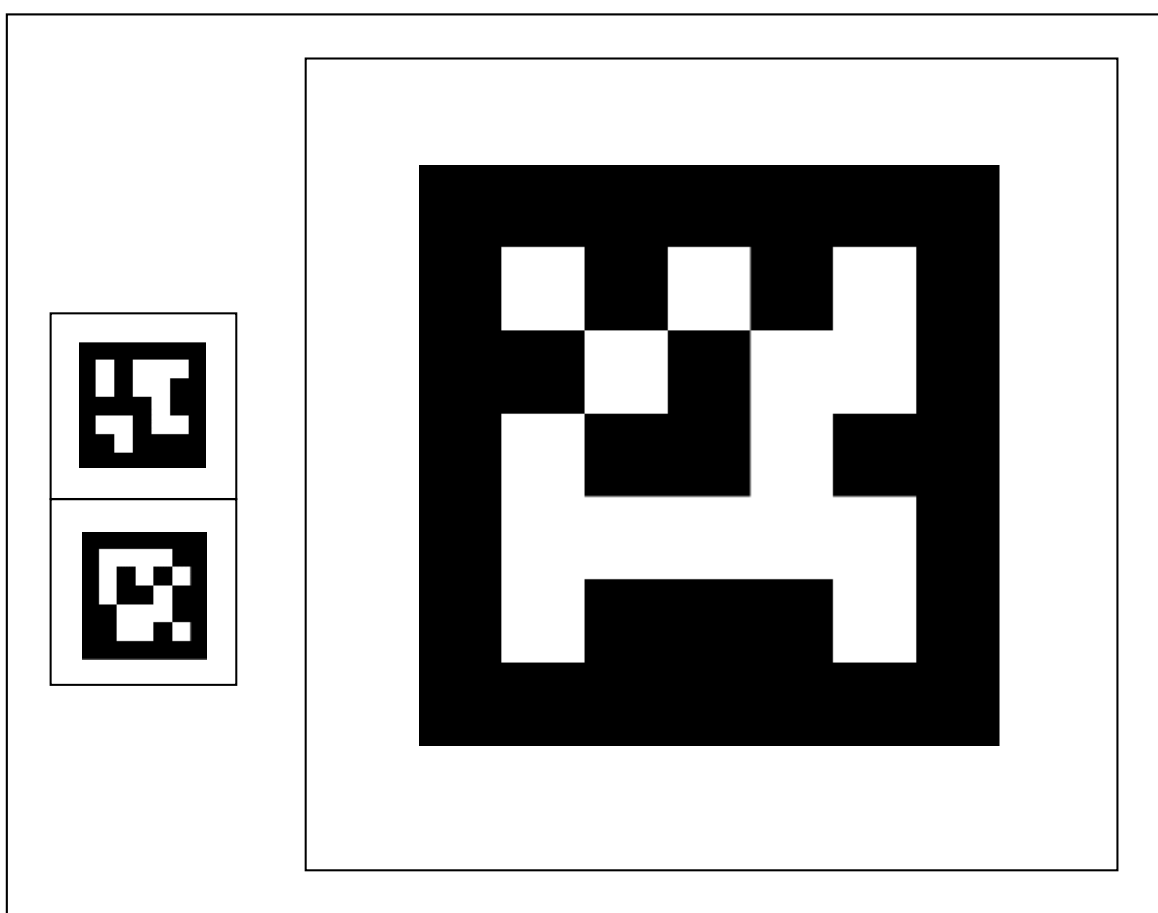


Fig. A.1 Marker bundle.

A.1 UAV Deployment Maneuver

The robot arm perform a series of movements to return the UAV from storage state to a horizontal position for deployment. The movements are designed to not interfere with the UGV and the manipulator chassis, and avoid collisions with the ground and different structures. The sequence of deployment is shown in Fig. A.2.

