FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Mission Control for Swarm Robotics in the Agriculture Context

João Tomás Salgado Souto Morais de Castro



Mestrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Filipe Baptista Neves dos Santos

July 31, 2024

© João Tomás Salgado Souto Morais de Castro, 2024

Resumo

Na natureza, e particularmente na agricultura, a polinização é fundamental para a sustentabilidade da nossa sociedade. A polinização é um processo vital subjacente à qualidade das plantações e culturas, responsável pela biodiversidade e padrões da flora. As abelhas desempenham um papel crucial na polinização natural, no entanto, as suas populações estão a diminuir. Os robôs podem ajudar e complementar os polinizadores na manutenção dos níveis de polinização enquanto os humanos trabalham para recuperar as populações de abelhas.

Este trabalho propõe a cooperação entre veículos aéreos não tripulados (UAVs) e um veículo terrestre não tripulado (UGV), de modo a aproveitar as vantagens deste trabalho colaborativo no contexto da polinização, chamado Pollinationbots. Para além do sistema robótico heterogéneo, foi desenvolvida uma arquitetura de comunicação ROS2 que proporciona um sistema modular que visa uma futura implementação de algoritmos de *obstacle-avoidance* e de condução autónoma do UGV. Neste trabalho, a arquitetura é explicada e discutida, destacando os princípios do design e o potencial de escalabilidade.

Além disso, o sistema utiliza comportamentos e metodologias presentes em *swarms* para implementar estratégias de polinização mais eficazes, garantindo uma polinização eficiente, mantendo a integridade ecológica do ambiente. A conceção do sistema é uma tarefa delicada, dadas as interações entre os diferentes agentes e a capacidade de coordenar os diferentes veículos em segurança. A arquitetura e o desempenho do sistema são apresentados e avaliados no simulador Webots, com foco no controlador de planeamento de trajetória e de *leader-follower*, examinando diferentes missões e cenários.

Resultados preliminares indicam que o Pollinationbots é uma solução viável para a polinização artificial, sendo os UAVs capazes de detetar alterações no ambiente e obstáculos no percurso do UGV, bem como identificar e abordar os pontos de interesse (POI) que devem ser visitados no contexto da missão de polinização.

Abstract

In nature, and particularly in agriculture, pollination is fundamental for the sustainability of our society. In this context, pollination is a vital process underlying crop yield quality and is responsible for the biodiversity and the standards of the flora. Bees play a crucial role in natural pollination; however, their populations are declining. Robots can help and complement pollinators in maintaining pollination levels while humans work to recover bee populations.

This work proposes cooperation between multiple unmanned aerial vehicles (UAVs) and an unmanned ground vehicle (UGV), leveraging the advantages of collaborative work for pollination, which is referred to as Pollinationbots. Besides the heterogeneous robotic system, we developed a ROS2 communication architecture that aims for a modular system for the future implementation of obstacle avoidance algorithms and autonomous driving of the UGV. In this work, this architecture is explained and discussed, highlighting the design principles and potential for scalability.

Moreover, the system employs swarm behaviors and methodologies to implement more effective pollination strategies, ensuring efficient pollination across various scenarios, while maintaining the ecological integrity of the environment. The system's design is a delicate and intricate task given the interactions between the different agents and the ability to coordinate the different vehicles safely. These challenges are analyzed and discussed. The architecture is presented and evaluated through the Webots simulator, focusing on path-planning and leader-follower behaviors, examining different missions and scenarios.

Preliminary results indicate that Pollinationbots is a viable solution for robotic pollination, with the UAVs being able to monitor possible changes in the environment and detect obstacles in the way of the UGV, as well as identify and address the points of interest (POI) that should be visited in the context of the pollination mission.

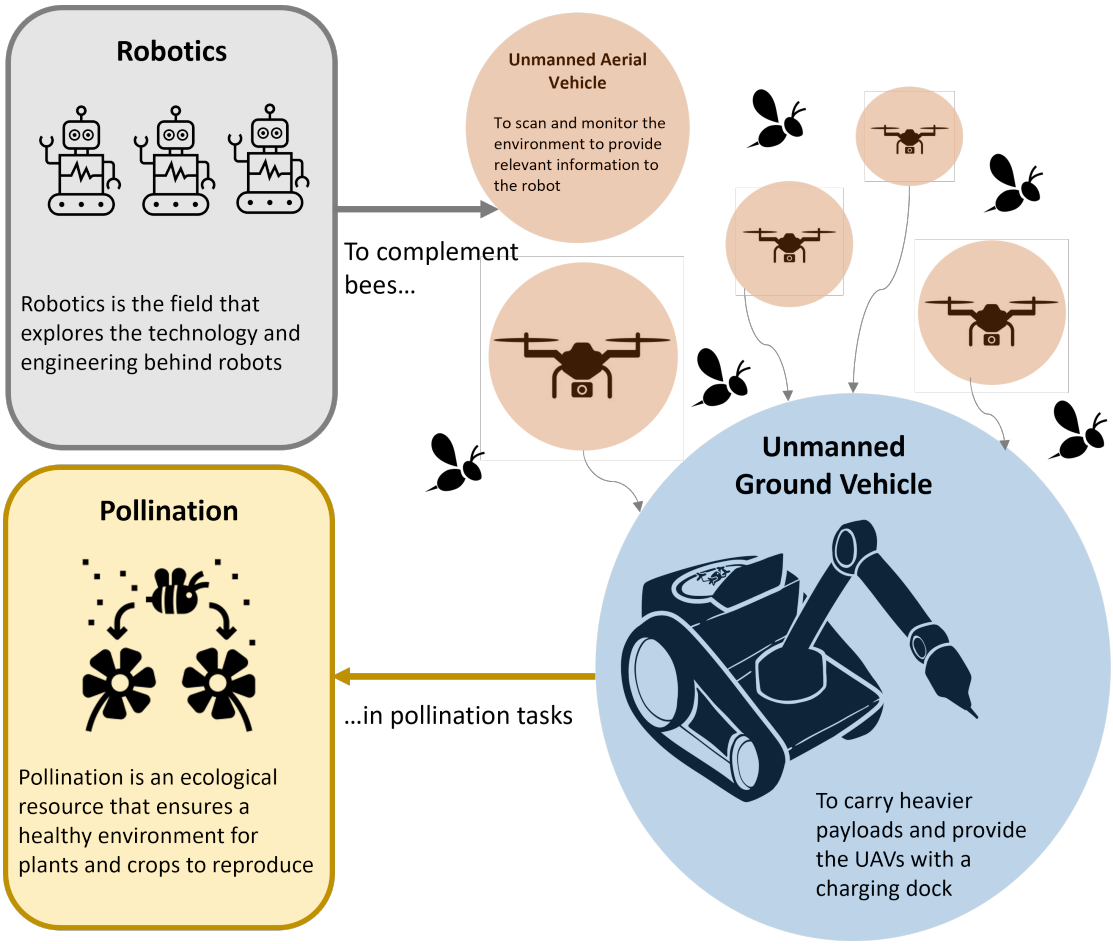


Figure 1: Robotic Solution composed by a UGV and several UAVs that cooperate to perform diverse missions in the agricultural context

United Nations Sustainable Development Goals

The 2030 Agenda for Sustainable Development is an ambitious agenda adopted by all the United Nations Members in 2015 that, at its core, comprises 17 Sustainable Development Goals (SDGs) that promote various dimensions of sustainable development in social, economic, and ambient environments. These objectives tackle worldwide issues, including poverty, inequality, and climate change, among many others. All goals are interlinked and address a specific subject matter - provide a good healthcare system, eradicate poverty and hunger, ensure that everyone obtains a quality education, and build a sustainable environment.

According to the United Nations Educational, Scientific and Cultural Organization, engineering has a crucial role in achieving the 17 SDGs. Engineering plays a vital role in connecting social needs with technological advancements. Addressing these challenges and goals in this work can redefine how this thesis is perceived and its potential to contribute to a more sustainable future.

This dissertation aims specifically for Goals 2 and 9 by promoting resilient agricultural practices (goal 2) and technological upgrading and innovating in agriculture (goal 9).

SDG	Goal	Contribution	Development indicators and metrics
2	2.1	This dissertation addresses a robotic solution that provides pollination services, ensuring sustainable agriculture production and crop quality. This way, it would be possible to boost food production and guarantee access for all people, especially people in vulnerable situations, to quality and nutritious food all year round.	This research aims to improve agriculture production, particularly in the context of specific plants that have a harder time being pollinated. Improving the quality of these products by assuring a consistent and homogeneous pollination would be the ultimate goal.
9	9.1	This work addresses a robotic heterogeneous solution composed of multiple UAVs and a UGV and several algorithms to improve the efficiency of its tasks. The main objective is to contribute to the scientific community with this project by innovating in this particular field and, hopefully, encouraging technological development and research.	This work will enable contributions to the existing literature, enriching this field with insights into the created robotic system. Implementing this technology in real-life scenarios and its commercial use would ensure that this solution could aim to contribute to this specific sustainable goal.

Agradecimentos

Primeiro, gostaria de agradecer ao meu orientador, Filipe Neves dos Santos, por sempre ter acreditado em mim e me ter incentivado a alcançar objetivos que me pareciam inatingíveis. As suas ideias e visão realista do projeto foram fundamentais para a concretização deste trabalho.

Agradeço ao Pedro Henrique Moura pela colaboração e acompanhamento no desenvolvimento desta dissertação. Sem as suas contribuições, o meu percurso e evolução académica não seria o mesmo.

Agradeço à TRIBE - Laboratório de Robótica e IoT para Agricultura e Floresta de Precisão Inteligente, por me ter acolhido e ajudado nas alturas mais difíceis. Palavras não fariam justiça ao apoio que me forneceram durante o meu percurso académico.

Agradeço aos meus amigos, por serem os pilares desta etapa da minha vida, todos os momentos, bons ou maus, que tenhamos atravessado, culminam neste projeto.

Finalmente, agradeço à minha família que lutou e me apoiou incondicionalmente durante estes 5 anos, vocês foram a força que tornou este percurso possível.

João Tomás Castro

"In the end,
you become what you always were
- a very big fish."

Will Bloom in Big Fish: A Novel of Mythic, 1998

Contents

1 Introduction **1**
- 1.1 Context . . . 1
- 1.2 Motivation . . . 1
- 1.3 Thesis Main Contributions . . . 2
- 1.4 Structure . . . 2

2 Background **4**
- 2.1 Swarm Robotics . . . 4
- 2.2 Swarm Dynamics . . . 5
- 2.3 Unmanned Vehicles . . . 7
  - 2.3.1 Unmanned Ground Vehicles . . . 7
  - 2.3.2 Unmanned Aerial Vehicles . . . 10
- 2.4 Simulators . . . 12
  - 2.4.1 Gazebo . . . 12
  - 2.4.2 Webots . . . 13
  - 2.4.3 ARGoS . . . 13
  - 2.4.4 Farming Simulator . . . 14
- 2.5 ROS2 (Robot Operating System) . . . 14

3 Bibliography Review **16**
- 3.1 Systematic Review Objectives . . . 16
- 3.2 Methodology . . . 17
  - 3.2.1 Eligibility Criteria . . . 17
  - 3.2.2 Selection Process . . . 18
- 3.3 Search Results . . . 19
  - 3.3.1 Path Planning . . . 20
  - 3.3.2 Applications . . . 21
  - 3.3.3 Mission Control . . . 21

4 Proposed Solution **22**
- 4.1 Requirements . . . 22
- 4.2 The Proposed Solution . . . 23
  - 4.2.1 UGV Low Level Control . . . 24
  - 4.2.2 UAV Low Level Control . . . 25
  - 4.2.3 Swarm Organization . . . 25
- 4.3 State Machine . . . 33

5 Simulation and Results **34**
5.1 Simulators 34
5.2 Webots 35
5.3 Simulation 36

6 Conclusions and Future Work **46**

List of Figures

1 Robotic Solution composed by a UGV and several UAVs that cooperate to perform diverse missions in the agricultural context v

2.1 Intra-organizational swarm interactions among ant[1] and bee[2] colonies. 4
2.2 Centralized *vs* Decentralized Systems 6
2.3 Moving Path Following 6
2.4 The kinematics schematic of skid-steering mobile robot [21]. 8
2.5 The kinematics schematic of skid-steering mobile robot [22]. 8
2.6 A front-wheel-steering vehicle with Ackermann steering [23]. 9
2.7 Free-body diagram of four-wheel with independent steering [24]. 10
2.8 Single-rotor[3] aerial vehicle 10
2.9 Multi-rotor[4] aerial vehicle 11
2.10 Flapping-wings[5] aerial vehicle 11
2.11 Fixed-wing[6] aerial vehicle 12
2.12 Simulation of an agricultural robot in Gazebo[7] 13
2.13 Simulation of a two-wheeled robot in Webots[8] 13
2.14 Simulation of a swarm system in ARGoS[9] 14
2.15 Simulation of agricultural machinery in Farming Simulator[10] 14
2.16 ROS 2 node interfaces [27] 15

3.1 PRISMA Workflow in the context of this systematic review 19
3.2 Obstacle visualization in 3D space using Artificial Potential Fields 20
3.3 Trajectories of UAVs flying through an obstacle-rich environment [36] 21

4.1 High-level diagram of the robotic system 23
4.2 ROS2 communication architecture of the robotic system 24
4.3 Leader-Follower formation with variable number of UAVs in the system 26
4.4 Assignment of POIs to UAVs 28
4.5 Robotic solution and a POI 28
4.6 UAV engaging in pollination mission and formation readjustment 30
4.7 UAV orientation according to formation 31
4.8 Collisions due to multiple mission management 31
4.9 Collisions due to assignment issues 32
4.10 UAV readjusmtent according to battery life 32
4.11 UAV State Machine 33

5.1 Webots mesh relative to Quinta do Seixo 35
5.2 UGV and UAV models in Webots 35
5.3 UAV Deployment 36
(a) UAV Deployment - Instance 1 36

(b) UAV Deployment - Instance 2 . . . 36
(c) UAV Deployment - Instance 3 . . . 36
(d) UAV Deployment - Instance 3 . . . 36
5.4 Hungarian Algorithm Simple Assignment . . . 38
(a) Hungarian Algorithm Deployment - Instance 1 . . . 38
(b) Hungarian Algorithm Deployment - Instance 2 . . . 38
(c) Hungarian Algorithm Deployment - Instance 3 . . . 38
(d) Hungarian Algorithm Deployment - Instance 4 . . . 38
5.5 Hungarian Algorithm with a pre-assigned vehicle . . . 39
(a) Hungarian Algorithm Pre-Assigned - Instance 1 . . . 39
(b) Hungarian Algorithm Pre-Assigned - Instance 2 . . . 39
(c) Hungarian Algorithm Pre-Assigned - Instance 3 . . . 39
(d) Hungarian Algorithm Pre-Assigned - Instance 4 . . . 39
5.6 Leader-Follower configuration in monitoring mission . . . 40
(a) Leader-Follower - Instance 1 . . . 40
(b) Leader-Follower - Instance 2 . . . 40
(c) Leader-Follower - Instance 3 . . . 40
(d) Leader-Follower - Instance 4 . . . 40
5.7 Pollination Mission with formation readjusting . . . 41
(a) Pollination Mission - Instance 1 . . . 41
(b) Pollination Mission - Instance 2 . . . 41
(c) Pollination Mission - Instance 3 . . . 41
(d) Pollination Mission - Instance 4 . . . 41
5.8 UGV and UAV models in Webots . . . 42
5.9 Complete Mission with 1 UAV . . . 43
(a) Complete Mission - 1 UAV - Instance 1 . . . 43
(b) Complete Mission - 1 UAV - Instance 2 . . . 43
(c) Complete Mission - 1 UAV - Instance 3 . . . 43
(d) Complete Mission - 1 UAV - Instance 4 . . . 43
5.10 Complete Mission with 2 UAVs . . . 43
(a) Complete Mission - 2 UAVs - Instance 1 . . . 43
(b) Complete Mission - 2 UAVs - Instance 2 . . . 43
5.11 Complete Mission with 3 UAVs . . . 44
(a) Complete Mission - 3 UAVs - Instance 1 . . . 44
(b) Complete Mission - 3 UAVs - Instance 2 . . . 44
(c) Complete Mission - 3 UAVs - Instance 3 . . . 44
(d) Complete Mission - 3 UAVs - Instance 4 . . . 44
5.12 Complete Mission with 4 UAVs . . . 45
(a) Complete Mission - 4 UAVs - Instance 1 . . . 45
(b) Complete Mission - 4 UAVs - Instance 2 . . . 45

List of Tables

2.1 Characteristics of the simulators 12

3.1 Literature review results 20

4.1 Camera Specifications 29

5.1 Error evolution when the UGV's velocity is 0 m/s 36
5.2 Error evolution when the UGV's velocity is 0 m/s 40

Abreviaturas e Símbolos

UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
MPF	Moving Path Following
ROS	Robot Operating System
PICO	Population - Intervention - Comparison - Outcome
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
PID	Proportional – Integral – Derivative
VL	Virtual Leader
POI	Point of Interest
FOV	Field of View

Chapter 1

Introduction

1.1 Context

As of today, robotics plays a crucial role in the context of agriculture, providing a safer and less labor-intensive workplace while maintaining the environmental integrity necessary for ideal agricultural production. In agriculture, pollination is an essential stage in the plant's reproductive cycle, greatly influencing crop quality and making it crucial for sustainable agriculture production [1]. According to the Food and Agriculture Organization (FAO), close to 75% of the world's crop production of fruits and seeds for human use depends on pollinators [2].

