Human-Swarm Robot Interaction with Different Awareness Constraints

Gabriel Kapellmann Zafra

1st Supervisor: Dr. Roderich Groß

2nd Supervisor: Dr. Andreas Kolling

Department of Automatic Control and Systems Engineering
The University of Sheffield

A Thesis Submitted for the Degree of
Doctor of Philosophy

February 2018

Random segments of code, that have grouped together to form unexpected protocols. Unanticipated, these free radicals engender questions of free will, creativity, and even the nature of what we might call the soul.

Dr. Alfred Lanning
"I Robot" (2004)

Abstract

Swarm robots are not yet ready to work in real world environments in spaces shared with humans. The real world is unpredictable, complex and dynamic, and swarm systems still are unable to adapt to unexpected situations. However, if humans were able to share their experience and knowledge with these systems, swarm robots could be one step closer to work outside the research labs. To achieve this, research must be done challenging human interaction with more realistic real world environment constraints. This thesis presents a series of studies that explore how human operators with limited situational and/or task awareness interact with swarms of robots. It seeks to inform the development of interaction methodologies and interfaces so that they are better adapted to real world environments. The first study explores how an operator with bird's-eye perspective can guide a swarm of robots when transporting a large object through an environment with obstacles. As an attempt to better emulate some restricted real world environments, in the second study, the operator is restricted from access to the bird's-eye perspective. This restriction limits the operator's situational awareness while they are collaborating with the swarm. Finally, limited task awareness was included as a additional restriction. In this third study, the operator not only has to deal with limited situational awareness but also with limited information regarding the objective. Results show that awareness limitations can have significant negative effects over the operator's performance, yet these effects can be overcome with proper training methods. Through all studies a series of experiments are conducted where operators interact with swarms of either real or simulated robots. In both cases, the development of the interaction interfaces suggest that careful design can support the operator in the process of overcoming awareness problems.

Acknowledgements

First I would like to thank my supervisor Dr. Roderich Groß for his support throughout my PhD studies. He introduced me to the world of swarm robotics and helped me in the transition to the research lifestyle. I am, as well, thankful for the excellent example that he was for my experience in scientific literature writing and proof-reading. Also, I would like to thank Andreas Kolling who acted as my second supervisor, as he provided valuable knowledge particularly at the start of my PhD studies and kept providing support through the four years.

Special thank you to everyone that I met from the office and lab. To Stefan Trenkwalder who was a good colleague and a great friend. I appreciate his support and advice with PhD related matters and everything else non related to it. To Nicole Salomons that without her collaboration I would have taken longer in my project. I also had the privilege of meeting and receiving support from Melvin Gauci, Fernando Perez Diaz, Yuri Kasubowski Lopes, Christopher Parrott, Matthew Doyle, Joao Vasco, Anil Ozdemir and Natalie Wood.

I would also like to thank the many friends and groups that became a part of my life. I am grateful with them that supported me in many different ways. I will single out Lucy Kemp who help me make my first years in Sheffield enjoyable and introduced me to the local culture. Also Anna Poptsi who during the final years offered me her unconditional support and company.

A special thanks to my family. To my brother and grandparents who have backed me up when they could. Special thank you to my father Gabriel Kapellmann Sanchez and my mother Patricia Zafra Ramirez who have made so many sacrifices and have always been there to help me and support me.

On a different note, I would thank the Science and Technology Council of Mexico (Consejo Nacional de Ciencia y Tecnología, CONACYT) for the scholarship support.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Every day, an increasing number of robotic systems get more involved with human activities. The aim of these systems is to assist humans in a broadening variety of roles on a daily basis. Furthermore, these systems are meant to exist, think and adapt on their own while interacting with humans within the same environments. There is a clear tendency to create intelligent autonomous robotic systems that should be able to team up with humans. For this reason, either as a roboticist (or any other related field) focusing on the engineering side or as a psychologist focusing on the human cognitive side, human-robot interaction [1, 2] has become an extensive and diverse area of research.

As research gets closer into creating more complex and better optimized robotic systems, the interaction between operators and robots should stay simple and casual, avoiding the need for the operators to learn new forms of interaction/communication. In simple words, the goal is to give any human the possibility to work with a fellow robot "just as if" it was another human entity (Fig. 1.1). There are, however, considerable challenges in the human-robot interaction (HRI) field regarding the robots, the operators and the interfaces (perception, management, planning, task execution, navigation, learning, etc.).

Usually, HRI is focused on an operator interacting with a single robot. However, if we could increase the number of robots the operator works with, the resulting systems could offer advantages like covering of larger working areas or faster task execution. With properly designed multi-robot systems (MRS), the operator controlling it could perform tasks that are typically executed by multiple humans. The time of execution for some tasks could even be reduced. The possible applications where multi-robot systems could become more

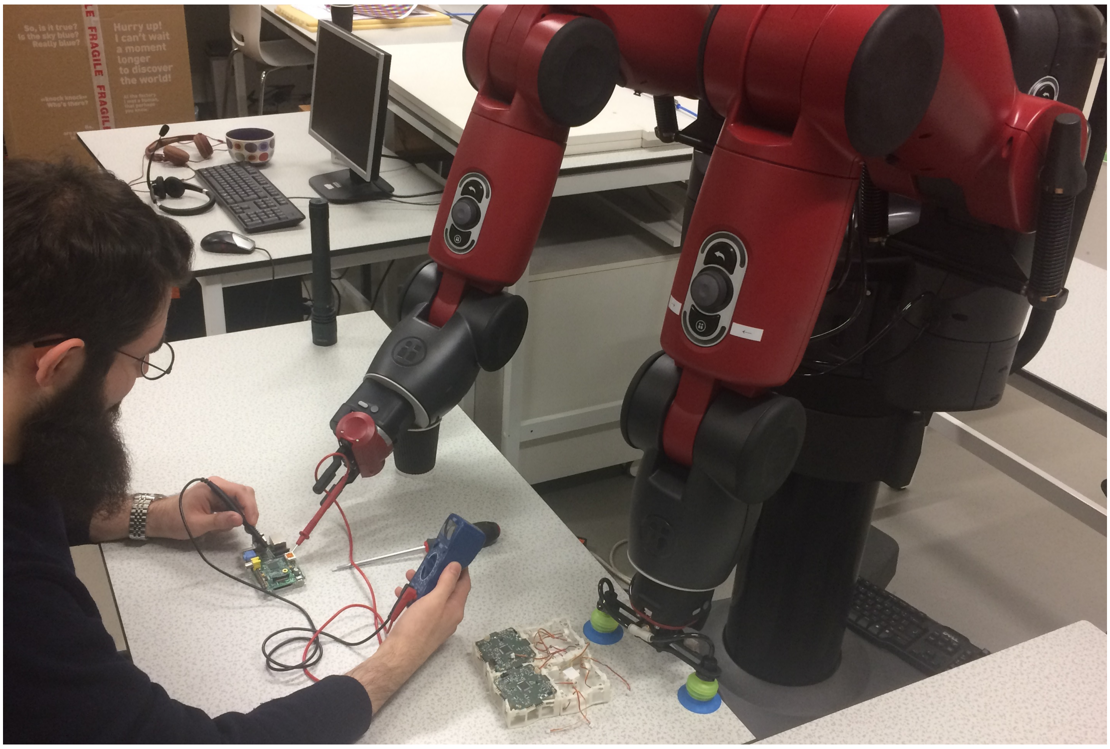


Fig. 1.1 Representation of a human engineer working along with a robot in repairing a circuit board.

functional in human environments fuels the need to explore new effective interaction methods. Examples of these applications range from social activities (cleaning or waitressing), security (patrol or threat response), industrial (storage organization or product transport), exploration (underwater, underground or space), construction (building or deployment), search & rescue (hazardous or dangerous environments) between others. As a consequence, more people will be able to interact with MRS.

Swarm robotic systems are a subset of MRS with characteristics that are widely observed in natural swarms. They take inspiration from the field of collective intelligence where complex behaviour emerges from the interactions between multiple individuals with each other and with their environment. This emerging intelligence, also known as swarm intelligence (SI), has been defined by Dorigo [3] & Bonabeau [4] as:

- "Swarm intelligence is an artificial intelligence technique based around the study of collective behaviour in decentralized, self-organized systems."
- "Swarm intelligence systems are typically made up of a population of simple agents interacting locally with one another and with their environment."

- “Although there is normally no centralized control structure dictating how individual agents should behave, local interactions between such agents often lead to the emergence of global behaviour.”

Swarm robots generally use local sensing as well as local communication, limiting their access to global state information. Because of their distributed social architecture, they are governed by a set of relatively simple individual rules. Yet, complex behaviours may emerge from the cooperation among the robots and of the interactions of the robots with their environment. The first artificial example of swarm behaviour was in a computer simulation program. This program was named “Boids” and was created by Craig Reynolds in 1986 [5]. The individuals of the virtual swarm followed a set of simple rules:

- Move in the same direction as your neighbours
- Remain close to your neighbours
- Avoid collisions with your neighbours

As first seen in “Boids”, and later through many other examples, swarm robots have the capacity to self-organize themselves. This ability could lead to novel solutions in problems like path finding in dynamic environments, exploration, search & rescue and support, to mention a few. In addition, these systems have certain advantages like scalability, fault tolerance and broad task coverage.

Swarm-intelligent systems have the potential to become a useful tool for solving complex problems [6]. However, their solution is not necessarily meant to be the best solution. Human intervention could be beneficial, for example, to adapt the swarm system behaviour and/or react to critical environment changes as well as to make “last minute” decisions in which human experience could be important [7].

However, introducing a human as an operator of a multi-robot system does not completely ensure that the team (humans and robots) will experience an improvement of performance. A poorly designed interface could compromise the performance of the multi-robotic entity rather than promote teamwork. In addition, the application of multi-robot control could imply the division of the operators’ attention between the robots. This would limit the interaction by the number of robots that the human operator could reasonably control. There are several details that make the interaction between multiple robots and humans a complex process [8]. For this reason, it is important to start exploring human-swarm robot interaction with realistic characteristics, imitating real world environments.

1.1 Motivation

The motivation for this thesis comes from the need of exploring the human-robot interaction domain applied to swarm robotics, particularly when the human operator has some awareness constraints that are typically encountered in real world applications. To take swarm systems out of controlled research environments and test them in real world. This to better understand such constraints and their effect on the interaction process. The presented studies will discuss two constraints and the possible implications of these over the interaction process.

The literature proposes several advantages of having multi-robot systems involved in the human world. One motivating reason involves safety and surveillance of and for humans in hazardous environments and/or hostile situations. Security missions as well as search & rescue missions are a constant priority [9–12] for multi-robot systems potential applications. It is natural that for these type of activities a robotic system that is scalable, fault tolerant and capable of doing parallel work in multiple different locations is desirable [3]. Swarm systems promise to have all these qualities and are aimed to be independent and robust. However, they are not ready to interact in the real world due to its unpredictability. Human experience and expertise could aide these systems to overcome such problems. For this to happen, the interaction techniques between the humans and robots need to be prepared to deal with awareness limitations that the human operator might suffer due to the nature of the real world environment.

A common problem in the real world is that constant and effective contact and communication with all the robots of a swarm would not be possible. However, if the operator could still effectively influence the swarm robot despite the restricted awareness and lack of information, human interaction could prove beneficial to increase the chances of mission success [13]. Human intervention in swarm robot dynamics could offer information about dangers and/or interesting areas within the working environment. Human expertise and experience may prove desirable to be involved in the decision loop. The human could compensate with complex decision making and problem solving while the swarm robot remains active and working in a defined task. For this reason, the strengths, limitations and requirements of high level supervisory control need to be explored in restricted environments [14].

1.2 Problem Definition

If swarm robots have emergent behaviours that have the potential to find solutions to problems on their own, why would human influence be needed? A limitation to make swarm systems behave as desired in real environments with humans is the assumption that those environments are static and obstacle free. However, the real world is dynamic and unstructured, especially in those locations where hazardous events have happened or might happen [15]. In addition, robotic systems will always have hardware limitations (limited range sensors, bandwidth limitations, communication difficulties, etc.) that can make a successful task completion even more complex.

The problem that this thesis attempts to tackle is how human influence can provide support and guidance to a swarm robot when limited by different awareness constraints. An operator with restricted awareness (such as restricted situational awareness or task awareness) should still be able to collaborate with a swarm robot. The aim is to study how these restrictions affect the human in helping the swarm to overcome unexpected event-related problems and/or complications. Human interaction should be focused on influencing the emergent behaviour of the robots rather than controlling them. The challenge is to allow this influence to be effective without increasing the complexity of the interfaces or the swarm [16].

The restriction of situational awareness can occur by lack of access to the bird's-eye view and can have direct effect on the systems' and operators' performance. Yet, even while the operator is restricted of explicit access to this global state information, the influence over the swarm should be delivered without compromising the performance of the system. In addition, lack of task awareness can occur by lack of explicit information and/or understanding regarding the task or main objective. The limitation of task awareness over the operator could have negative effects over the operator, like restricting the ability to identify additional information regarding the working environment. Therefore, the need to understand the impact of these restrictions and explore solutions to overcome them becomes evident, as real environments often present this kind of limitations.

Furthermore, to obtain the full potential of swarm robotic systems in hazardous situations (exploration, rescue & safety missions), how both awareness levels (situational & task awareness) can affect the human perception needs to be further explored with new methodologies and more realistic approaches.

1.3 Contributions

- **Control of a swarm robot behaviour with full awareness**: We propose an interaction strategy that enables an operator with full awareness to influence a swarm of robots. The operator controls a designated leader robot and defines a set of dynamic way-points to influence the overall behaviour of a the swarm. The interaction is made directly with the leader robot via one of two possible portable devices that simplifies the communication process and minimizes the distractions for the operator. We analyse and compare the performance of the two portable devices, one with a pure GUI-tactile interface (a mobile phone) and the other with a combination of voice and tactile commands (the Google Glass). Through a series of trials with real robots, the strategy is proved to be reliable and convenient for manipulation of the swarm robot behaviour.

- **Interaction with limited situational awareness**: Evidence was provided suggesting that human-swarm robot interaction with limited situational awareness can be effective. This study explores how an operator with limited situational awareness can collaborate with a swarm of robots while lacking the bird's-eye view. An analysis shows that the designed interface allowed the operator to influence multiple robots through a random robot leader selection method. Training experience proved to be a crucial factor in the operator's performance as analysis of behavioural differences revealed that trained operators learned to gain superior situational awareness. When given only local information, however, untrained operators did not perform significantly better than random interactions.

- **Operator resilience to restricted task awareness**: A systematic investigation is presented of how a human operator attains different levels of task awareness under different information constraints. We provide a statistical analysis of the results that show how the human operator task awareness level is directly linked to the task information provided. It suggests that explicit mention of the task objectives is needed so that the human influence provides the swarm robot with significant help that will increase its performance.

