REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministry of Higher Education and Scientific Research
Mohamed Khider University – BISKRA
Faculty of Exact, Natural and Life Sciences
Computer Department

Order number: IVA /M2/2020

Memory

Presented to obtain the academic master's degree in

Computer Science

Course: Images and Artificial Life

Swarm robotics

Par :

MILOUDI AISSA

Supported Put the date of the defense, before the jury composed of :

Cherif Foudil	Professor	Supervisor

Acknowledgements

I would like to express my sincere gratitude to my dissertation director Professor Cherif Foudil. I thank him for having supervised, guided, helped and advised me, I would like to thank all the members of the jury, who did me the honor of carefully studying my work. I address my sincere thanks to all the professors, speakers and all the people who guided me by their words, writings, advice and criticism during my research.

I thank my very dear parents who have always been there for me," You have sacrificed everything. for your children sparing neither health nor effort. You have given me a beautiful model of hard work and perseverance. I am indebted to an education that I am proud of and I thank my large family for their encouragement.

I would like to thank all my Friends, for their friendship, unconditional support and their encouragement and trust.

ملخص

روبوتات السرب هي مجال ناشئ من الروبوتات، والذي يركز على أنظمة متعددة من الروبوتات المنظمة ذاتيا، وغالبا ما تتميز أعداد كبيرة من الوحدات. حاليا، تم إنشاء عدد من التطبيقات لتلبية الحاجة المتزايدة للروبوتات سرب في حياتنا اليومية. وتشمل المثلة المواقع في النشطة الترفيهية والفنية مثل الرسم أو الموسيقى أو حتى الرقص. سنقدم في هذه الورقة دراسة نموذجية للأنظمة متعددة الروبوتات، وهي تشكيل الانماط، المستوحاة من الظواهر المهمة الموجودة في الكائنات الحية (مثل الحيوانات والنباتات). تشكل تحديا لحل مشكلة تنظيم مجموعة من الروبوتات في تشكيل أنماط كاملة بأشكال بسيطة مثل السداسي أو الخطوط أو التوزيع الموحد. تم استخدام تصميم وإعداد وحدات التحكم لأنظمة روبوت سرب للحصول على الشكال بطريقة منظمة في وقت قصير مع أجهزة استشعار مختلفة. نهجنا مستوحاة من الطبيعة، ول سيما القوى الميكانيكية المشاركة في الدراسات الخلوية الداخلية، من أجل إنتاج أشكال منظمة ذاتيا من التجميع باستخدام الروبوتات. استخدمنا محاكاة تجريبية بنا على محاكي أرغوس. حيث يسمح لنا بدراسة جوانب السلوك الجماعي المنظم.

كما تم استخدام نوعين من الروبوتات الروبوت الطائر والروبوت الماشي وذلك بتطبيق استخدام التواصل بين الاسراب وتطبيق الفيزياء الصناعية نموذج نابض مثبط الشبكة وتمت انشاء بعض النماذج والسناريوهات كمثال تطبيقي على الارغوس باستخدام لغة البرمجة ++سي.

Resumé

Les robots Swarm sont un domaine émergent de la robotique, qui se concentre sur plusieurs systèmes de robots auto-organisés, comportant souvent un grand nombre d'unités. Actuellement, un certain nombre d'applications ont été créées pour répondre au besoin croissant de robots en essaim dans notre vie quotidienne. Les exemples incluent les lieux d'activités récréatives et artistiques telles que la peinture, la musique ou même la danse. Dans cet article, nous présenterons une étude typique des systèmes multi-robots, la formation de motifs, inspirée de phénomènes importants rencontrés dans les organismes vivants (tels que les animaux et les plantes). C'est un défi de résoudre le problème de l'organisation d'un groupe de robots en formant des motifs complets avec des formes simples telles que des hexagones, des lignes ou une distribution uniforme. La conception et la configuration des contrôleurs pour les systèmes de robots en essaim ont été utilisées pour obtenir les formes de manière organisée en peu de temps avec différents capteurs. Notre approche s'inspire de la nature, et en particulier des forces mécaniques impliquées dans les études cellulaires endogènes, afin de produire des formes d'assemblage auto-organisées à l'aide de robots. Nous avons utilisé notre simulateur pilote sur un émulateur Argos. Cela nous permet d'étudier les aspects du comportement de groupe organisé.

En outre, deux types de robots, le robot volant et le robot marcheur, ont été utilisés en appliquant l'utilisation de la communication entre les essaims et l'application de la physique industrielle, un modèle de réseau retardateur de ressort.

Table of content

General Introduction

For more than 50 years, researchers have worked towards the development of autonomous mobile robots. Just as stationary manipulation robots have revolutionized factory assembly lines, mobile robots have the potential to change the way humans live and work. The ability to move around allows mobile robots, for example, to clean the environment, to fetch and deliver goods, to examine hostile or remote areas, to search and rescue victims in a disaster scenario – in short: to accomplish strenuous, repetitive, or dangerous tasks that previously had to be done by human workers.

Swarm intelligence systems consist typically of a population of relatively simple agents which interact only locally with each other and with their environment, without global knowledge about their own state and of the state of the world. Furthermore, the agents follow often very simple rules and exhibit, to a certain degree, random interactions between each other. A recurrent property of these systems is the emergence of “intelligent” global behaviors, without centralized control/Swarm intelligence is often inspired from the behavior of social insects and from other groups of animals. Examples include ant colonies bird flocking animal herding colony of bacteria and fish schooling.

Multi-robot systems are systems composed of several mobile robots where a large number of simple robots, relative to the complexity of the mission, is coordinated to accomplish a task. Thanks to their robustness and modularity, they can perform complex tasks more efficiently than single robots. Although SRMs are being intensively studied today, they are still in limited industrial use. This chapter presents an overview of SRMs, highlighting their advantages, limitations, challenges and taxonomy. It then presents an analysis of some academic contributions to highlight how SRMs are characterized and experimented and highlights the main tools used for SRM work. Finally, he concludes with a presentation of the challenges preventing the deployment of SRMs in more industrial applications.

The main idea behind swarm robotics is the study of how to coordinate large groups of relatively simple robots through the use of local rules. It focuses on studying the design of a large number of relatively simple robots, their physical bodies and their control behaviors to achieve a specific task that is beyond the capability of a single robot Swarm robotics is closely related to the idea of swarm intelligence and shares its interest in self-organizing decentralized systems. It offers several advantages for robotic applications such as scalability, flexibility and robustness due to redundancy In its early age, swarm robotic was involved in mimicking intelligent swarming behaviors of social animals such as foraging aggregation flocking, cooperative transport of objects and self-organized patterns formation Recently, with the tremendous progress being made in this area, researchers are focusing primarily on how a swarm robot system can be involved in our real life. Today, they can be effectively involved in military (e.g., collective bomb detection, cooperative research and exploration), logistics (e.g., managing warehouses and products delivery to customers), agriculture (e.g., seeding, harvesting and grains storage) and emergency (e.g., rescuing in disasters), etc.

The objective of our work is to propose a simple solution to the training control task for a swarm robotic system. Using a virtual spring damper mesh interaction model to realize basic geometric formations such as circle, line, taking into account the execution time for the task using two different sensors.

Swarm robotics is a relatively new field of research inspired by intelligence and robotics in

swarm.in chapter 1 we introduced swarm robotics and its relation to the natural world like insects and animals , we talked about its vast properties (scalability, robustness,flixibilty,adaptivness,Self-organization,Parallel functionality)and its probabilistic ,deterministic ,artificial evolution ,morphogenesis-based and artificial physics approaches.

In chapter 2, we defined that Argos (Autonomous Robots Go Swarming) is a multi-physics robot simulator. It can simulate large-scale swarms of robots of any kind efficiently. We talked about the architecture of Argos core and its schematic form and we mention some robots types like (eye-bot, foot-bot, key-bot,E-puck) , and it worth to mention the existence of other simulators like Gazebo, player/stage..etc.,

In chapter 3, we dive into coordination management as a central task of SR & MRS technology, we talked about the quality of the coordination and its overall performance in the system. We moved to the physics arterial approach we choose spring damper mesh as a model we touched its mathematical questions to control swarm robots.

In chapter 4, we explained the basic interstation of Argos platform simulation we saw the xml files (containing the design of the arena and its component) control files C++ (containing the functions that control the bots), we talked about some important parts of bots (sensors & actuates) at the end of the experiment we represented the result of the simulation at several steps (initial state, secondary state, final state) in relation to time and position

Chapter 1 Introduction of swarm robotics

1.1 Introduction

In computing and artificial intelligence (AI), swarm intelligence (SI) is a field in which considered an important concept with emerging properties, which designs and studies efficient calculation methods for solving complex problems. [1]

It is inspired in particular by the behavior of biological systems such as real swarms or insect colonies. Swarm intelligence is the collective behavior of nature decentralized self-organized systems. It is natural or artificial. The SI concept is applied in many works in artificial intelligence. [1]

Gerardo Beni introduced this concept in 1989, in the context of systems cellular robotics. They consist of a population of simple agents that follow very simple rules by interacting locally with each other and with their environment. In the absence of a centralized control structure dictating the behavior of individual agents, it functions randomly and the interactions between these agents lead to the emergence of a global "intelligent" behavior, unknown to agents individual. Ant colonies, bird drying, breeding, growth Bacterial and fish farming are natural examples of Swarm Intelligence. [1]

Robotics is a scientific and engineering discipline that is focus on the understanding and use of artificial, embodied capabilities. [1]

In the late of 1980s, multi-robot systems (MRS) was specifically introduced to address the lack of information processing capacity and many other aspects of single mobile robots that cannot perform special tasks like those needing cooperation and collaboration between groups of robots. [2]

Swarm Robotics is another way to deal with the coordination of multi-robot frameworks, which comprise of substantial quantities of for the most part basic physical robots. It is assumed that a coveted aggregate conduct rises up out of the cooperation between the robots and connections of robots with nature. [2]

1.2 Overview of swarm robots

This is an emerging field of research in which the theory of swarm intelligence is applied to multi-robotic system swarm robotics is a sub-domain very particular of collective robotics which studies how to coordinate large groups relatively simple robots, using local rules. [3]

It focuses on the study of the design of a large number of relatively simple robots, their physical bodies and their control behaviors. Since its introduction in 2000, several successful experiments have been carried out have been completed. [3]

1.3 Social motivation and inspiration of insects

The collective behavior of social insects, such as the honeybee dance, the construction of the wasp nest, the construction of the path that follows the ants, has long been considered as strange and mysterious aspects of biology. The researchers have demonstrated that individuals do not need representation or knowledge sophisticated to produce such complex behaviors. In social insects, individuals are able to exchange information without no help or leader to guide them; for example, to communicate the location of a food source, an area of favorable food search or the presence of a danger to their companions. This interaction between individuals is based on the concept of locality, where there is no knowledge of the overall situation. [4]

The implicit communication resulting from changes in the environment is called Stigmergy. Insects change their behavior because of the changes brought by their partners into the environment. This can be seen in the termite nest building, where changes in worker behavior are determined by the structure of the nest. [4]

Organization emerges from interactions between individuals and between individuals and the environment. These interactions spread throughout the colony and, as a result, the colony can solve tasks that could not be solved by a single individual. These Collective behaviors are defined as self-organizing behaviors. The theories on self-organization, borrowed from the fields of physics and chemistry, can be used to explain how social insects manifest a complex collective behavior resulting from interactions between individuals. [4]

Self-organization it has based on a combination of the following four basic rules: feedback positive, negative feedback, randomness and multiple interactions [4]

1.4 Swarm Intelligence and Stigmergy

Swarm intelligence as an innovative distributed intelligence approach for the optimization problems, as well as specific kinds of general problem solving relies on the ideas of emergence. It uses social swarming behaviors observed in nature as a blueprint to design complex emergent systems. Depending on the underlying concept in nature, such as bird flocks, bee swarms, or ant colonies, different categories of swarm intelligence systems can be identified. Ants and ant colonies, as the most popular technique, will be discussed here in some detail. [16]

1.5 The Genius of Natural Swarm Systems

In nature, collective complex behaviors exhibited by grouping of insects or animals are generally associated to the term of “swarm” [3]. This last and refereeing to Hinchey, Sterritt, and Rouff [4], is defined as “images of large groups of small insects in which each member performs a simple role, but the action produces complex behavior as a whole”. This means that the observed complex or macroscopic behavior of the whole swarm system is produced from the combination of the simple (microscopic) behaviors of the numerous simple individual entities that constitute the swarm system. The entities have the ability to achieve significant results as a team resulting from their interactions with the environment, and their local interactions between each other [5].

In our daily life, there exist many kinds of natural grouping systems that can produce interesting and unexpectedly complex collective behaviors, which have been, became sources of inspiration for many research domains. The most relevant ones are (**Bird flocking, swarming of bees, Fish schools, Ants’ colonies**).