Understanding pollination is the key to ensuring quality crops while preserving the ecosystem's balance [3]. Pollination is the act of deploying pollen grains from the male anther to the female flower stigma. Incomplete and variable pollen delivery directly correlates to a reduced yield mean, decreasing homogeneity among crops with a greater dependence on pollinators.

This thesis explores the benefits of the cooperation between ground and aerial vehicles to provide a more efficient and cost-effective alternative to vehicle-mounted devices while being automated, providing less labor-intensive procedures. Different swarm concepts and methodologies are addressed to optimize tasks performed by the system.

1.2 Motivation

Insects are the primary pollinators responsible for producing quality food, fiber, and medicine crops. In this regard, more than 90% of the world's top 107 crops are pollinated by bees [4], with honey bees (*Apis mellifera*) being the primary pollinators [5]. Furthermore, honey bees present greater variability in their foraging behavior, mostly caused by the sunflower's nectar sugar concentration [6]. This aspect contributes to a greater plant genetic diversity and increases adaptability in changing environments.

With the decline of bee population [7] and a lack of efficient solutions to complement the pollination process, there is an urgent need for technological advances in this field. Improving artificial pollination techniques may increase pollination efficiency and sustainability. Manual

artificial pollination requires a human operator to use a tool to connect the anthers of the male flower with the female flower's stigma. This method is proven to be effective but is also labor-intensive and, therefore, less cost-effective. On the other hand, vehicle-mounted devices, despite automating the process of pollination, are not as accurate as manual pollination methods, which causes much pollen to be wasted and, as a result, increases costs [8].

There have been multiple proposals to explore the potential of UAVs and UGVs as artificial pollinators [9]. UAVs have the potential to tackle this issue but face a major drawback: battery life. Associating multiple UAVs working collaboratively as swarms with UGVs may be a possible solution. Having UGVs support UAVs by carrying specific payloads or even allowing them to dock and recharge can make a difference in terms of optimizing battery life. The main goal is to enhance this process and to provide these services with a low environmental impact.

1.3 Thesis Main Contributions

This work explores the existing literature as it dives into the world of simulators and analyzes real-life scenarios in the context of artificial pollination. The main goal is to implement a swarm architecture into a heterogeneous robotic system to explore the benefits of this collaborative work in the context of agriculture while validating our solution through the simulator.

In regards to the vehicle controllers, it is necessary to assess the demands and constraints of the system. Given the limited battery life of the UAVs, a cautious analysis must be done to preserve the battery life while optimizing the vehicle's performance.

For the simulation, firstly, it is crucial to determine the ideal simulator to validate the system. Highly complex and computationally expensive solutions may not provide the best performance in the presence of multiple robots, while simplistic simulators may not represent the dynamics of real-world interactions accurately.

This thesis main contributions are:

- Systematic review on the domain of swarm robotics;
- A virtual environment simulator with two types of vehicles (ground and aerial) in the context of pollination;
- A swarm mission and a leader-follower controller for multiple heterogeneous robots;

1.4 Structure

The structure of this document will be the following:

- Chapter 2 presents the theoretical aspects addressed in the context of the work;
- Chapter 3 discusses the methodologies used for the literature search and bibliographic review;

- Chapter 4 describes the control and architecture behind Pollinationbots;
- Chapter 5 shows the implementation and simulation of the robotic system and discusses the obtained results;
- Chapter 6 sets out the most relevant conclusions to be considered in the thesis.

Chapter 2

Background

This chapter presents the fundamental concepts and tools that will be considered during the development of this work. The presentation of these topics improves the bibliographical research and the understanding of the theoretical knowledge of the scientific domain where the work is contextualized. Firstly, the concept of swarm robotics will be explained. Then, a classification of the different types of vehicles outlining their advantages and disadvantages. Finally, a survey will be assessed regarding the simulators and frameworks used in the context of robotics.

2.1 Swarm Robotics

The concept of swarm robotics has its roots in nature, where natural-based swarm behaviors have emerged. Some examples of this occurrence are ant and bee colonies (Figure 2.1). The underlying concept rests on the principle that while one entity may not possess great problem-solving capacity, the group as a whole can solve complex tasks. This phenomenon is only possible due to effective information exchange and a high level of coordination among the individuals [10].



Figure 2.1: Intra-organizational swarm interactions among ant[1] and bee[2] colonies.

[1] https://zeeshanusmani.com/2019/11/27/swarm-intelligence/
[2] https://medium.com/@vegasbees/will-bees-join-another-hive-e5acca0bbca4

Some of the main characteristics of swarms are:

- **Scalability**: Agents can join or leave the swarm without disrupting it. Being able to adapt to different sizes of population is a beneficial property in a real-time system.
- **Versatility**: The individuals in the system can have multiple roles and perform various tasks according to the group's needs.
- **Robustness**: Even if some agents cease their functions, the swarm remains functional, although it may impact the system's performance.

These characteristics can be transposed into robotic scenarios. Decentralized and autonomous control [11] provides multiple possibilities for implementing a scalable and robust system. If an agent of the system ceases its functions and leaves the swarm, the group is able to reorganize itself and fill the gap left by the unit. The same happens when a new agent arises. When one unit recognizes and interacts with an external agent, it is unnecessary to communicate this event to the rest of the swarm; this new member is now obeying the rules by which the swarm is governed.

The Ant Algorithm [12] is useful for optimizing complex control problems in robotic swarms. This algorithm is based on the concept of stigmergy [13]. Stigmergy is based on the principle that the work performed by an agent influences how the rest of the swarm will act and perform its actions. This allows systems to understand how many units should be deployed to carry out a specific task.

2.2 Swarm Dynamics

In swarm systems, there are 2 types of control:

- Centralized Control
- Decentralized Control

In centralized control, one single entity has all the information regarding the other units. All the actions are determined by the leader of the swarm. In decentralized control, each unit has information regarding its whereabouts and states. Each member needs to effectively communicate its data and, collectively, a decision is made [14]. Through Figure 2.2, in the centralized case, the yellow node represents the leader of the set of agents. This entity has access to all the information of the remaining agents, as it is represented by the lines connecting each agent and the leader. The leader is in charge of all the followers's actions (black nodes).

Concerning the decentralized case, there are only black nodes with connections to the nearest neighbors. This represents the fact that, in decentralized systems, no agent has all the information from the set of individuals and is not in charge of making global decisions. Each unit has to react according to its surroundings.

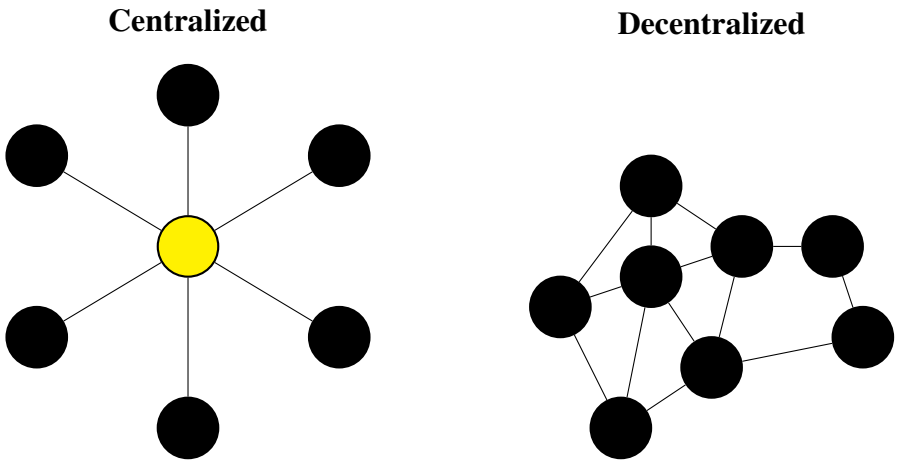


Figure 2.2: Centralized *vs* Decentralized Systems

Regarding heterogeneous robotic systems, in these scenarios, a pre-assignment of who the leader and follower will be is necessary. A static assignment of these roles may bring some downsides. In a system composed by a UGV and a UAV, if the terrain is stable and relatively smooth, the UGV might take the leadership, on the other hand, if there is an irregular terrain where the UGV does not have crucial information regarding the surroundings, then the UAV (due to its altitude) should take the leadership [15].

Global shape formation is a key aspect of swarm intelligence. In nature, birds fly together while forming a "V" shape to reduce air resistance, ants form a line to optimize the path to a food source. Shape formation is usually associated with the optimization of a specific task, such as mapping and exploring an area [16].

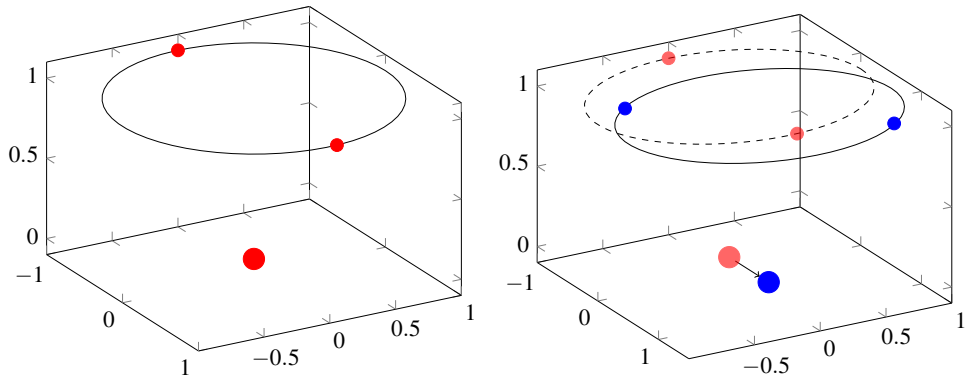


Figure 2.3: Moving Path Following

Moreover, in the formation strategy, to assist tasks such as mapping and monitoring, one possible implementation would be to integrate a Moving Path Following [17] approach where a UAV would follow and converge upon a predetermined moving path within a fixed referential frame.

On the left side of Figure 2.3 are represented one UGV (the larger red dot below the two smaller red dots) and two UAVs (the smaller red dots). The UAVs trajectory is defined by the

circular line. The image on the right shows a future state of the system where the new positions of the vehicles are defined by the blue dots. The UAVs maintain the fixed trajectory around the UGV, this way, they can assist it in any possible task. The predetermined path of the UAVs is dynamic and should be modelled according to the specific tasks developed by them.

This way, the UAVs and UGV could assist and complement each other. Given the fact that the UAVs have a limited battery life, in case they need to charge, given their proximity to the UGV, it would be possible for them to dock and recharge. The ground vehicle would also benefit from this association since the UAV would be able to map the surroundings finding possible obstacles or regions of interest that would not be possible to identify in the UGV frame [18].

2.3 Unmanned Vehicles

To develop a heterogeneous robotic system, it is necessary to understand what types of unmanned vehicles exist, as well as their characteristics and constraints.

2.3.1 Unmanned Ground Vehicles

UGVs are characterized by their robustness and stability. Their configurations can impact the types of terrain they can traverse, the payload they can carry, and their maneuverability. The configuration of the steering system is an essential domain to address, given its impact on the vehicle's locomotion [19].

Some of the more common steering schemes are:

- Skid steering
- Articulated steering
- Coordinated steering
- Independent steering

Skid Steering

Skid steering is extensively used in wheeled and tracked vehicles. In this configuration, steering is enabled by controlling the wheels' velocities on the chassis' opposing sides. The difference in the velocities of the two sides defines the vehicle's turning radius. The wheels do not change orientation with respect to the chassis.

These vehicles are highly robust and flexible; hence, they are used in military, agriculture, and space exploration (Figure 2.7).

However, creating kinematic and dynamic models is a complex process given this locomotion scheme. Accounting for accurate models describing wheel slippage in wheeled vehicles requires certain considerations of the interactions between the wheels and the terrain [20].

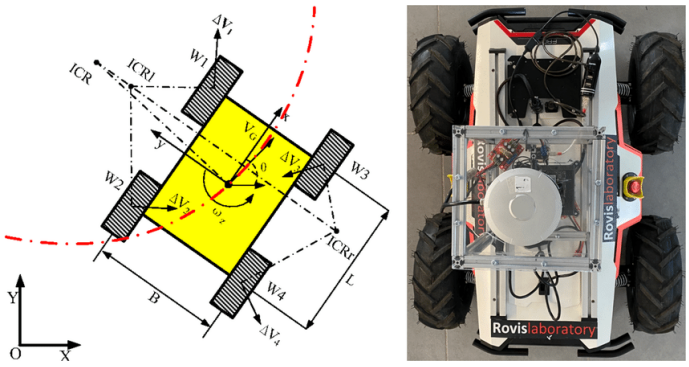


Figure 2.4: The kinematics schematic of skid-steering mobile robot [21].

Articulated Steering

In these configurations, the vehicle comprises two frames (front and rear). The rear frame steers about a specific pivot point that connects the two frames.

There are two types of articulated steering:

- Passive Articulated Steering - this configuration relies on the natural motion of the vehicle and the mechanical link to steer the vehicle.
- Active Articulated Steering - the two frames are connected by an active revolute joint that changes the angle between the two frames.

While active articulated steering provides better maneuverability and performance, it is quite expensive and demands a complex control system. Passive articulated steering offers a simpler and cost-effective solution with lower responsiveness and adaptability.



Figure 2.5: The kinematics schematic of skid-steering mobile robot [22].

Coordinated Steering

This methodology, also known as synchronized steering, is applied in multi-axle vehicles and is based on the coordination of the front and rear axles to optimize the vehicle's performance through mechanical coupling. The Ackermann steering mechanism is a case of coordinated steering. Some of the advantages of this configuration are:

- Increases controllability of vehicles by using shorter tie rods;
- Prevents the vehicle from slipping outward while turning the vehicle.

The improved maneuverability of this steering configuration makes it adequate for large-axle vehicles, such as trucks and agriculture machinery.

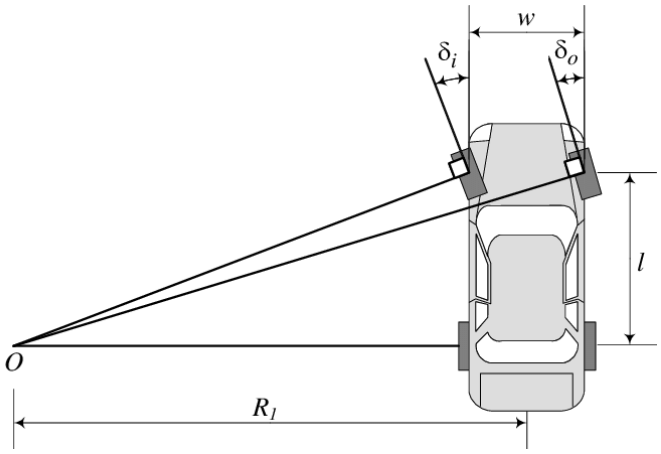


Figure 2.6: A front-wheel-steering vehicle with Ackermann steering [23].