- **Design and development of a managment algorithm**: An algorithm named "Gossip Algorithm" has been designed and applied to the interaction process. It addresses the problem of the operator to organize a robotic swarm via a randomly selected robot that temporarily acts like a leader. Different to the algorithm reported in [17], the proposed algorithm also allows the operator to dynamically organize and manipulate the swarm. For example, it allows the operator to request a count of the robots, and assign them into subgroups and to different tasks. This algorithm was tested in the simulator environment and further applied and validated on real robots.

- **Enki simulator upgrade**: In this thesis, the simulation platform used for the second and third studies was based on the Enki simulator [18]. We developed new add-ons to the simulation platform that made the simulator more useful for experimental research. These additions included: A user interface for simulated and real robots, the logging of all activities from robots and users, and an add-on able to interpret the logged data and replay the robots movements in the environment. The revised simulator has been made open to the public under the GPU license on GitHub [19].

1.4 Case Studies Preview

This thesis explores the implications that certain constraints in the awareness of the operator have on the interaction between a human and a swarm robot through three case studies. This section provides a brief introduction to each case study and the addressed challenges.

1.4.1 Human-Robot Swarm Interaction with Full Awareness

This case study explores the use of a portable device [20] to interact with a swarm robot in a full awareness state. For this study a collective transport behaviour [21] was selected as the main task. A leader architecture where a human operator influences the robots [22] to push an object to a specific point is implemented. The human operator uses a portable device (either the Google Glass or a mobile phone) to interact with the swarm. The challenge is to allow the robots to negotiate obstacles during cooperative transport without needing to increase the swarm robot complexity [16].

1.4.2 Human-Robot Swarm Interaction with Limited Situational Awareness

This case study limits the full awareness level from the previous study by restricting the operator to access to the birds-eye view. An aggregation behaviour [23] was selected as the main task in an environment with physical barriers. The study is focused on how an operator with limited situational awareness [24, 25] can collaborate with a swarm robot by controlling random units from the swarm as leaders [26]. The interface controls were sufficient for the operator to aid the autonomous robots overcome the lack of situational awareness. An analysis of behavioural differences revealed that operators who received training learned to gain superior situational awareness.

1.4.3 Human-Robot Swarm Interaction with Limited Situational and Task Awareness

This case study adds some further restrictions to the last study. An object clustering behaviour [27] was presented to the operators as the main task. However, the experimenter's main task was to evaluate if the operators were able to achieve TA without being affected by the limitation of SA [28, 29], and with different levels of task information [30]. The swarm robot is located in an environment with a hidden room but is not prepared to execute any kind of exploration behaviour, placing the focus of the study on determining the amount of extra information (from the mission objective and the robots [31]) and the explicitness needed so that the operator attempts to explore the environment.

1.5 Publications

This thesis represents the author's work and has led to two publications in academic conferences:

1. Kapellmann-Zafra, G., Chen, J., & Groß, R. (2016). Using Google Glass in Human–Robot Swarm Interaction. In 17th Conference Towards Autonomous Robotic Systems, TAROS 2016 (pp. 196-201). Springer.

2. Kapellmann-Zafra, G., Salomons, N., Kolling, A., & Groß, R. (2016). Human-Robot Swarm Interaction with Limited Situational Awareness. In 10th International Conference on Swarm Intelligence, ANTS 2016 (pp. 125-136). Springer.

In addition, the author's work also led to one contributed publication:

1. Salomons, N., Kapellmann-Zafra, G., & Groß, R. (2016). Human management of a robotic swarm. In 17th Conference Towards Autonomous Robotic Systems, TAROS 2016 (pp. 282-287). Springer.

1.6 Outline

This thesis is organized as follows:

Chapter 2 provides background information about human-robot interaction, multi-robot human interaction and swarm robot-human interaction, followed by a review of interaction strategies and user interfaces regarding swarm robots systems. It also presents a review of two subsets of the awareness concept: situational awareness and task awareness.

Chapter 3 presents an interaction strategy between a human operator and a robot swarm with full awareness. The interaction was tested using two portable devices (Google Glass & mobile phone). This study has been previously presented in [32].

Chapter 4 introduces a situational awareness constraint to the interaction process. In this study, an operator attempts to help a robot swarm solve a task without access to global state information. This study has been previously presented in [33].

Chapter 5 focuses on a similar human interaction scheme as with the last chapter but introduces a task awareness constraint. In this study, the operator is asked to support the swarm robot in solving a task. However, the operator receives limited information regarding the mission.

Chapter 6 concludes the thesis and outlines ideas to continue with future research.

Chapter 2

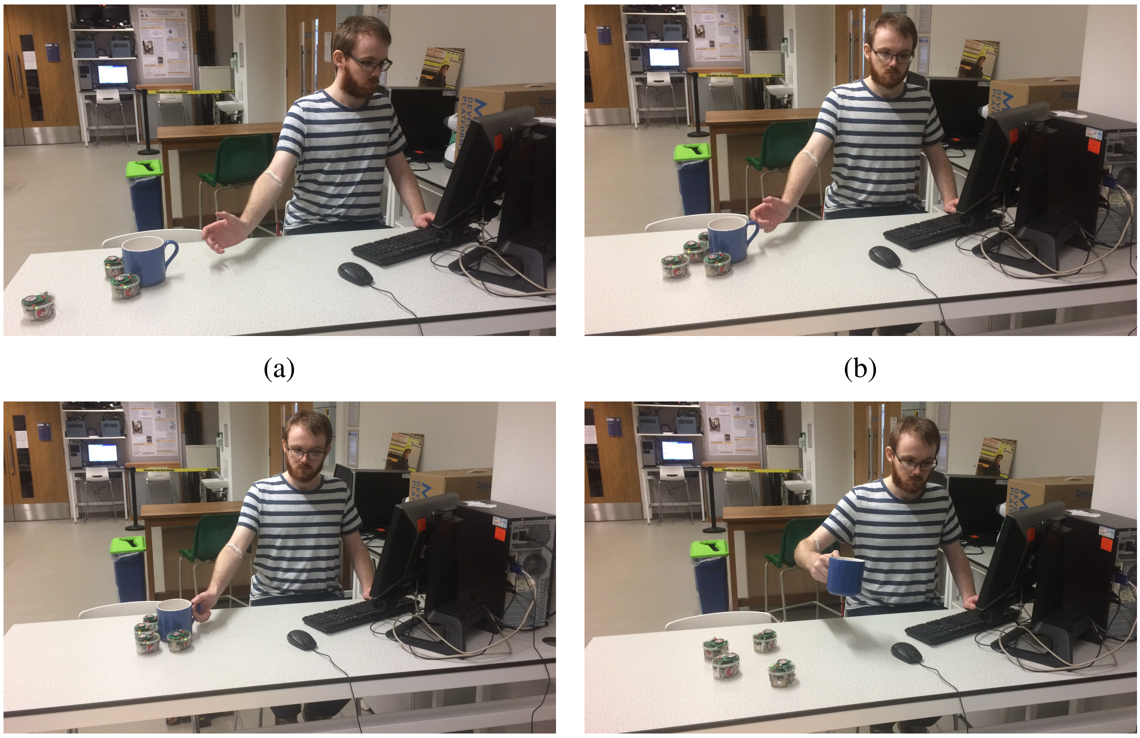
Background and Related Work

This chapter presents a review of the current literature regarding human interaction with robots. Section 2.1 provides some general introduction to swarm robots systems. Sections 2.2 and 2.3 provide some general context about human-robot interaction in single robot and multi-robot systems. Section 2.4 is more focused on human-swarm robot interaction and provides background for interaction strategies (2.4.1) and a concise overview of bandwidth limitations (2.4.2) and the *Neglect Benevolence* concept (2.4.3). Section 2.5 presents current interaction interfaces and the different types of input (2.5.1) and output (2.5.2) techniques that they use. Finally, section 2.6 provides HRI information regarding the concepts of "Situational Awareness" (2.6.1) and "Task Awareness" (2.6.2).

2.1 Swarm Robots

Robotic swarm systems are a subset of multi-robot systems [31, 34]. These systems are normally comprised of large numbers of robots that rely on distributed and decentralised algorithms to collaborate. They aim to be robust to failure due to their emphasis on robustness, scalability (from a few units to thousands or million of units), flexibility and self-organization [35, 36]. Robotic swarms share all of these motivations, but place special interest in robots that act (physically) independently of each other as agents [37].

Swarm robots have a decentralised control approach. In order to coordinate, the robots need to interact with each other and their surroundings. They base their coordination on local interactions and self-organization [38]. It is because of these interactions between



(a) (b) (c) (d)

Fig. 2.1 A representation of a human working with a computer while requesting the robots to transport his mug closer with a simple hand gesture. a) The user moves his hand requesting the robots to transport the mug. b) The robots group next to the mug and start pushing it towards the user. c) The user finally grabs the mug. d) The user lifts up the mug and the robots return to their initial position.

robots and the environment that the desired behaviours emerge from within the system [39] to accomplish complex tasks [40].

A group of multiple robots could carry out tasks that are impossible for a single robot as well as actuate in different places at the same time [34]. An individual robot might be slower to accomplish a task, is unable to multi-task, and has a single point of failure. In contrast, swarm robots could distribute the workload, execute multiple tasks simultaneously and be resistant to failure, as one lost unit would not affect the entire system. Additionally, if a human was introduced in the control loop, the individual robot would also have to allocate some time and resources to attend the human operator requests, while a swarm system would possibly deploy a reduced number of units without affecting the rest of the swarm. These advantages support the need to tackle the challenge of effectively integrating human operators with robotic swarms.

Fig. 2.2 An example of a robotic platform (*Zeno* robot at the University of Sheffield) used to study human-robot interactions and robot ethics.

Nevertheless, a problem with swarm systems is that the attractive features of their social structure also make the interactions with (external) users complex (see Figure 2.1). Human-Swarm Robot Interaction (HSI) is a domain focused on studying methodologies, phenomena and applications of swarm robots in joint systems with humans. In some cases, human navigation increases the chances of a robot swarm to succeed in solving a problem [13]. Depending on the accuracy of a decision taken by the swarm and its outcome, the human operator could choose to provide an input [41]. However, it is still not clear how these distributed systems can be effectively controlled by a human operator. Many of the challenges that HSI brings cannot be addressed with HRI metrics only [42], making evident the urge to explore human requirements, strengths and limitations for supervisory control in collaborative systems [14].

2.2 Human-Robot Interaction

The field of Human-Robot Interaction (HRI) is a subset of Human-Computer Interaction (HCI) [43]. It is focused on designing, understanding and evaluating robotic systems to be used by humans [1]. Sheridan et al. [30] proposed a division of four different mayor levels of interaction methods between humans and robots depending on the nature of the task: *Social Robotics*, *Human Supervised Robotics*, *Automated Robotics* and *Teleoperated Robotics*.

2.2.1 Social Robotics

A social robot interacts and communicates with humans following social behaviours. These robots start appearing more often in daily life and humans have increasing chances of interacting with them. Different robotic platforms have been proposed to study the social behaviour of robots in society (Fig. 2.2). As the desired effect is to provide support to humans while generating some kind of social relationship, concepts like responsiveness become a crucial factor for effective communication [44]. For example, *Maggie* is a robot used to study social robot behaviour [45]. The study attempts to research social interactions with the use of a human-friendly platform. The aim was to develop a robotic platform that functions more as a partner rather than just as a tool. Similarly, there has been more focus on the concept of using robots as social companions [46]. Robots that act less as assistants, machines or servants and act more as friends. These platforms generate new questions concerning the robots' social skills.

2.2.2 Human Supervised Robotics

Many robotics platforms have been designed as tools for different environments (home use, mining, agriculture, exploration, military, security, industrial, entertainment). Depending on the robot's objective, the interaction model could change and be adapted to the task. Augmented reality (AR) has been tested to help a human reconfigure and perform path planning in industrial robots [47]. A good example for exploration missions is the Mars rover *Opportunity* or Mars Exploration Rover-B (MER-B) which despite being semi-autonomous is supervised through tele-operation control by a human team [48]. In social environments, the collaboration between a random human and a robot can become complicated. Inside an office, an assistant robot could attempt to predict the operator's desires and intentions while lacking explicit communication [49]. The entertainment industry has widely used the tracking and mimicking of the human body as an interaction technique [50]. Some studies attempt to improve the tracking systems to measure human body movements [51]. Robotic applications use speech recognition combined with head pointing gestures [52] to develop interaction models to be used in homely environments. This allows robots to assist humans with daily tasks in highly used environments like a kitchen [53].

2.2.3 Automated Robotics

As robots become more common in all environments, more and more robots will be prone to be operated by non-expert operators and will need simple interaction methodologies [54]. Complex interaction methodologies seem to be a constant problem as interfaces are constantly being developed by robotics experts to be used by robotics experts [55]. There is a need for development of new interfaces able to deal with unexpected failures, fragmented information and untrained users not familiar with robotics terminology. Attempts to simplify the interaction process have as aim to give the robot system the ability to learn from its operator. In [56] an interactive interface is able to learn hand gestures using Hidden Markov Models (HMM) by repeated demonstration. This approach seems to be continuously explored as in [57] where the interface is simple enough so that a child is able to teach a robot how to write.

Furthermore, automated robotic systems where humans act like passengers or clients have the need to include some form of interaction between both, humans and robots. For example, in a security system that is controlled by artificial intelligence (AI), an important question that needs to be asked is: Should the robot use its “own judgement” in taking actions where human life’s are involved or should humans be left inside the control loop [30]. It is important to define where humans would best fit in the control loop. At this point, interaction may become passive as AI could learn from a human without the need for the operator to provide direct feedback. In [58] an action-conditioned video prediction model is tested for unsupervised learning, eliminating the need for a human to monitor the system.

2.2.4 Teleoperated Robotics

In non-routine tasks, like hazardous or inaccessible environments, HRI plays an important role [31, 54]. The need for effective teleoperation interfaces becomes evident when humans cannot be present in the working environment. Moreover, understanding human perception and its psycho-physiological factors, particularly in dangerous situations, requires new methodologies and different approaches of HRI [59]. These robotic systems are not meant to replace humans, but instead offer new capabilities to process information and react to adversities [12]. Nevertheless, robot-assisted rescue tasks still have a 2:1 human to robot ratio, placing two people at risk per tool.