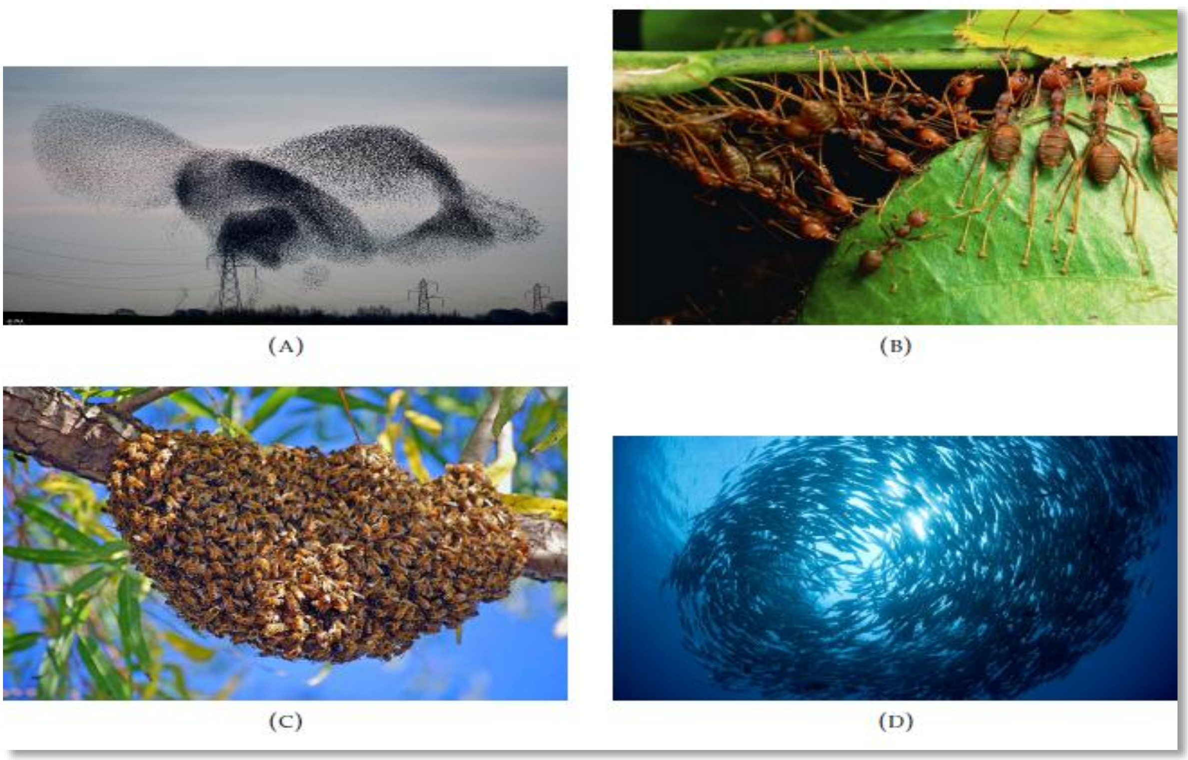

Figure 1.1: Examples of natural swarm systems [5]

1.6 Swarm Intelligence Systems: Properties

A swarm intelligence system is characterized by a set of special features such as:

- **Robustness**: Kitano as a property that allows a system to maintain its functions despite external and internal perturbations defines this feature. It means that the system should still perform even if some individuals fail. [6]
- **Adaptiveness** (Flexibility): Adaptiveness is a basic biological phenomenon, where by an organism becomes better suited to its habitat. This means that the system has the ability to adapt to any changing environment. [6]
- **Scalability**: This means that the high levels of system functionality should be maintained even though the size of individuals is increased. The behavior of the whole swarm shouldn't be perturbed by adding a new individual which can only influence the behavior of a few others. In artificial systems, this is extremely significant since the performance of a scalable system can be increased by simply increasing the size without having the need to reprogram. [6]
- **Self-organization (decentralized control):** Widely well-known in biological systems such as cells, organisms and groups that possess a large number of subunits. The individual subunits are working as a group without neither local supervision nor central control. [6]
- **Parallel functionality**: This is possible in a swarm system as different operations at different places at the same time can be performed by different individual entities. This helps to make an artificial swarm intelligence system more flexible, and enables it to powerfully self-organize and perform different aspects of a highly complex task. [6]

1.7 Multi-robotics

Multi-robot systems (MRS) are born to overcome the lack in information processing capability and many other aspects of single robots that are not capable to dial with special tasks, which in order to be efficiently completed need cooperation and collaboration between groups of robots [7]. Since its introduction in the late 1980s, various works (such as: cellular robotics, collective robotics, and distributed robotics) have been issued to describe group of simple physical robots collaborating together to perform specific tasks. MRS have also achieve a great success and made a great progress in many areas such as cooperative transportation and aggregation, environmental monitoring, search-and-rescue missions, foraging, and space exploration in such tasks, even the simplicity in design and the low-cost in productivity, as well as the increase in capabilities, flexibility, and fault tolerance advantages gained when using multi-robots instead of a single one. However, with the new arising challenges such as decentralization in control and self-organization, researchers in multi robotic field begun to make attention to the increase progress known in swarm intelligence systems giving birth to the new sub-domain of research called "swarm robotics". [8]

1.8 Swarm robotics and the difference with multi robotics

Swarm robotics is a very particular and peculiar sub-area of collective robotics in which swarm intelligence techniques are applied. The 2000 year has witnesses the first project "swarm-bot" that has been marked as the real period of the development of swarm robotics. The inventor of ant colony algorithm Marco Dorigo shared the project, and it aimed to study new approaches to the design and implementation of self-organizing and self-assembling artifacts. [9]

Dorigo ET Alones of the founders of swarm robotics gave a definition to this research domain as follow: "Swarm robotics can be loosely defined as the study of how collectively intelligent behavior can emerge from local interactions of a large number of relatively simple physically embodied agents". [10]

The main idea of the approach behind this field of research is to build relatively many small and low-cost robots that are supposed to accomplish the same task as a single complex robot or a small group of complex robots [11].

The approach also takes into account studying the design of robots (both their physical bodies and their controlling behaviors) in a way that a desired collective behavior emerges from the inter-robot interactions and the interactions of the robots with the environment [12].

Further, as the key properties of a typical SI system can be applied to either MRS and Swarm robot systems (SRS), a set of criteria has been highlighted by Sahin [14] to overcome the confusions raised about the use of the term "swarm" and the overlapping meanings applied to multi-robot research. Dorigo and Shahin set criteria that are not meant to be used as a checklist, rather they help evaluating the degree to which SR might be applied and how it might be different from other MRS, are described as follow

1. Autonomy: A SR system is made up of autonomous robots that are able to physically interact with the environment and affect it. [8]
2. Large number: A SR system should be consisted of limited homogeneous groups of robots in which each group contains of large number of members. Hence, highly heterogeneous robot groups tend to fall outside swarm robotics. [8]
3. Limited capabilities: A SR system is composed of robots relatively incapable or inefficient to carry out tasks on their own but they are highly efficient when they cooperate. [8]

4. Scalability and robustness: A SR system should be scalable and robust. Increasing the number of unites will improve the performance of the overall system and on the other hand, reducing some units will not yield to a breakdown of the system. [8]
5. Distributed coordination: In SRS, the coordination between robots is distributed, each robot should only have local and limited sensing and communication abilities. Based on this set of criteria, SRS are more beneficent than MRS which might be used whenever several robotic platforms are applied to achieve a mission. [8]

- **Differentiation**

	Swarm robotics systems	**Multi-robotics systems**
Population size	Variation in great range	Small
Control	Decentralized and autonomous	Centralized or remote
Flexibility	High	Low
Scalability	High	Low
Environment	Unknown	Known or unknown
Motion	Yes	Yes
Typical applications	Post-disaster relief, military application, dangerous application	Transportation sensing robot football

Figure 1.2: Comparison of swarm robotics systems and multi-Robotics Systems. [8]

1.9 Approaches on aggregation schemes in swarm robotics

1.9.1 Signal-based methods

Its biological counterparts inspire these methods where special signals are used to activate the aggregation process. [4]

In a study on "aggregation based on signals", a light source was used to aggregate a robotic system around an object and to transport the object collaboratively to another objective. By following simple behaviors (for example, finding light and following light), the task aggregation process was completed without any explicit communication mechanism and propose an infrared (IR) based method to regulate the size of an aggregate created by a robotic system allowing each robot to approach the size of the aggregate and decide to join him or leave him accordingly.

They used an aggregation approach based on benchmarks for solving the problem of collective decision-making in a system of robotics in a swarm. [4]

Recently, one of the most important "cue-based" aggregation models has been inspired by the collective behavior of honeybees, who prefer the picked at a temperature of 36° C. The BEECLUST model was the first algorithm that imitated this behavior; a progressive light source was used to generate a grouping behavior in a swarm robotic system. [4]

Different variations of the model have been suggested to improve the performance of the aggregation process. [4]

They are have proposed a new aggregation algorithm in which a dynamic velocity and a time of comparative waiting times were introduced in the original BEECLUST model, which contributed to a significant improvement in aggregation time. [4]

They proposed another one adaptive variant of BEECLUST, in which the original algorithm has been extended to automatically adapt to all lighting conditions. [4]

1.9.2 Self-organized methods

In methods based on self-organization, aggregation models are obtained by using simple rules of local interaction between individuals. In the subsections we highlight the different approaches proposed so far in the literature. [7]

-Probabilistic approach

Most of the work in this approach used a finite state probabilistic machine to control the behavior of the swarm. For example, they have adopted a probabilistic approach, inspired by the cockroach model, to achieve aggregation using a swarm of 20 robots physical Alice in homogeneous environments. A similar work by Nikolaus Correll has shown that, when using probabilistic aggregation rules, a minimum combination of communication range and travel speed was necessary to obtain a single group of aggregates Onur Soysal suggested a Probabilistic aggregation method in which a finite state machine was used to combine a set of simple behaviors, including avoiding an obstacle, to approach, repel and wait. [7]

-Deterministic approach

In this approach, robots usually build a connected visibility graph and ensure that it is maintained at all times Ando used an algorithm in this sense to study the aggregation in a group of mobile robots with a detection range limited. The formation of the graph in these algorithms was based on the assumption that robots were able to measure both the distance and angle of their neighbors. However, Gordon, Wagner have been able to obtain such an aggregation using only measuring the angle of the robot's neighbors. The aggregation performance of the latter Algorithm were subsequently improved, by introducing an additional capacity of coarse distance detection to differentiate if neighboring robots were close or distant. In another study, De Gennaro used the Laplacian matrix to allow each robot to create its own proximity graph. The associated control was fully decentralized and the simulated results demonstrated that the model was effective and even increased the connectivity of the entire swarm. [7]

-Approach to artificial evolution

In some studies, self-organized aggregation models have been addressed using artificial evolutionary techniques. For example, aggregation with simple robots, called s-bots, has been studied. In this study, general solutions to the aggregation problem have been found to have been developed using a scalable robotic mechanism. The method was able to produce clustering behaviors with static behavioral strategies and dynamic. In another study Gauci proposed two algorithms, one reactive controller without memory and a recurrent controller with memory for studying aggregation into a swarm of e-puck robots. The algorithms were based on a technique of classical scalable programming and used a simple binary sensor with a range sufficient detection to obtain error-free aggregation. The results of the simulation and experiments have shown that aggregation to a cluster has been successfully achieved. However, a sufficiently long range in the binary sensor was required to obtain a precise aggregation. [7]

-Morphogenesis-based approach

Biological morphogenesis, including its internal genetic and cellular mechanisms, is recently become a source of inspiration for many multi-robotic studies. This has given rise to morphogenetic robotics as a new field of emerging robotics research to study the self-organization of swarm robots or modular. To solve the problem of aggregation models in this context, Guo have established a metaphor between multi-cellular systems and multi-robot systems in order to propose a decentralized algorithm based on GRN (Gene Regulatory Network) for the construction of multi-robot shapes. Thanks to this NRM model, several robots can perform a self-organization autonomous in different predefined forms and an adaptive self-reorganization in different dynamic environments. Later, the authors proposed two extensions of the original model: by introducing, a form representation free to create more complex 2D or 3D patterns with a b-spline structure rational, and by adopting a hierarchical gene regulation network H-GRN (H-GRN). It is a model for the adaptive generation of multi-robot models and training in changing environments. [7]

-Approach to artificial physics

This approach, which belongs to bio-inspired methods, is based on the observation of the physical. It was first introduced by William M Spears as physico-mimetic framework (or artificial physics). To control the behavior of the entire swarm system, the frame uses virtual physics forces generated by the interactions of robots. The framework was able, by using two types of laws of force physical: the Newtonian law of force and the Leanar-Jones law of force, to lead to large groups of aggregation agents moving in a desired formation such as a hexagonal network. In addition, the framework is extended to manage training in motion at through the obstacle fields. [7]

Spring-based virtual control models have recognized a significant interest in the in recent years. In these models, virtual spring forces are applied to maintain a desired distance between the aggregated robots, while providing a certain amount of flexibility to structure and flexibility to movement. Donghwa Jeong have developed a dynamic model based on a virtual spring damper for a system of artificial swarms, in which algorithms for dispersion formation and the aggregation of lines are proposed to achieve attractive and repulsive forces between the artificial agents and their neighbors. The dispersion algorithm is based on elements of trigonal planning without a leader while the line formation algorithm uses line elements paired with a temporary leader. [7]

1.10 Fields of application of swarm robotics

1.10.1 Tasks cover a large area

The swarm robotics system is distributed and specialized for tasks requiring a large area of space, for example, tasks cover large areas. An example of this would be simple to search for and collect several targets in an open area. The swarm tries to search with group cooperation to speed up the search. The area can be very large and the swarm can benefit from parallel research with several small groups located in the detection zones of robots. [10]

1.10.2 Dangerous tasks for the robot

Thanks to scalability and stability, the swarm provides redundancy to handle tasks dangerous. The swarm can suffer a considerable loss of robots before the work is done interrupted. Robots are very cheap and are preferred for areas that damage probably the workers. In some tasks, robots may be unrecoverable and the use of complex and expensive robots is therefore economically unacceptable, while robotics in swarm with cheap individuals can provide solutions reasonable. [10]

1.10.3 Tasks require a population of scale

The workload of some tasks may change over time, and the size of the swarm should be reduced according to the current workload for high efficiency, both by in terms of time and economy. For example, to eliminate oil leaks after tank accidents, the swarm should maintain a high population when the oil leaks quickly at the beginning of the task and gradually reduce the number of robots when the source of the leak is blocked and the leak area is almost completely cleaned. [10]

1.10.4 Tasks require redundancy

The robustness of swarm robotic systems benefits mainly from the redundancy of the swarm, i.e. the withdrawal of some robots does not have a significant impact on the performance. Some tasks focus on the result rather than the process, i.e. say that the system must ensure that the task will be successfully completed, mainly in the form of increasing redundancy. [10]

1.11 Swarm Robotics Problems Focus

In the last decade, swarm robotics researches has known a significant progress due to the advantages gained when using such technology to solve many problems that are beyond the capabilities of classical multi-robots systems. The problems involve in swarm robotics research can be classified into those mainly based on the patterns (e.g. aggregation, cartography, migration, self-organizing grids, deployment of distributed agents and area coverage), those focused on the entities in the environment (e.g. Searching for the targets, detecting the odor sources, locating the ore veins in wild field, foraging, rescuing the victims in disaster areas and), and those mostly Hybrid of the two previous problems (e.g. cooperative transportation, exploring a planet and navigating in large area). Brambilla Illustrates another classification of the problems involved in swarm robotics based on the collective behavior problems focus. [8]

1.12 Conclusion

Swarm robotics is a relatively new field of research inspired by intelligence and robotics in swarm. This is the result of the application of techniques from swarm intelligence to multi-robotics. In this chapter, we have presented an overview of swarm robotics as well as the theories based on this field to better understand specialty research.