Independent Steering

In independent steering, each wheel is steered individually. The change in the vehicle's heading can be controlled by modulating each wheel's steering angle. This allows for greater vehicle maneuverability and, given the coordination of driving and steering, leads to low energy consumption.

A vehicle with this configuration has improvements in trajectory performance due to the steering of the rear wheels, which also contributes to its lateral stability. Although this steering mechanism has complex actuation and coordination control, it provides great maneuverability for mobile robots.

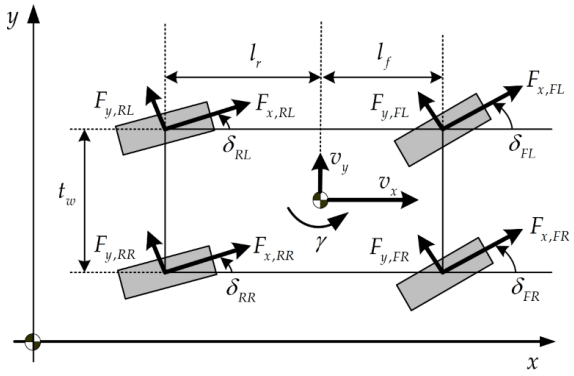


Figure 2.7: Free-body diagram of four-wheel with independent steering [24].

2.3.2 Unmanned Aerial Vehicles

Regarding UAVs, there is no specific standard classification. Various classifications are used based on parameters such as speed, weight, and size. The selected classification is based on the main propulsion configurations that UAV employ [25]:

- Single-rotor
- Multi-rotor
- Flapping-wings
- Fixed-wing

Single-rotor

Single-rotor aerial vehicles are practical in situations where they are required to lift heavier payloads. Since it is only necessary to power a single propeller, its efficiency is greatly increased. The main downside is its flight stability and directional control due to only having one rotor (Figure 2.8).



Figure 2.8: Single-rotor[3]aerial vehicle

[3] https://www.roboticgizmos.com/prodrone-pdh-03/

Multi-rotor

Multi-rotors are characterized by having exceptional flight stability, allowing them to control pitch, yaw, and roll with ease. The battery life is inferior to that of drones with only one rotor having a limited flight time (Figure 2.9).



Figure 2.9: Multi-rotor[4]aerial vehicle

Flapping-wings

These aerial vehicles are inspired by nature, mainly by birds and insects. With the help of mechanical wings, these vehicles can fly but have a very complex control system and relatively low robustness (Figure 2.10).

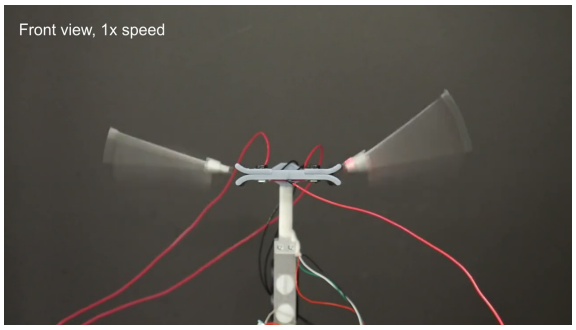


Figure 2.10: Flapping-wings[5]aerial vehicle

Fixed-wing

Fixed-wing are often used in mapping tasks of large terrain extensions due to their increased battery life and endurance while providing a satisfactory payload. It is a very flexible system with reduced costs associated (Figure 2.11).

[4] https://siberriavi.artstation.com/projects/Zo1e0
[5] https://www.bristol.ac.uk/news/2022/february/flapping-wing-robots.html



Figure 2.11: Fixed-wing[6]aerial vehicle

2.4 Simulators

Simulating our system and understanding how it would translate to the real world is essential to understanding how it would behave in a real-life scenario.

Most swarm robotic applications are confined to laboratory facilities. As a way to explore and exploit this field, generally, it is necessary to resort to simulators. This section will explore the state-of-the-art simulators, detailing their characteristics.

Table 3.1 shows the characteristics of the different types of simulators explored:

Table 2.1: Characteristics of the simulators

Simulator	Computational Power	Performance	Physics Engine
Gazebo	High	Poor	ODE (Open Dynamics Engine)
Webots	Medium-High	Good	ODE (Open Dynamics Engine)
ARGoS	Very Low	Very good	Customizable
Farming Simulator	High	Good	GIANTS (Proprietary)

2.4.1 Gazebo

Gazebo is an open-source three-dimensional simulator that allows the simulation of multiple robots with high fidelity regarding real-world physics. It can accurately represent and simulate gravity, torque, friction, etc. Supporting the implementation of multiple sensors, objects, and cameras, essential features to understand the behavior of specific devices without implementing them in the real world (Figure 2.12).

[6] https://dominotheory.com/project-replicators-switchblade-problem/



Figure 2.12: Simulation of an agricultural robot in Gazebo[7]

2.4.2 Webots

Webots is a multi-platform 3D robot simulation software with several programming interfaces such as MATLAB, C, ROS, Python, etc. Its physics are generated in ODE (Open Dynamics Engine), where it is possible to adjust the collision physics and body dynamics presented in the simulator.

It can produce interactive environments and provides a complete development environment for programming, modeling and simulating (Figure 2.13).

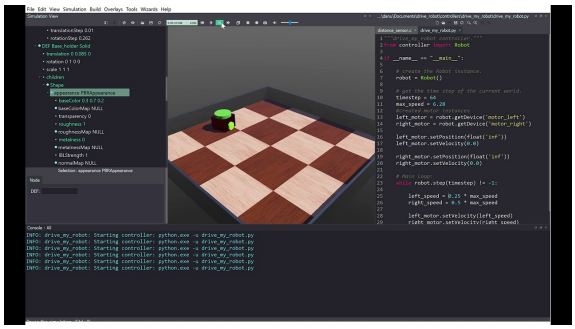


Figure 2.13: Simulation of a two-wheeled robot in Webots[8]

2.4.3 ARGoS

ARGoS is a large-scale robot simulator that can efficiently simulate many robots (Figure 2.14). It was developed to provide a simulator that managed to support large-scale robot swarms in different environment dynamics.

[7] https://www.youtube.com/watch?app=desktop&v=98KOrmXzLks
[8] https://ardupilot.org/dev/docs/sitl-with-webots-python.html
[9] https://www.designboom.com/technology/haldewag-plan-bee-drone-03-03-2017/

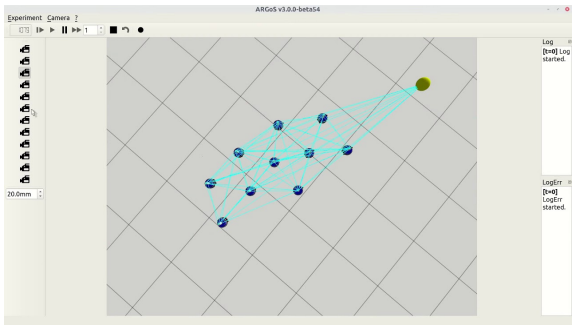


Figure 2.14: Simulation of a swarm system in ARGoS[9]

2.4.4 Farming Simulator

Farming Simulator is a software initially designed to be a game (Figure 2.15). However, with the number of features and upgrades the physics engine has received, it may become a robot simulator that can be used to test realistic scenarios in the context of agriculture.



Figure 2.15: Simulation of agricultural machinery in Farming Simulator[10]

2.5 ROS2 (Robot Operating System)

Robotic frameworks are crucial components in the process of making a functional system. Breaking down complex structures into smaller parts is essential in developing a robot. ROS2 was used to create an interface for communicating with the simulator and facilitating communication among system agents [26].

ROS2 consists of independent computing processes called nodes. Communication among nodes is performed through a publisher/subscriber service. Each node should be responsible for a unique purpose, such as controlling the wheel motors. Topics are communication channels that allow different system parts to have access to vital information. Nodes can publish and subscribe to any number of topics.

Robustness, reliability, and modularity are essential features in this specific robotic system, and for the system to meet these particular standards, ROS2 was used.

[10] https://tecmasters.com.br/farming-simulator-ganha-dois-novos-modos/

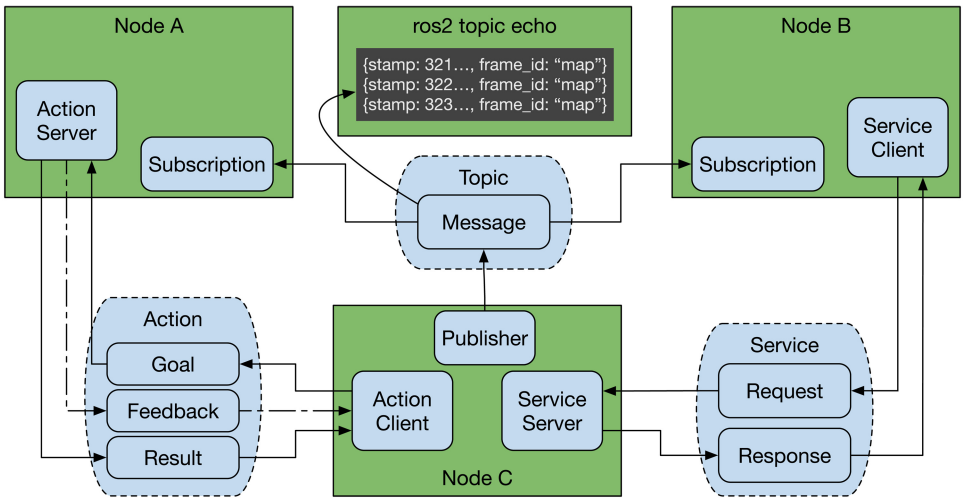


Figure 2.16: ROS 2 node interfaces [27]

Topics are not the only communication method in ROS2, other approaches involve:

- **Services**: This communication method is based on a call-and-response model rather than the publihsher-subscriber. Topics allow to receive continuous data streams, while services provide data when they are specifically called by a node.
- **Actions**: Actions behave similarly to services; they are composed of three parts: the goal, feedback, and result. Once an action is called, a certain value is expected to be achieved. The action will provide constant feedback on the current value of the specific parameter until it achieves the desired goal.

Chapter 3

Bibliography Review

This chapter covers the bibliographic review on the development of a heterogeneous robotic system composed by a UGV and multiple UAVs associated to agricultural tasks. The first section will address the main questions regarding the proposed topic. Afterward, the implemented methodology, criteria and search strategies will be discussed. Finally, the selection process and result analysis will take place.

3.1 Systematic Review Objectives

This review will focus on the literature relative to heterogeneous and swarm robotic systems. In essence, the main questions that this section intends to cover are:

- What are the main applications of UGVs and UAVs in the context of agriculture?
- How is communication carried out between the swarm?
- What are the path planning strategies implemented?
- What are the limitations of the system?
- How is the performance of the procedures evaluated?
- What are the controller strategies for agriculture's heterogeneous/swarm robotic systems?

By framing the review in the Population - Intervention - Comparison - Outcome (PICO) framework to understand the approaches and results this section is interested in:

- **Population**: heterogeneous robotic systems
- **Intervention**: Mapping, monitoring and path planning
- **Comparison**: *not applicable to the current study*
- **Outcome**: Implemented methodologies

3.2 Methodology

Systematic reviews diverge from traditional reviews in multiple ways. Traditional reviews often are descriptive and focus on a set of articles chosen by the author, introducing some bias and subjectivity into the research and analysis [28]. The most common standard is the PRISMA statement, a standard that was originally designed for health interventions but has demonstrated that it is suitable for different domains [29].

This section intends to explore this standard, detailing the research strategy conducted for this study. A brief discussion and analysis of the results will take place.

3.2.1 Eligibility Criteria

Regarding this systematic review, only the publications between 2018 and the day of the survey were considered. The remaining articles would be eligible if the following requirements were verified:

- The publication is written in English.
- The publication is older than 2018.
- The publication is open access.

After carefully analyzing the publications, each one is assessed and verified to see if it complies with the objectives of the work. Each quality question is given an evaluation:

- Yes (1.0)
- Partially (0.5)
- No (0.0)

The publications with a score lower than 2.0 were excluded from the final review. The quality assessment was developed according to the following questions:

- Does the paper refer path planning strategies?
- Does the paper refer to obstacle avoidance?
- Does the paper mention the applications of the robot?
- Does the paper mention how is communication performed?
- Does the paper refer how the performace of the procedures is evaluated?
- Does the paper refer the control strategies for heterogeneous/swarm robotic system configurations in agriculture?

Regarding the survey, 3 different databases were utilized to conduct this study:

- IEEE Digital Library [30]
- ISI Web of Science [31]
- Scopus [32]

In these databases, the most effective way of obtaining publications according to the areas of interest is through a query. Producing this query is an essential step in the systematic review. The query used is this study was the following:

("swarm robot" AND ("uav" OR "ugv" OR "heterogeneous robot*")) AND ("mapping" OR "monitoring")*

According to this query, there is an interest among publications that elaborate on topics such as swarm robotics, hence the "*swarm robot**", when associated with keywords such as "*uav*", "*ugv*" and "*heterogeneous robot**". The aim of using these technologies in the context of agriculture would be in tasks related to mapping and monitoring, hence the "*mapping*" and "*monitoring*".

3.2.2 Selection Process

The query was then inserted into the databases and 57 publications were reported, meeting the requirements established in the subsection *Eligibility Criteria* (Subsection 3.2.1). The distribution of publications among the databases is the following:

- Scopus has 26 publications.
- IEEE Digital Library has 25 publications.
- Web of Science has 6 publications.

Figure 3.1 represents the process performed for this systematic review, according to the PRISMA standard. Initially, with the help of Parsifal, all the duplicated publications across the databases were removed. Then the title and abstract of each document were carefully analyzed to identify the publications that matched the area of interest of this systematic review. Finally, after reading the publications in their entirety, an evaluation was carried out according to the quality assessment questions proposed.

This analysis provides a robust foundation regarding the current state-of-the-art swarm technologies, identifying possible gaps in the existing literature and highlighting areas where advances in the field should be made.

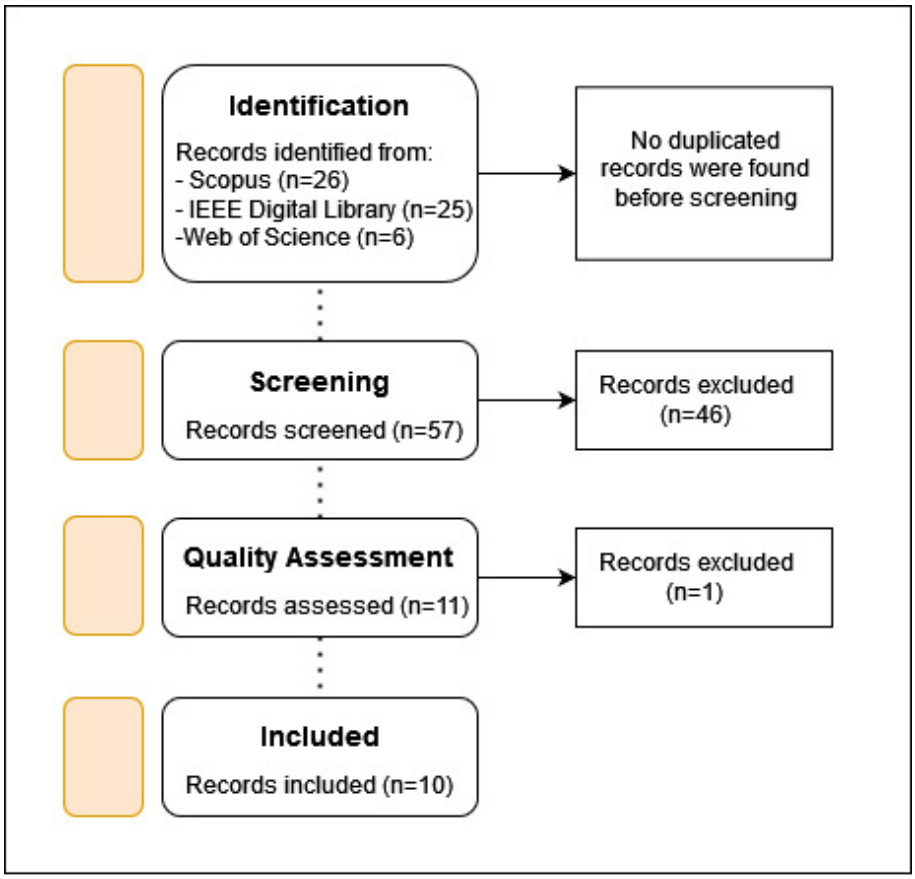


Figure 3.1: PRISMA Workflow in the context of this systematic review

3.3 Search Results

Concerning the existing literature, three main topics were focused on according to the desired characteristics of our robotic solution: path planning, its application, and mission control.