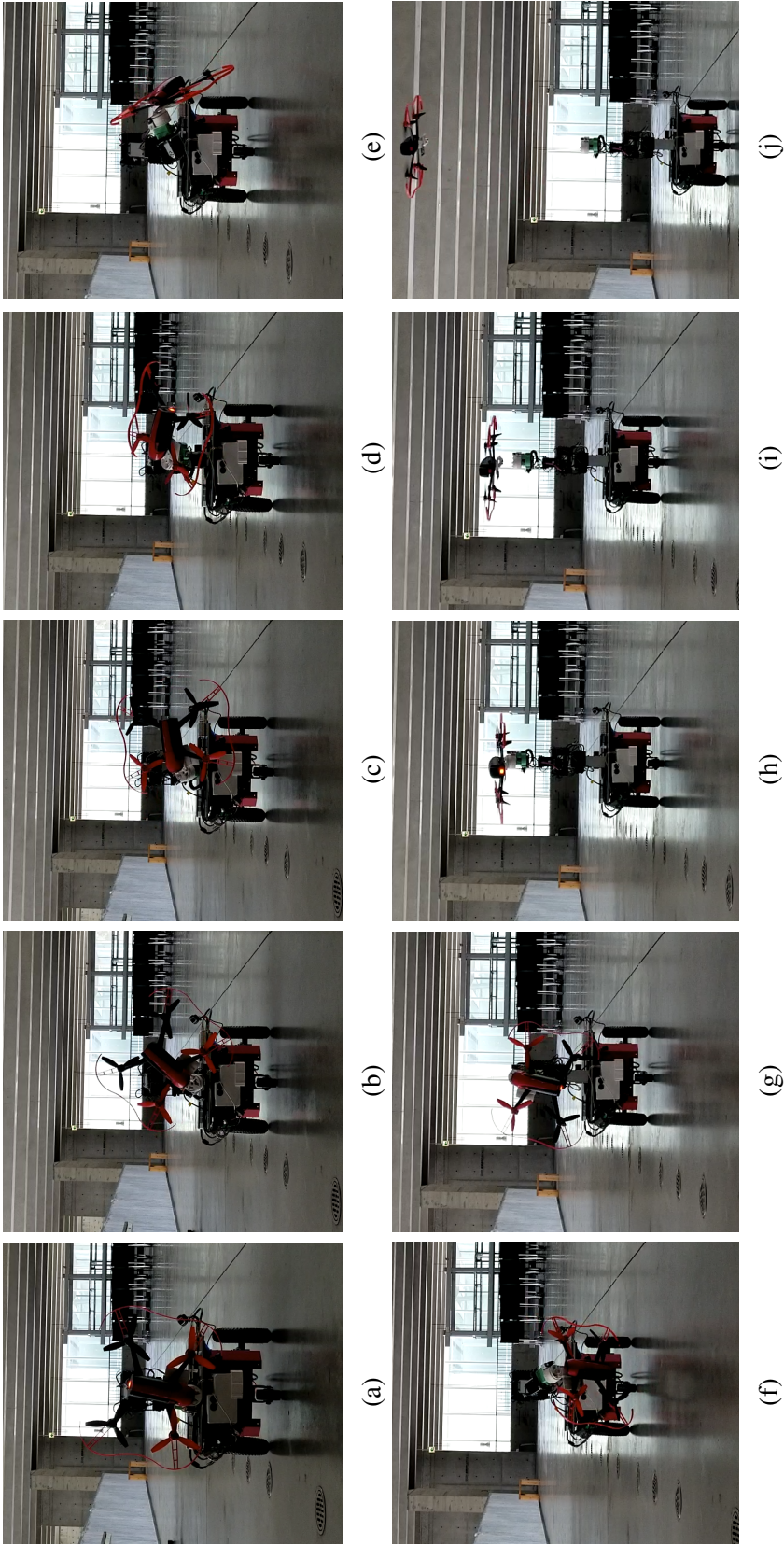


Fig. A.2 Sequence of photographs for the deployment maneuver.

Appendix B

Technical Drawings

B.0.1 Robot Arm

The precise dimensions, motor references, and location of the elements in the designed robot manipulator are shown in the technical drawing of Fig. B.1.