Chapter 2 Platform Simulation (ARGoS)

2.1 Introduction

Argos (Autonomous Robots Go Swarming) is a *multi-physics* robot simulator. It can simulate large-scale swarms of robots of any kind efficiently. You can customize Argos easily by adding new plug-ins [12]. Argos is a discrete-time simulator for multi-robot systems acting in 3D spaces. The main code is entirely written in C++ and is all based on the use of free software libraries. Each robot, and, more generally, each entity active in the 3D space, is a seen as composition of a physical structure, a set of sensors and actuators, and a controller module. The simulator includes an extensive set of sensors and actuators, and in particular all those necessary to allow the realistic simulation of the three types of robots composing the swarmanoid. The simulator also includes multiple physics engines that allow dealing with the simulation of the physics of movements and collisions according to different levels of realism and computational effort. During the same simulation run, different physics engines can handle the physics of different groups of robots. [12]

2.2 Swarm Robotic Simulation Platforms

- Player/stage is a combined package of open source software tools developed for robot and sensor applications.
- Gazebo It includes an accurate simulation of the physics of rigid bodies (realistic sensor feedback and possible interactions between objects can then be generated).
- ARGoS is a new pluggable multi-physics engine allowing to simulate in real time the massive and heterogeneous robotics of swarms.

- It is easy to implement and use.
- multiple physics engines can be used in a single experiment,
- the robots can migrate from one to the other in a transparent way.
-ARGoS can also be implemented in parallel in the simulation.

- Enki It is a robot simulator based on fast 2D physics written in C ++ It is also able to simulate the kinematics, collision, sensors and cameras of robots working on a flat surface.
- USARSim (Unified System for Automation and Robot Simulation)

-is a high-fidelity multi-robot simulator originally developed for the Search and Rescue (SAR) activities of the Robocup competition.

-It is based on a widely used game engine, Unreal Engine 2.0.

The simulator takes full advantage of high-precision physics, noise simulation and numerous engine geometries and models. [13]

2.3 Platform Simulation (ARGoS)

ARGoS a novel open source multi-robot simulator. ARGoS was developed within the EU-funded Swarmanoid project1, which was dedicated to the study of tools and control strategies for heterogeneous swarms of robots. Simulation is central to the study of swarm robotics for several reasons. In general, simulation allows for cheaper and faster collection of experimental data, without the risk of damaging the (often-expensive) real hardware platforms. [14]

ARGoS extensibility is ensured by ARGoS' highly modular architecture robots, sensors, actuators, visualizations and physics engines are implemented as user-defined modules. Multiple implementations of each type of module are possible. The user can choose which modules to utilize in an experiment through an intuitive XML configuration file. To obtain scalability, the ARGoS architecture is multi-threaded and is designed to optimize CPU usage. Performance can be further enhanced by choosing appropriate modules. For instance, there are many possible models for each specific sensor or actuator, characterized by differences in accuracy and computational cost. Each model is implemented into an ARGoS module the architecture of ARGoS is depicted in Figure 2.1, and below a short description about its main modules. [14]

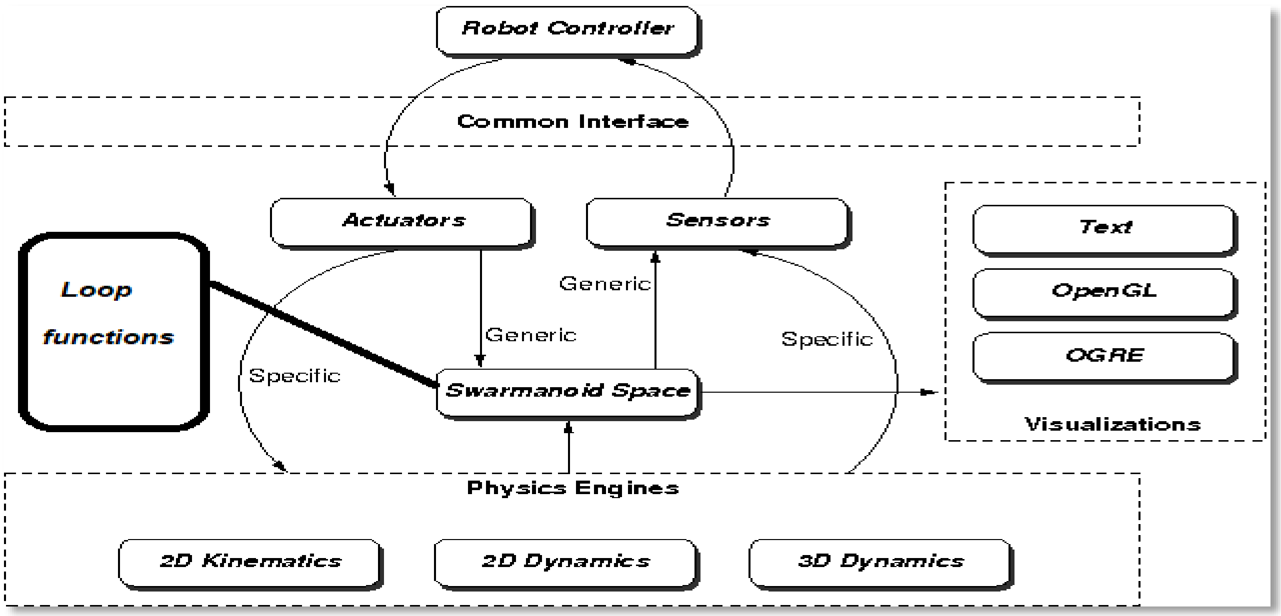

FIGURE 2.1: Schematic description of the ARGoS architecture [14]

• **Swarmanoid space**: This module is the core component of ARGoS architecture. It is a set of entities, which is considered as a central warehouse containing all the relevant information (i.e. position and orientation of robots or obstacles) about the state of the simulation. [14]

• **Sensors and actuators**: Sensors are plug-ins that read the state of the simulated 3D space, with accessing only to specific types of entities to perform their calculations (i.e. a range sensor module needs to access information about embodied entities only). Analogously, actuator plug-ins update the state of the core component via writing into the components of a robot for example. [14]

• **Physics engines**: With the Physics engines modules, the state of the embodied entities can be updated during an experiment by running multiple engines in parallel. This can be achieved by assigning each physics engine to a different part of the embodied entities. [14]

• **Visualizations**: visualizations modules are rendering mechanisms that make an output representation of the state of the simulated 3D space. By default, ARGoS offers an interactive graphical user interface based on Qt and OpenGL. However, a high-quality rendering engine based on POV-Ray can also be used. [14]

• **Controllers**: controller is a plug-in that interact with the core component through sensors and actuators. It is an implementation of the individual behavior of an entity (i.e. a robot). Currently, it can be implemented in ARGoS using C++ or lua programing languages. [14]

• **Loop functions**: Loop functions are functions hooks that are defined by user, and which can be placed in precise points in the simulation loop. For example, they can be defined at initialization time, or before and after the execution of the update phase. Through this feature and at each simulation step, the physics engine and its state can be queried and modified, while data could be also collected for further visualization and analysis. [14]

Furthermore, ARGoS has built-in models for several well-known robots such as foot-bot, e-puck, kilobot and flybots, etc. To run an ARGoS based simulation experiment, two main components must be provided: a set of controllers and a configuration file. The controllers are user codes that include implementations of the individual behavior of the robots, and optionally, specific functions to be executed in different parts of ARGoS to interact with the running experiment. Currently, they can be implemented using C++ or lua programing languages. The configuration file is an XML file in which a description of the structure of the simulated environment is provided, it contains all required information to set up simulated entities such as the arena, the robots, the physics engines, etc. [14]

2.4 ARGOS Robotique Platform

To discuss the modularity of ARGoS, we describe how a robot is modeled in ARGoS. The robot we chose for this case study is the football-bot. Football is a ground-based robot that moves with a combination of wheels and tracks (called treels). The robot is equipped with sensor and actuator and allows interaction with the environment.[17]

2.4.1 Foot-bot

The foot-bot (Figure 2.2) is a two wheels differential mobile robot of about 17 cm of diameter and 29 cm of height, designed and built within the context of the SWARMBOTS project. The foot-bot can move using a combination of wheel sand tracks (called treels), and it comes with various sensors and actuators that allow interaction with the surrounding environment. Thereafter, a list of the most relevant ones used in swarm robotics studies:

• **Twelve (12) RGB LEDs** composing a ring that surrounds the robot body, and through witch colored patterns can be displayed to the other robots.

• **An Omni-directional camera** that can be used to perceive colored objects displayed by other robots (up to a distance of approximately 50 cm).

• **Four(4)** ground sensors placed under the chassis can be used to perceive markers or holes on the ground.

• **Twenty four (24) IR** sensors for obstacle and proximity detection.

• **A range-and-bearing communication device**, called RAB, for exchanging messages between robots within a limited range.

• **A gripper connector** for allowing the foot-bot to perform physical connections such as gripping abject or robot-to-robot connections.

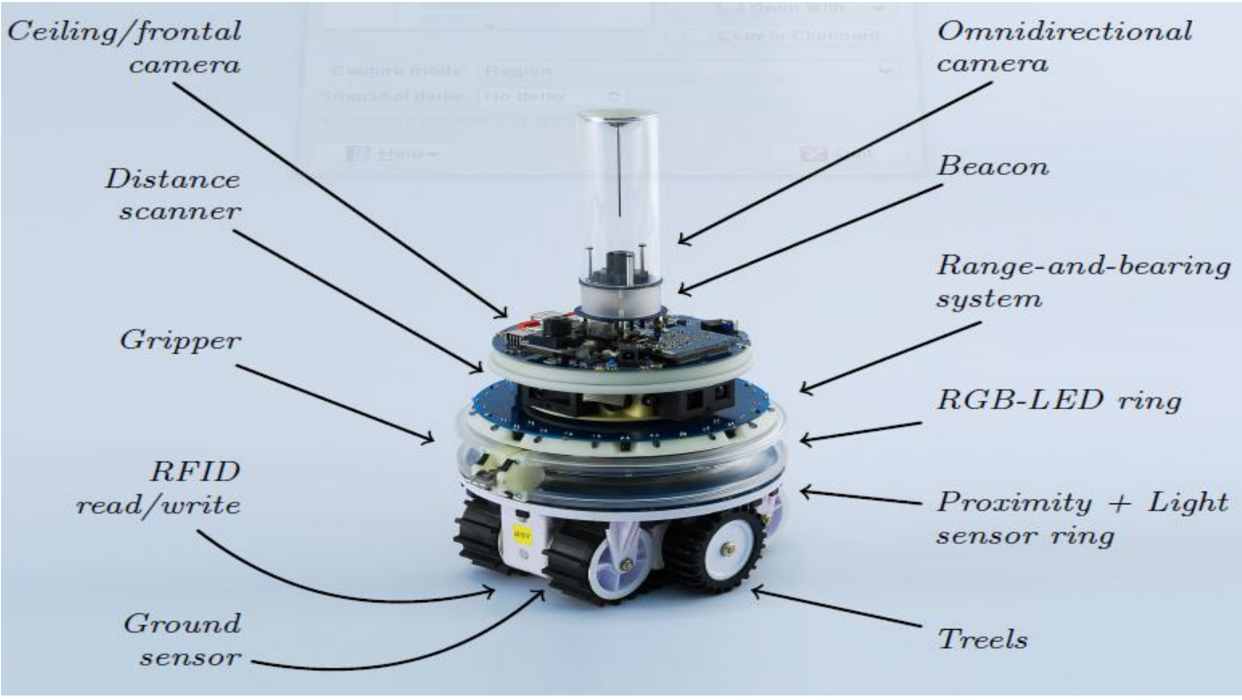

Figure 2.2 the foot-bot [17]

2.4.2 E-puck

The e-puck (Figure 2.3) is a wheeled cylindrical robot of approximately 7.4 cm of diameter and 5.5 cm of height. Mondada et al. eventually designed it for educational researches purpose. Alike the foot-bot, the e-puck is equipped with a variety of sensors, and whose mobility is ensured by a differential drive system. The e-puck has nearly the same sensors/actuators of the foot-bot and below a list of the main ones used in most of the swarm robotics studies [15].