Understanding path planning algorithms in the context of robotic swarms is relevant, given the configuration of our system. Swarm systems require meticulous path planning to deal with possible interactions among the group. The application and environment where these approaches are used are essential to determine if these technologies are relevant and can spur innovation in a specific field. The architectures and mission controllers adopted in the literature are fundamental to understanding the system's organization and how the swarm interacts. Table 3.1 compiles this analysis.

Table 3.1: Literature review results

Reference	Path Planning	Application	Mission Control
[33]	✓	✓	✓
[34]	✓	✓	✓
[35]			✓
[36]	✓	✓	✓
[37]	✓	✓	✓
[38]	✓		✓
[39]	✓	✓	✓
[40]	✓		✓
[41]	✓		

3.3.1 Path Planning

Several approaches were analyzed in the field of path planning. We focused on strategies that were not dependent on heavy computational calculations to preserve the battery life of our system. Methodologies such as Reinforced Learning [34], Self Organized Maps (SOM) [41] or Graph-based AI Planners [36] were strategies that did not suit our requirements as they are based on machine learning algorithms and demand complex calculations. Probabilistic approaches like Markov Random Fields [38] also require a lot of computational resources. We focused on algorithms including Artificial Potential Fields [33] (Figure 3.2) for their efficiency and obstacle avoidance abilities.

The *Carrot-Chasing* algorithm [42] is also a viable option. This algorithm is a geometric strategy that consists on steering the UAV towards a Virtual Leader (VL) [43] located on the path. The UGV can be in charge of the path generation to improve the efficiency of the UAVs and preserve their battery life.

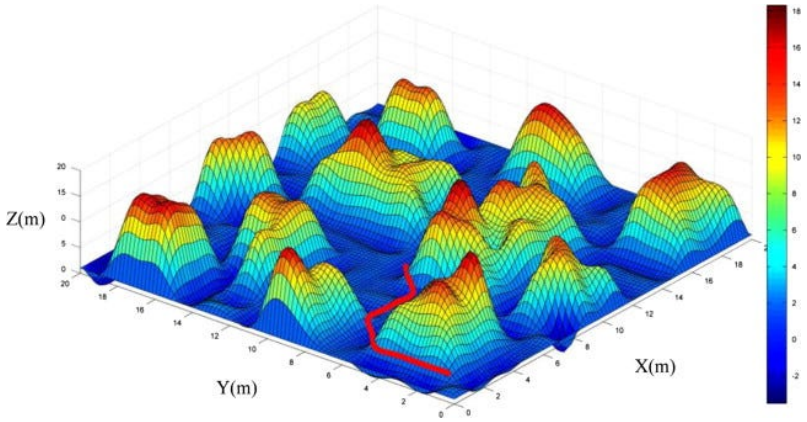


Figure 3.2: Obstacle visualization in 3D space using Artificial Potential Fields

3.3.2 Applications

The background in which these swarm applications are implemented is diverse, showing the versatility of these methodologies (Figure 3.3). From space exploration [34], agriculture [35] to search and rescue missions [39], swarm robotics is an alluring field that can lead to improvements in multiple areas.

Agriculture is a domain that would benefit from swarm-based approaches to tasks like monitoring or mapping, providing efficient coverage of large areas in a short time due to the collaborative work of the robots.

Swarm robotics can bring several advantages to various fields due to its scalability, adaptability, and robustness, improving the efficiency and accuracy of the operations.



Figure 3.3: Trajectories of UAVs flying through an obstacle-rich environment [36]

3.3.3 Mission Control

In the domain of mission control, various architectures were presented, according to the context in which they were implemented. Decentralized robotic swarms with behaviors inspired by biological systems [37] is an interesting approach that allows for a more robust and scalable system. Extreme systems that prioritize collecting the maximum amount of information in the shortest time possible require complex mission strategies. Visiting certain locations can have different weights associated [39] representing the importance of visiting that location in the current scenario.

Effective information exchange among the swarm is essential to ensure the system's success. Although the protocols used to establish communication among the agents of the swarm are not explicitly mentioned, there are key ideas in common in the universe of publications regarding communication:

- A safe and reliable communication system is important to ensure the consistency of information exchanged among the agents; and,
- The number of agents it is possible to detect and communicate with should be accounted as a factor in control laws.

Chapter 4

Proposed Solution

This chapter presents the intended requirements for this project and analyzes the architecture, explaining in detail each functional block, as well as how each part of the system is interconnected and its communication flow.

4.1 Requirements

The main objectives of this work were:

- Create a virtual environment simulator with two types of vehicles (ground and aerial) in the context of pollination;
- Develop a swarm mission and leader-follower controller for multiple heterogeneous robots.

The system must meet several key requirements to implement these outlined goals successfully. These key requirements can be categorized into two different groups:

1. **Functional Requirements**
    - The simulation must present real-life scenarios and dynamics, as well as the interactions between the vehicles and the environment;
    - The simulation must include a ground vehicle and multiple aerial vehicles;
    - The system must present real-time coordination and be responsive to any changes in the environment.
2. **Non-Functional Requirements**
    - The system must be modular to allow the future implementation of upgrades or features to the solution.
    - The system must be robust and ensure continuous operation.

4.2 The Proposed Solution

The robotic solution is comprised of a UGV as the operation's leader and multiple UAVs whose purpose is ultimately to aid the UGV by mapping the terrain and transporting the pollen from the male to the female flower. This solution was given the name Pollinationbots.

The system will adopt a centralized approach where the UGV can access parameters such as the position and velocity of all swarm agents. This way, the UGV handles most of the control and computing, improving the efficiency of the UAVs and preserving their limited battery.

Figure 4.1 presents a high-level diagram of the system considering one UAV.

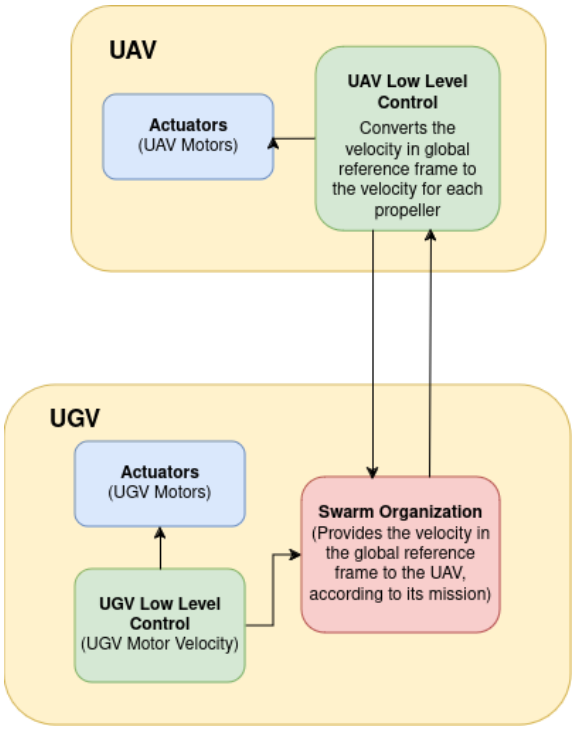


Figure 4.1: High-level diagram of the robotic system

In this specific example, only a UGV and a UAV are considered to facilitate its comprehension. Regarding the system, three main domains were considered:

- **Low Level Control** - This block computes the desired velocities for the motors to perform specific movements.
- **Actuators** - In this block, the motors are actuated with the desired velocity provided by the Low Level Control block.
- **Swarm Organization** - This block manages the formation of the swarm, according to the needs of the system and the mission. It provides the desired global velocities to each UAV in the system.

Communication Architecture

To create an interface to communicate with the simulator and facilitate communication among system agents, we used the Robot Operating System 2 [26]. Effective communication is the foundation of working systems, especially in robotic environments. Robust communication protocols are key to an optimized robotic system to address real-time decision-making and prevent any possible conflicts in the system [44]. ROS2 provides a modular and robust framework that supports the system's prerequisites. Using ROS2 nodes enables us to address the system's scalability more efficiently, given that adding a different node does not influence past implementations realized in different nodes.

Figure 4.2 shows the ROS2 communication architecture of the robotic system, associating each functional block to the high-level diagram blocks (Figure 4.1).

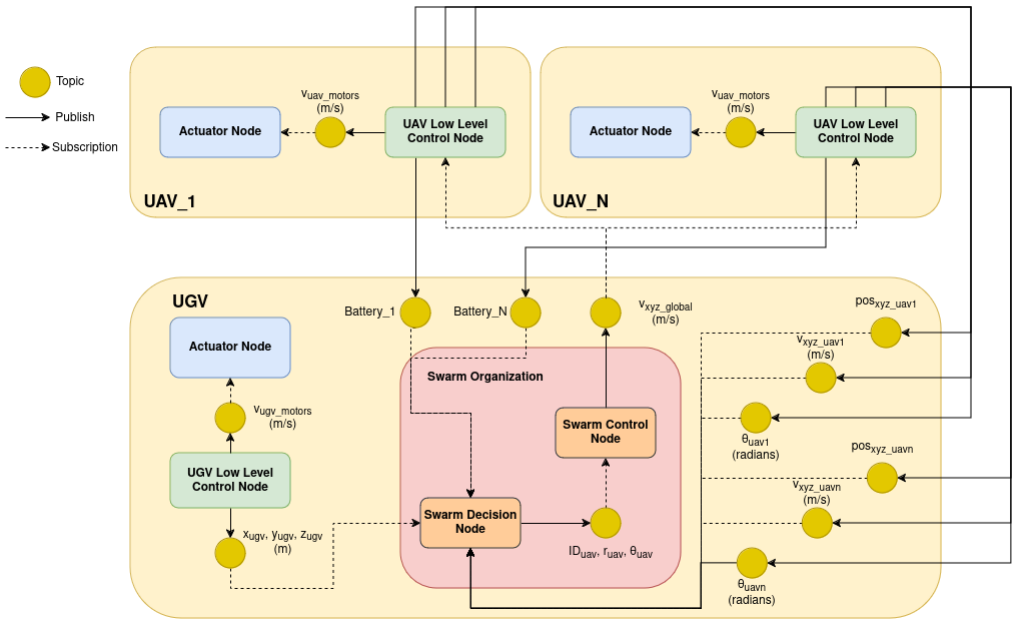


Figure 4.2: ROS2 communication architecture of the robotic system

4.2.1 UGV Low Level Control

The low level control of the UGV converts the desired global linear and angular velocity into the velocity for each of the motors. This four-wheeled vehicle follows the differential drive configuration (Figure 2.7), where the velocity of each wheel follows the following equations (Equation 4.1 and 4.2).

$$v_{left_wheels} = \frac{v_{linear} - (v_{angular} * r_{axis})}{r_{wheel}} \tag{4.1}$$

$$v_{right_wheels} = \frac{v_{linear} + (v_{angular} * r_{axis})}{r_{wheel}} \tag{4.2}$$

where r_{axis} represents half of the axis length between the two wheels, r_{wheel} represents the wheel's radius, and v_{linear} and $v_{angular}$ represent the desired linear and angular velocity.

This node also publishes information regarding its position on the topic to which the Swarm Decision node is subscribed to. The positions that the UAVs adopt depend on the current position in the leader-follower controller, so the information regarding the whereabouts of the UGV is extremely important.

4.2.2 UAV Low Level Control

In this node, it is necessary to transform the global frame velocities received through the topic into motor velocities for the UAV to perform the desired trajectories.

To achieve this, we used a PID controller. In the context of this work, we already had a PID controller that took the current and desired velocities of the systems and outputted the desired velocities for each propeller of the UAV.

Furthermore, this node would publish the UAV's position, velocity, orientation, and battery into the topics subscribed by the Swarm Organization nodes to feed the swarm controller and to be able to supervise the behavior of each UAV.

4.2.3 Swarm Organization

In this block, all the control relative to the positions of each UAV is managed. The control laws, optimization algorithms, and computer vision techniques will be addressed.

Control

Regarding the system's control, each position will be defined according to the desired *radius* and *angle* calculated in the Swarm Decision Node. This approach is denominated as Carrot-Chasing algorithm [42]. This algorithm is a geometric strategy that consists on steering the UAV towards a Virtual Leader (VL) [43] located on the path. To achieve this, a reference position is created based on the system's leader location, in this case, the UGV. Equations 4.3 and 4.4 represent the desired position:

$$x_{uav} = x_{ugv} - radius \cdot \cos(\theta_{ugv} + angle) \tag{4.3}$$

$$y_{uav} = y_{ugv} - radius \cdot \sin(\theta_{ugv} + angle) \tag{4.4}$$

where *radius* is the distance from the UGV to the VL, and *angle* is the difference in orientation in regards to the front of the robot.

This implementation enhances the process of swarm management and distribution of agents in the formation. By reducing the parameters responsible for generating the positions of the UAVs to *radius* and *angle*, it simplifies the control and coordination of the swarm while allowing rapid reconfiguration of the assigned positions.

To track the desired signal, we consider the agent in a 2D plane, having $x = (x_1, x_2)$ as the states of the system and $u = (u_1, u_2)$ as the control inputs, with simple integrator dynamics (Equation 4.5):

$$\dot{x} = u \tag{4.5}$$

where x represents the UAV's position and $\dot{x}$ its derivative, the velocity.

The feedback controller (Equation 4.6) is given by:

$$u = -K\tilde{x} + \dot{x}_d, \qquad \tilde{x} = x - x_d \tag{4.6}$$

where K is a 2×2 positive-definite matrix gain, $\tilde{x}$ is the 2D tracking error and x_d is the desired state. In this specific case, the derivative of the desired state corresponds to the UGV's velocities.

To compute the real-world positions, firstly, it is necessary to create a matrix with the radius and angle correspondent to each desired position (Algorithm 1):

Algorithm 1 Calculate VL's Parameters

Require: Number of UAVs $\rightarrow N$, 2D array A of size $N \times 2$, Radius $\rightarrow R$
Ensure: $N > 0$, $R > 0$
1: **for** $i \leftarrow 0$ to N **do**
2: $\quad A[i][0] \leftarrow R$
3: $\quad A[i][1] \leftarrow \frac{2\pi}{N} \cdot (i+1)$
4: **end for**
5: **return** A =0

Having the parameters stored, it is possible to calculate the desired position for the UAVs (Equation 4.3 and 4.4). The resulting positions and formations can be observed in Figure 4.3, where each red point represents a distinct UAV and the blue point represents the UGV.