Also, the extended arm with the total reach length is shown in Fig. B.2. In the same figure, the details for the length of each link, motor sizes, and other dimensions are shown.

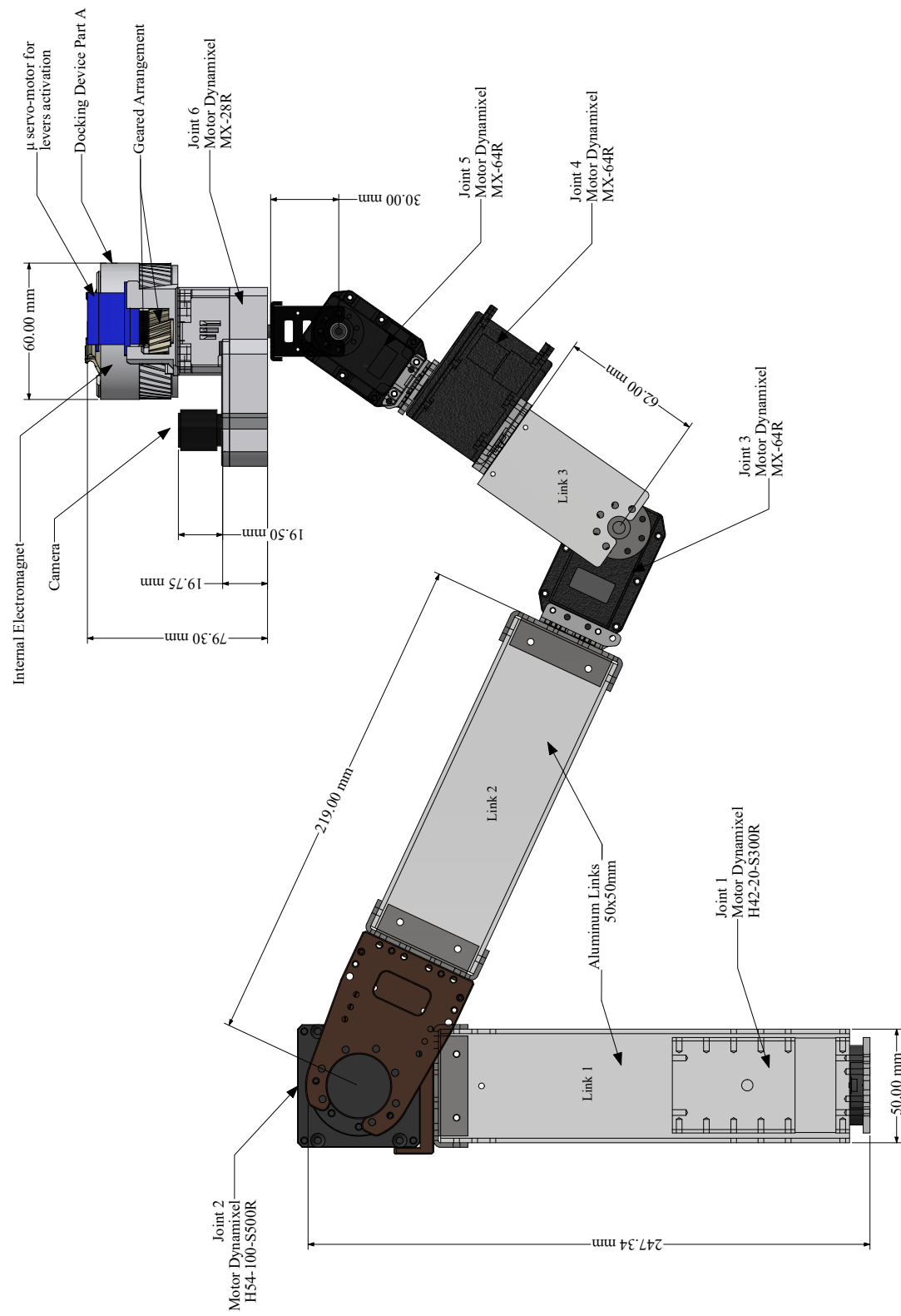


Fig. B.1 Detailed technical drawing of the robot manipulator with the Part A of the docking device. Links 1 and 2 are built with aluminum square pipes; they are designed especially with great longitude for more extended vertical reach.