Some search & rescue oriented studies [20, 60–62] have searched for different approaches to interact with rescue robots. An eye tracking gesture control (gaze gesture) was developed to teleoperate a drone in dangerous environments [63], with its control having some overlap with natural eye inspection patterns. Augmented reality is used to provide feedback in a virtual/augmented manner as a support tool for a human operator [20]. This way, the extra graphical information could be useful in hazardous tasks like search & rescue, however the operator would need to have direct visual contact with the robots. Similarly, the SAGE interface mainly focuses on maintaining the operator's situational awareness [64]. The multiagent system (MAS) infrastructure proposed in [10] is a strategy mainly used in a simulation environment that relates to sensor fusion and interface design for effective robotic control. However, there is still need to develop a strategy with the capacity to deal with hardware in the real world, aimed to protect the human operators while performing any amount of rescue/protection tasks. The real world environments would be dynamic with restricted access points, bandwidth limitations, with restricted communication capabilities, limited to general awareness and with multiple working agents (robots and humans).

2.3 Human Multi-Robot Interaction

Human interaction with Multi-Robot Systems (MRS) has been more explored over the last decade. It is highly motivated by the benefits that human intelligence merged with teams of cooperative robots could offer. Also, these systems could help reduce the required manpower for a given task. In addition, effective interaction with multi-robot systems could decrease the workload (or raise the "fan-out") of the human operator. The "fan-out" concept relates to the amount of robots that a human operator can endeavour to control. A proposed method to measure interaction effectiveness suggests to compare the fan-out and the robots' effective time to be used as a metric to compare different interaction designs [65].

Design of effective interfaces between large amounts of robots and a human operator can be a challenging task, from placing the human operator directly as a leader to designing gesture based languages between the human and the robots [66]. Some designs attempt to put the human operator directly in the field. An example of this approach is where a MRS and a human work together in transporting heavy objects [22]. Other designs use mobile devices through vision-based control and pose estimation [67]. Augmented reality has become a key tool for human interaction with large amounts of robots [20]. It has been used for debugging and/or monitoring of MRS as in [68].

Security and surveillance missions are potentially part of the most interesting application areas for MRS. A surveillance mission with multi-UAV systems that attempt to achieve the best situation-aware patrolling route while minimizing communication latencies could improve the way security and rescue teams work [69]. Rescue robotics could also be strongly benefited from MRS. Nevertheless, complete autonomous multi-robot teams in hazardous environments are still not realistic, so human interaction has always been needed up to this point. However, reducing the number of human operators while maintaining control would improve the efficiency of search & rescue missions [70]. These tasks could be done by multiple robots in an autonomous manner and the managing task could be done by a human operator, either independently or with autonomous support.

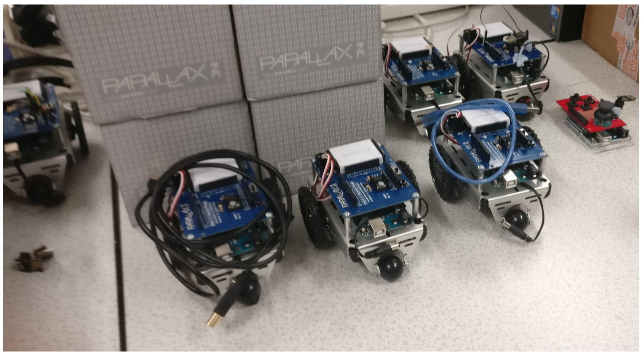
2.4 Human-Swarm Robot Interaction

There have been several proposals suggested for the implementation of human interaction with robot swarms. Some of these proposals are through controlling units as a leader [26], with haptic interactions [71] or with body gestures [72]. A recent survey on human interaction with robotic swarms is available in [73], and a more dedicated survey of HSI applied to UV's systems is available in [74].

One of the most frequently studied challenges in human-swarm interaction has been the design of appropriate control inputs for swarm robots. Four basic control approaches have been distinguished [73]:

1. Selection of specific algorithms/behaviours

2. Changing parameters from the algorithms



(a)

(b)

Fig. 2.3 a) A small group of Boe-Bot robots that will work as a swarm. b) A group of Kilobot robots belonging to the 900 robots swarm from The University of Sheffield.

3. Influence through environment factors

4. Control through a selected subset of the swarm

Human intuition is still the main tool for the design and development of swarm robots [37]. With proper design of the system architecture it is possible to achieve scalability of performance without altering the architecture and/or behaviour with any kind of human intervention [75]. However, it is highly likely that this could only be true under ideal conditions (like in simulations e.g. [76–78, 8]), as real environments have multiple unpredictable parameters (light, friction, temperature, vibrations, etc.). Real world systems have to face these problems in addition to their own related hardware problems (battery failure, overheating, components failure, etc). A human operator could attempt to compensate for unpredictability of the environment with high level leadership by managing responsibilities and adapting the workload and the team's task performance through collaboration [79].

A real swarm robot model would be limited for the operator to observe and/or communicate with a subset of the swarm [80]. It would also allow some kind of restricted influence of the robots based on limited local interactions and to be supervised on an abstract level. It would be able to stimulate the emergence of adaptable behaviours in crowded places [15], perform some human-like path planning [81] and avoid obstacles in dynamic environments [8]. All of this would be needed for the robotic system to become practical in real world applications [31]. Furthermore, in those real world applications, monitoring of the robot swarm behaviour may not be the operator's primary or secondary task.

Studies focusing on the human interface controls usually assume access to the position of all the robots in an interface, generally referred to as the "birds-eye view" [82]. Many developed interfaces work with the assumption of some sort of access to direct visual contact either from the operator or from a control device (e.g. [83, 84, 8]). This is a benefit that real world systems cannot rely on. For example, in the selection of sub-teams from a UAV swarm with an un-instrumented operator through body gestures, the robots would require to have direct visual contact of the operator to be able to receive the instructions [85]. Another example involves an overhead camera inside the control loop working as a tracking system [86].

The problem of obtaining and visualizing the information about the state of the swarm has been studied less than swarm controls. Most HSI interfaces have been developed by robotics experts but have not succeeded in being user friendly to domain experts [54]. For this reason, meaningful self-organised feedback mechanisms would be required for non-expert operators to be able to use all the capabilities that a swarm system could offer [87]. However, this self organized swarm feedback should not be based on the constraint of having direct visual contact [88], but powerful enough to offer effective general awareness.

Teleoperation of a robot using the "single-leader network" [89] architecture and their effect on the remaining swarm has been studied in different ways, for example, by using a robot leader to determine what kind of flocking and motion behaviours can be generated from different human inputs [90]. This control architecture can also be mixed with other types of controls (like switching algorithms) to create hybrid controls that allow human operators to teleoperate leaders and switch the swarm algorithm [13] to guide it and form particular geometric forms [89]. However, the assumption that the human operator has constant visual contact with all the robots constrains the systems to be used within a controlled environment (like in a research laboratory). In addition, visual contact with all the robots at once might directly affect the psychophysiological state (stress, anxiety, happiness) of the operator and their perception of the swarm behaviour [76, 84].

A general introduction to swarm robotics [34] lists some robotic platforms that have been frequently used in swarm robotics studies. Some of these platforms are the *Kobot* robot [91] and the *e-puck* robot [92], this last one being the one used in the experiments of this thesis. Some newer platforms are the *Kilobot* robot [93] that was specifically designed for swarm research (see Fig. 2.3b), or the open-source *Zooids* robot [94] that in its case was specifically designed for HSI. All of these platforms have some hardware differences that challenge user interfaces to find effective communication methods that can work with their different hardware configurations.

2.4.1 HSI Strategies

For current human-robot teams, exploration and search & rescue applications are the least favoured in terms of reality gap when related to MRS. As mentioned in Section 2.2, one aspect of current robot-assisted teams is the 2:1 human to robot ratio, placing two people at risk per robot [12]. Swarm robotic systems could dramatically improve this ratio but the interaction techniques need to be improved. A problem related to multiple HSI studies is that they assume that the human operator has some kind of robust access to all agents and to the birds-eye view. A difference with real world applications is that human operators may have limited access to some agents, and may only be aware of the location of a small subset of robots or none at all.

To improve the interaction techniques, first it is important to classify the different forms of control that the operator can have over a swarm robot. This organization would be focused into how the operator can provide input to the robots. It should be designed for the operator to be able to manage the robots as a whole while abstracting any details regarding the individual robots [25, 95]. The survey previously mentioned in Section 2.4 refers to four basic interaction approaches [73]: algorithm switching, parameter changing, environment influence and selection of leaders.

- **Algorithm & Behaviour Selection**: Using this technique the operator can provide input to the robot swarm through selecting pre-established behaviours. For example, in [96, 60] the human operator could exert his influence by choosing from a set of behaviours (*Stop, Come, Rendezvous, Deploy, Random, Heading and Leave*). The performance relied on the operator's ability to choose the appropriate behaviours and the right repetition frequency. Another example explores the switching between a predator and leader behaviour, yet depending on the complexity of the mission objective, the interaction might need to be exerted by multiple humans or design specific collective structures [97].

- **Parameter changing**: Using this technique the human operator can modify a single or multiple parameters of the swarm. This way, the emergent behaviours are influenced indirectly, and the effects of this influence appear when the swarm interacts with itself and/or the environment. Modifications of these parameters have the potential to generate big changes in the properties and organization of a group behaviour [98]. For example, in [99] group control of a swarm of UAVs was implemented through the modification of four parameters: *Conformity*, *Sociability*, *Dedication*, and *Disposition*. This high level control allowed the operator to calibrate the swarm behaviour even during use.

- **Environment influence**: Using this technique the operator can influence the behaviour of a swarm by altering part of the environment, either in a virtual or physical manner. An example of a virtual modification has been explored with the implementation of the "Gravity Point Method" [7]. With it, the human operator can place virtual attraction and repulsion points in any location of the environment to guide the robot swarm. Similarly, as an example of physical manipulation, the operator can guide a swarm via pheromone-inspired control by spreading different types of pheromones, establishing desirable *Keep Away* and *Stay In* areas [100]. Potential real world applications of this interaction technique could be used to guide nano-particles in wide search spaces [101], navigate microrobots through different fluids [102], or even manipulate object positions [103].

- **Selection of leaders**: Using this technique the operator has the ability to gain control of a robot or a subset of robots from the swarm. This way, the operator has to focus on a reduced number of robots simultaneously. Leader-based influence has been the most explored technique to influence a multi-robot system. This type of interaction needs to be scalable and reliable, as the objective to have more available working robots would only succeed if they can be effectively interacted with [75]. There are multiple examples in the literature of leader-based swarm interaction. In [104], leaders can be dynamically selected over a GUI and the operator can adjust three different aspects of their interactions: *leader density, sensing error,* and *method of information propagation*. Another example uses a very simple (and intuitive) control through standardized Web Services [105]. This interaction technique offers an interface that allows external operators to gain remote control over a robot avatar, thereby influencing the swarm. It is the objective of this technique to provide the operator with the ability to inject "expert" knowledge into the system without the need of manipulating the entire swarm [13].

The challenge of coordinating the actions of a swarm system with reliable interoperability between the human operator and the robots would require good communication and teamwork models [10] to overcome faulty cooperation between robots and humans. Properly designed swarm robots will allow an increase of their performance and a balance between autonomy and human influence. High levels of automation would enhance system performance but have low resistance to system failures. On the other hand, low level of automation would promote the operator's situational awareness and failure robustness, but could affect task performance [106, 74]. Furthermore, autonomous learning methods could be implemented to allow a human operator to exert real-time control over a swarm [107].

2.4.2 Bandwidth Limitations

As swarm robots start to grow in size, the need to explore communication methods that are reliable with limited bandwidth becomes more important. Bandwidth limitations make MRS's difficult to monitor and/or influence. Furthermore, it is because of the distributed qualities of collective systems that HSI can be even more challenging [25]. It is virtually impossible for a human operator to simultaneously control and/or supervise every agent of a swarm robot in an individual manner. In addition, if the communication infrastructure that enables the interaction is constrained in terms of latency and/or bandwidth [24], the operator's situational awareness [108] and the understanding of the systems' actions, behaviours and/or task progress [109] can be considerably affected. These constraints make a real-world HSI system unreliable in most situations [42].

The impact of these limitations has been studied with operators that only have access to limited information about the robots [104, 110]. Human operators have been able to learn and adapt to swarm dynamics and accommodate to a "medium bandwidth scheme". Swarm members communicate between them and then provide the operator with averaged members data. This scheme shows that providing information of every single agent in the swarm does not improve the interaction performance [82, 110]. Also, this scheme gives the opportunity to control a distributed system through the selection of a single or multiple leaders acting as intermediaries between the operator and the robots [104, 26].

The studies that are focused on controlling the swarm usually assume access to the birds-eye view and present the position of all swarm robots in an interface (some examples are [7, 13]). A real-world application would experience complications and delays when transferring that amount of data, most likely caused by limited and unreliable bandwidth [111]. These delays could have a significant impact over the control performance [112].

On the other hand, the problem of obtaining and visualizing the information about the state of the swarm has been studied somewhat less than swarm controls. Early attempts to monitor distributed systems have tried to use augmented reality (AR) techniques [113]. Nevertheless, AR demands high bandwidth robustness for the detection and processing of visual tags. Simulated environments that limit the access to the state information of the swarm with bandwidth or latency restrictions provide a more realistic working scenario, making it more similar to the real world [114, 110]. It is possible, with the right interface, to allow the human operator to detect overall system state with limited amount of information [24]. In [88] an attempt to use minimal bandwidth and extract useful information from a robot swarm was presented. The importance of this study was that it was performed with real robots.

For real world applications, the communications network in hazardous environments is most likely not to be stable and the available bandwidth may be intermittent or reduced [12]. Exploration and search and rescue missions may present complex communication limitations [110]. In dangerous environments, where the operator has restricted or non-existent access to the area, wireless communication would be the main (if not the only) channel of communication. Yet, wireless connectivity rises multiple issues like, attenuation due to adverse weather conditions [115], the presence of noise in the signal causing error rates and/or link outage [116] or coverage, routing and jamming holes in the network [117], to mention a few. Added to this, if the amount of data that needs to be exchanged between the robots and the operator is big [118], the question of whether the technology and interfaces are ready to face such environments in the real world rises.

2.4.3 Neglect Benevolence

Neglect benevolence [114, 119] is a relative new concept that studies the dynamic nature of emergent behaviours. Most swarm algorithms require time to converge to a stable emergent behaviour. If their dynamics (swarm algorithms) get disturbed by the interaction with an external entity, like a human operator, the convergence may be delayed or interrupted and the performance may be negatively affected. Hence, some swarms may benefit from a period of neglect forcing the human operator to give some time before applying a new command [114].

There can be positive effects to learning the neglect benevolence dynamics of a system. A human operator can learn to adapt the timing of the applied commands to take advantage of the neglected time [33, 119]. The operator could choose the best time to provide a new equilibrium point redirecting the swarm from one natural stable state to another. In these cases, it could be beneficial for the operator to delay the input to minimize the overall time required for the system to converge [119]. Consequently, the autonomy of a robotic system is measured by its neglect time and the amount of the operator's attention that it needs [120].