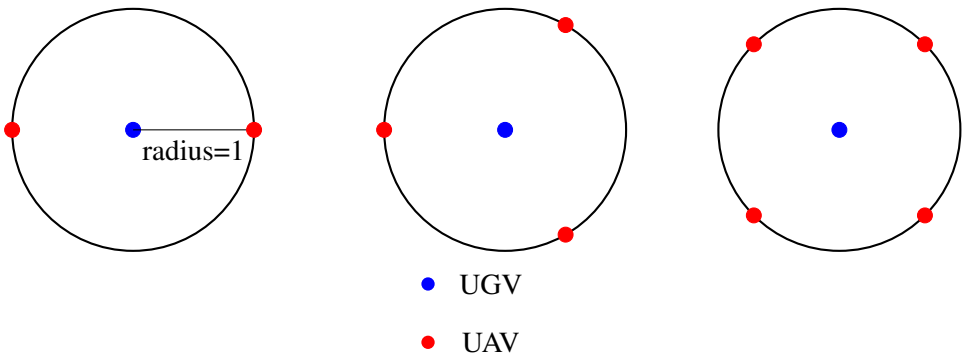


Figure 4.3: Leader-Follower formation with variable number of UAVs in the system

Finally, each UAV must be assigned to the desired position. Since the aim is to minimize the total distance the UAVs cover to assume the desired formation, the Kuhn-Munkres Algorithm [45], also known as the Hungarian Algorithm, was used to achieve this. This method is used to find the

maximum-weight matching in bipartite graphs, which is called the assignment problem. This way, optimizing the system's control and improving its robustness would be possible since the system would adapt to new positions efficiently according to the number of UAVs performing monitoring missions.

To use this algorithm, initially, it is necessary to create an $N \times N$ adjacency matrix with the distance between each vehicle and the desired points (Algorithm 2).

Algorithm 2 Calculate Adjacency Matrix

Require: 2D array A of size $N \times N$, a 2D array B of size $N \times 2$ with the global coordinates of the desired points, a 2D array C of size $N \times 2$ with the UAVs' coordinates, and the number of UAVs $\rightarrow$ N

for $i \leftarrow 0$ to N **do**
  for $j \leftarrow 0$ to N **do**
    $A[i][j] \leftarrow \sqrt{(C[j][0] - B[j][0])^2 + (C[j][1] - B[j][1])^2}$
  end for
end for
return A =0

This adjacency matrix will be the cost function used in the Hungarian Algorithm. In the specific case where the number of robots exceeds the desired points, an extra column is added to obtain a square matrix. The extra values will equal zero so that they do not influence the overall cost.

Finally, the Hungarian algorithm is implemented (Algorithm 3):

Algorithm 3 Hungarian Algorithm

Require: Cost matrix C of size $n \times n$

Subtract the row minimum from each row of the cost matrix
Subtract the column minimum from each column of the cost matrix
Verify the number of lines through the rows and columns that cover all zeros in the matrix
while the number of covering lines is less than n **do**
  Find the smallest uncovered value in the matrix
  Subtract this value from all uncovered elements and add it to each column that is crossed out
  Verify the number of lines through the rows and columns that cover all zeros in the matrix
end while
Choose the entries in the matrix that have a "0", taking into account that each robot can only be attributed 1 desired point =0

In Figure 4.4, the assignment of each UAV according to the Hungarian Algorithm can be observed.

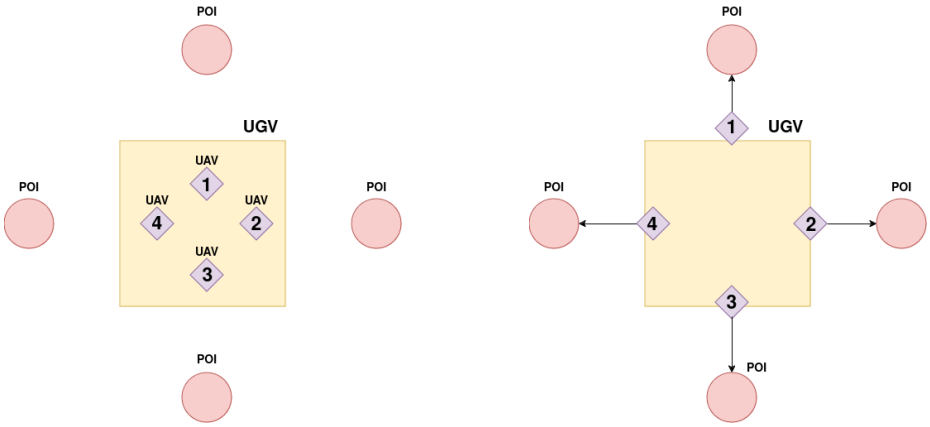


Figure 4.4: Assignment of POIs to UAVs

Object Detection

In the context of the swarm pollination mission, once a UAV detects a Point of Interest (POI), the system agent must approach it to simulate the pollen being collected from a male flower or deployed in a female flower.

Using an object detection algorithm would allow the randomization of any possible points in the simulator, providing greater robustness to the algorithm by dynamically adapting to the environment the system is in.

In the simulator, the flower is represented by a yellow ball with 0.05 meters of radius to cause some contrast between the simulator's environment and the POI (Figure 4.5).

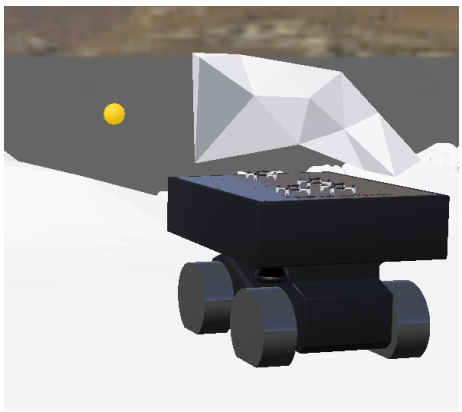


Figure 4.5: Robotic solution and a POI

Firstly, to locate the POI, it is necessary to identify it. To this end, each UAV has a camera attached to scan the environment for any possible "flowers". The specifications of the camera implemented in the simulator are the following:

Table 4.1: Camera Specifications

Specification	**Details**
Resolution	640 x 640 pixels
Field of View	-45 to 45 degrees
Projection	Planar
Frame Rate	10 fps

With the camera set up, the POI detection is addressed. Having a clear contrast between the environment and the POIs, enables the use of the Image Thresholding Method [46].

Image thresholding is an image segmentation technique used to differentiate objects or foreground pixels from background pixels. To achieve this, a mask is created where the pixels between a certain range of desired colors are set to white and the rest to black. Once the mask is obtained, various approaches can be considered to detect the contours of the POI using OpenCV [47].

Algorithm 4 Image Thresholding

Require: Image I (BGR format), lower HSV threshold $L_y = (0, 200, 200)$, upper HSV threshold $U_y = (50, 255, 255)$

1: Convert I from BGR to HSV: $I_{HSV} \leftarrow \text{cvtColor}(I, \text{COLOR_BGR2HSV})$
2: Apply thresholding to get a binary mask: $M \leftarrow \text{inRange}(I_{HSV}, L_y, U_y)$
3: Find contours in the binary mask: $\text{contours} \leftarrow \text{findContours}(M, \text{retr_external}, \text{chain_approx_simple})$
4: **for** each contour c in contours **do**
5: Calculate the bounding box: $B \leftarrow \text{boundingRect}(c)$
6: Draw the bounding box on the image: $\text{rectangle}(I, B, \text{color} = (0, 255, 0), \text{thickness} = 2)$
7: **end for**
8: **return** I =0

The bounding box is considered to solve the mapping problem from an image to the 3D space using Inverse projection [48]. The inverse projection consists of transforming image coordinates to world coordinates using the following steps:

1. Given the target image coordinates, in this specific case, the center of the bounding box, the normalized image coordinates are computed using the Intrinsic Matrix:

$$\begin{bmatrix} x_{norm} \\ y_{norm} \\ 1 \end{bmatrix} = \mathbf{K}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

With the Intrinsic Matrix of the camera being:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x and f_y are the focal lengths in x and y directions, c_x and c_y are the pixel coordinates of the camera optical center.

2. The world coordinates are then computed through the normalized image coordinates using the Inverse Projection equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{R}^{-1} \left(\mathbf{s} \begin{bmatrix} x_{norm} \\ y_{norm} \\ 1 \end{bmatrix} - \mathbf{t} \right)$$

where **R** is the rotation matrix, **t** is the translation vector, and **s** is a scaling factor that adjusts depth.

To calculate **s**, since the diameter of the POI is known, it is possible to obtain the scaling factor by dividing the real-world diameter by the number of pixels the object's width corresponds to in the image.

Pollination Mission

With the location of the POI, the UAV is ready to approach the "flower". The control of the movement of the UAV in this task is based on the same control laws as the formation control. After all, the location of the POI and the desired points of the formation are, inherently, a formation. The major difference relies on the fact that the location of the POI does not need to be constantly updated in relation to the movement of the UAV.

When a POI is detected, the Hungarian Algorithm can still be computed with the desired points of the new formation, given the fact that one UAV must engage in the pollination mission, and the location of the POI (Figure 4.6).

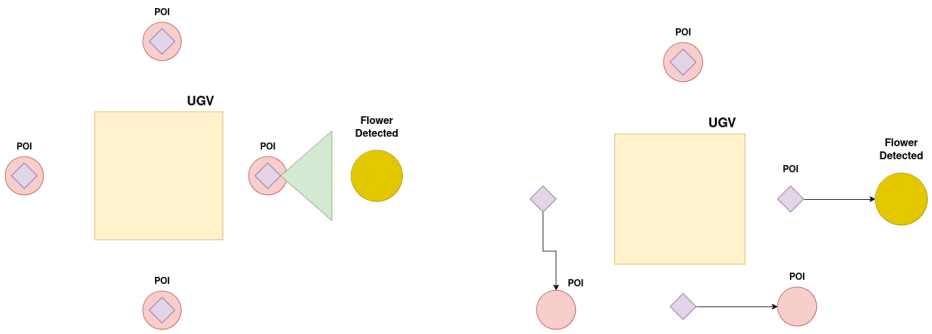


Figure 4.6: UAV engaging in pollination mission and formation readjustment

The orientation of each UAV must also be addressed. In the case where the location of the "flowers" are completely unknown, the orientation of the UAVs will match the angle their attributed in the formation (Figure 4.7) to maximize the amount of area scanned.

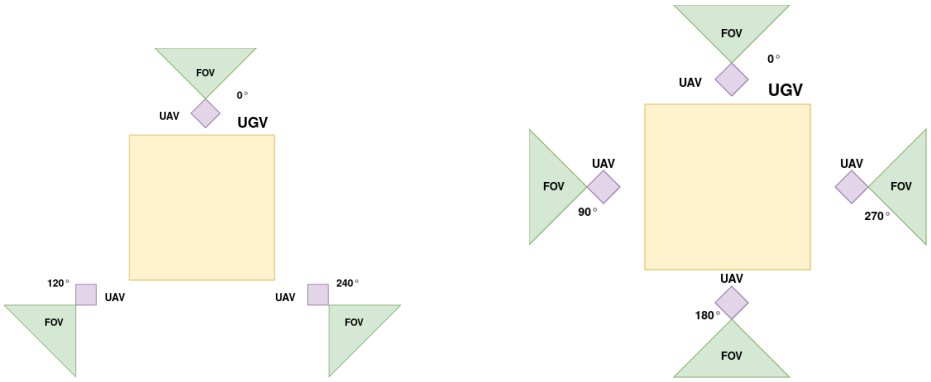


Figure 4.7: UAV orientation according to formation

Obstacle Avoidance

In terms of obstacle avoidance, safety measurements are implemented to prevent collisions among agents of the swarm. When a member enters a pollination mission and the formation is readjusted, possible collisions can occur. To avoid this, if the agent in the monitoring mission is at a certain threshold distance from the vehicle in the pollination mission, it will stop the movement in the x and y-axis until this threshold value is reached again (Figure 4.8).

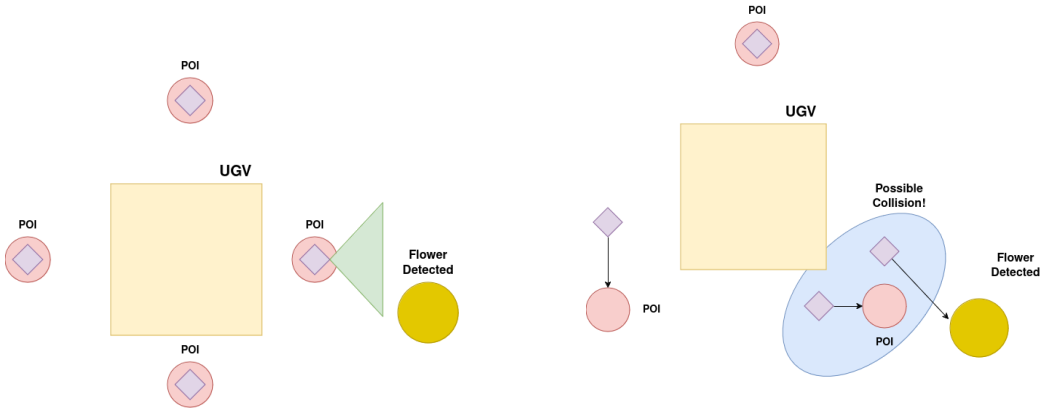


Figure 4.8: Collisions due to multiple mission management

A different scenario can occur when all the UAVs are docked in the UGV and they are deployed to the desired POIs. In this specific case, the optimization provided by the Hungarian Algorithm can optimally assign the positions without any collisions since the desired monitoring positions are fairly spread apart and the movement of the UAVs is in the outwards direction from their starting point. If the UAVs were assigned to the closest POI, possible impacts could occur, depending on

their initial position (Figure 4.9). The optimization does not resolve any possible collisions but prevents a large number of instances where collisions would occur.

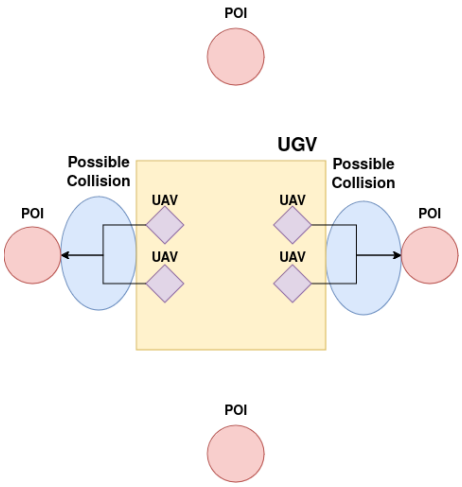


Figure 4.9: Collisions due to assignment issues

Battery Life

Regarding the battery of each UAV, once the UAVs enter a monitoring or pollination mission, their battery starts decreasing. The implementation of this feature into the simulator is very important to be able to simulate the system as accurately as possible.

Once the battery of any UAV drops below a certain value, the UAV returns to the ground vehicle where it docks and recharges its battery. The formation of UAVs is able to adapt in this case, similarly to the context of the pollination mission (Figure 4.10).

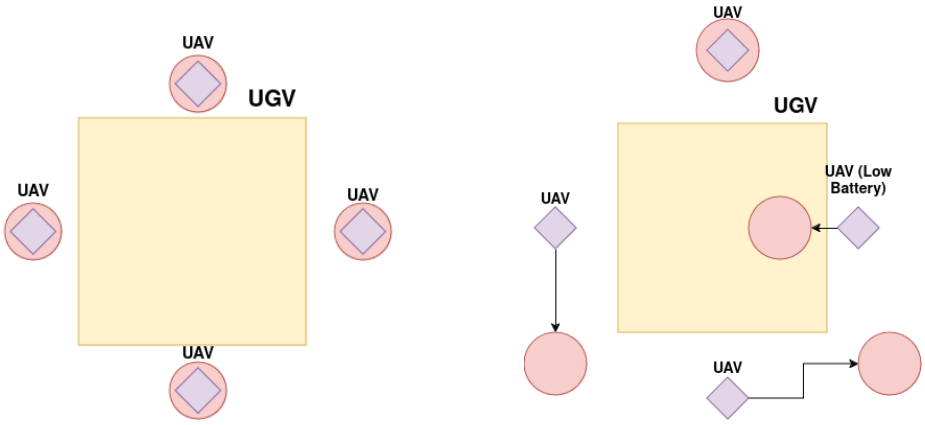


Figure 4.10: UAV readjusmtent according to battery life

Concerning the pollination mission, if the vehicle deployed to perform the monitoring mission is below a threshold battery value, it will automatically return to the UGV to recharge, and a different UAV will be assigned to this task.