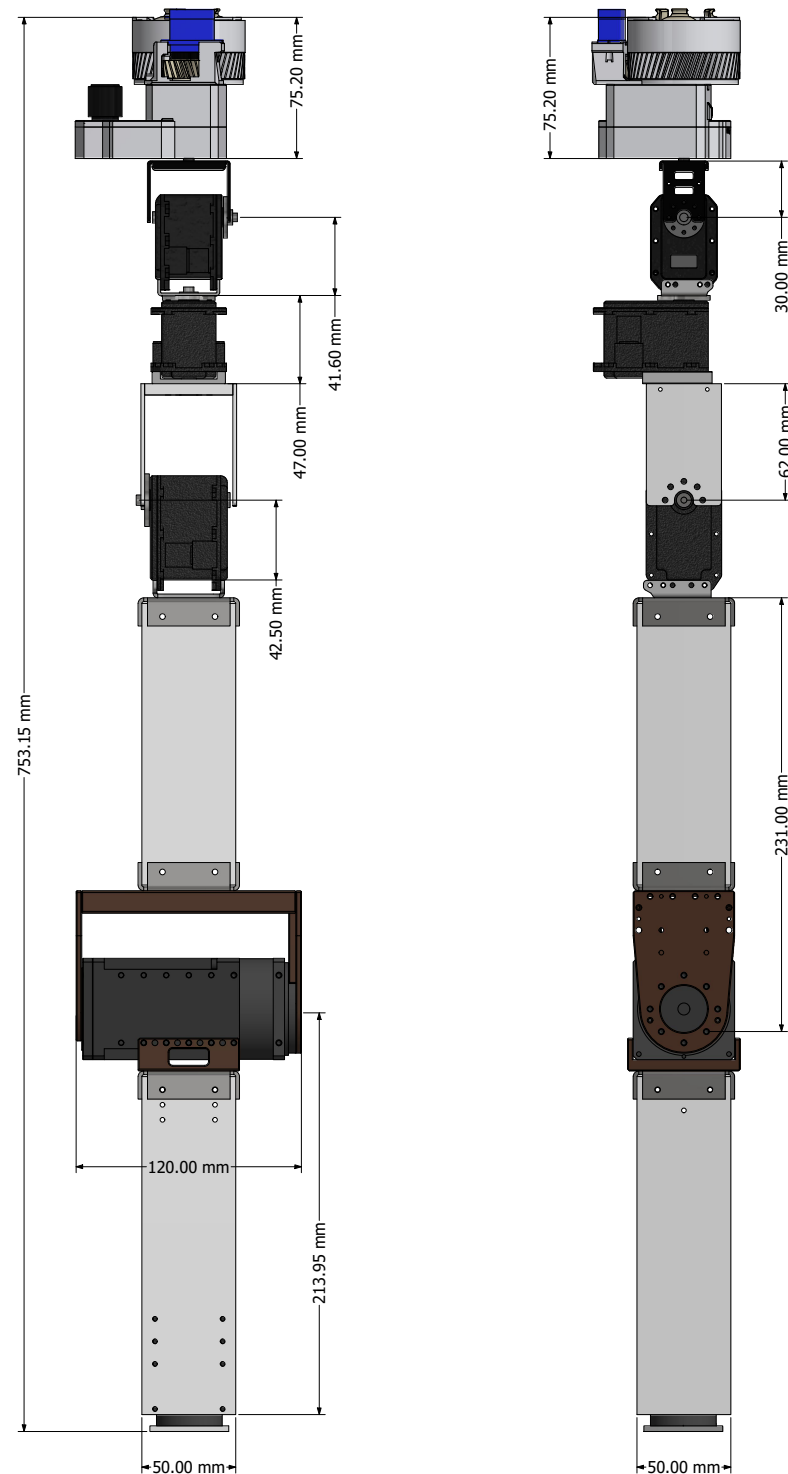


Fig. B.2 Detailed technical drawing of the robot manipulator while fully extended. The total length for reaching objects vertically is around 75.3cm.