As in a collective transport experiment, the time that the operator waits before executing another location command gives the swarm enough time to react and transport an object to a defined initial location [32]. Other examples include a human operator attempting to interact with a swarm robot through different kind of high-level attractors [25, 24].

In contrast to the concept of neglect benevolence, the *neglect tolerance* concept refers to the quality of a system which deteriorates due to periods of neglect [120]. The more often the system receives attention from the operator, the better the performance it achieves. In this case, for convenience to the operator, increasing a robot's trusted intelligence can increase its neglect tolerance and thus increase the operators' fan-out [65].

2.5 Interfaces

Multiple different interfaces have been proposed, developed and tested attempting to establish some kind of effective interaction between a human operator and a robot swarm. A useful and effective interface needs to be designed so that a human operator can dynamically communicate high level intentions to a robot swarm and to be able to close the interaction communication loop between a human operator and a robotic system. It should also be able to receive and interpret the swarm feedback and present it in an understandable way to the operator. The human should be able to input information (commands, objectives, requests, etc.) to the system and to receive feedback (environment information, system state information, data processed information, etc.) from it.

Parallel to the design of user interfaces, comes the design of methods to evaluate the effectiveness of different user interfaces. An example of this is a cognitive model that was designed to test multiple types of graphical interfaces [121]. It attempts to predict how an interface will be used and to understand the cognitive effort of the human operator while interacting with the robots. Nevertheless, this model was created only for graphical user interfaces, and needs to be expanded to include all other types of interaction methodologies.

There have been multiple different channels explored to provide (input) information to the robots [122], however the human operators usually get feedback from them in a visual way [123]. Similarly to the input of information to the robots, the feedback could be provided to the operator through different channels [122]. Table 2.1 presents the four main input and feedback channels for HSI [122, 123].

Table 2.1 Interaction modes for input and output of information from the operator perspective.

Interaction Channels		
Input		*Feedback*
Graphic Controls	-	Graphic Indicators
Gesture	-	Visual
Speech	-	Audio
Haptic	-	Haptic

2.5.1 Operator Input

- **Graphic Controls**: A graphical user interface (GUI) is the most prevalent channel to send information to a swarm robot. The SAGE interface was designed for the Multi Autonomous Ground-robotic International Challenge (MAGIC) and attempted to interact with multi-robot systems through high level manual control [64]. Other GUIs can even create, edit and test virtual environments as well as connect to real robots [124]. Similarly, it has become more common to use portable devices that connect directly with the robots and through some type of remote control modify their positions [125, 21]. Finally, some independent interfaces try to include adaptive algorithms to automatically help the operator manage the commands issued as part of the control task [126].

- **Gesture**: The detection of gestures as the source of commands has been explored in many areas. One of the first devices used to acquire gesture commands was developed by Microsoft and was named as the “Kinect” [50]. It has been used as an interactive sensor with deictic and body gestures to control a multi-robot system [72]. Gestures have been explored as a tool to select agents within a swarm of robots [29]. Furthermore, these attempts have tried to give the swarm the ability to learn the command gestures from a human operator that acts like a teacher. The gestures are meant to be learned through a repetition process with the help of a consensus protocol [41].

- **Speech**: Interaction through speech can be a complicated task. It has been tested in the selection of sub-teams within a swarm of robots [77] as well as to give commands to update the sub-team objectives [127]. However, this technique is hard to work with because of the nature of speech itself. Any robot/device would have complex hardware and software requirements in order to be able to do proper speech recognition and recognize spoken commands.

- **Haptic**: This interaction methodology is related to the sense of touch. The Phantom Omni device [128] was developed specifically for this type of interactions (haptic interactions). Some studies have used it as the main participant input source [123, 129, 71], for instance to enable a single operator to control and interact with a swarm of robots [129, 71].

2.5.2 System Feedback

The process of receiving and interpreting feedback information from a swarm robot can be complex. The amount of raw data can be difficult to process and interpret, especially if a human operator has to do the interpretation by themselves. It is usually easier for a human operator to interpret and trust visual (graphical) feedback, followed by auditory feedback and lastly textual feedback [130].

- **Graphic Indicators**: Using a GUI has also been the most used channel to receive and interpret feedback from a swarm robot. The SAGE interface provides the ability to monitor the robots' positions through four displays [64]. Augmented reality has been used as a 3D interpretation of the systems' feedback, providing also some kind of spatial information about the environment [88]. Otherwise, the presentation of the feedback is delivered in 2D and limited to some kind of graphic interpretation [124]. Some adaptive algorithms have been tested to automatically determine a presentation layout that fits to the operator based on some user defined configurations and the history of data presentation for similar tasks [126].

- **Visual**: This type of feedback is the most commonly used if there is no other kind of interpreted feedback channel. It involves the human operator having direct visual contact with the robots and interpreting the behaviours and responses depending on their movement and actions (some significant examples are [129, 94, 77, 125]). Some other interfaces that are specially designed for visual feedback involve LED matrices that create patterns depending on the message to be communicated [131].

- **Audio**: There have been a limited number of tests in which the human operator receives feedback in audio form. It has been tested as part of a visual interface, which provides the operator with audio and tactile feedback as secondary cues [132]. However, there is still no study that uses only audio as the main feedback from a swarm robot.
- **Haptic**: The previously introduced Phantom Omni device is also able to generate some physical force as feedback. Swarm robot manipulability information can be relayed to the human operator via feedback forces through this haptic device [71, 123]. Simple tactile cues mixed with other types of communications (like text and audio) seem to increase the operator's awareness of the surroundings [132]. This increase of situational awareness could be of great use in systems that work in hazardous environments [133].

2.6 Awareness Levels

Based on the *Cambridge Dictionary* [134] definition, "awareness" is:

"The knowledge that something exists, or understanding of a situation or subject at the present time based on information or experience."

There are multiple types of awareness levels [135], the two types that are related to this thesis are *situational awareness* (SA) and *task awareness* (TA). Enabling a human operator to keep proper awareness of the system stimulates the level of trust in it and should not affect the operator's workload [136]. For this thesis, the discussed literature is constrained specifically to SA and TA in robotic systems.

2.6.1 Situational Awareness

Early studies define situational awareness (SA) as a state of knowledge of the environment and the surroundings [137]. It involves the operator being aware of what is happening and how information, events and actions will affect their objectives immediately and in the near future. An up-to-date review about SA in individuals, teams and systems [138] separates the process to acquire SA from the decision making process. Even the "best-trained decision makers" would make wrong decisions if they lacked partial or full SA [139].

It is important to notice that SA is not about the user knowledge, but about the portion of it pertaining to the state of a dynamic environment. The process to achieve SA is defined as situational assessment, and is related to acquiring and/or maintaining SA [140]. This process is composed of three different activities: perception, comprehension, and projection.

Some studies state that swarm robots could provide extra information to a human operator to achieve better levels of SA [31]. Other studies state that human operators achieve better performance when they have access to the birds-eye view than with other type of visual feedback [141]. However, there is still need to study the impact over the adaptation of operators when totally removing access to global state information (such as position). This case can be referred to as the swarm being "out-of-sight". Such interaction schemes are pointed out as desirable [37, 31], however, it is not clear what the cost of such an interaction scheme would be with regard to the operator's ability to observe and control the swarm effectively.

Despite the lack of out-of-sight awareness understanding, multiple studies assume that the operator has some kind of access to all robots [97, 125], unlike in practice where a human may only know the locations of a small subset of agents or even none at all. Throughout the literature there are multiple mentions of research motivated by real world applications, for example, in search and rescue missions [64, 60, 59, 61]. As previously mentioned, one aspect of actual robot-assisted search & rescue systems places two people at risk per robot [12]. However, for swarm technology to improve this ratio, interfaces and interaction techniques need to be improved.

Proper human interfaces still lack effective means to keep SA when the birds-eye view is missing. An operator may have SA for short periods of time but lose it eventually as information flows or stops [142, 54]. For example, in teleoperated robots, interface failures (degraded video image, low bandwidth, reduced frame rate, reduced resolution, etc.) may degrade the operator's situational awareness and consequently, the performance [108, 62]. SA is critical for an operator to make effective decisions especially if in-the-loop of the control [143]. Lack of this type of awareness may cause operators not to detect critical system errors and lack of knowledge on how to restore the system's functionality [106]. Furthermore, operators should be able to predict when an event will occur as time is also an element of situation awareness [142].

2.6.2 Task Awareness

The concept of task awareness (TA) has been defined as the operator's awareness of what tasks have been completed and which ones are still unfinished. It is the knowledge of task queuing and task responsibility (referring to who is responsible for the performance of each task) and the team members' understanding of how a task will be completed [144].

Team and task awareness requires common understanding, however this does not mean that all team members should have the same understanding, but that every member should have a similar mental model [30] (awareness and trust) of the main objective [145]. Task awareness runs at two levels [146]. At a global level where the selection of a main objective is presented, and to a local level where identification of specific minor tasks involved in the realization of the main objective is required. Task awareness is a conscious representation of a selection process, so a sign of task awareness could only be found when learners have identified extra minor tasks about it. In addition, if the human operator does not understand how the robot team is organized, the trust in the system's efficiency will be diminished [70] even if the operator's task is relatively easy [147].

Freedom of mobility of the operator improves the feeling of presence and the workspace awareness, but affects the efficiency of the task performance [148]. In addition, a human operator cannot adequately focus on more than a handful (low fan-out ratio [65]) of simultaneous moving objects [42]. Making the robots autonomous could always help the operator to overcome this problems, however an increase in the autonomy level of the robots will have a trade-off in the operator's understanding of the system [14]. For this reason, an appropriate equilibrium of autonomy between the human operator and the swarm robot and a reliable interaction framework are needed [87, 149].

Chapter 3

Human-Swarm Robot Interaction with Full Awareness

In Section 1.4.1, multiple works were discussed that approach the swarm robot-human interaction challenge in different ways. This chapter explores the effects that the influence of a human over a swarm system could have, particularly under the condition of full awareness. For this condition we position the human in the same environment where the controlled swarm robot is working. We introduce a strategy where the human influences the swarm through a selected robot acting as a leader. The human operator uses a portable device that acts as a communication interface with the robots.

In this study the interaction process occurs with one of two portable devices: A mobile phone and the Google Glass. The mobile phone uses the standard GUI interface, while the Google Glass introduces voice recognition techniques. This approach proposes a structure where the human operator (which will be refereed to as "the user") uses one selected robot of the swarm as a leader. Through the leader-robot configuration this study attempts to understand how the user can guide a swarm of robots in accomplishing a task. It is important to mention that such task can only be accomplished with multiple similar robots collaborating between them.

To achieve this, we used as an example scenario a cooperative transport task with real robots. The cooperative transport of large objects by groups of comparatively small robots is a canonical task studied in collective robotics [150–153]. Chen et al. [21] proposed an occlusion-based cooperative transport algorithm that does neither require the robots to communicate with each other, nor to consistently perceive the goal. We study how a human operator interacts with only a single robot of a swarm, yet gains control over the entire swarm.

The structure of this chapter is as follows. Section 3.1 presents the methodology and a formulation of the problem (3.1.1). Section 3.1.2 introduces the robot platform. Section 3.1.3 explains the user interface architecture and an overview of the used mobile device and of the Google Glass. Section 3.2 presents the obtained results followed by the conclusions of the chapter (3.3).

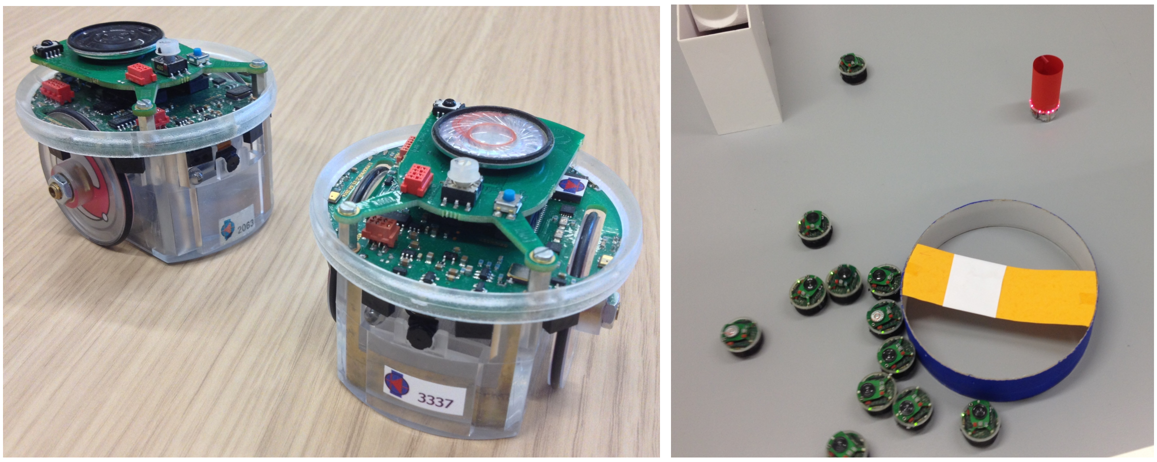
3.1 Methodology

In this section, we define the problem and describe the robotic platform and the developed graphical user interface (GUI). We also introduce the interaction devices, their qualities and restrictions. We describe the general specifications of the interaction interface (GUI) and how it was adapted for the mobile phone and the Google Glass.

3.1.1 Problem Formulation

Section 2.4.1 discussed some useful, but challenging, strategies used to interact with a swarm of robots by a human operator. It is clear that multiple approaches to human interaction with robot swarms have been explored. In [73], two major types of interaction are mentioned: *indirect* and *direct*. In indirect interaction, the human operator interacts through external strategies (e.g. beacons or markers) or environmental influences (e.g. ambient light or noises) [22]. During direct interaction, the human operator interacts through direct access to the swarm, either fully or partially [125, 72]. In this study, we explore how human operators can collaborate with a swarm of robots through portable devices. The main distinctive aspect of the presented system is that the human operator needs to interact with only a single robot, and yet gains control over the cooperative actions of the entire swarm.

As a practical example for this case study, a proposed occlusion-based cooperative transport algorithm by Chen et al. [21] was used. In it, the swarm aims to transport an object from a starting location to another position. First, the robots in the swarm start looking for the object. Once the object has been found, the robot pushes it in a perpendicular manner to



(a) (b)

Fig. 3.1 (a) The real e-puck robot. (b) A group of e-puck robots pushing the object towards the (red robot) dynamic goal.

the objects' surface, but only if it had no line of sight of the goal. In this situation, the robot's view of the goal is assumed to be occluded by the object. This algorithm requires the robots to neither communicate with each other, nor to consistently perceive the goal.