4.3 State Machine

The system was developed based on Finite State Machines [49]. This methodology provides greater modularity in case the system suffers any change or feature integration and predictability by having clearly defined states and transitions.

In Figure 4.11, the UAV State Machine is presented.

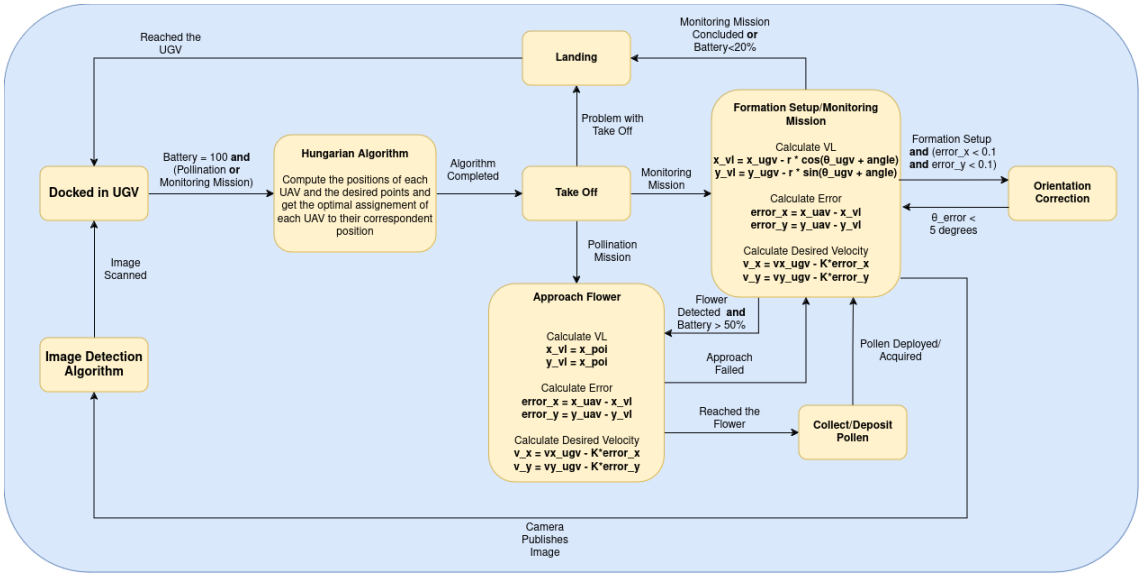


Figure 4.11: UAV State Machine

Chapter 5

Simulation and Results

In this chapter, the simulation of the system and its performance will be demonstrated and evaluated. Firstly, the characteristics of the simulator and its scenario are explored. Several tests are conducted regarding the accuracy of the system and its time to perform specific tasks. Furthermore, different configurations of the system will be assessed.

5.1 Simulators

Simulating our system and understanding how it would translate to the real world is essential to prevent any possible mishaps and to perceive how it would behave in a real-life scenario.

Several simulators, such as Gazebo, ARGoS, Farming Simulator, and Webots, were compared, and the last one was adopted. Although Webots can be computationally inefficient, it supports multiple programming languages, has a programmable physics engine and a ROS2 bridge to perform the communications [50].

Overall, simulators provide a safe way of developing, testing, and evaluating the performance of robotic systems while minimizing costs and environmental impact [51].

When choosing the simulator, the main factors considered were:

- **Performance** - Computational expensive simulators may not be suited for the simulation of swarm architectures given the high number of vehicles integrated in the system;
- **Realism** - A simulator that can represent real-world dynamics accurately ensures that once the system is integrated into the real world, it will behave similarly to the expected from the simulator;
- **Integration and Compatibility** - Integrating the system with the Robotic Operating System 2 (ROS2) was a prerequisite of the system.

5.2 Webots

To simulate the system, firstly, it was necessary to generate the world. The scenario used in this context was a ground mesh that represented Quinta do Seixo (Figure 5.1), so it was possible to integrate the system into an agricultural environment.

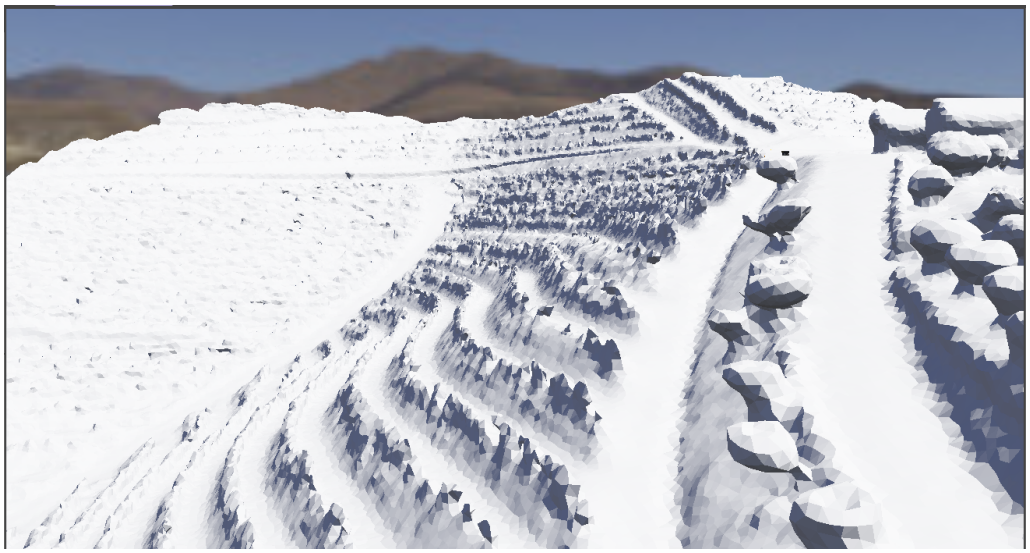


Figure 5.1: Webots mesh relative to Quinta do Seixo

Although artificial pollination methodologies are not usual in vineyard scenarios, this environment offers a realistic terrain where it can accurately simulate the robotic system.

With regards to the UGV and UAVs used to represent the solution, to simulate the UGV, the Summit-XL Steel model was used with some slight modifications in the wheels since the original model has omnidirectional wheels. The UAVs are based on the Crazyflie model with the addition of some sensors (Figure 5.2).

For future work, integrating specific robot models in this scenario is crucial to ensure that the simulated system represents the real system as precisely as possible.

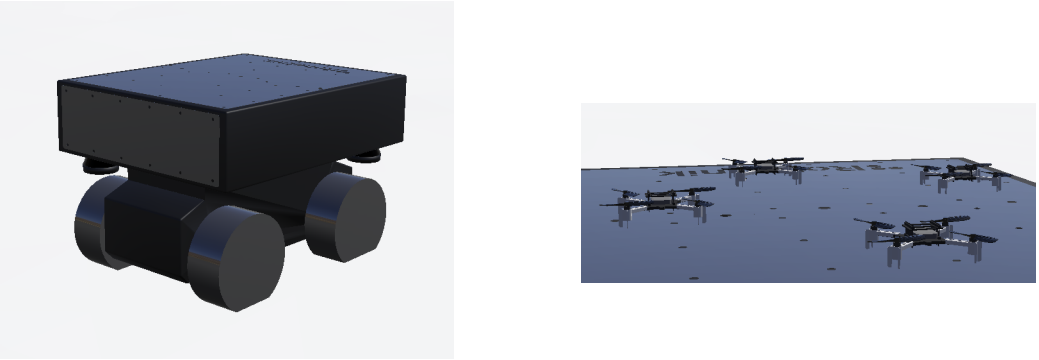


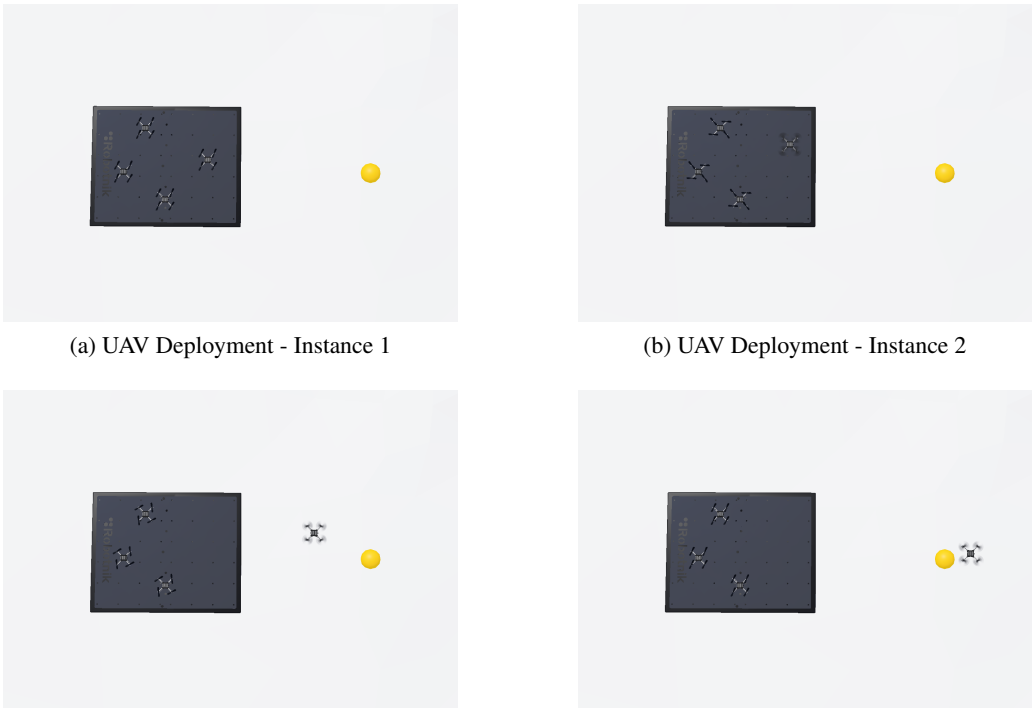
Figure 5.2: UGV and UAV models in Webots

5.3 Simulation

Feedback Controller

The first simulation scenario presented is the deployment of a UAV to a specific position (Figure 5.3). The main focus of this test is to verify how long it takes to complete this task and its performance when varying specific control parameters.

In this case, the VL position is generated with *radius* equal to 1 meter and *angle* equal to 0 radians, concerning the robot's orientation, and proportional constant (Equation 4.6), $K = 0.02$.



(a) UAV Deployment - Instance 1

(b) UAV Deployment - Instance 2

(c) UAV Deployment - Instance 3

(d) UAV Deployment - Instance 3

Figure 5.3: UAV Deployment

Under these conditions, the movement of the UAV is relatively smooth, with no overshoot. By increasing the value of K, the UAV travels faster to its desired position; however, it overshoots the predicted location.

Table 5.1 shows the error evolution according to the VL's position according to the proportional constant value:

Table 5.1: Error evolution when the UGV's velocity is 0 m/s

	t=0s		t=5s		t=7.5s		t=10s	
Error in	K = 0.02	K = 0.05	K = 0.02	K = 0.05	K = 0.02	K = 0.05	K = 0.02	K = 0.05
X	-0.800	-0.800	-0.716	-0.725	-0.102	0.066	0.063	0.224
Y	0.029	0.027	0.120	0.103	0.041	0.024	-0.011	-0.014

The UAV starts from the same position in both scenarios, hence the same initial error. In t=2.5s, the error in the y-axis increases due to the terrain's irregularities and some slippage that can occur when the UAV is taking off. At 7.5s, the UAV with a $K = 0.02$ is still trying to reach the desired location, whereas the UAV with $K = 0.05$ has already passed through the VL's position. In the last instance, the first UAV is still cautiously approaching the desired location, and the second vehicle is still overshooting it, as it is possible to observe by the increase in the error on the x-axis.

To prevent any possible undesired behaviors, the proportional constant in the feedback equation is kept at 0.02 for the following scenarios.

Hungarian Algorithm

The second scenario evaluated is the Hungarian Algorithm assignment methodology. Ensuring that the correct assignment is being made is very important. To this end, the assignment is done by minimizing the total distance traveled by the UAVs. The first case addressed is when 4 UAVs are assigned to the monitoring formation composed of 4 VLs using Algorithm 1.

In this scenario, the 4 VLs' and 4 UAVs' positions were the following:

1. VL 1: Radius = 1, Angle = 0
    - VL 1 X Coordinate: 11.38
    - VL 1 Y Coordinate: 8.27
2. VL 2: Radius = 1, Angle = PI/2
    - VL 2 X Coordinate: 12.40
    - VL 2 Y Coordinate: 9.26
3. VL 3: Radius = 1, Angle = PI
    - VL 3 X Coordinate: 11.41
    - VL 3 Y Coordinate: 10.28
4. VL 4: Radius = 1, Angle = 3PI/2
    - VL 4 X Coordinate: 10.4
    - VL 4 Y Coordinate: 9.29

1. UAV 1:
    - X Coordinate: 11.29
    - Y Coordinate: 9.46
2. UAV 2:
    - X Coordinate: 11.60
    - Y Coordinate: 9.32
3. UAV 3:
    - X Coordinate: 11.20
    - Y Coordinate: 9.25
4. UAV 4:
    - X Coordinate: 11.40
    - Y Coordinate: 9.12

When computing Algorithm 2, the resulting cost matrix is:

$$\mathbf{C} = \begin{bmatrix} 1.1879 & 1.1209 & 0.8236 & 0.9128 \\ 1.0604 & 0.8028 & 0.9786 & 1.1990 \\ 0.9909 & 1.2002 & 1.0487 & 0.8007 \\ 0.8462 & 1.0081 & 1.1538 & 1.0154 \end{bmatrix}$$

Due to the initial positions of each UAV, there is a clear independent minimum value in each column that represents the POI to which the UAV with the corresponding row should be deployed. To make sure, Algorithm 3 is followed:

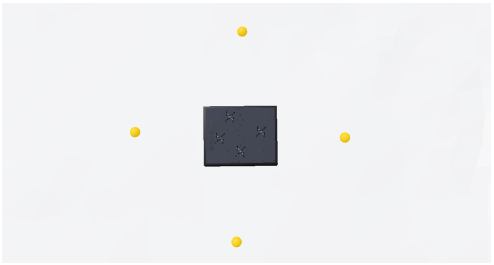
1. Step 1 - Subtracting the minimum value in each row to the other values of the row:

$$\mathbf{C} = \begin{bmatrix} 0.3643 & 0.2973 & 0 & 0.0892 \\ 0.2576 & 0 & 0.1758 & 0.3962 \\ 0.1902 & 0.3995 & 0.2480 & 0 \\ 0 & 0.1619 & 0.3076 & 0.1384 \end{bmatrix}$$

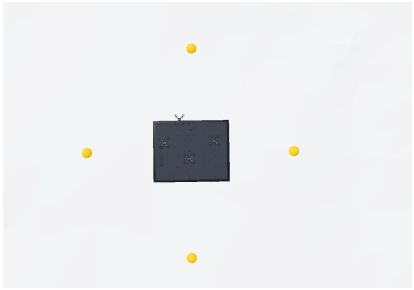
2. Step 2 - Subtracting the minimum value in each column to the other values of the column (since each column has a zero after Step 1, the matrix will be the same).

3. Step 3 - Draw lines that cover all the zeros in the matrix using the least amount of lines: since there is only one zero in each column, the minimum number of lines is 4, which corresponds to the matrix dimension; therefore, the algorithm has concluded. The row and column where the zeros are placed correspond to the specific UAV and the POI assigned to it.

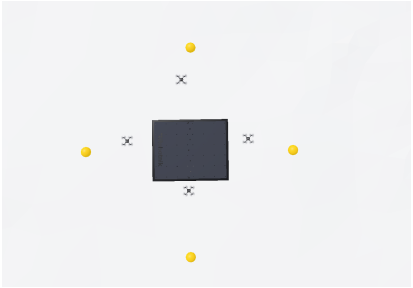
In Figure 5.4, it is possible to visualize the assignment and deployment of the vehicles toward their respective POIs.