B.0.2 Docking Device

Fig. B.3 details complete the docking device with different perspectives and description of main components. Fig. B.4 presents the Part A of the docking device with a cross section to indicate the corresponding internal components.

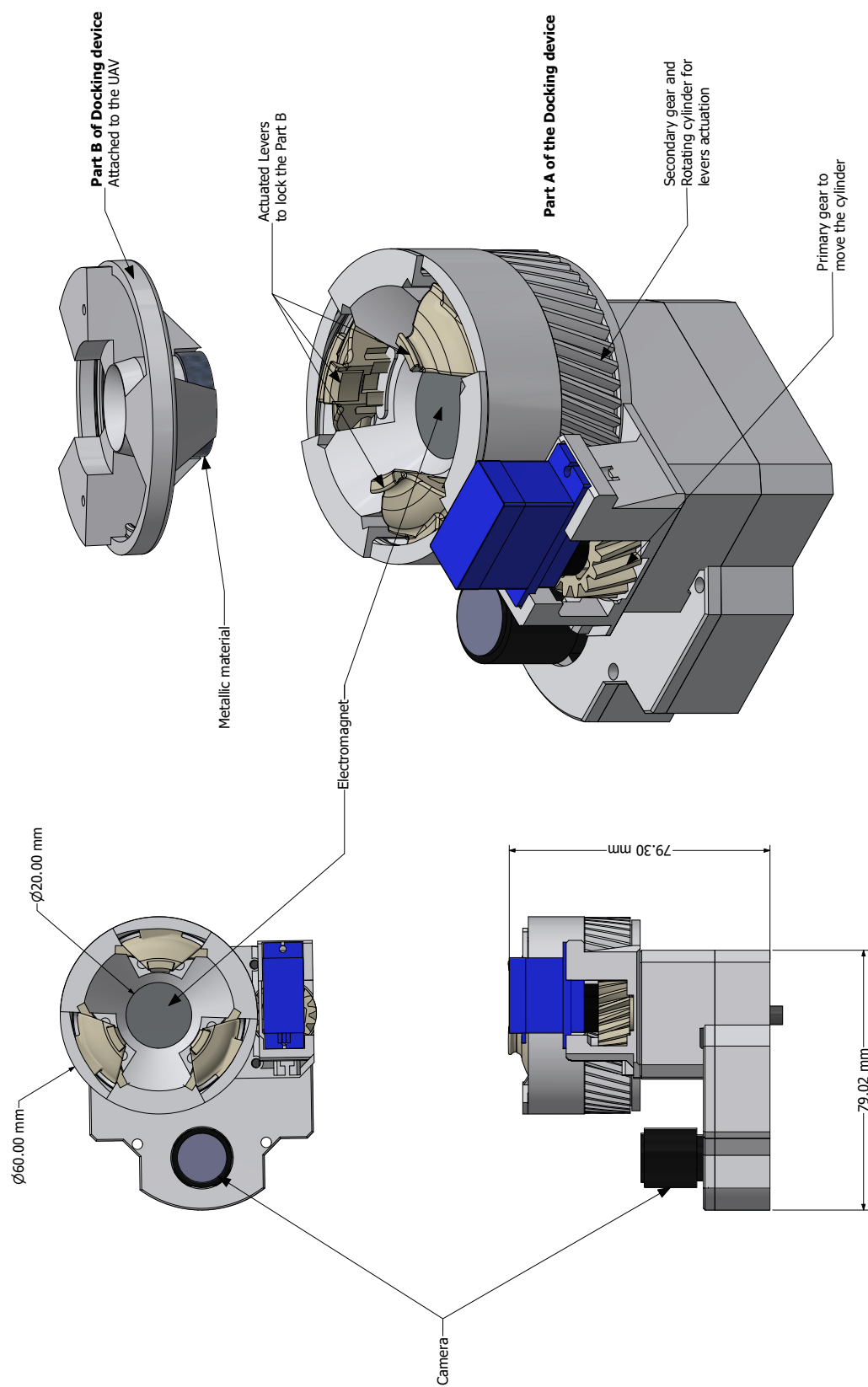


Fig. B.3 Detailed technical drawing of the docking device.