• **Four (4) RGB LEDs** evenly disposed around the e-puck body and which can be lighten up independently.

• **A Bluetooth antenna** to allow the robot to establish wireless radio communication with up to any eight (8) other Bluetooth devices at the same time.

• **Eight (8) Infrared emitters** disposed approximate around thee-puck body, and which might be used to detect obstacles and receive encoded messages from other robots.

• **An extension module** for range and bearing communications, which is used to provide local communication capabilities to the robot.

• **A frontal RGB camera** that can be used for visual perception.

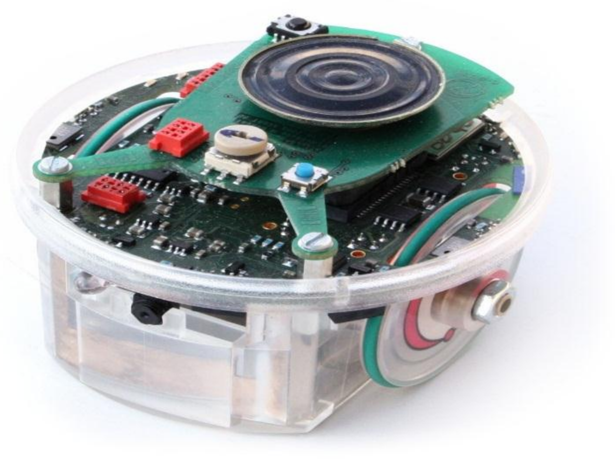

Figure 2.3 : e-puck [15].

2.4.3 Eye-bot

The Eye-bot hovering platform, shown in figure 2.3, has been specifically designed for collective indoor operation and to meet the requirements of the Swarmanoid project, by implementing the dimensioning design method. [17]

The Eye-bot structure is fabricated from quasi-isotropic carbon fiber plate, which is designed for high-strength and light weight. The body is 54 cm tall with a base diameter of 50 cm. In order to attach to ferrous ceilings, a celling attachment and detachment mechanism is installed at the top of the body. This mechanism allows the Eye-bot to have an extended perching time up to 3 hours, while maintaining a bird's-eye view. At the base there are four Himax CR2805 coaxial brushless motors that are fitted with twin counter-rotating APC 7x5E/EP inch propellers. A detailed explanation of the structure is provided in appendix A.1. [17]

The Eye-bot electronics consists of several distributed control boards and an embedded computer. The distributed control, includes an eight channel high speed (500 Hz) brushless motor controller, flight controller, relative positioning controller and autonomous controller. The autonomous controller runs the flight control algorithm, the relative positioning controller runs the 3-D sensing algorithm, which produces the range, bearing, elevation and proximity sensing information used by the autonomous controller. The flight controller is in charge of the attitude estimation and stability controllers this flight controller outputs the eight motor speed commands used by the brushless motor controller. A detailed explanation of the custom designed control electronics is provided in appendix A.2. [17]

	Diameter	Endurance	Motor-propeller Efficiency	Payload	Structure Weight	Battery Weight
Design:	50 cm	10 min	4.3 g/W	600 g	952 g	449 g
Real:	50 cm	9.9 min	4.4 g/W	600 g	897 g	451 g

Table 2.4: Comparison between the design and the real platform performance [17]

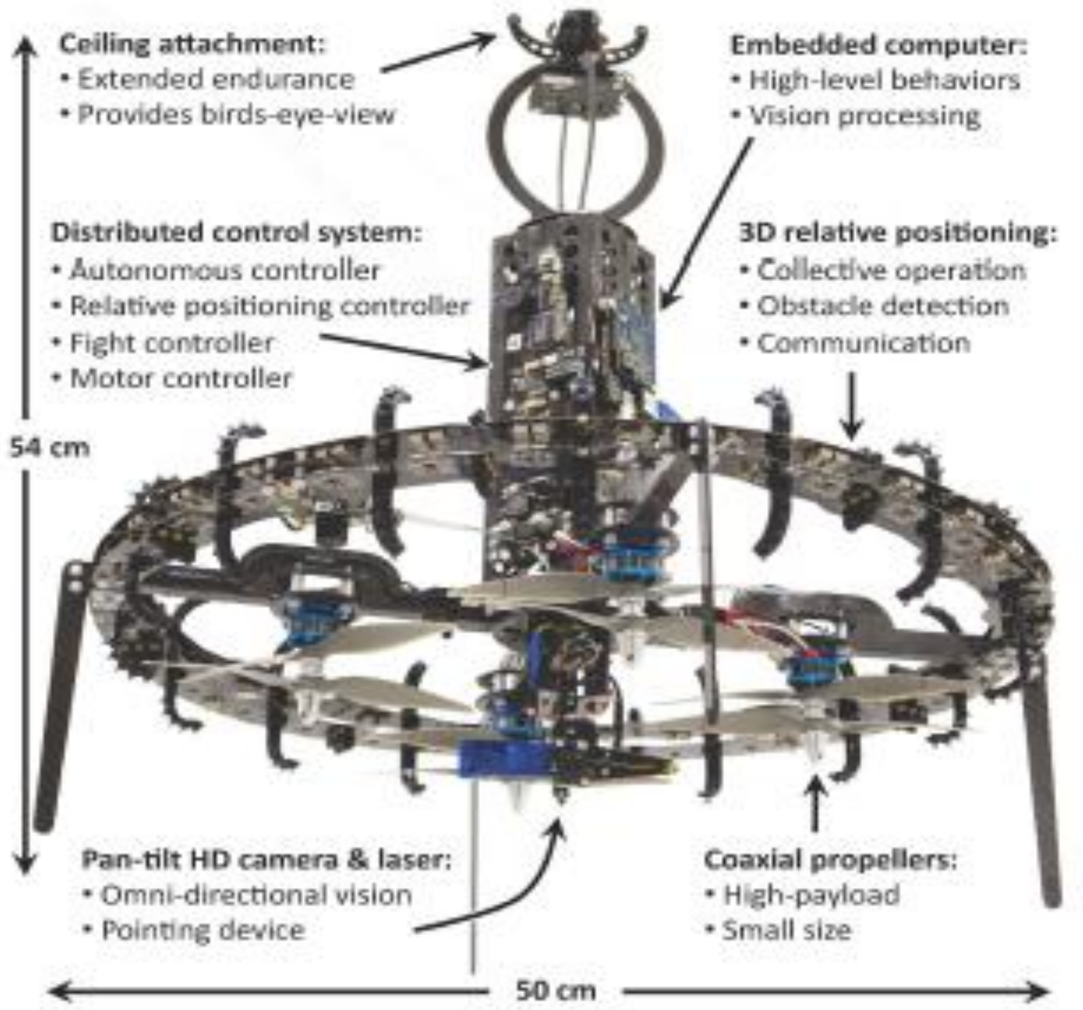

Figure 2.5: The Eye-bot [17]

- A flying robot custom designed for indoor collective operation, fitted with a 3-D relative positioning sensor (base perimeter) and ceiling attachment mechanism (top), having a diameter of 50 cm and a maximum takeoff weight of 2 kg including a payload of 600 g [17]

2.5 On-board Sensing/Actuating System

In this section, we give more details about the on-board sensing/actuating systems Used throughout this thesis.[17]

2.5.1 Infrared range and bearing sensors

Both the foot-bot and the e-puck robots are equipped with a board that allows them to communicate locally without the need of any external reference (For more specifications about this device readers can refer to Roberts [16] for the foot-bot robot and to Gutiérrez et al. [17] for the e-puck robot). With this communication system, a robot can communicate with its neighbors, measuring at the same time both the range and the bearing (orientation) of the sender (Figure 2.4). The range and the bearing communication board is composed of 12 infrared sensors/actuators that allow sending and receiving messages within a communication range of 6m maximum and in 12 different directions. The communication range with which infrared signals might be sensed are adjustable in real time. The particularity of this infrared communication module is that the same message can be sent either in all directions, or specifically in one direction through setting witch sensors/actuator pair is used to send the message.[17]

.

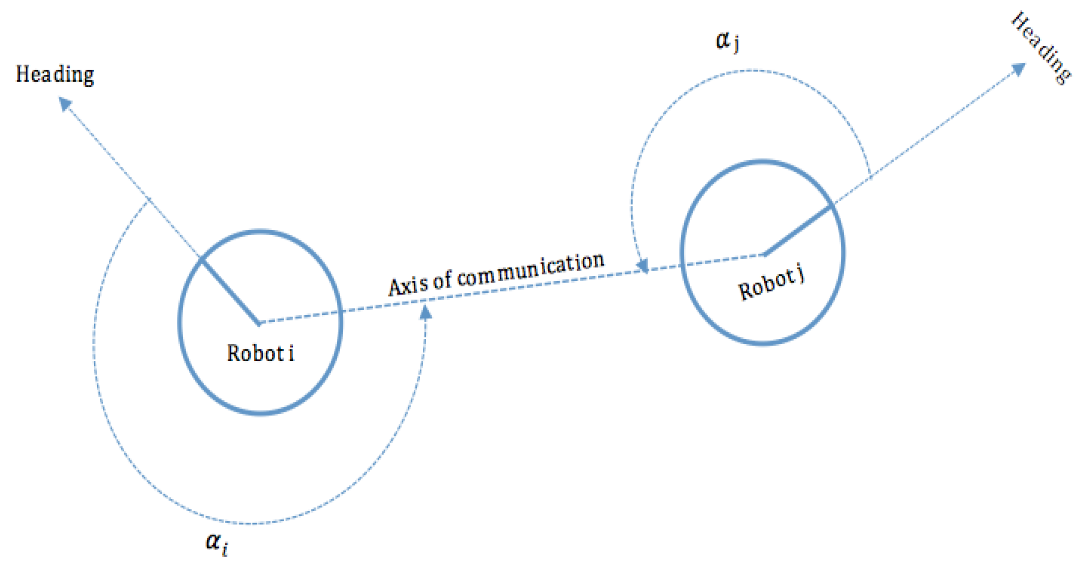

FIGURE 2.6: The relative position measurements within the communication system.[17]

In ARGoS, the user can configure a range and bearing communication device as a module added to the list of sensors/actuators node in the xml configuration file. If any specific additional parameters are configured to this device, the behavior of the sensor will be perfect meaning that any message can be received within the configured communication range, and that the relative position (distance and bearing values) of the emitter can be precisely measured. However, several parameters are available to simulate more accurately the range and communication device as it is in the real world. For example, the user can set the max_packets parameter to limit the number of messages allowed to be received in one control time-step cycle. The user can also add Gaussian noise of the form N (0, s2) to the range and bearing measures through setting the noise_std_dev parameter. Loss of data during communication can be also simulated by setting the loss probability parameter, meaning that packets are susceptible to be lost with the configured probability. Finally, it is also possible to reproduce in simulation the very noisy nature of the range measure of the physical Robots by activating the real_range_noise parameter.[17]

2.6 Conclusion

ARGoS a novel open source multi-robot simulator. ARGoS extensibility is ensured by ARGoS' highly modular architecture robots, sensors, actuators, visualizations and physic. The robot we chose for this case study is the foot-bot and eye-bot and E-puck.

Chapter 3 Swarm Robotics Coordination

3.1 Introduction

Coordination management is a central task of SR & MRS, the quality of the coordination has a direct impact on the overall performance of the system. This coordination can be defined in two ways. (Static: (offline), Dynamic (online, reactive)). [19]

The coordination has thus been described by Thomas W. Malone as "the set of additional activities that need to be performed in a multi-agent environment and that a single agent pursuing the same goals would not accomplish ".The choice of the coordination mechanism to be implemented varies according to the field of use of the system, the number of agents that make it up and their tasks to be carried out, the environment and its characteristics (stability, predictability, etc.). This mechanism aims at managing the dependencies that exist between the activities of the agents which are often incompletely specified. [18]

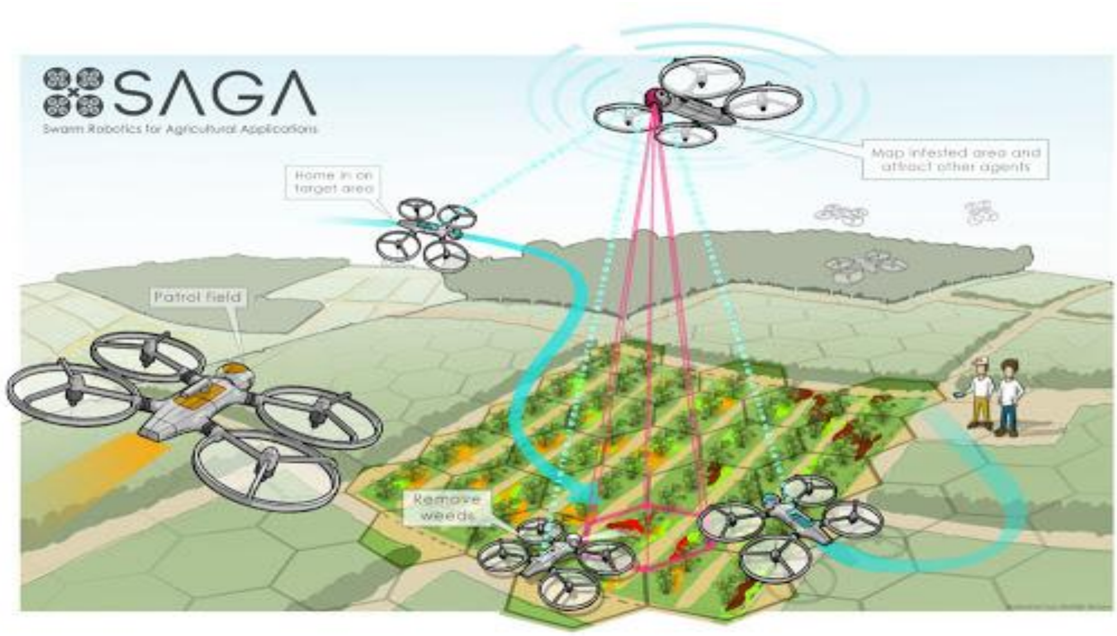

FIGURE 3.1 coordination swarm drones [20]

3.2 Why coordinate?

Coordination of actions is necessary for four main reasons:

1.Incomplete information: agents need information and results that only other agents can provide.