A limitation of the occlusion-based cooperative transport algorithm is that it assumes that the environment is free of obstacles. If the line of sight between a robot and the goal is occluded by anything but the object (e.g., walls, other objects, other robots, etc.), the strategy will not work. In this chapter we explore a way to overcome this limitation by adding the help of a human operator to the swarm system. While the robots' sight may be occluded by obstacles in the environment, a human operator has the advantage of the bird's-eye view. Through a pre-defined leader robot, that acts like a dynamic goal, the operator can guide the other robots to push the object and avoid any obstacles (as in Fig. 3.1b). This way the swarm can focus on the physical manipulation, while the operator focuses on the overall guidance.

3.1.2 Robotic Platform

The robotic platform that was used is the e-puck robot [92] (Fig. 3.1a). It was designed mainly for education and research purposes. It is a differential wheeled robot which can move backward and forward at different speeds with a maximum of 12.8 cm/s. It is 5.5 cm high, 7.0 cm in diameter and weighs 150 g. Its main processor is a dsPIC30F6014A microcontroller which belongs to the 16 bit architecture family. Its processing speed is 30 MIPS (Million

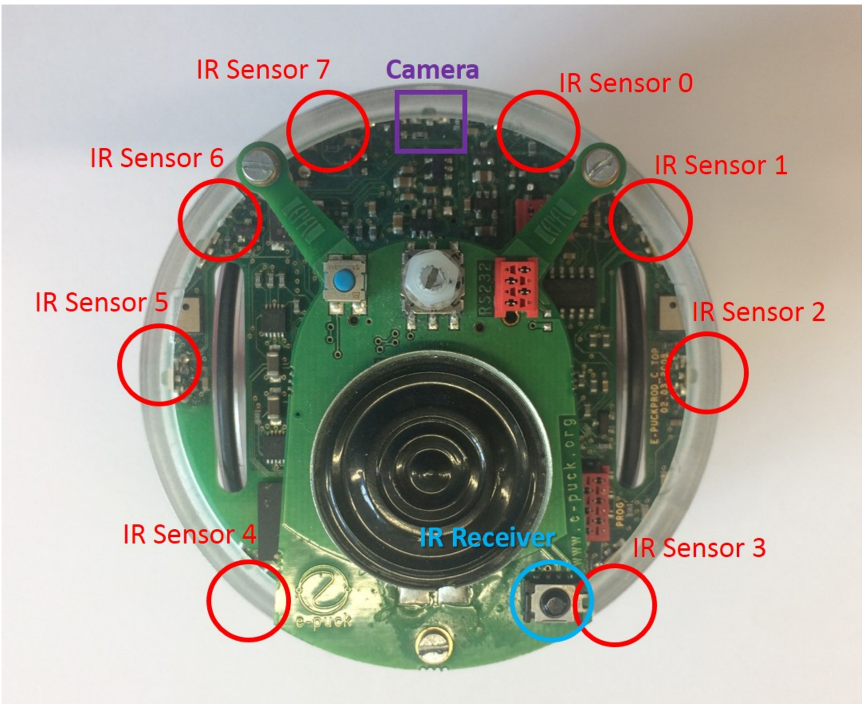


Fig. 3.2 A picture of the real epuck taken from the top. In it, the location of every IR proximity sensor, the IR receiver and the color camera are marked.

Instructions Per Second). I has 8KB of RAM memory and 144KB Flash ROM memory. It also has a Bluetooth antenna that can emulate a bidirectional serial port and supports a baud rate of 115.2 KB.

The Camera

The e-puck has a directional color camera mounted in its front (Fig. 3.2). The camera is hard wired to the dsPIC and is configured through the I2C protocol. The camera is able to provide 8 bit RGB color images of 640x480 pixels. However, the e-puck memory can neither store that amount of data nor can the microcontroller process it fast enough. For this reason, the images are sub-sampled to 40x15 pixels per frame. With this sub-sampling, the e-puck is able to process an average frame-rate of 18.5 fps.

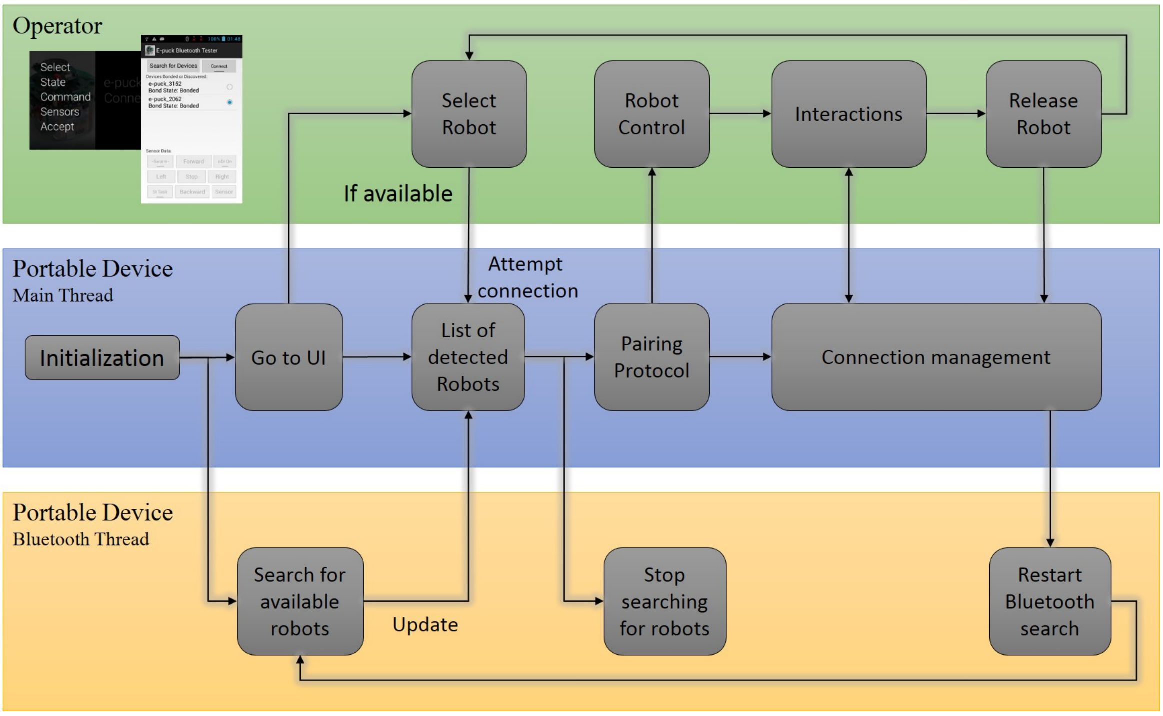


Fig. 3.3 User interface software architecture representation.

The IR Sensors

There are eight infrared (IR) proximity sensors distributed around the e-puck body (Fig. 3.2). Each one of these sensors has an IR LED and an IR transistor. These sensors are able to measure the distance to a nearby object to a range of about 8 cm. Above the IR sensors there are also 8 red LEDs that can be programmed individually. On the top part of the robot there is a modulated infrared receiver. With the latter the e-puck can decode a modulated signal from a conventional TV remote control.

3.1.3 User Interface

The interaction between the human and an agent of the robot swarm occurs through a GUI. This interface is meant to provide the user with the ability to select and control an individual robot from the swarm, and through this robot influence the members of the swarm. The same interface was used with two portable devices: A Cubot C9+ mobile phone and the Google Glass. Both devices work with an Android operating system, but have some hardware differences. Because of this, two different development tool-kits were used: The Android

SDK, that is of general use for development within all Android devices, and the GDK (Google Development Kit), that is an add-on to the Android SDK used for development of Google Glass applications.

The interface uses bluetooth technology from the mobile devices as the communication channel with the robots. This is the only wireless protocol that the e-puck robots have. Despite the possibility of connecting up to 7 simultaneous devices/robots (or active slaves), this interface was limited to just one. This constraint was set based on the design of the interaction process that was developed.

The GUI architecture is shown in Fig. 3.3. The green rectangle contains the user actions, the blue rectangle contains the actions performed by the main GUI thread and the yellow rectangle contains the actions performed by a parallel thread that is in charge of managing the bluetooth connections. Once initialized, the bluetooth thread starts searching for nearby robots and the GUI is loaded so that the user can request for a connection with one of the detected robots. From Fig. 3.3 we present the generic actions that the user can perform as many times as needed:

- **Select Robot**: Select and attempt a connection with any of the listed robots.
- **Robot Control**: Receive ID and control of the connected robot.
- **Interactions**: Give direct motion commands to the robot, change the behaviour of the robot and request information directly from the robot's sensors.
- **Release Robot**: Disconnect and/or change robot.

When a user starts the connection process, the system first checks if there is an existing ongoing connection. If the connection channel is free then it identifies if the selected robot has been previously paired. If not, the pairing process starts automatically and afterwards finalizes the connection. A closer look at the GUI is available in Appendix A, where we present the structure of the classes and the most important functions.

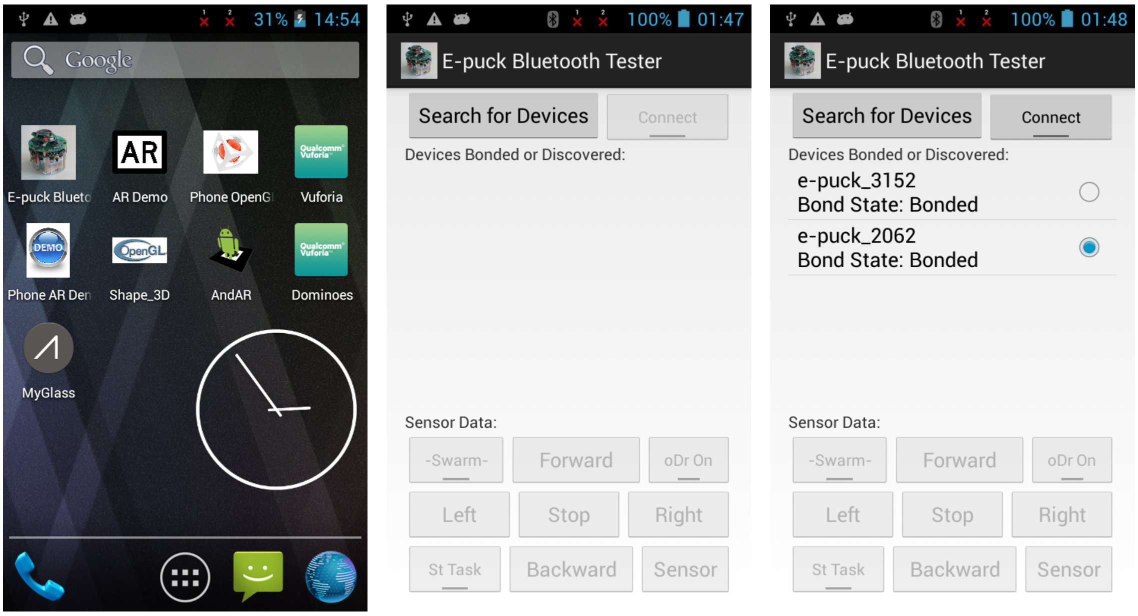
Android Device

The GUI was designed to be functional with any portable device that works with an Android OS (phone or tablet). For the experiments of this study the used mobile phone model was the Cubot C9+ (as seen in Fig. 3.4) which has the following hardware specifications:

Fig. 3.4 Picture of the Cubot C9+ phone used in the study.

- OP Version – Android 4.2.2
- ARM Cortex-A7 Dual Core 1.2 GHz
- Graphics ARM Mali-400 MP 500 MHz
- 256 MB RAM + 512 MB ROM
- 2.0 MP back camera and 0.3 MP front camera
- 4.0 Inch Capacitive Screen, 800 x 480 pixels
- Bluetooth 2.0

Fig. 3.5 presents three example screenshots from the mobile phone. As in Fig. 3.5a, the GUI can be reached via the icon located at the top left corner. The way the user interacts with the interface is mainly through the phone's touch screen. The GUI was designed as an application for mobile devices. As soon as the user opens the GUI, it activates the bluetooth device. The user has the option to start searching for an available robot as seen in Fig. 3.5b. When at least one robot is found, its ID is listed and the user has the option to establish a



(a) (b) (c)

Fig. 3.5 Snapshots taken from the Android phone interface. a) The portable device main menu. b) The portable device when it starts searching for available e-puck robots. c) The portable device when it has already found some e-puck robots.

connection. This is done by selecting the chosen robot from the list and selecting "Connect" as in Fig. 3.5c.

When a connection to a robot has been established, the nine buttons at the bottom of the GUI (Fig. 3.5b and Fig. 3.5c) become active. In Fig. 3.6 we present a diagram with all the possible actions that the user has and we provide further explanation of each of them:

- *Motion Commands (Forward, Backward, Left, Right, Stop)*: Gives the operator the ability to move the connected robot in different directions.
- *Swarm/Leader*: Swaps the behaviour of the robot. "Leader mode" - Enables the operator to teleoperate the robot. "Swarm mode" - Enables the robot to behave like any other unit from the swarm.
- *Overdrive On/Off*: Instructs the robot to accept or ignore any commands issued by a remote control.[1]
- *Sensor*: Delivers one reading of all the IR sensors in the screen.

[1] In the experiment, all robots get activated simultaneously by issuing a signal via an IR remote control.

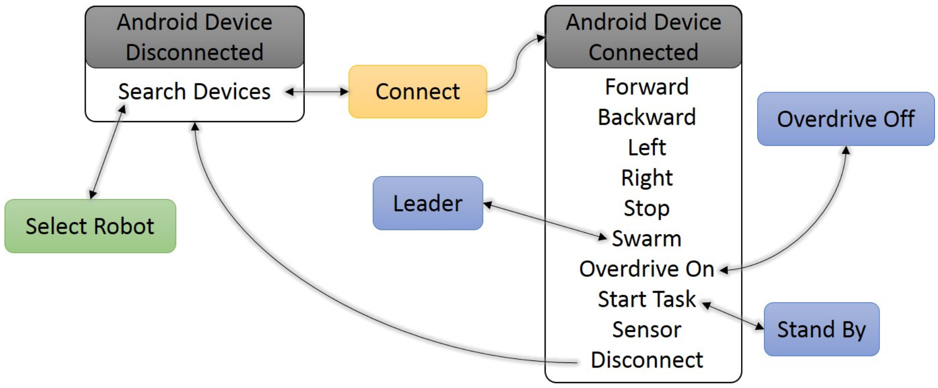


Fig. 3.6 Diagram of the architecture of the Android mobile device software.

- *Start Task/Stand By*: Swaps the active status of the robot. “Start Task” - Enables the robot execute the selected behaviour. “Stand By” - Allows the operator to change some configurations of the robot (Swarm/Leader mode, Overdrive and Connect/Disconnect).
- *Disconnect*: Terminates the wireless connection to the robot.