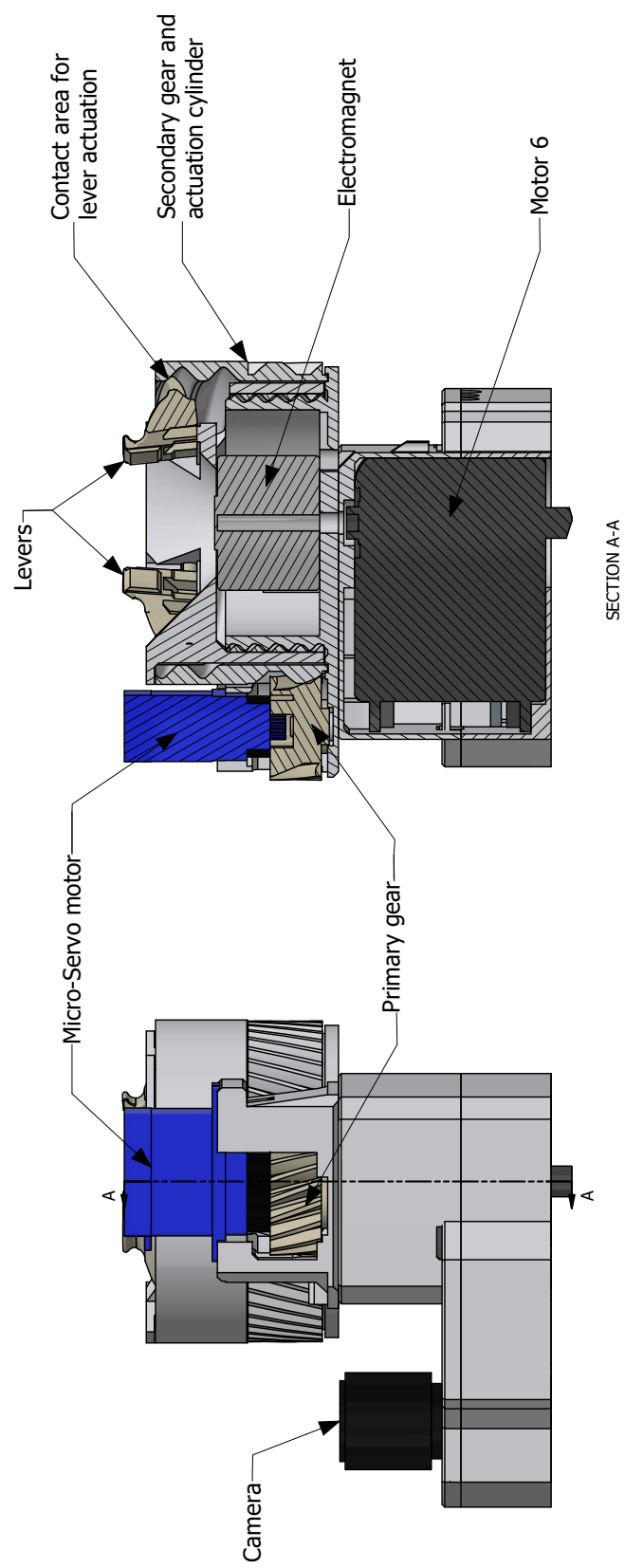


Fig. B.4 Detailed technical drawing of the Part A of the docking device with cross section. The area of contact of the actuation cylinder with the levers is indicated.