2. Resources are limited: resources are reduced and several agents use them (time, space, energy, money or tools). It is therefore necessary to share these resources in order to optimize the actions to be carried out, while trying to avoid possible conflicts.

3. Optimize costs: coordinating actions also reduces costs by eliminating unnecessary actions and avoiding redundant actions.

4. Moving beyond goal dependency: enabling agents with distinct but mutually dependent goals to meet those goals and get the job done.

Eventually taking advantage of that dependency. [18]

3.3Application of swarm robotics in the real world:

Yazdani Hasen cited the different applications of swarm robotics in the real world:

- Rescue missions: Swarms of robots could be sent to places where help could not be reached, saving lives.
- Mining tasks: A swarm could cover a minefield and the robots would be sacrificed on the mines. Swarms robots can be used in the army to form an autonomous army.
- Autonomous surveillance and environmental monitoring: in order to study environmental parameters, search for survivors and locate sources of danger such as chemical or gas spills, toxic pollution, pipe leaks, radioactivity.
- Wide area coverage: Robots can disperse and perform monitoring tasks, for example in forests. This can be useful for detecting dangerous events, such as a chemical spill. In military applications.
- Agriculture: A decentralized monitoring/mapping scenario and implementation of a use case for the detection and mapping of weeds in a field by a group of small unmanned aerial vehicles.
- Toxic waste clean-up: Search and rescue and collection of field samples are important applications of swarm robots.
- Exploration and mapping: would save time and money.
- Medical field: the use of nano-robots moving in human veins and arteries (for example, to fight certain types of cancer).
- Dangerous tasks: Many industries use dangerous objects such as burners, chemicals, the manufacture of nuclear weapons, etc. In this case, the use of swarms can reduce the danger in such types of industries. [21]

3.4 Models and approaches:

Aggregation models are obtained using simple rules of local interaction between individuals. In the following subsections, we highlight the different approaches proposed so far in the literature (Probabilistic approach, Deterministic approach, Morphogenesis-based approach, Approach to artificial evolution, Approach to artificial physics).[3]

3.5 Artificial Physics Approach:

- This approach, which belongs to the bio-inspired methods, takes inspiration from the observation of Physics. It was firstly introduced by Spears as a physi-comimetics (or an artificial physics) framework. To control the behavior of the whole swarm system, the framework makes use of virtual physics forces generated from the interactions of the robots. The framework was able, through using two types of physics force laws: Newtonian force law and Leaner-Jones force law, to drive large groups of aggregating agents moving into a desired formation such as a hexagonal lattice further the framework is extended to handle moving formations through obstacle fields.[19]
- Derived from this approach, different virtual physics forces laws have been so far applied in the literature. For example, Howard, Matari´c, and Sukhatme used virtual electric charges to model the deployment of robots into an unknown area, Moeslinger, Schmickl, and Crailsheim applied repulsive and attractive virtual forces to investigate flocking behavior within swarm robotics. Gasparri, Priolo, and Ulivi adopted also an attractive/repulsive virtual force model to study aggregation in a swarm of multi-robot systems based on local interaction. This model was later extended to cope with actuator saturation by Gasparri and to integrate obstacle avoidance by Leccese. In another work, Hashimoto suggested a control algorithm for a robotic swarm basing on the center of gravity of the local swarm, and this through making use of virtual forces, local forces and an advancing force law. 19]

Moreover, virtual spring based control models have recognized a significant interest in the last years. In these models, virtual spring forces are applied to maintain a desired distance among aggregated robots with providing some flexibility to the structure and smoothness to the movement. As instance in the work of Shucker and Bennett, a fully Distributed Robotic Macrosensor (DRM) control mechanism, involving a set of aggregation formation algorithms based on virtual spring mesh connectivity, is proposed to deploy a huge number of swarm robots system. The flexibility and the fault-tolerance of the aggregation formation mesh is guaranteed through introducing a novel algorithm based upon an acute-angle test used to create a mesh ofacute triangles. Bezzo and Fierro suggested a fully decentralized switched spring mesh model to investigate a multi-robots wireless communication navigation problem, which consists of moving the robots of the system in a swarming manner with maintaining communication connectivity while searching a moving target in a two-dimensional obstacle populated environment. The authors seek specifically to find the shortest path between a base station and one or more users (mobile target) that generate attractive potential fields around their center. [19]

- As an extension of virtual spring models, Virtual spring-damper based models have been also used in several studies. Some applications can be found in the work of Dewi, Risma, and Oktarina where a wedge navigation formation is created by a flock of robots using simple virtual spring-damper model between the leader and the followers; the authors only use RF communication system and distance sensors between follower robots. In the work of Urcola the authors presented a virtual structure based on spring-damper elements to control a navigation system composed of leader and followers robots in formation movement. The navigation system can adapt the formation to the environment. Jeong and Lee developed a virtual spring damper based dynamic model for an artificial swarm system, in which a dispersion and line aggregation formation algorithms are proposed to realize at-tractive and repulsive forces between the artificial agents and their neighbors; the dispersion algorithm is based on trigonal planner elements without a leader whereas the line formation algorithm use paired line elements with an interim leader. [19]

3.6 The proposed models (spring damper mesh):

We demonstrate a method for self-organization and leader following of nonholonomic robotic swarm based on spring damper mesh. By self-organization of swarm robots we mean the emergence of order in a swarm as the result of interactions among the single robots. In other words the self-organization of swarm robots mimics some natural behavior of social animals like ants among others. The dynamics of two-wheel robot is derived, and a relation between virtual forces and robot control inputs is defined in order to establish stable swarm formation. Two cases of swarm control are analyzed. [23]

3.7 Dynamics of robot swarms

Dynamics of robot swarms the most commonly used robots in swarm research are nonholonomic two-wheel robots. The robot is shown in Figure 3.2. To describe the dynamics of a two-wheel robot, we could use Euler–Lagrange's equations with multipliers. Unfortunately, one of the drawbacks is the necessity of decoupling the multipliers from robot torques. To avoid having to decouple multipliers we will use different formalism using Maggie's equations [24, 25], derived from the principle of virtual work.

$$\sum_{j=1}^{n} C_{ij}\left[\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E}{\partial \dot{q}_i}\right) - \frac{\partial E}{\partial q_i}\right] = \Theta_i, \quad i = 1, \ldots, s, \quad j = 1, \ldots, n, \qquad (1)$$

Where s is the number of independent parameters in the generalized coordinates q j equal to the numbers of the system degrees of freedom. The generalized velocities and the right-hand sides of Eq. (1) can be expressed explicitly by

$$\dot{q} = \sum_{i=1}^{s} C_{ij}\dot{e}_i + G_j, \tag{2}$$

$$\sum_{i=1}^{s} \Theta_i \delta e_i = \sum_{i=1}^{s} \delta e_i \sum_{j=1}^{n} C_{ij} Q_j. \tag{3}$$

Where e˙i , i = 1,...s, are called kinetic parameters and in this case they are equal to α˙ 1, α˙ 2, i.e., the rotation angles of the two wheels. [24, 25]

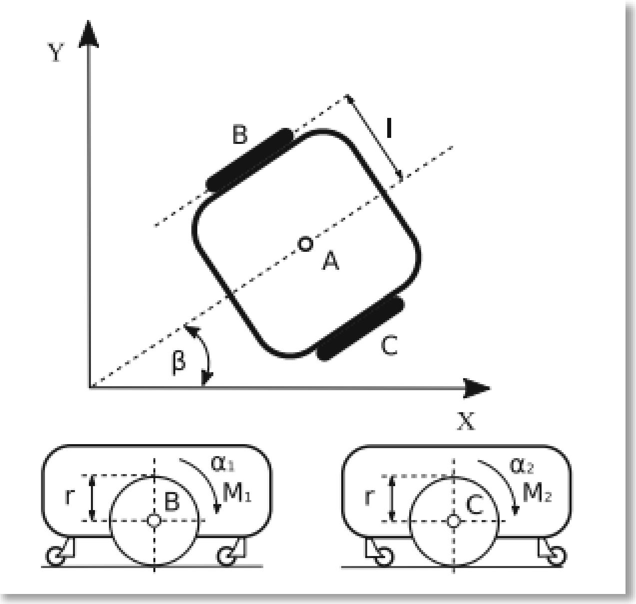

Figure. 3.2 Two-wheel robot schematic [24, 25]

The generalized coordinates and generalized velocities for two-wheel robot are

$$q = [x_A, y_A, \beta, \alpha_1, \alpha_2]^T, \quad \dot{q} = [\dot{x}_A, \dot{y}_A, \dot{\beta}, \dot{\alpha}_1, \dot{\alpha}_2]^T \tag{4}$$

Where β is the angle between the line perpendicular to the segment joining the wheel centers and the x-axis. For the described robot, we have two degrees of freedom and the independent generalized coordinates are the wheels rotation angles α1, α2. Using kinematics analysis, we determine the velocities relationships as follows

$$\dot{\beta} = \frac{r}{2l}(\dot{\alpha}_1 - \dot{\alpha}_2),\ \dot{x}_A = \frac{r}{2}(\dot{\alpha}_1 + \dot{\alpha}_2)\cos\beta,\ \ \dot{y}_A = \frac{r}{2}(\dot{\alpha}_1 + \dot{\alpha}_2)\sin\beta \tag{5}$$

Where r is the radius of the wheels and l half the distance between the wheel centers. The total kinetic energy E is equal to the sum of the kinetic energies of both wheels and robot chassis:

$$E = E_{w1} + E_{w2} + E_r \tag{6}$$

$$\begin{aligned} E_r &= \frac{1}{2}m_r v_A^2 + \frac{1}{2}I_{zr}\dot{\beta}^2, \\ E_{w1} &= \frac{1}{2}m_w v_B^2 + \frac{1}{2}I_{zw}\dot{\alpha}_1^2 + \frac{1}{2}I_{xw}\dot{\beta}^2, \\ E_{w2} &= \frac{1}{2}m_w v_C^2 + \frac{1}{2}I_{zw}\dot{\alpha}_2^2 + \frac{1}{2}I_{xw}\dot{\beta}^2, \end{aligned} \tag{7}$$

Where mw, mr, Izw, Ixw, Izr are, respectively, the masses and the moments of inertia of wheels and robot chassis with respect to the x- and z-axes. Using Eqs. (2) and (5) we can calculate the coefficients Ci j, G j , and with Eqs. (1) and (3) we determine the generalized forces Q j in the following form

$$\begin{array}{llll} C_{11} = \frac{r}{2}\cos\beta, & C_{21} = \frac{r}{2}\cos\beta, & G_1 = 0, & Q_1 = (2m_w + m_r)\ddot{x}_A, \\ C_{12} = \frac{r}{2}\sin\beta, & C_{22} = \frac{r}{2}\sin\beta, & G_2 = 0, & Q_2 = (2m_w + m_r)\ddot{y}_A, \\ C_{13} = \frac{r}{2l}, & C_{23} = -\frac{r}{2l}, & G_3 = 0, & Q_3 = 2m_w l^2\ddot{\beta} + 2I_{zw}\left(\frac{l}{r}\right)^2\ddot{\beta} + 2I_{xw}\ddot{\beta} + I_{zr}\ddot{\beta}, \\ C_{14} = 1, & C_{24} = 0, & G_4 = 0, & Q_4 = I_{zw}\ddot{\alpha}_1 \\ C_{15} = 0, & C_{25} = 1, & G_5 = 0, & Q_5 = I_{zw}\ddot{\alpha}_2. \end{array} \tag{8}$$