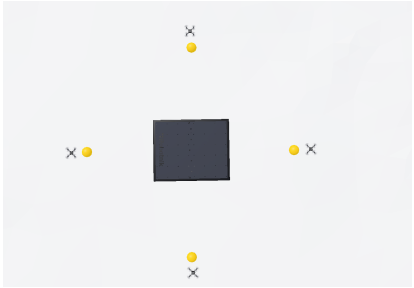
(a) Hungarian Algorithm Deployment - Instance 1



(b) Hungarian Algorithm Deployment - Instance 2



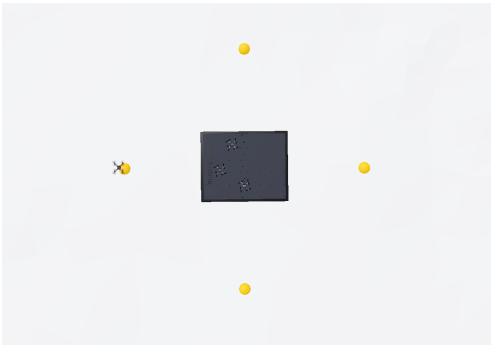
(c) Hungarian Algorithm Deployment - Instance 3



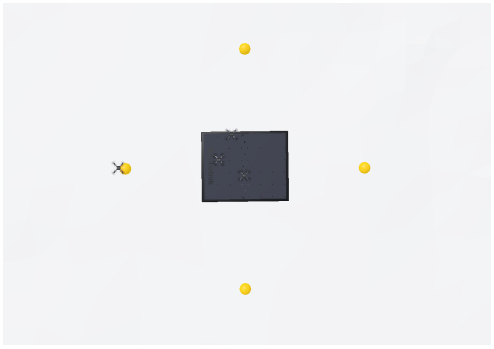
(d) Hungarian Algorithm Deployment - Instance 4

Figure 5.4: Hungarian Algorithm Simple Assignment

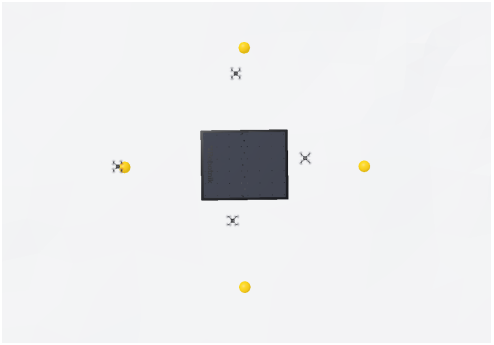
If a UAV is already assigned to a specific POI, the UAVs can adapt once the monitoring mission is assigned to them. In Figure 5.5, a UAV is already assigned to a desired position, and the remaining agents enter a monitoring mission.



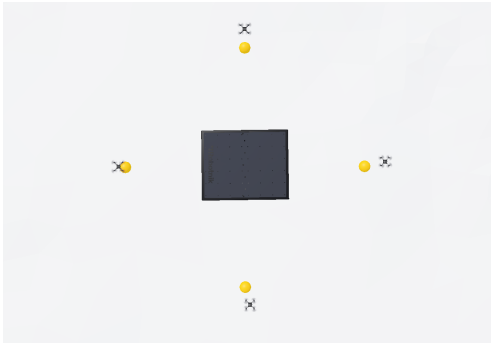
(a) Hungarian Algorithm Pre-Assigned - Instance 1



(b) Hungarian Algorithm Pre-Assigned - Instance 2



(c) Hungarian Algorithm Pre-Assigned - Instance 3



(d) Hungarian Algorithm Pre-Assigned - Instance 4

Figure 5.5: Hungarian Algorithm with a pre-assigned vehicle

Leader-Follower

The system can organize its formation and optimize the path to achieve its desired VL position. However, to be able to complete the monitoring mission fully, the Leader-Follower controller must be fully implemented. That is, when the UGV moves, the formation must maintain itself cohesively and follow it.

To achieve this, in the feedback controller (Equation 4.6), the value $\dot{x}_d$, representing the UGV's velocities, is appended. So, besides correcting the displacement error through the proportional part of the equation, the actual velocities of the UGV are added to account for his movement in the terrain.

Moreover, the VL equations (Equation 4.3 and 4.3), in addition to the *radius* and *angle* variables, also depend on the current position of the UGV. If the UGV's position is updated, so are the VL's positions; consequently, the UAVs' proportional term will increase, and they will start correcting their position and move towards the new desired position.

Figure 5.6 presents the monitoring mission using a Leader-Follower configuration.



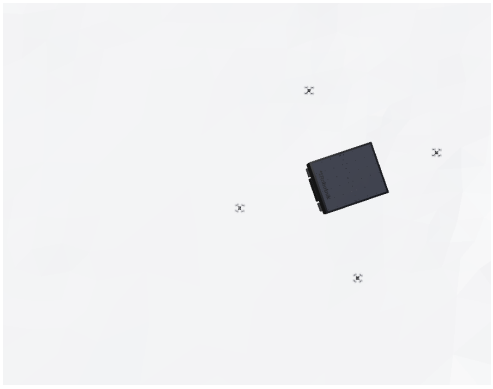
(a) Leader-Follower - Instance 1



(b) Leader-Follower - Instance 2



(c) Leader-Follower - Instance 3



(d) Leader-Follower - Instance 4

Figure 5.6: Leader-Follower configuration in monitoring mission

The system's performance while having the leader (UGV) execute a curvilinear trajectory in a monitoring mission is evaluated in Table 5.2.

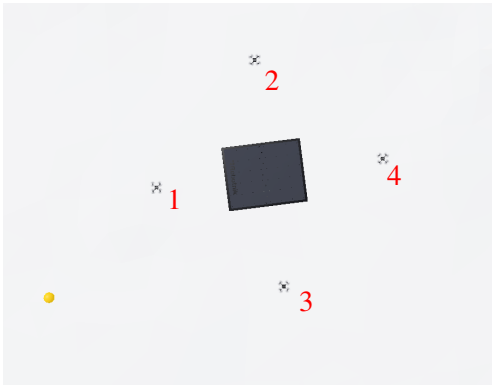
Table 5.2: Error evolution when the UGV's velocity is 0 m/s

	t=0s		t=5s		t=7.5s		t=10s	
UAV	Error in X	Error in Y	Error in X	Error in Y	Error in X	Error in Y	Error in X	Error in Y
1	-0.025	-0.002	-0.221	-0.015	-0.109	-0.039	-0.107	-0.043
2	-0.049	-0.001	-0.152	-0.037	-0.051	-0.008	-0.057	-0.052
3	-0.027	-0.002	-0.266	-0.025	-0.162	-0.030	-0.111	-0.003
4	-0.027	-0.002	-0.307	-0.019	-0.169	-0.037	-0.132	-0.080

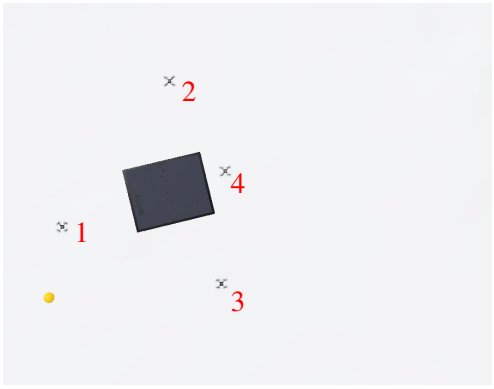
In t = 0s, the UAVs start in their assigned monitoring positions. Afterward, a significant increase in their error is registered since the UGV starts performing its trajectory. At 7.5s, the UAVs are able to reduce the error relative to their desired position, but eventually, the error stagnates at around 0.1 meters. The UAVs cannot reduce the error any further since the VL is constantly being updated, considering the UGV is always moving. If the UGV stops, then the VLs would be constant, and the UAVs could minimize the error to the desired position. To mitigate this phenomenon, in the future, a predictive controller that would consider the movement of the vehicle, as well as the possible time loss in communication protocols between robots, could be implemented.

Pollination Mission

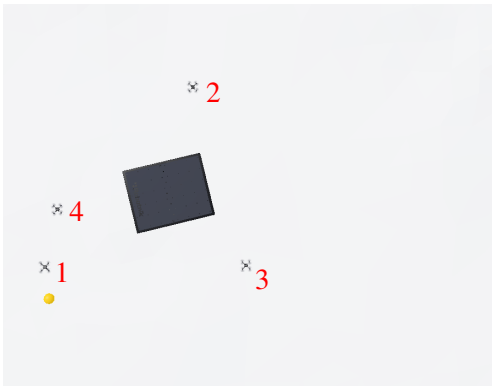
In the presence of a POI, the system must visit this point to collect or pollute pollen. To this end, a computer vision algorithm (Algorithm 4) was made to detect and locate these POIs. Moreover, when the assigned UAV is deployed in this mission, the formation must adjust itself to deal with the loss of an agent in the monitoring mission. Once the pollination mission finishes, the agent reintegrates the swarm, and the formation is readjusted (Figure 5.7).



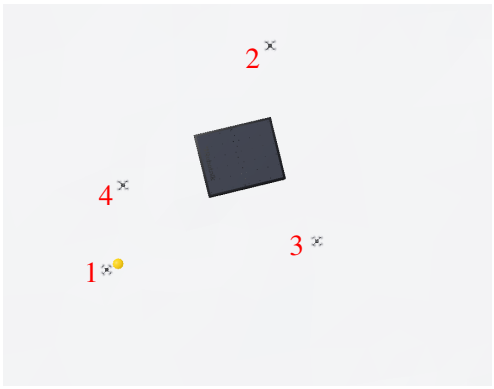
(a) Pollination Mission - Instance 1



(b) Pollination Mission - Instance 2



(c) Pollination Mission - Instance 3



(d) Pollination Mission - Instance 4

Figure 5.7: Pollination Mission with formation readjusting

Adjusting the formation according to the number of UAVs in the system and each UAV's current mission is an important characteristic of the solution to maximize coverage and efficiency.

This approach enhances mission flexibility, improving reliability and adapting to dynamic environments. The Hungarian Algorithm's assignment methodology, when handling multiple POIs, enables the system to not only efficiently cover all the desired locations but also optimize the UAVs' battery life by reducing the total distance traveled to complete it.

Complete Mission

In this last scenario, a course with 6 POIs (representing the flowers) was created to evaluate the performance of the robotic system (Figure 5.8). Initially, the UAVs will be deployed in a monitoring mission, and as the vehicle progresses, the system must approach all the POIs in the context of the pollination mission. The UGV is controlled externally to traverse the course.

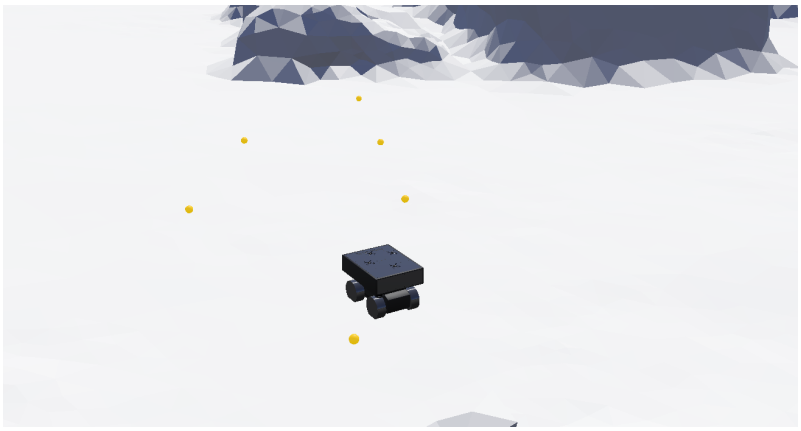


Figure 5.8: UGV and UAV models in Webots

Different configurations with different numbers of UAVs were considered in this context to understand how this factor would impact the success and efficiency of the mission. To better comprehend the impact of the number of agents in the formation, the configurations were tested according to the number of POIs that were successfully visited.

In the first scenario, the results were the following:

- **1 UAV** - POIs visited: 3

  With solely one UAV performing the mission (Figure 5.9), several problems were raised:

  1. The UAV's orientation matched the robot's orientation; therefore, the limited FOV meant some POIs were undetected. A specific POI was placed behind the initial position of the UGV, so it was never detected.
  2. In specific situations, the UAV was deployed to a pollination mission, and had it stayed in a monitoring mission, it would have detected a different POI, but due to the overload of tasks, it could not perform all the missions.

  Overall, the system's performance with a single UAV was inefficient and suboptimal. Moreover, when the UAV was deployed to a pollination mission, no UAVs were in a monitoring mission to account for possible obstacles or changes in the environment, endangering the system.



(a) Complete Mission - 1 UAV - Instance 1



(b) Complete Mission - 1 UAV - Instance 2



(c) Complete Mission - 1 UAV - Instance 3



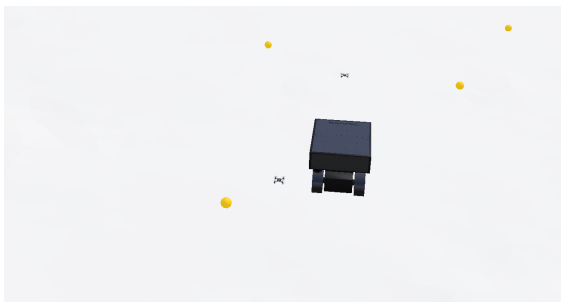
(d) Complete Mission - 1 UAV - Instance 4

Figure 5.9: Complete Mission with 1 UAV

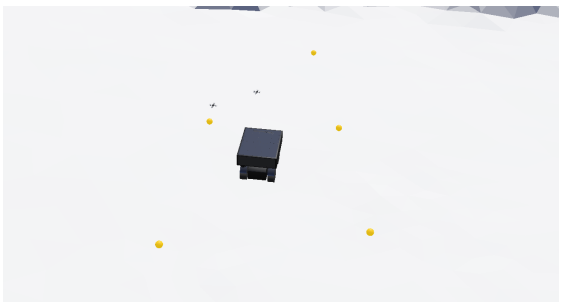
- **2 UAVs** - POIs visited: 4

  In this example, the POI placed behind the UGV in the initial position was detected; however, the FOV is still very limited, and only 4 POIs were detected.

  With this configuration (Figure 5.10), if the UAV ahead of the robot engages in a pollination mission, the UAV on the other end of the robot takes its place. This is very beneficial in monitoring possible obstacles in the UGV's way while a UAV is performing the pollination mission.



(a) Complete Mission - 2 UAVs - Instance 1



(b) Complete Mission - 2 UAVs - Instance 2

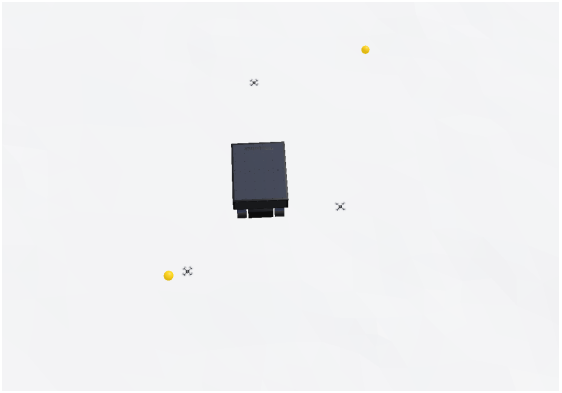
Figure 5.10: Complete Mission with 2 UAVs

- **3 UAVs** - POIs visited: 6

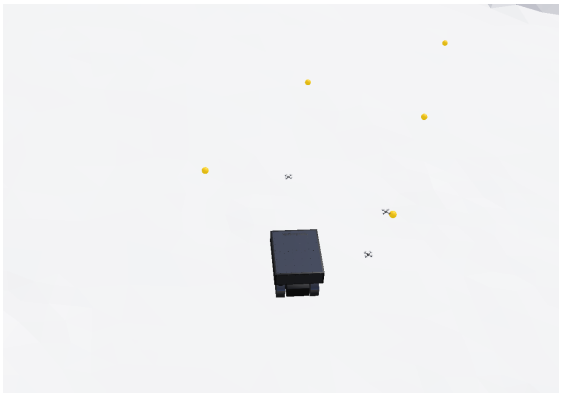
  This is the first configuration able to detect all the POIs (Figure 5.11). The total FOV is now significant and enabled the system to perform the mission accordingly.