Google Glass

The Google Glass was developed by “Google Inc.” as an optical head-mounted display. It was released to the public in the UK on May 15th, 2014 and discontinued from January 15th, 2015. It works as a modern hands-free add-on but with more processing power and hardware capabilities. It has wireless connectivity through Wi-Fi and Bluetooth 4.0LE technology. It can display images and video through a small interferometric screen. This screen is positioned at the right upper side of the visual range of the user. It lacks an internal speaker or any auxiliary output to connect to external speakers. However, it has a bone conduction transducer that delivers the audio signals from the left side of the headset directly to the user auditive system. Some other common additional sensors like accelerometers and magnetometers are also present. The Google Glass hardware specifications are:

- Glass OS (Google Xe Software)
- 1.20 GHz Texas Instruments OMAP 4430
- 2GB RAM + 16GB ROM (12 GB of usable memory)
- Display: Prism Projector (640×360 pixels)

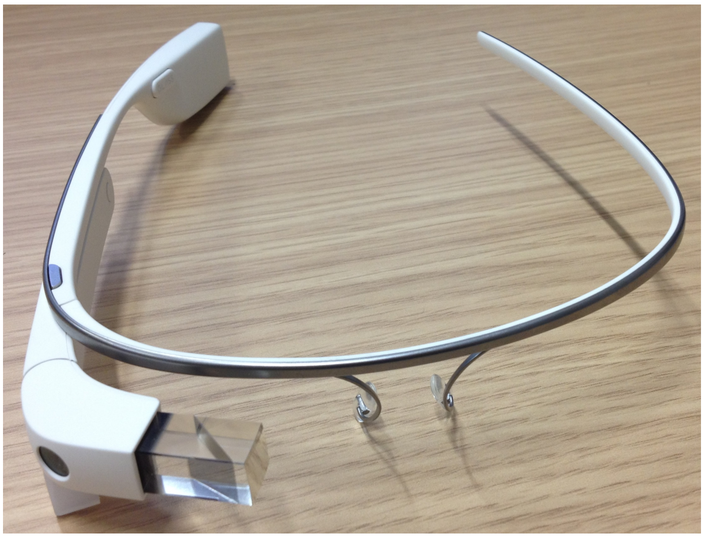
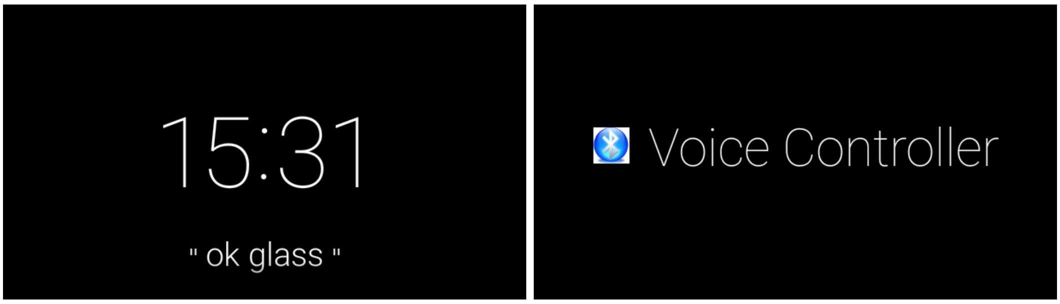


Fig. 3.7 Picture of the Google Glass used in the study.

- Photos 5.0 MP, Videos 720p
- Audio by Bone Conduction Transducer
- Inputs: Microphone, ambient light sensor, & Touchpad
- Sensors: Accelerometer, gyroscope & magnetometer
- 570mAh Internal lithium-ion battery
- Wi-Fi - 802.11 b/g 2.4 GHz & Bluetooth 4.0LE

To save battery and prevent heating, the main processor is dynamically underclocked. Depending on the circumstances, the processing speed can get down to 300MHz, being very low in comparison to the nominal speed reported that is 1.2 GHz.

Despite the underclocking procedure, because of the compact size of the glasses and the multiple devices it contains, the glasses suffer of overheating problems. When multiple components are simultaneously activated, the produced heat rises considerably. Even when only the bluetooth device and the display were activated, after a short period of constant use, the Google Glass would get seriously overheated. Despite this problem, the glasses performed acceptably during the 10 minute duration of each trial. Nevertheless, after every trial, the glasses had to be turned off to cool down.



(a) (b)

Fig. 3.8 a) Main Google Glass screen. b) Application *Voice Controller* icon.

To give the operator the ability to influence the swarm with the Google Glass, we adapted the GUI from 3.1.3. It is essentially the same as the one used with the Cubot 9+ mobile phone (3.1.3) but works with touch and voice commands.

Once the glasses are turned on, there are two ways to wake them up from sleep mode. Either by tapping once on the white body (the touchpad) or by briefly tilting the head back up to 35 degrees in reference to the normal head position for a fraction of a second.

After the wake-up, the interactions with the GUI can be performed via touch gestures and/or voice commands. Instructions issued via voice command require a sequence of words; the user needs to say *OK Glass*, followed by the name of the menu or instruction. On the other hand, with the touchpad, the Google Glass can detect a tap over it as a click and any swipe motion with one or two fingers. The GUI architecture is described in Fig. 3.10.

Once the Google Glass is initialized and the main screen (Fig. 3.8a) appears, the user needs to say “ok glass” + “Voice controller” meaning that the application *Voice controller* is the one requested to be loaded. If the user decides to use touch gestures instead, after the main screen is presented the user only needs to tap once to go into the Google Glass main menu and search between the applications for the icon corresponding to the designed GUI (Fig. 3.8b). As soon as the *Voice Controller* application is loaded, the glasses start searching for available devices. As there might be multiple different bluetooth devices, a filter checks specifically for e-pucks. If no robot is found, the screen remains black but the glasses keep searching. When a robot is detected, its particular information appears on screen. If more robots are discovered, they are stored in an array of screens that can be scrolled with touch gestures or with a mixture of voice commands and head movement.

Figure 3.9 shows a set of screenshots of the different menus and options of the *Voice Controller* application. Initially, the user needs to select a desired robot. Then the *Main* menu for disconnected robots is presented (Fig. 3.9a). By choosing the *Select* instruction, users can go back to the list of robots that were detected by the Bluetooth device. In this list the user can see all the detected robots, their ID and their connection state (Fig. 3.9b). They then can select a robot via the *Accept* instruction. Once a robot is selected, a connection to it can be selected via the *State* menu (Fig. 3.9c).

Once connected, the Google Glass automatically updates the options in the robots list (Fig. 3.9d) and the *Main* menu (Fig. 3.9e). Similarly, the *State* menu changes, depending on the current status of the robot. The updated options in the *State* menu are the same as in section 3.1.3, except for some minor changes: The *Commands* menu and the *Sensors* menu are now located in the *Main Menu*.

When a user achieves connection to a robot the initial state will be as "active". The current *State* menu (Fig. 3.9f) gives the user the option to put the robot in "Stand-By" mode. When in "Stand-By" mode the *State* menu changes depending on the parameters of the robot. If the robot is configured as "Swarm agent" the *State* menu looks like in Fig. 3.9g. At this point the user can change the robot into "Leader" mode or into "Active" mode. On the contrary, if the robot is configured as "Leader" the *State* menu appears as depicted in Fig. 3.9h. Similarly, the user can modify the robot mode into "Swarm agent" mode or "Active" mode.

Finally, the *Commands* menu (Fig. 3.9i) and the *Sensors* menu (Fig. 3.9j) contain direct commands for the robot. They will only appear in the *Main* menu when the robot is in "Leader" mode. Through them, the user can control the motion of the robot and obtain instant sensor readings from the robot IR sensors.

3.2 Experimental Setup

This section explains and analyses the selected activity[2] that was used for testing the leader interaction scheme. The task was the same as in [21] using the same algorithm for the occlusion-based cooperative transport controller also detailed in [21]. It consists of a robot swarm pushing an object towards a goal position.

[2] A published version of this study was presented in the conference TAROS 2016.

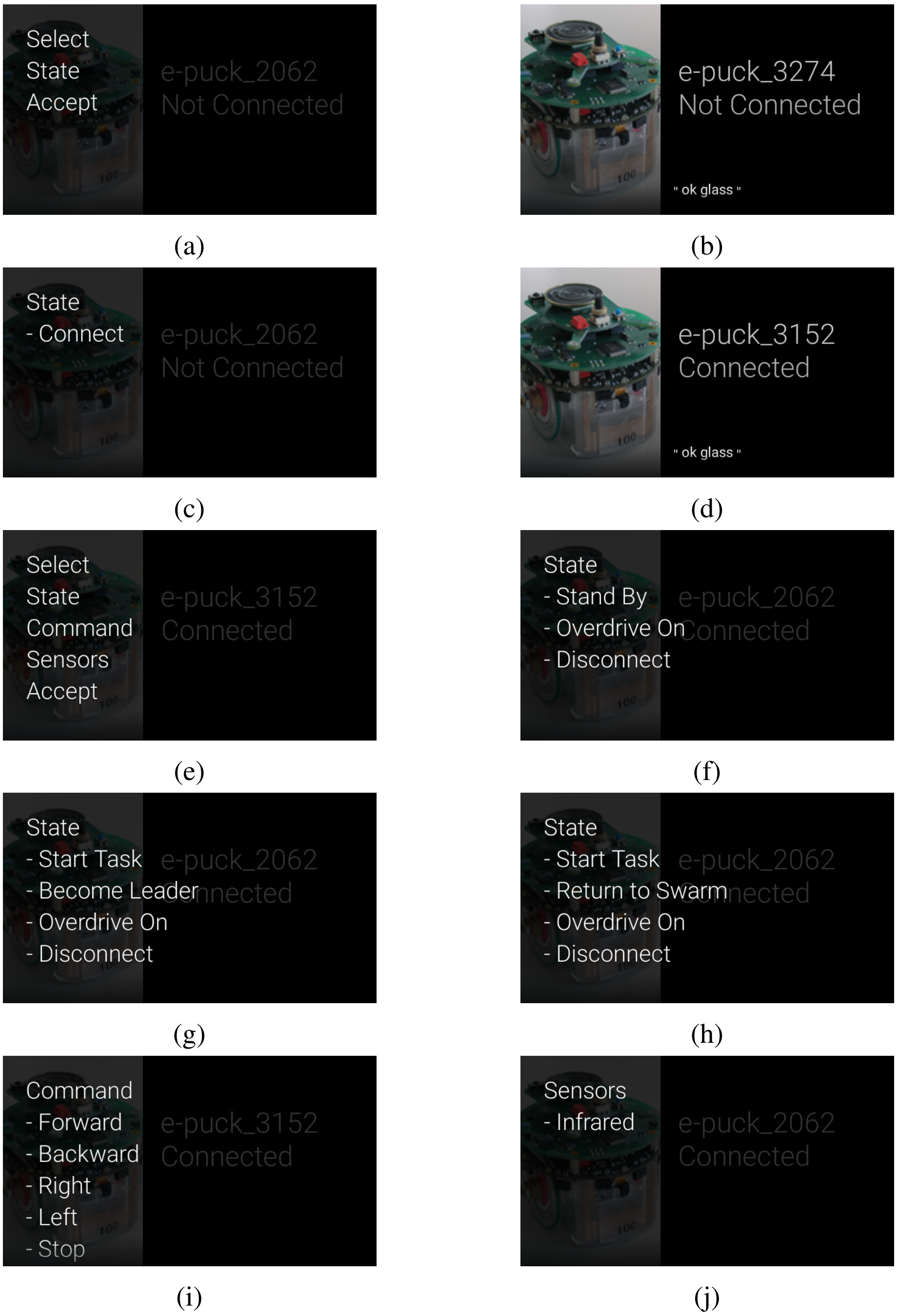


Fig. 3.9 Snapshots taken from the Google Glass interface: a) Disconnected "Main" menu. b) Disconnected Selection list. c) Disconnected "State" menu. d) Connected list. e) Connected "Main" menu f) Connected "State" menu while active. g) Connected "State" menu during "Stand-By" as swarm unit. h) Connected "State" menu menu during "Stand-By" as leader. i) "Commands" menu. j) "Sensor" menu .

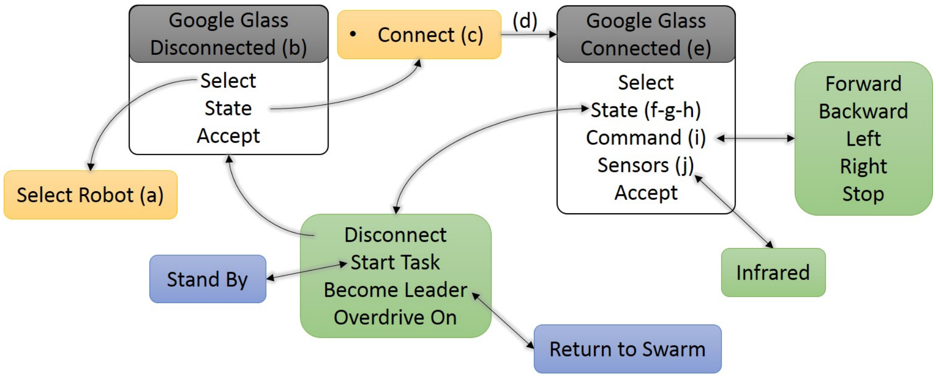


Fig. 3.10 Diagram explaining the architecture of the Google Glass software. The letters next to the instructions and arrows make refference to the screenshots presented in Fig. 3.9.

The experimental environment was a rectangular arena of 400×225 cm. It had two obstructing walls, each of 112 cm side length, which is half of the arena width. An example of the described environment is shown in Fig. 3.12a. The object to be transported was a blue cylinder of 42 cm diameter as seen in Fig. 3.1b. It can be seen that the object is taller than the robots and also considerably bigger in size as well as heavier. The partial goal for the swarm is the leader robot and is controlled by a human operator. The initial positions of all the elements can be seen in Fig. 3.11.

The robots initially move randomly through the environment, avoiding walls and each other using their proximity sensors. Once a robot detects the blue object with its camera, it moves directly to it. As soon as the robot is in contact with the object, it performs one revolution over its own axis scanning the environment for the goal which is assumed to be of red colour. If the goal is not visible, the robot turns its orientation pointing to the object and pushes it for a fixed duration. Afterwards, the robot will repeat the process starting from the scanning for the goal to the pushing of the object. In case the robot is next to other fellow robots also pushing the object, then the repeat scan is not necessary and is skipped. Otherwise, it follows the object's perimeter and approaches it from a different angle. Full details of the controller are reported in [21].