The generalized forces and coefficients Ci j , G j in (8) and Eq. (1) yield the following the Maggie equations

$$\begin{aligned} (2m_w + m_r)\left(\frac{r}{2}\right)^2(\ddot{\alpha}_1 + \ddot{\alpha}_2) + (2m_w l^2 + 2I_{xw} + I_{zr})\frac{r}{2l}(\ddot{\alpha}_1 - \ddot{\alpha}_2) + I_{zw}\ddot{\alpha}_1 &= M_1 - N_1, \\ (2m_w + m_r)\left(\frac{r}{2}\right)^2(\ddot{\alpha}_1 + \ddot{\alpha}_2) - (2m_w l^2 + 2I_{xw} + I_{zr})\frac{r}{2l}(\ddot{\alpha}_1 - \ddot{\alpha}_2) + I_{zw}\ddot{\alpha}_2 &= M_2 - N_2. \end{aligned} \tag{9}$$

If we write Eq. (9) in matrix form, we get

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{F}(\dot{\boldsymbol{q}}) = \boldsymbol{U}, \tag{10}$$

Where M is the inertia matrix, F (q˙) is the friction function, and U is the wheels torque vector

$$\boldsymbol{M} = \begin{bmatrix} a_1 + a_2 + a_3 & a_1 - a_2 \\ a_1 - a_2 & a_1 + a_2 + a_3 \end{bmatrix},$$
$$\boldsymbol{F}(\dot{\boldsymbol{q}}) = \begin{bmatrix} a_4 \mathrm{sgn} \dot{\alpha}_1 \\ a_5 \mathrm{sgn} \dot{\alpha}_2 \end{bmatrix},$$
$$\boldsymbol{U} = [M_1, M_2]^{\mathrm{T}} \tag{11}$$

The parameters a1,..., a5 are calculated using the relations

$$a_1 = (2m_w + m_r)\left(\frac{r}{2}\right)^2, a_2 = \left(2m_w l^2 + 2I_{xw} + I_{zr}\right)\frac{r}{2l}, a_3 = I_{zw}, a_4 = N_1 f_1, a_5 = N_2 f_2, \tag{12}$$

where N1, N2 are the loads applied to wheels 1 and 2, while f1, f2 are the coefficients of rolling friction of both wheels. Robot control inputs are wheels angular acceleration q¨ = [¨a1, a¨2] T and wheels angular velocities q˙ = [˙a1, a˙2] T given by the equations

$$\dot{\alpha}_1 = \frac{v_A}{r} + \dot{\beta}\frac{l}{r}, \quad \dot{\alpha}_2 = \frac{v_A}{r} - \dot{\beta}\frac{l}{r} \tag{13}$$

3.8 Swarm control method

Swarm control method Swarm members are linked together with springs and dampers connected in parallel, Figure 3.2. We assume the robots have mass m and are connected via springs with stiffness k and dampers with damping coefficient c. The length of the springs at rest is equal to d0. Virtual springs and dampers are connected in the centers of mass of each robot, depicted in what follows as point A. Every link between the robots exerts a force Fi j equal to the sum of elastic spring and viscous damper forces such as discussed in [26.27].

We assume that the spring and damper are linear and their forces are proportional, respectively, to the displacement and relative velocity between spring ends.

$$F_{ij} = F_{Sij} + F_{Dij} = k\left(d_{ij} - d_0\right) + c\frac{\mathrm{d}}{\mathrm{d}t}\left(d_{ij} - d_0\right), \tag{14}$$

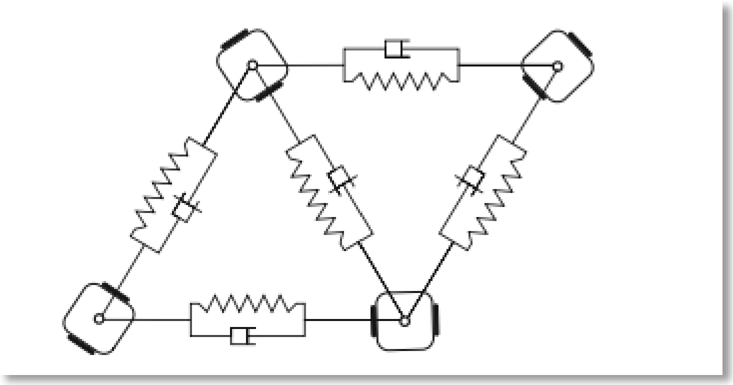

Figure. 3.3 Spring dampers mesh connecting mobile robots

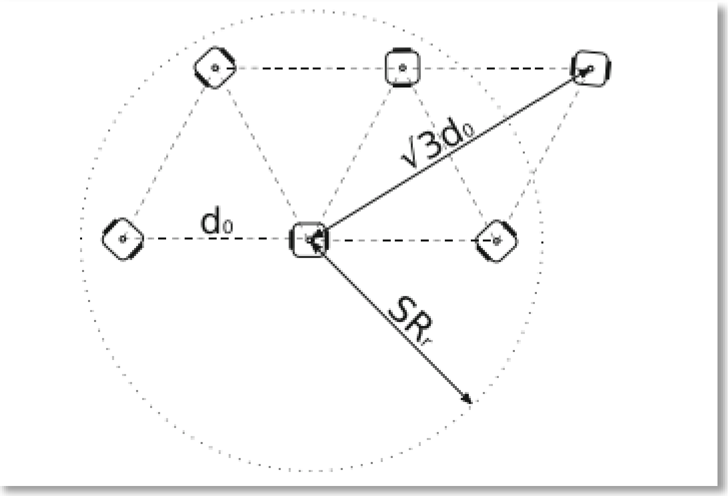

Figure3.4 Sensing range in nearest neighbors link connection

Where k stands for the spring stiffness coefficient, c is the damping coefficient, and di j is the current length of the spring connecting the robots I and j. Each robot has limited sensing range S Rr which determines how many spring damper links are present; the number of links is, in this way, equal to the number of visible neighboring robots. To establish links with only the nearest neighbors, the sensing range has to meet the inequality

$$d_0\sqrt{3} > SR_r > d_0. \tag{15}$$

For example, setting the sensing range at the value 1.2d0 results in creating links between only nearest neighbors forming the swarm in the equilateral triangle formation shown in Fig. 3. Let us note that the sensing range is quite similar to the notion of horizon used in the peridynamics, see [28], or the interaction range in the lattice dynamics The nearest neighbors do not remain the same all the time, in general. The total force exerted on a robot i is equal to the sum of all link forces between robot i and its visible neighbors [29].

$$\overline{F_i} = \sum_{j\in N_r} \overline{F_{ij}} = \sum_{j\in N_r} \overline{F_{Sij}} + \overline{F_{Dij}}. \quad (16)$$

We conclude that we require values of linear velocity Va and angular velocity β˙, to specify robot control inputs, having the net force Fi as known value. To determine the values of Va, β˙ we first have to find the corresponding forces causing the robot to move forward or backward (force Fy i) and rotate around the point A (force Fx i). The origin of the vectors Fy i, Fxi is in the point D. If we know the values of link forces and their orientation, we can find the x and y components of the total force Fi given by

$$\begin{aligned} F_x &= \sum_{j\in N_r} F_{ij}\cos\alpha_{ij}, \\ F_y &= \sum_{j\in N_r} F_{ij}\sin\alpha_{ij}. \end{aligned} \quad (17)$$

The vectors Fy i, Fx i are the components of the vector Fi j in a rotating frame of reference around the point A by an angle $\phi i = \beta i - \pi /2$. The addition of - π 2 brings Fy i (instead of vector Fx i) in the direction of the velocity vector Va. The values Fy i, Fx i are calculated using the following equation

$$\begin{aligned} F'_x &= -F_x\sin\varphi_i + F_y\cos\varphi_i, \\ F'_y &= F_x\cos\varphi_i + F_y\sin\varphi_i. \end{aligned} \quad (18)$$

Using the above equations the values of the linear velocity vA and the angular velocity β˙ are determined by

$$\begin{aligned} v_A &= \int \frac{F'_y}{2m_w + m_r}\,dt, \\ \dot{\beta} &= -\lambda F'_x, \end{aligned} \quad (19)$$

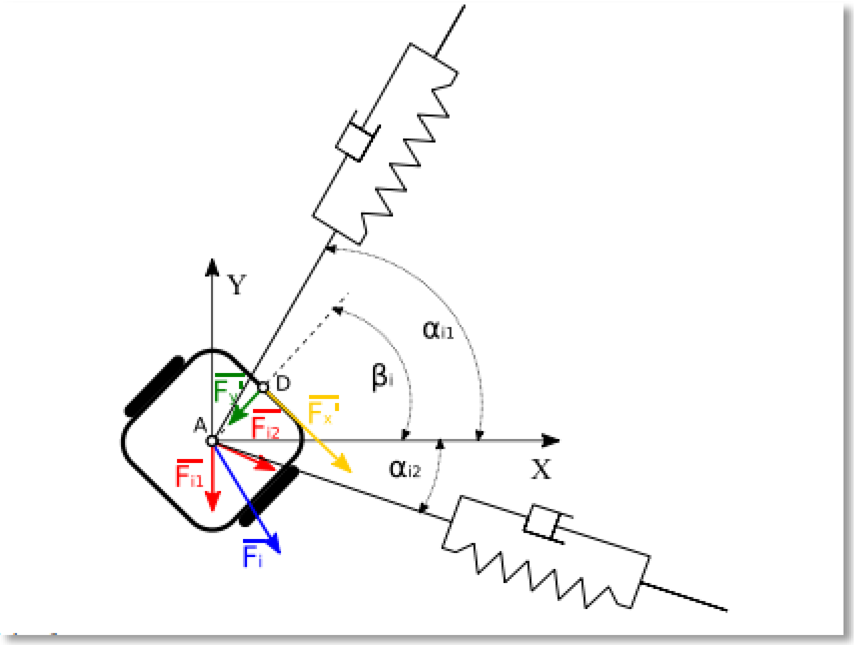

Figure. 3.5 Robot i driving forces

m_w (kg)	m_r (kg)	I_{zr} (kgm^2)	I_{xw} (kgm^2)	I_{zw} (kgm^2)	N (N)	l (m)	r (m)	f (m)	k (N/m)	c (Ns/m)	λ (−)	d_0 (m)
0.05	0.4	0.08	0.01	0.03	22	0.058	0.032	0.004	7	1	4	1
0.05	0.4	0.08	0.01	0.03	22	0.058	0.032	0.004	15	1.5	4	1

Table 3.1 Simulation parameters

Where λ is the gain constant. The gain constant λ allows one to control the robots' rotation speed around the point A, aligning the robot frame with the direction of the total force Fi acting on the robot. Moreover, the damping coefficient c and stiffness coefficient k can be set by the designer so that the values of Va and $\dot{\beta}$ are adequate for the task at hand.

3.9 Conclusion

In this chapter reported the analysis of a swarm robot coordination to control the coordination among individuals in the swarm. The relationship between a direction of motion in the coordination and exchange of information and reported more information of coordination, and i reported the approaches of swarm like approach Artificial Physics and choose spring damper mesh model while explaining the details of the model and explaining the mathematical relationships to it.

Chapter 4 Simulation Result

4.1 Introduction

Simulation by ARGoS is a multi-robot simulator called ARGoS (Autonomous Robots Go Swarming). Extensibility is ensured by ARGoS' highly modular architecture robots, sensors, actuators, visualizations and physics engines are implemented as user-defined modules. Multiple implementations of each type of module are possible

4.2 scenario

The first step consists in the creation of the ARINA with an Argos file and we create the 8 robots 4 fot-bot and 4 eye-bot. These last ones are posed randomly with the random function and each robot contains its contolleur file and After the ecxecusion all the eye-bot robot goes up and turns we go on a point for example point(0.0.0) and all the fot-boot robot make communication between them to make the assembly.

4.3 Simulator (ARGOS):

We present ARGoS , a novel open source multi-robot simulator. The main design focus of ARGoS is the real-time simulation of large heterogeneous swarms of robots. Existing robot simulators obtain scalability by imposing limitations on their extensibility and on the accuracy of the robot models. By contrast, in ARGoS we pursue a deeply modular approach that allows the user both to easily add custom features and to allocate computational resources where needed by the experiment. A unique feature of ARGoS is the possibility to use multiple physics engines of different types and to assign them to different parts of the environment. Robots can migrate from one engine to another transparently. This feature enables entirely novel classes of optimizations to improve scalability and paves the way for a new approach to parallelism in robotics simulation. Results show that ARGoS can simulate about 10,000 simple wheeled robots 40% faster than real-time. To maximize simulation speed, the ARGoS architecture is designed to exploit modern CPU architectures.[22]

4.4 Using ARGoS: Basics

To use ARGoS, you must run the command argos3. This command expects you to provide two kinds of input:

-An XML configuration file, typically with extension .argos

- User code compiled into one or more libraries

The configuration file contains all the information to set up the arena, the robots, the physics engines, the controllers, etc. User code includes robot controllers and, optionally, hook functions to be executed in various parts of ARGoS to interact with the running experiment.[22]

ARGS File:

framework [required]Sets up internal parameters of ARGoS itself, such as the number of threads to use, the base random seed, the length of the experiment and the length of the integration step. [22]

controllers [required] Contains the list of user-defined controllers and their configuration. [22]

arena [required] Contains the list of entities to add to the arena at the beginning of the experiment and their initial positions. [22]

- Actuator
- Sensor

physics_engines [required] Configures the physics engines to use: which physics engines, how they are connected to each other, etc.

media [required]Configures the media to use.

visualization [optional] Contains the visualization set up.

loop_functions [optional] Contains the configuration of the loop functions. [22]