  Possible POIs that the UAV ahead of the robot might have missed due to being in a pollination mission were now covered by the UAVs in the back. Once the UGV was ahead of these POIs, the UAVs were able to detect and address them.

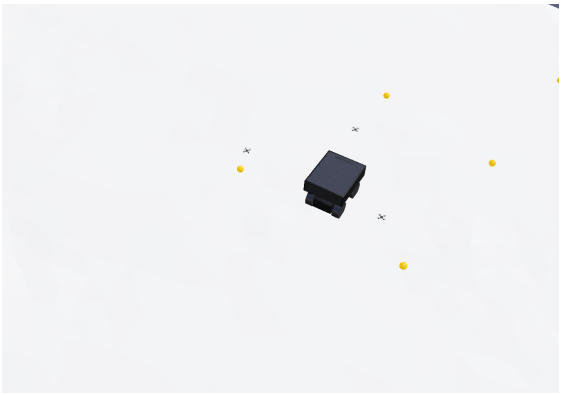
  In this specific course, the system had at least one UAV that was always monitoring the path.



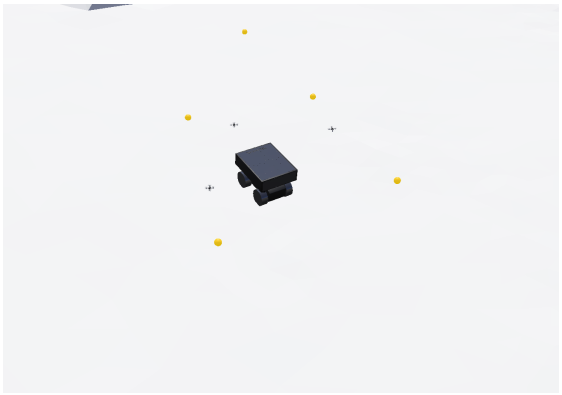
(a) Complete Mission - 3 UAVs - Instance 1



(b) Complete Mission - 3 UAVs - Instance 2



(c) Complete Mission - 3 UAVs - Instance 3



(d) Complete Mission - 3 UAVs - Instance 4

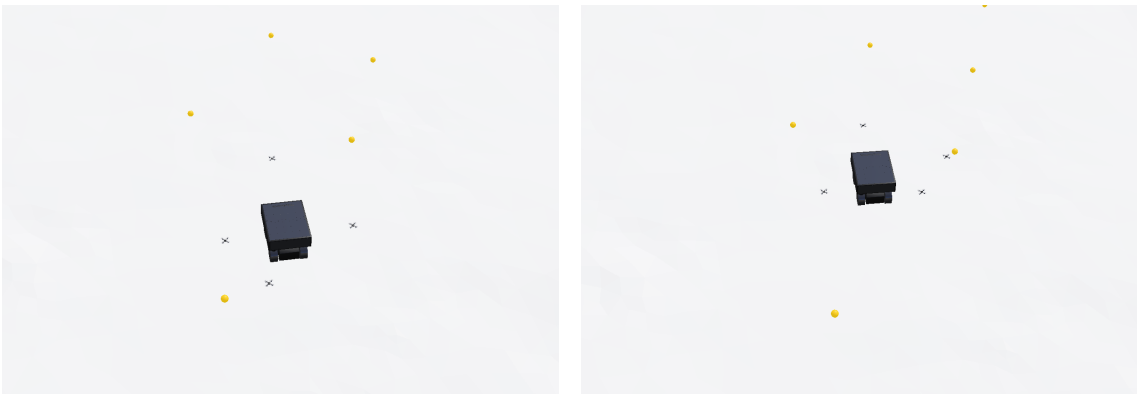
Figure 5.11: Complete Mission with 3 UAVs

- **4 UAVs** - POIs visited: 6

  With 4 UAVs, the system can detect all the POIs as well. The only difference in the scenario with 3 UAVs is that, in this specific course, at least 2 UAVs were in monitoring missions. This allows the system to monitor not only the front but also the back of the UGV.

  The total FOV when all the UAVs are in monitoring missions is greatly increased in this scenario, providing extra safety towards environmental changes or unexpected events.

  Figure 5.12 presents the system with the described configuration.



(a) Complete Mission - 4 UAVs - Instance 1 (b) Complete Mission - 4 UAVs - Instance 2

Figure 5.12: Complete Mission with 4 UAVs

Chapter 6

Conclusions and Future Work

Artificial pollination is a process that complements and aids pollinators, providing greater quality and uniformity towards agricultural production.

This thesis explores the implementation of swarm methodologies into a heterogeneous robotic system to complement artificial pollination. This collaborative work may be a precise and cost-effective alternative to current manual and vehicle-mounted artificial pollination techniques.

The robotic solution could be an alternative to current artificial pollination methods, benefiting from the implementation of swarm methodologies, which enhance the efficiency and adaptability of our system. The current robotic system allows for better monitoring due to the organization of the system while providing targeted and precise pollination. Although the UAVs still need adjustments to track their desired position in the monitoring mission, the results are promising.

The scenario and context in which the system is integrated are essential to understanding the swarm configuration that should be used, as well as the number of UAVs required to efficiently perform the designated mission, considering that having redundancy in these systems may be a benefit.

The robotic system can manage both monitoring and pollination missions while optimizing the task assignment to increase battery life and efficiently carry out the mission.

In future work, it is necessary to develop an obstacle-avoidance algorithm that could deal with potential obstacles not only in the monitoring mission but also in the pollination mission. Moreover, the formations that should be adopted in different scenarios should also be studied and implemented. Finally, it is intended to transpose the robotic solution from the simulated environment to a real-life system and test it in the field.

Although swarm robotics is an under-explored field in the context of artificial pollination, some approaches appear promising in robotic pollination. Efficient management of the UAVs is the main priority in these scenarios given their limited battery life.

Facing the recent decline of the bee population, we must innovate and create new technologies to assist them in pollination tasks while maximizing our efforts in recovering this species.

Bibliography

[1] Lucas A. Garibaldi, Marcelo A. Aizen, Alexandra M. Klein, Saul A. Cunningham, and Lawrence D. Harder. Global growth and stability of agricultural yield decrease with pollinator dependence. *Proceedings of the National Academy of Sciences*, 108(14):5909–5914, 2011.

[2] World bee day. https://www.fao.org/world-bee-day/en. Accessed: 2023-11-13.

[3] Arindam Das, Sayan Sau, Manas Kumar Pandit, and Koushik Saha. A review on: Importance of pollinators in fruit and vegetable production and their collateral jeopardy from agro-chemicals, 2018.

[4] Alexandra Maria Klein, Bernard E. Vaissière, James H. Cane, Ingolf Steffan-Dewenter, Saul A. Cunningham, Claire Kremen, and Teja Tscharntke. Importance of pollinators in changing landscapes for world crops. *Proceedings of the Royal Society B: Biological Sciences*, 274:303 – 313, 2007.

[5] Clare E. Aslan, Christina T. Liang, Ben Galindo, Hill Kimberly, and Walter Topete. The role of honey bees as pollinators in natural areas. *Natural Areas Journal*, 36(4):478–488, 2016.

[6] Atif Idrees, Ziyad Abdul Qadir, Amin Ul Hasnat, Ayesha Afzal, Saboor Ahmad, Muhammad Anjum Aqueel, Zhigang Li, Ahmed Rady, Shahbaz Ali, and Jun Li. Effectiveness of honeybee (apis mellifera) visit on the pollination of different sunflower cultivars. *Journal of King Saud University - Science*, 35(7):102837, 2023.

[7] Eduardo E. Zattara and Marcelo A. Aizen. Worldwide occurrence records reflect a global decline in bee species richness. *bioRxiv*, 2020.

[8] Isabel Pinheiro, Filipe Santos, António Valente, and Mário Cunha. Robotic pollinating tools for actinidia crops. *Biology and Life Sciences Forum*, 27(1), 2023.

[9] Chizhao Yang, Ryan Watson, Jason Gross, and Yu Gu. Localization algorithm design and evaluation for an autonomous pollination robot. pages 2702–2710, 10 2019.

[10] Erol Sahin. Swarm robotics: From sources of inspiration to domains of application. volume 3342, pages 10–20, 01 2005.

[11] G. Vásárhelyi, Cs. Virágh, G. Somorjai, N. Tarcai, T. Szörenyi, T. Nepusz, and T. Vicsek. Outdoor flocking and formation flight with autonomous aerial robots. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3866–3873, 2014.

[12] Ding Yingying, He Yan, and Jiang Jingping. Multi-robot cooperation method based on the ant algorithm. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706)*, pages 14–18, 2003.

[13] Francis Heylighen. Stigmergy as a universal coordination mechanism i: Definition and components. *Cognitive Systems Research*, 38:4–13, 2016.

[14] Divya Srivastava, Roxanna Pakkar, Austin Langrehr, and Chaska Yamane. Adaptable uav swarm autonomy and formation platform. pages 1–6, 03 2019.

[15] Violet Mwaffo, Jishnu Keshavan, Tyson Hedrick, and Sean Humbert. Detecting intermittent switching leadership in coupled dynamical systems. *Scientific Reports*, 8, 07 2018.

[16] Donghwa Jeong and Kiju Lee. Dispersion and line formation in artificial swarm intelligence. 06 2014.

[17] Tiago Oliveira, A. Pedro Aguiar, and Pedro Encarnação. Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems. *IEEE Transactions on Robotics*, 32, 08 2016.

[18] Martina Mammarella, Lorenzo Comba, Alessandro Biglia, Fabrizio Dabbene, and Paolo Gay. Cooperative agricultural operations of aerial and ground unmanned vehicles. pages 224–229, 11 2020.

[19] Dimitrios Apostolopoulos. Analytical configuration of wheeled robotic locomotion. 06 2001.

[20] Tianmiao Wang, Yao Wu, Jianhong Liang, Chenhao Han, Jiao Chen, and Qiteng Zhao. Analysis and experimental kinematics of a skid-steering wheeled robot based on a laser scanner sensor. *Sensors*, 15:9681–9702, 05 2015.

[21] Bogdan Trasnea, Cosmin Ginerică, Mihai Zaha, Gigel Macesanu, Claudiu Pozna, and Sorin Grigorescu. Octopath: An octree-based self-supervised learning approach to local trajectory planning for mobile robots. *Sensors*, 21:3606, 05 2021.

[22] Eero Väljaots and Raivo Sell. Energy efficiency profiles for unmanned ground vehicles. *Proceedings of the Estonian Academy of Sciences*, 68:55, 04 2019.

[23] Anna Bourmistrova, Milan Simic, Reza Hoseinnezhad, and Reza Jazar. Autodriver algorithm. *Journal of Systemics, Cybernetics and Informatics*, 9:8, 01 2012.

[24] Yonghwan Jeong and Seongjin Yim. Model predictive control-based integrated path tracking and velocity control for autonomous vehicle with four-wheel independent steering and driving. *Electronics*, 10:2812, 11 2021.

[25] Chris Dinelli, John Racette, Mario Escarcega, Simon Lotero, Jeffrey Gordon, James Montoya, Chase Dunaway, Vasileios Androulakis, Hassan Khaniani, Sihua Shao, Pedram Roghanchi, and Mostafa Hassanalian. Configurations and applications of multi-agent hybrid drone/unmanned ground vehicle for underground environments: A review. *Drones*, 7(2), 2023.

[26] Yuya Maruyama, Shinpei Kato, and Takuya Azumi. Exploring the performance of ros2. pages 1–10, 10 2016.

[27] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074, 2022.

[28] Matthew J Page, David Moher, Patrick M Bossuyt, Isabelle Boutron, Tammy C Hoffmann, Cynthia D Mulrow, Larissa Shamseer, Jennifer M Tetzlaff, Elie A Akl, Sue E Brennan, Roger Chou, Julie Glanville, Jeremy M Grimshaw, Asbjørn Hróbjartsson, Manoj M Lalu, Tianjing Li, Elizabeth W Loder, Evan Mayo-Wilson, Steve McDonald, Luke A McGuinness, Lesley A Stewart, James Thomas, Andrea C Tricco, Vivian A Welch, Penny Whiting, and Joanne E McKenzie. Prisma 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ*, 372, 2021.

[29] Ricardo Sousa, Heber Sobreira, and A. Moreira. A systematic literature review on long-term localization and mapping for mobile robots, 10 2022.

[30] Ieee xplore. http://ieeexplore.ieee.org. Accessed: 2024-01-03.

[31] Web of science. https://www.webofscience.com/wos/woscc/basic-search. Accessed: 2024-01-03.

[32] Scopus. https://www.scopus.com/. Accessed: 2023-11-13.

[33] Carlos Carbone, Dario Albani, Federico Magistri, Dimitri Ognibene, Cyrill Stachniss, Gert Kootstra, Daniele Nardi, and Vito Trianni. *Monitoring and Mapping of Crop Fields with UAV Swarms Based on Information Gain*, page 306–319. Springer International Publishing, 2022.

[34] Andrew Barth and Ou Ma. Cooperative behavior of a heterogeneous robot team for planetary exploration using deep reinforcement learning. *Acta Astronautica*, 214:689–700, 2024.

[35] Sofia Hustiu, Marius Kloetzer, Alejandro López-Martínez, and Cristian Mahulea. Whitening of greenhouse's roof using drones and petri net models. In *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8, 2022.

[36] Wolfgang Hönig, James A. Preiss, T. K. Satish Kumar, Gaurav S. Sukhatme, and Nora Ayanian. Trajectory planning for quadrotor swarms. *IEEE Transactions on Robotics*, 34(4):856–869, 2018.

[37] Afzal Ahmad, Viktor Walter, Pavel Petráček, Matěj Petrlík, Tomáš Báča, David Žaitlík, and Martin Saska. Autonomous aerial swarming in gnss-denied environments with high obstacle density. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 570–576, 2021.

[38] Malintha Fernando. Online flocking control of uavs with mean-field approximation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2021.

[39] Armin Sadeghi, Ahmad Bilal Asghar, and Stephen L. Smith. On minimum time multi-robot planning with guarantees on the total collected reward. In *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 16–22, 2019.

[40] Etienne Petitprez, François Guérin, Frédéric Guinand, Florian Germain, and Nicolas Kerthe. Decentralized coordination of a multi-uav system for spatial planar shape formations. *Sensors*, 23(23), 2023.

[41] Takuma Sakamoto, Stéphane Bonardi, and Takashi Kubota. A routing framework for heterogeneous multi-robot teams in exploration tasks. *IEEE Robotics and Automation Letters*, 5(4):6662–6669, 2020.

[42] Bartomeu Rubí, Ramon Pérez, and Bernardo Morcego. A survey of path following control strategies for uavs focused on quadrotors. *Journal of Intelligent Robotic Systems*, 98, 05 2020.

[43] Meng Zhou, Zihao Wang, Jing Wang, and Zhe Dong. A hybrid path planning and formation control strategy of multi-robots in a dynamic environment. *J. Adv. Comput. Intell. Intell. Informatics*, 26:342–354, 2022.

[44] T. Balch and R.C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.

[45] Haibin Zhu, MengChu Zhou, and Rob Alkins. Group role assignment via a kuhn–munkres algorithm-based solution. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 42(3):739–750, 2012.

[46] M. Sezgin and Bulent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13:146–168, 01 2004.

[47] Opencv. https://opencv.org/. Accessed: 2024-06-25.

[48] Marco Callieri, Paolo Cignoni, Massimiliano Corsini, and Roberto Scopigno. Masked photo blending: Mapping dense photographic data set on high-resolution sampled 3d models. *Computers Graphics*, 32:464–473, 08 2008.

[49] Fred B. Schneider. Implementing fault-tolerant services using the state machine approach: a tutorial. *ACM Comput. Surv.*, 22(4):299–319, dec 1990.

[50] Olivier Michel. Cyberbotics ltd. webots™: Professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):5, 2004.

[51] Sokratis Tselegkaridis and Theodosios Sapounidis. Simulators in educational robotics: A review. *Education Sciences*, 11(1), 2021.