The objective of the swarm is to push an object from a starting position of the arena to another. However, there is no direct line between the starting point and the desired finish point as the arena has some obstacles between them. The human operator controls the leader attempting to influence the swarm and help it manoeuvre the object through a clear path. Because of this, the objective of the operator is to help the swarm transport the blue object from one corner to the other, avoiding the wall obstructions. The operator and the leader

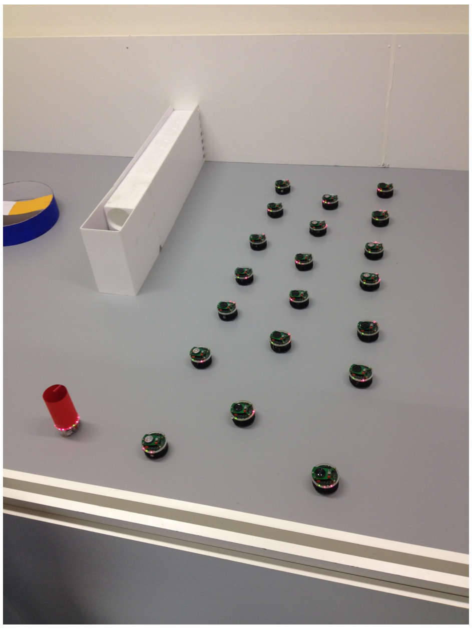


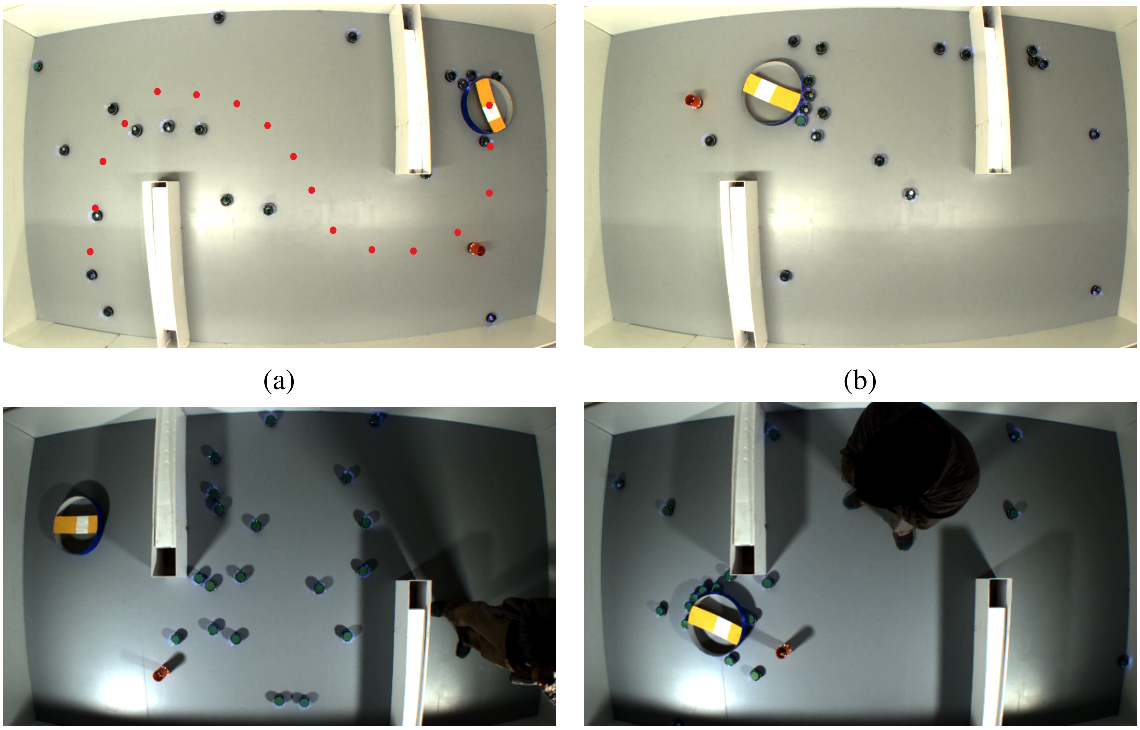
Fig. 3.11 Initial positions of the swarm robots, the leader robot and the object for every trial.

robot interact via the portable device. It could either be the mobile phone (*Cubot 9C+* as seen in Fig. 3.4) or the Google Glass (Fig. 3.7).

A set of multiple trials were performed. Initially, the object to be transported was put in one corner (either top right or bottom left in Fig. 3.12a). In all the trials, 21 e-puck robots were used, from which 20 of them were acting as working robots (searching and pushing the object) and 1 of them as the leader. The leader robot is equipped with a red cylinder to be recognized by the other robots as a goal. It acts as the dynamic goal in order to lead the transporting motion of the swarm. In addition, the leader robot activates the red LEDs along its perimeter. To make the appearance of the working robots more uniform, they were fitted with a black “skirt” around their body (Fig. 3.1b).

3.3 Discussion

In this chapter we proposed that an operator with full awareness makes use of one of two hands-free devices (the Cubot C9+ mobile phone and the Google Glass) as tools to gain control over a robot swarm in the context of a cooperative transport task. A series of experiments were performed where it was possible to dynamically modify the goal to which the object was being transported, enabling the system to negotiate through obstacles as



(a) (b) (c) (d)

Fig. 3.12 A human operator, (in (a-b) outside the captured region) wearing the Google Glass, guides a swarm of robots that is transporting a circular object through an arena with obstacles. The swarm consists of 1 leader robot (in red) and 20 follower robots. a) Red markers indicate the example trajectory of the object throughout a trial. b) Four follower robots push the object towards the position of the leader. c) Four follower robots pushing the object while the human operator is present in the arena. d) The human operator directs the leader robot to a convenient position at the beginning of the trial.

the operator had direct visual contact with the robots and the object throughout the trials. This feedback helped the operator to manoeuvre the leader robot at an appropriate pace, in response to the object's displacement. In all trials, the human operator was able to lead the pushing swarm along a trajectory using one of the portable devices.

Fig. 3.13 shows some screenshots of one of the recorded videos from the trials. The video is available at [154]. A documentary, show-casing the experiment, featured in the *Daily Planet* program of the Discovery Channel in 2015. In this particular example the human operator was communicating with the leader robot through the Google Glass.

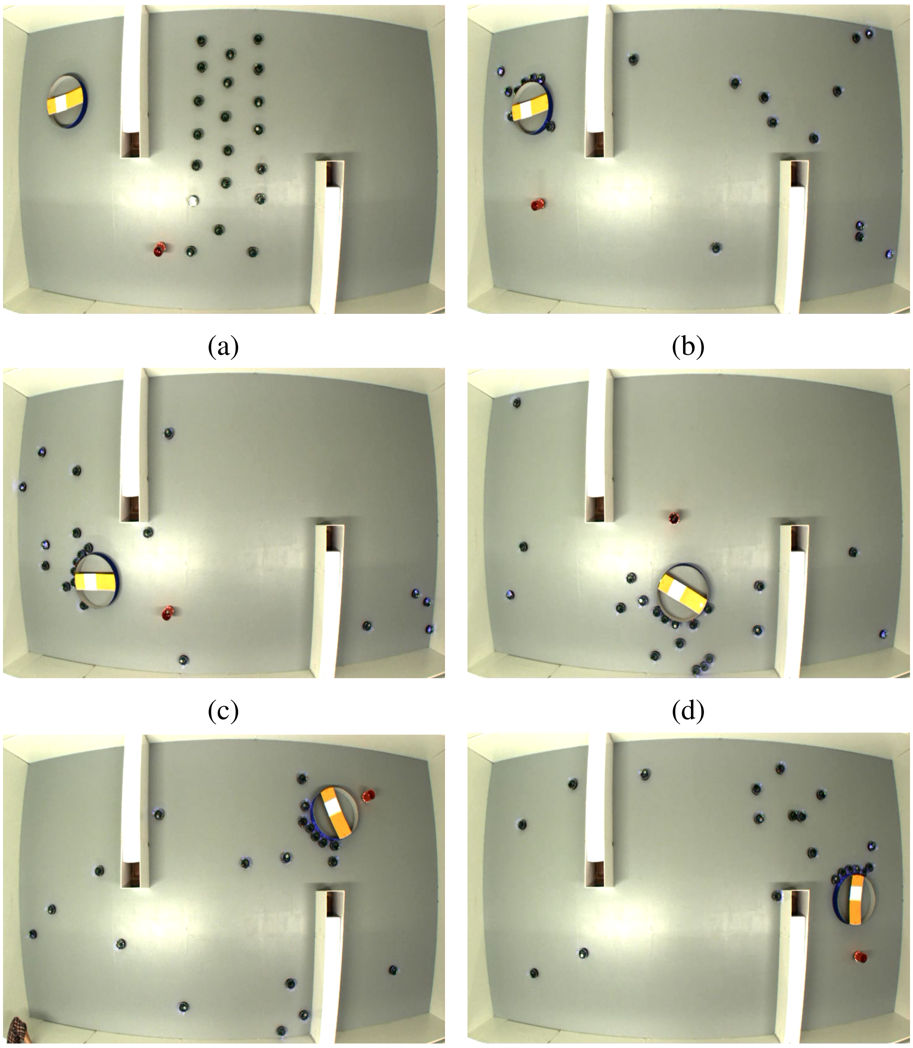
Through physical experiments we demonstrated that the operator's interactions resulted in a positive global feedback to the system. Our conclusions are supported by the fact that in every trial the object was pushed (transported) to the main goal position successfully.

The system had very low communication requirements for the robots. They did not need to explicitly communicate with each other, and the operator only communicated with the leader robot via the portable device using simple commands. Yet, the operator had enough influence over the entire swarm, and was able to direct the collective force such that the object moved in the desired direction.

While the mobile phone gave the operator the ability to influence the swarm with a GUI, the Google Glass interface allowed the human operator to influence the swarm via either touch or voice commands. The mobile phone GUI proved to be stable and reliable, yet the operator had to lose visual contact with the leader robot when operating it. On the other hand, the voice command option turned out to be preferable by the operator. This option gave the operator the impression of having a more direct communication and allowed a hands-free interaction process. Nevertheless, despite the ease-of-use of the Google Glass interface, it presented some performance issues. The two main problems were overheating and poor performance of the in-built display. When the overheating reached critical levels, it had the potential of even blocking the Google Glass. Both problems caused delays on the response time from the leader robot when receiving new commands.

During the trials, the working robots were moving slower than the leader robot. This gave enough time for the human operator to react and command the leader robot. If the swarming robots were moving and/or reacting substantially faster, the reaction of the leader would need to be faster too. This would increase the complexity of the task for the human operator, especially if the need to direct the leader robot and avoid obstacles simultaneously was still there. An alternative approach for this problem would be to develop a semi-autonomous leader, which avoids obstacles by itself while getting high-level direction input by the human.

This chapter explored the interaction benefit that a swarm robot can have from a human operator. However, real world environments do not offer operators with full awareness benefits, on the contrary the real world constraints much more an operator than what has been studied. For this reason, within the next chapters, we focus on how an operator can interact with a swarm robot without having access to the bird's-eye view of the environment. In other words, by adding some constraints to the operator's situational awareness.



(a) (b)

(c) (d)

(e) (f)

Fig. 3.13 Snapshots taken from one of the trials through different instants: a) 0:00 minutes. b) 2:00 minutes. c) 5:20 minutes. d) 7:30 minutes. e) 10:00 minutes. f) 13:00 minutes.

Chapter 4

Human-Robot Swarm Interaction with Limited Situational Awareness

In this chapter we present a study which considers further restrictions to the interaction protocol used in Chapter 3. Now, the human operator will not have access to the global positions of the robots at any moment. This limited situational awareness (SA) state will directly affect the way the operator receives any feedback. This led to the investigation of how to utilize local sensory information effectively. The only way the operator could receive feedback about the swarm state was through local sensor information from the leader robot and from its local cluster. This is more in line with the nature of distributed systems in which global state information can be difficult to obtain. In addition, this simulates in a better way a real-world scenario.

An example of such scenarios can be found in search and rescue missions, where keeping visual contact with each member of the swarm would be infeasible. For the experiments of the study presented in this chapter, a 3-D computer simulation environment with a conceptual robot were used.

The structure of this chapter is as follows. Section 4.1 presents the formulation of the problem (4.1.1) followed by an introduction to the virtual robot platform and the simulation software (4.1.2). In section 4.1.3 the swarm behaviours are explained and then section 4.1.4 introduces the developed user interface. Section 4.2 presents the experimental setup and the participant classification. Finally, in section 4.3 we present the results followed by a final summary (4.4).

Fig. 4.1 The e-puck miniature mobile robot in the simulation.

4.1 Methodology

4.1.1 Problem Formulation

We now focus on the study of the interaction between a human and a swarm of robots within a distributed scheme. In this scheme, a human operator interacts with a swarm of robots solely through a GUI. The operator can establish contact with any random robot of the swarm at a time and modify certain actions and/or parameters of it and its local cluster. In consequence, this chapter focuses on the interaction performance and limitations with SA constraints.

This study explores the performance that a human operator achieves with restricted SA while supporting a robot swarm in the execution of a task. The task from the previous chapter would be too complicated and time consuming for conducting a comprehensive study with human participants. For this reason we decided to change the task to a simpler one. The selected task was similar to the one presented by Gauci *et al.* [23] where a swarm of robots are required to aggregate into a single cluster within a given time period. By default, the robots execute the aggregation behaviour also presented in [23]. Unlike [23], we consider environments with obstructions and robots which have limited range sensors, both of which can prevent aggregation. An operator with full birds-eye view would be able to support the swarm to overcome the obstruction problems. However, we focus on the impact that the operators performance suffers when they are restricted of the global view (the birds-eye view).

(a) (b)

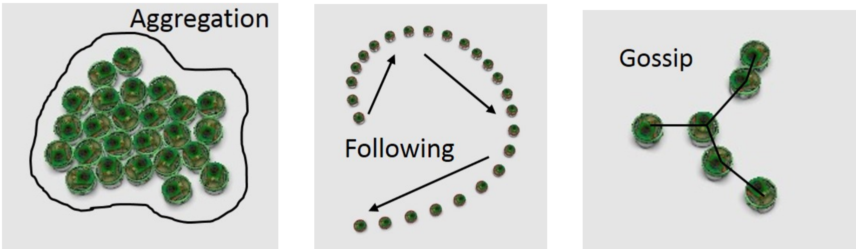
Fig. 4.2 a) Image taken by an e-puck robot in the Enki simulator environment. b) Another image with a section of a wall in it.

4.1.2 Robot and Simulation Platform

In this study we used an open source physics simulator. It is known as “Enki” and was developed by S. Magnenat and collegues at EPFL [18]. Enki treats the kinematics and dynamics of rigid objects in two dimensions. Space is represented continuously and Enki automatically resolves collisions among objects. Physics calculations are updated 625 times per second and the robots execute their control cycle every 0.08 s.