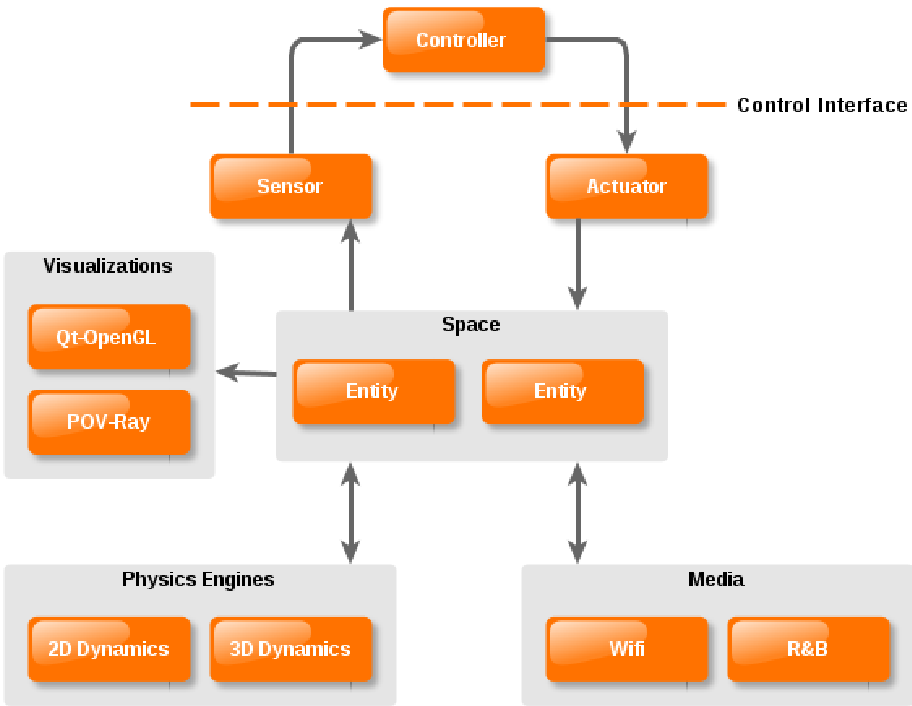

Figure 4.1 The general architecture of ARGoS [22]

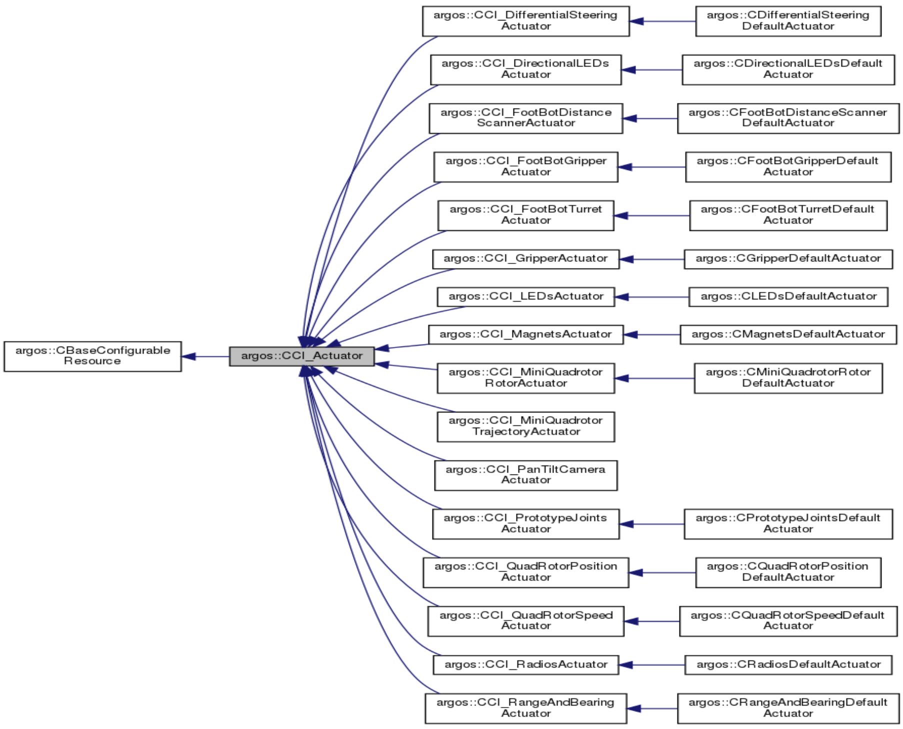

Figure 4.2 Actuator (Eye-bot Foot-bot robot) [22]

Inheritance diagram for argos::CCI_Sensor:

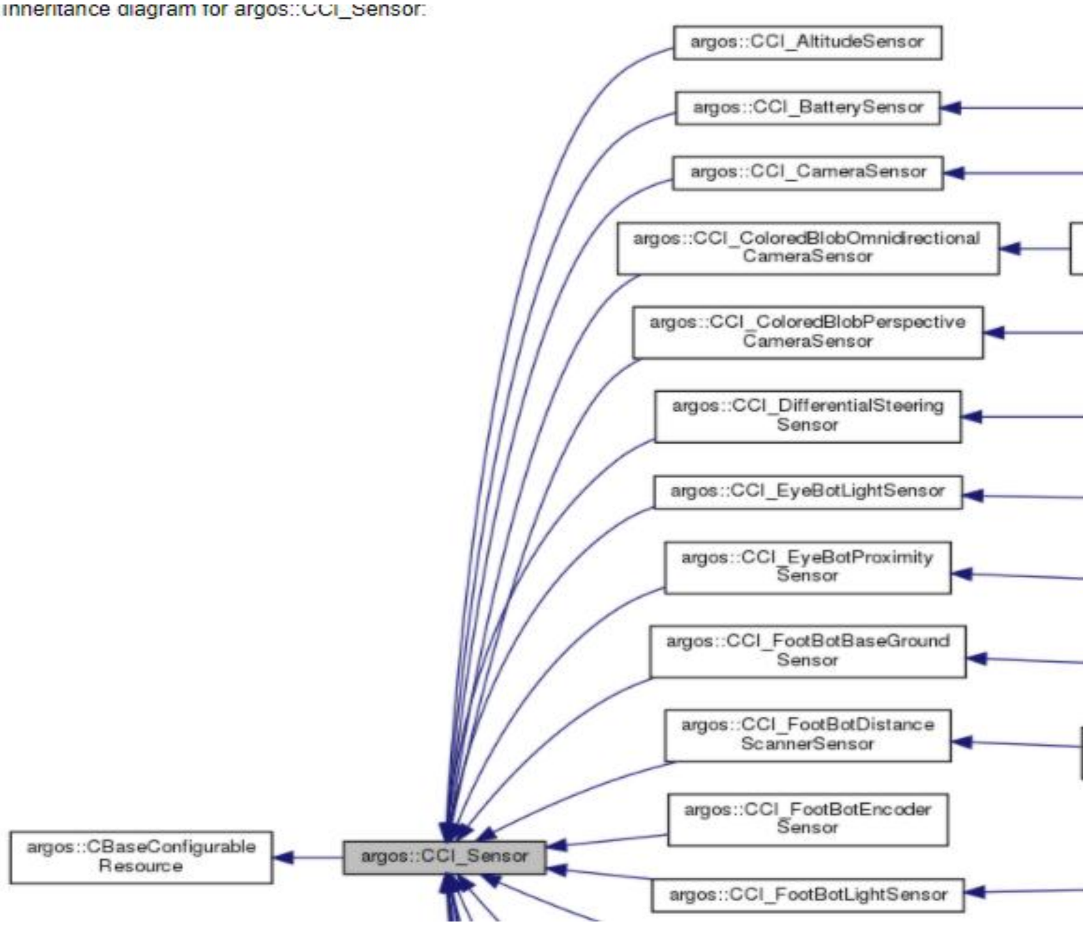

Figure 4.3 Sensor (Eye-bot Foot-bot robot) [22]

4.5 Simulation results

Initial state (Figure 4.4)

In order to evaluate the performance of the proposed solution, we have carried out several simulation experiments using up to 8 bots (4 eye-bots, 4 foot-boots). We simulate the experiments using the Argos simulator.

At the beginning Argos simulator will start to create the arena and loading the 8 bots by the xml files Argos will start the compilation process and the execution process at the same time all of the robots will be at the initial start position.

-the foot-bot bots will be placed randomly, whereas the eye-bot bots will be placed in the following positions R1 (0.5.0.0) R2 (-0.5.0.0) R3 (0.0.5.0) R4 (0.-0.5.0) which these values coordinate to the center of a circle

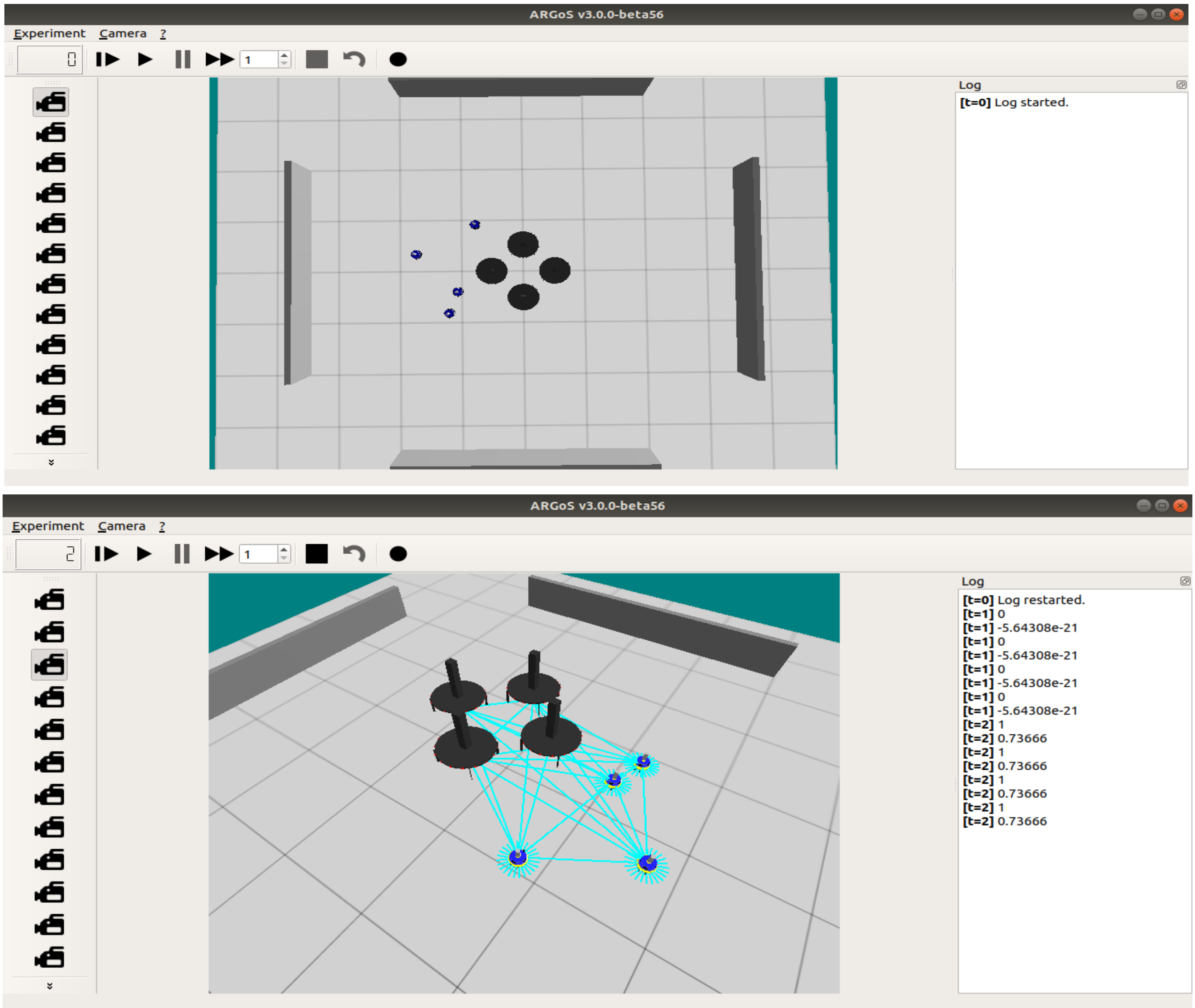

Figure 4.4 initial simulation state

Secondary state (Figure 4.5)

* The foot-bots will be created at a random positions and they will start moving with help of actuates the bots will use the sensors like proximity sensor ,position sensor and other sensors to coordinate each type of bots that transport inside the arena and modulate the shapes (circular shapes , linear shapes) . By using spring damper mesh model to place the bots in the same line leaving equal spaces between them.

* after their creation The eye-bot start taking off at the height of 3,0 and they move so that they get placed at the circumference of the circular shape positioning in the diameter R leaving equal spaces between them.

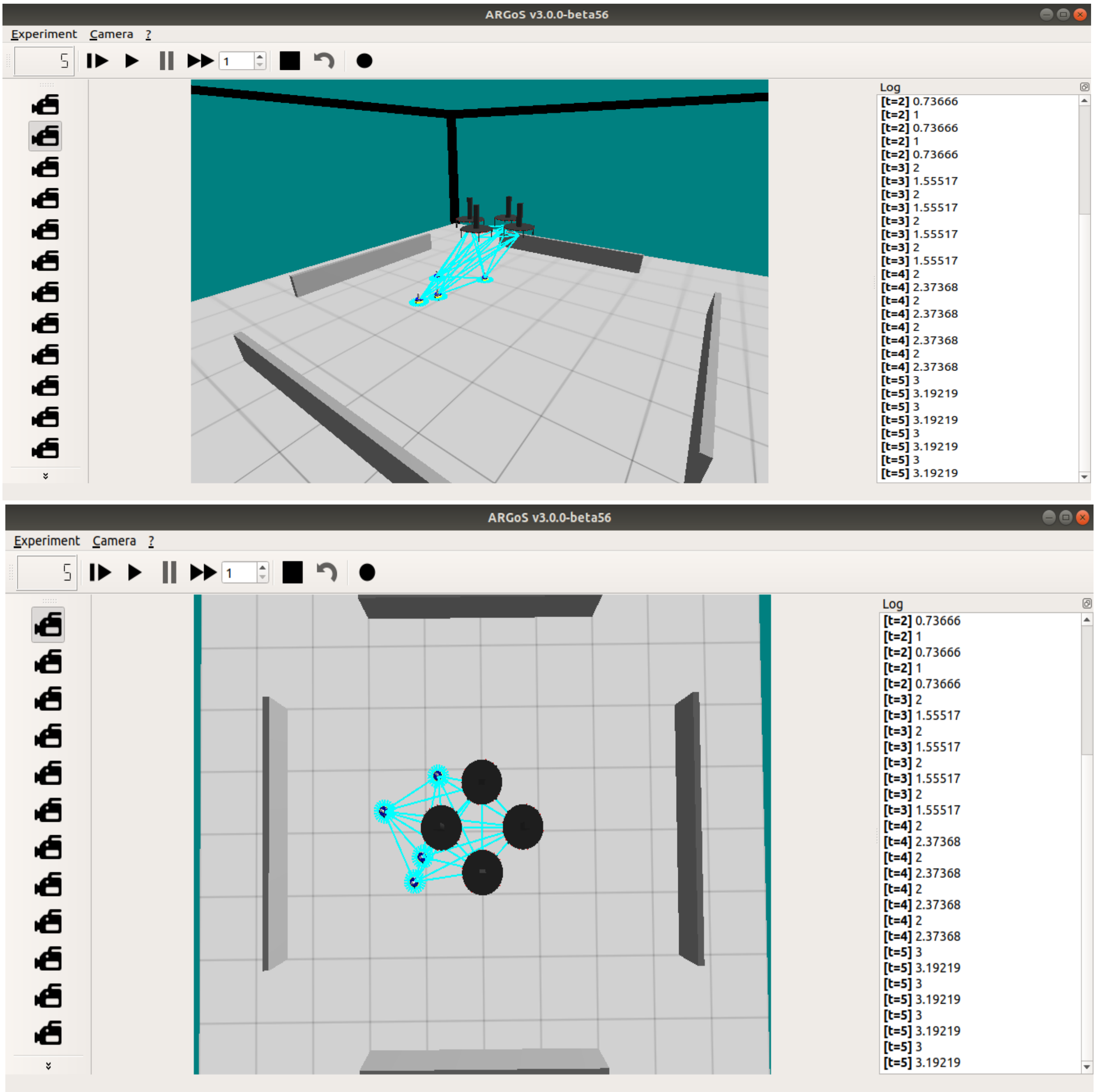

Figure 4.5 secondary simulation step

Final state (Figure 4.6)

* Finally, the foot-bot will take its final linear shape by using the proximity sensor and the position sensor, so they avoid each other and organize themselves in a straight line.

* Eye-bot taken place around the circumference respecting a mathematical circle equation and leaving equal spaces between them and keep their speed constant.

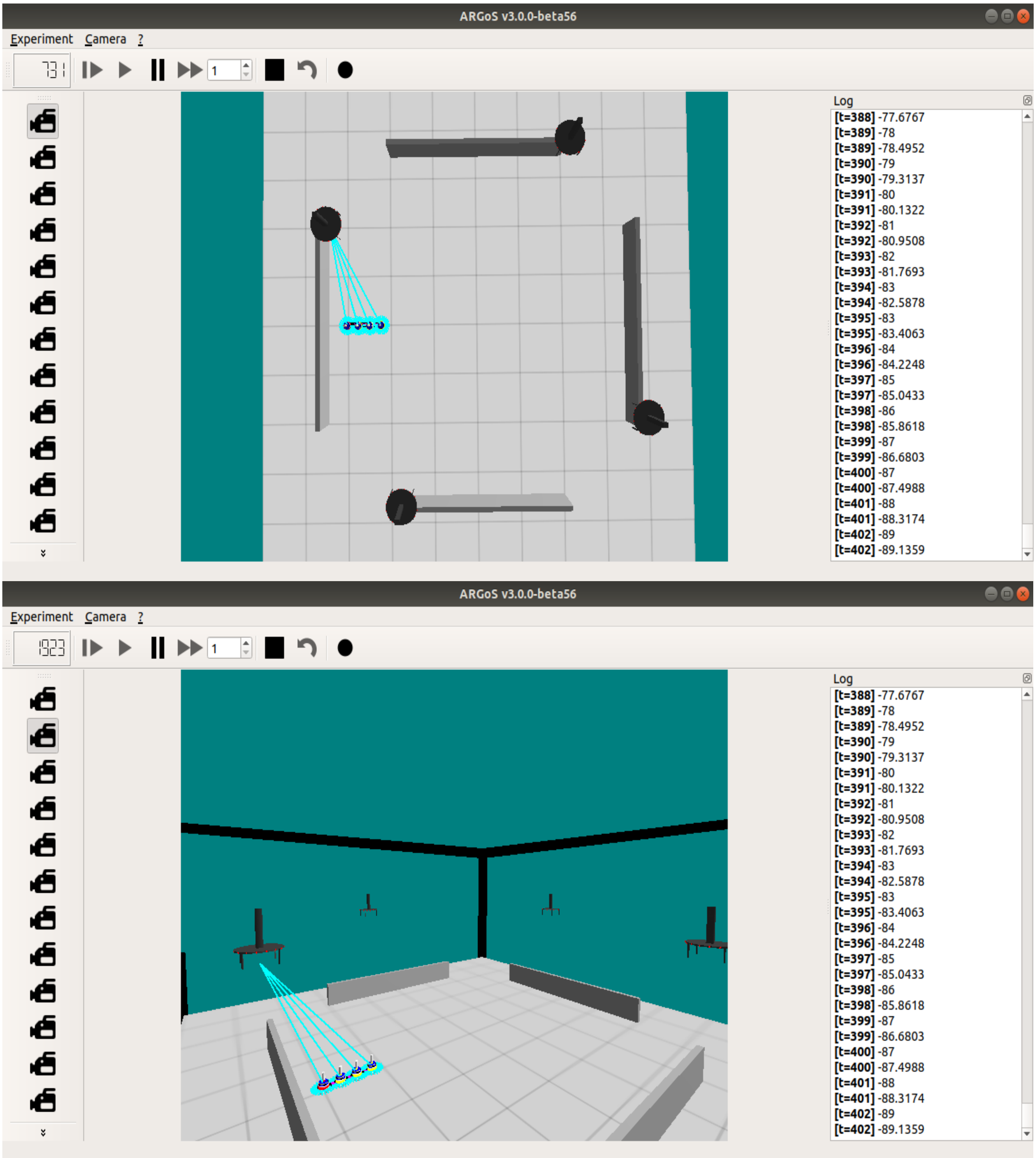

Figure 4.6 Final simulation step

4.6 Conclusion

In this chapter, we have presented the tools necessary for the validation of our approach. On began with the ARGOS simulator which is used by most researchers in this field. The main architecture as well as the robotic platform (foot-bot) and the (eye-bot) that will be used also the analysis of a swarm robot coordination to control the coordination among individuals in the swarm, the relationship between a directions of motion in the coordination. These materials and methods are essential for the synthesis of controllers in order to study collective behavior within a cloud of robots. In addition, we have discussed all materials required in terms of detection and activation systems onboard robot that will be particularly used to implement the controllers proposed.

General Conclusion

Swarm robotics is a very interesting and ambitious research in a sub-domain, which mainly seeks to integrate the theory behind swarms in nature to multi-robotic systems.

Swarm robotics models are fundamentally inspired by the following behaviors collectives of social animals such as birds, ants and bees. They are generally applied in studies dealing with collective task problems. As For example, training schemes and specifically self-organized aggregation schemes are part of the complex issues that interest the literature on swarm robotics.

This issue was addressed in this thesis by designing synthesized controllers for significant collective behaviors in swarm robotics. In particular, the controllers have been successfully applied to the study of model training tasks robotic self -organization in swarms.

The controllers are mainly based on an artificial spring damper mesh model that is inspired by

Biomechanical properties - such as spring and damer mesh. We have shown that the basic controller is able to realize simple geometric formations within two types of robotic platforms in swarm: foot-bots and eye-bots with success.

Swarm robots will change the way humans engage with the world, this change created by a complex technology that made human life easier and faster, we can find this kind of technology (Swarm robotics) in many fields and domains such as the development in medicine, agricultural, manufacturing, sport, and arms race.

References

[1] Redfield, Signe (2019). "A definition for robotics as an academic discipline". Nature Machine Intelligence. 1 (June 2019): 263–264. doi:10.1038/s42256-019-0064-x. Retrieved November 20, 2019.

[2] Zhiguo Shi et al. “A survey of swarm robotics system”. In: Advances in Swarm Intelligence (2012), pp. 564–572.

[3] Jon Timmis, Paul Andrews, and Emma Hart. “On artificial immune systems and swarm intelligence”. In: Swarm Intelligence 4.4 (2010), pp. 247–273.

[4] Michael G Hinchey, Roy Sterritt, and Chris Rouff. “Swarms and swarm intelligence”. In: Computer 40.4 (2007).

[5] Amelia Ritahani Ismail. “Immune-inspired self-healing swarm robotic systems”. PhD thesis. University of York, 2011.

[6] Hiroaki Kitano. “Biological robustness”. In: Nature Reviews Genetics 5.11 (2004), pp. 826–837.

[7] Zhiguo Shi et al. “A survey of swarm robotics system”. In: Advances in Swarm Intelligence (2012), pp. 564–572.

[8] Hui Keng Lau. “Error Detection in Swarm Robotics: A Focus on Adaptivity to Dynamic Environments”. PhD thesis.02/2012 University of York,

[9] Marco Dorigo et al. “The swarm-bots project”. In: International Workshop on Swarm Robotics. Springer. 2004, pp. 31–44.

[10] Marco Dorigo et al. “Evolving self-organizing behaviors for a swarm-bot”. In Autonomous Robots 17.2 (2004), pp. 223–245.

[11] Heiko Hamann. Space-time continuous models of swarm robotic systems: Supporting global-to-local programming. Vol. 9. Springer Science & Business Media, 2010.

[12] Aleksandar Jevti´c and Diego Andina. “Swarm Intelligence and Its Application in Swarm Robotics”. In: Proceedings of the 6th WSEAS International Conference on Computational Intelligence, Man-machine Systems and Cybernetics.

CIMMACS'07. Tenerife, Canary Islands, Spain: World Scientific, Engineering Academy, and Society (WSEAS), 2007, pp. 41–46. URL: http://dl.acm.org/citation.cfm?id=1984502.1984510.

[13] Amanda JC Sharkey. "Swarm robotics and minimalism". In: Connection Science 19.3 (2007), pp. 245–260.

[14] Erol ¸Sahin. "Swarm robotics: From sources of inspiration to domains of application". In: International workshop on swarm robotics. Springer. 2004, pp. 10–20.

[15] Hui Keng Lau. "Error Detection in Swarm Robotics: A Focus on Adaptivity toDynamic Environments". PhD thesis. University of York, 2012.

[16] Alvaro Gutiérrez et al. "An open localization and local communication embodied sensor". In: Sensors 8.11 (2008), pp. 7545–7563.

[17] James F. Roberts Enabling Collective Operation of Indoor Flying Robots Lausanne April 2011.

[18] Ferber. J. Les systèmes multi-agents. Vers une intelligence collective. InterEditions, Paris, 1995.

[19] Pierre KANCIR DOCTORAL THESIS « Méthodologie de conception de système multi-robots : de la simulation à la démonstration » 2018

[20] https://robohub.org/ ver(08-09-2020)

[21] IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 1, PP 46-47 www.iosrjournals.org .Ver. II (22sep-2020),

[22] www.argos-sim.info /ver(8-09-2020)

[23]Jakub Wiech ·Victor A. Eremeyev ·Ivan Giorgio Virtual spring damper method for nonholonomic roboticswarm self-organization and leader following https://doi.org/10.1007/s00161-018-0664-4/ .ver (222-09-2020)

[24] Giergiel, J., Zylski, W.: Description of motion of a mobile robot by Maggie's equations. J. Theor. Appl. Mech. ˙ 43(3), pp. 511–521.

[25] Gutowski R.: Mechanika Analityczna, 1971, PWN, Warszawa.

[26] Urcola, P., Riazuelo, L., Lazaro, M., Montano, L.: Cooperative navigation using environment compliant robot formations. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2008, pp. 2789–2794, IEEE

[27] Balkacem, K., Foudil, C.: A virtual viscoelastic based aggregation model for self-organization of swarm robots system. TAROS 2016: Towards Autonomous Robotic Systems, pp. 202–213, Springer

[28] Madenci, E., Oterkus, E.: Peridynamic Theory and Its Applications. Springer, New York

[29] Born, M., Huang, K.: Dynamical Theory of Crystal Lattices. Clarendon Press, Oxford (1988).