The robotic platform that we used was the same (the e-puck [92]) as in Chapter 3 but in its virtual representation mode within the *Enki* environment. Enki has a built-in model of this miniature mobile robot as shown in Fig. 4.1. The robot is represented as a disk with a diameter of 7.4 cm and a weight of 152 g. As a differential wheeled robot, each wheel can move backward and forward at different speeds with a maximum of 12.8 cm/s.

Each virtual robot can simulate the color camera, providing a horizontal field of view of 56 degrees and was limited to a maximum distance range of 150 cm. Fig. 4.2 shows two example frames as taken by a robot in the simulation environment. In addition, we assume that the robots can use part of their camera as a binary sensor. This sensor is mainly used to detect other robots in its direct line-of-sight. The sensor value can be $S = 1$ if it detects another robot and $S = 0$ otherwise. Likewise to the real robot, the virtual model also has eight infra-red (IR) sensors distributed around its body and a Bluetooth communication device. These sensors allow the human operator to interact with the virtual robots and also to receive real-time feedback.



(a) (b) (c)

Fig. 4.3 The three available algorithms: a) Aggregation. b) Follower. c) Gossip.

4.1.3 Swarm Behaviors

The human operator has the option to dynamically change the leader robot behaviour by switching between multiple algorithms. These changes are relayed between locally connected robots causing a change in their behaviour as well and consequently, of the locally connected section of the swarm. When a robot executes a particular algorithm, we refer to that algorithm as being in the corresponding mode (e.g. aggregation mode). Three algorithms were available for the operator to choose. They are presented below:

Aggregation Algorithm

Algorithm 1 Aggregation Algorithm

```
while true do
    if RobotDetected then
        Velocity pair (1,-1)
    else
        Velocity pair (-0.7,-1)
    end if
end while
```

The Aggregation algorithm is shown in Algorithm 1. It is identical to the one reported in [23]. Each robot is able to detect whether another robot is in its direct line of sight or not. It maps this binary sensor input onto a pair of constant wheel velocities. For simplicity we state the velocity values after scaling them to $(-1,1)$. If another robot is detected, the velocity pair is $(1,-1)$; the robot thus turns clockwise on the spot. Otherwise, the scaled velocity pair is $(-0.7,-1)$; the robot thus moves backward, following a clockwise circular trajectory. As shown in [23], this simple algorithm leads to the overall aggregation of the swarm (Fig. 4.3a), provided the sensing range is sufficiently large and no obstacles are present in the environment.

Follower Algorithm

Algorithm 2 Follower Algorithm

```
while true do
    if RobotDetected then
        Velocity pair (1,1)
    else
        Velocity pair (1,-1)
    end if
end while
```

The Follower algorithm is shown inAlgorithm 2. It uses the same line-of-sight sensor and reactive control architecture as the aggregation algorithm. The wheel velocity constants are however different. If another robot is perceived, the robot moves straight forward $(1,1)$, attempting to approach the detected robot; otherwise, the robot rotates anti-clockwise on the spot $(-1,1)$. This causes the robots to follow each other in a linear fashion (Fig. 4.3b). The robots follow the first robot they detect, so there is no particular leader. However, if the operator takes control of one robot, the rest will follow it as a temporal leader.

Gossip Algorithm

The gossip algorithm allows the operator to take “administrative" control of the swarm that is locally connected. It provides some tools for the operator to attempt to regain some SA. The algorithm’s pseudo-code is presented in Algorithm 3. It prevents the selected robot from changing its position, yet the operator has control over the orientation (Line 6) and can monitor the sensors activity (Line 9). The robot requests all other robots in its neighborhood to stop. These requests get relayed, so that all locally ‘connected’ robots finally stop. A graphical representation of this relay action can be seen in Fig. 4.4. Only in this mode the operator is able to obtain a count (Line 12) of the connected robots. To release the robot from the gossip mode the operator can switch algorithm (Line 15) or release the robot, the later returns the robot to the last algorithm.

More details of the implementation of this algorithm and the count operation can be found in Appendix B. It is also important to mention that the Gossip algorithm was the first

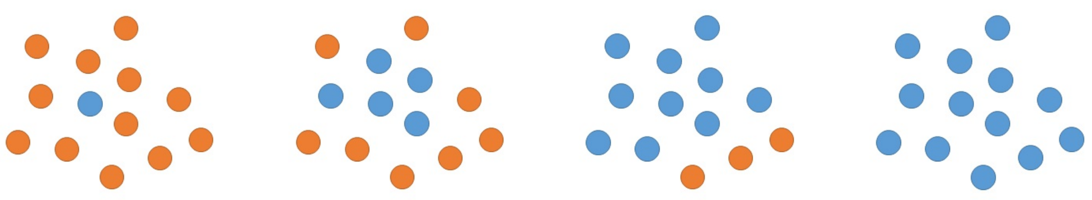
Algorithm 3 Gossip algorithm

Stop all activity
Relay *Gossip* Command
Input:
switch *Command* **do**
    case *Rotate*
        Rotate orientation
        goto *Input*.
    case *Sensors*
        Sensors activity
        goto *Input*.
    case *Count*
        Cluster count
        goto *Input*.
    case *Algorithm*
        Switch Algorithm
        Relay *Command*
End;

step to the development of a more complex algorithm named the *Management Algorithm* and more details of this upgraded version can be found in [155].

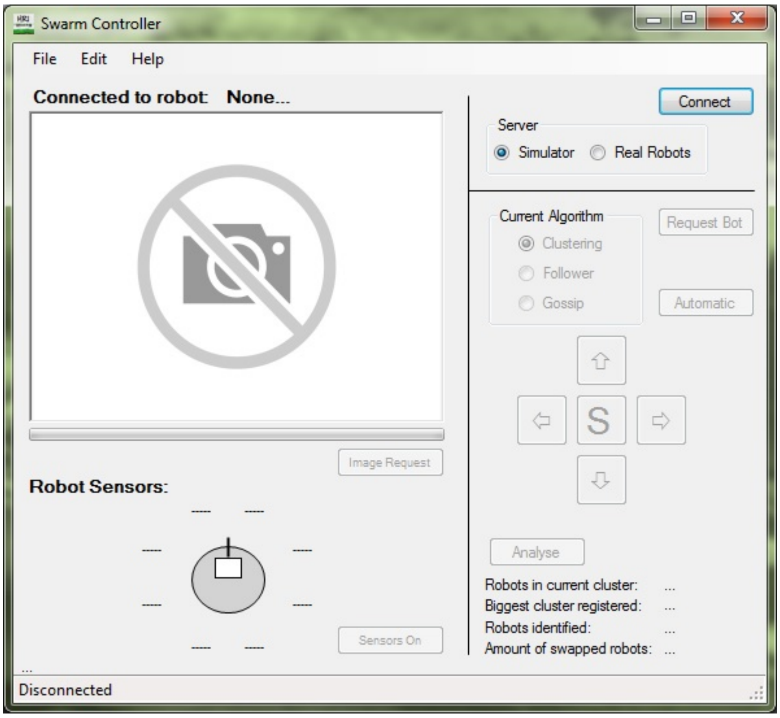
4.1.4 User Interface

The interaction between the human operator and the swarm robot occurs through a keyboard and the GUI shown in Fig. 4.5a. The GUI was designed to work with either real robots or

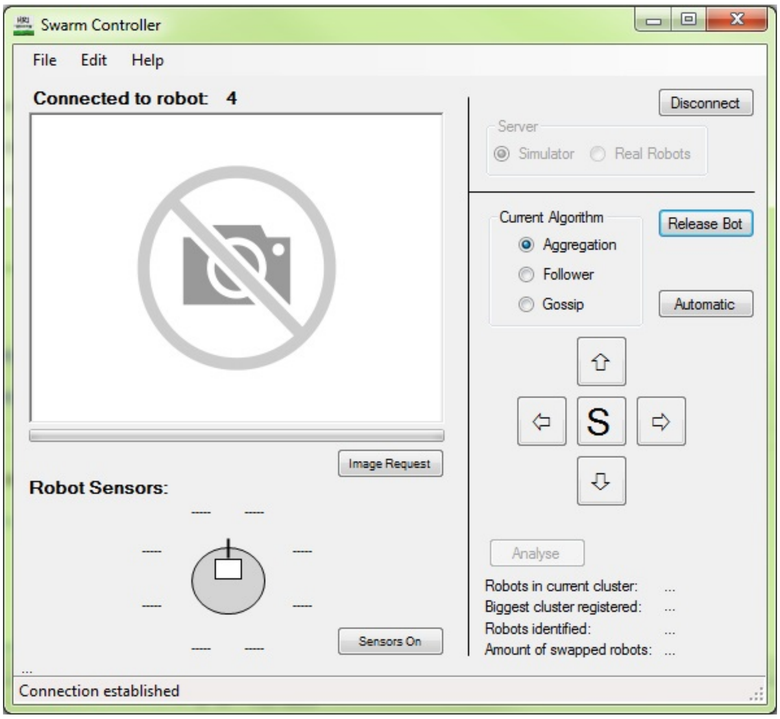


(a) (b) (c) (d)

Fig. 4.4 The gossip algorithm relays a command to the nearest neighbours. a) The leader robot receives a new command from the operator. b) It relays it to the nearby robots. c) The nearby robots relay the message to their locally connected robots. d) Finally all robots have relayed the message to all the locally connected robots.



(a)



(b)

Fig. 4.5 a) GUI that the participants used in the human-robot swarm interaction study. b) GUI when connected to a robot.

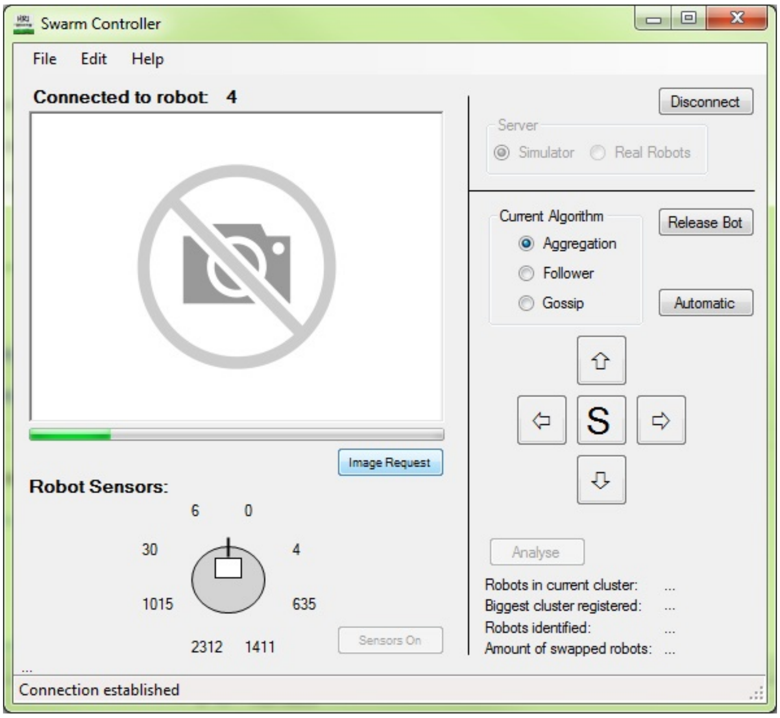
with simulated robots. When working with real robots, the GUI connects through a bluetooth connection directly to a detected available robot. In this case, to establish a connection with the Enki simulator environment, the GUI connects through a local network. By pressing the "Request Bot" button in the GUI, the human operator can start a connection with one robot at a time. Every time a robot is requested, a random robot is chosen, each with the same probability, and a connection is established.

Once connected (Fig. 4.5b), the operator is acknowledged of the robot's (unique) identification number and of its currently active algorithm. From this point, the operator can receive feedback from the robot sensors or influence its motion/behaviour. The operator has three options to obtain information from its sensors:

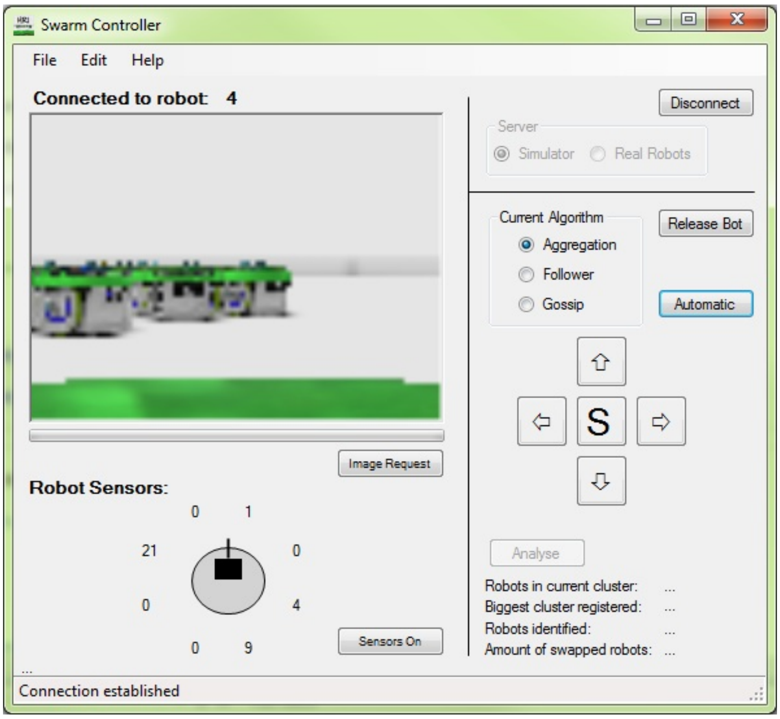
- *Monitoring of the proximity sensors*: With the "Sensors On/Off" button, the operator can activate and/or deactivate the sensor feedback. When activated, the sensor's data is updated periodically. The operator is then able to see the raw values of the proximity sensors. Each value is positioned in the location of the sensor that is reading each value. An example of the sensor readings can be seen in Fig. 4.7a.
- *Monitoring of the binary line-of-sight sensor*: This sensor's data is also updated periodically. It is activated at the same time as the proximity sensors and with the same "Sensors On/Off" button. When activated, the operator can observe the status of the binary line-of-sight sensor, indicating whether another robot is detected. The graphical representation is a square located inside the circle representing the robot. Fig. 4.7a shows an example of this sensor been activated (when a robot is detected, the square turns into black color).
- *Requesting an image of the camera*: By clicking on the "Image Request" button, the user is shown a 80x60 pixels snapshot as taken from the robot's camera. An example picture is shown in Fig 4.2a. A 1 s average delay occurs between the request (Fig. 4.6a) and display of such image (Fig. 4.6b). This delay emulates the time that the Bluetooth protocol would take to transfer the data.

To keep the simulation of the bandwidth limitations as real as possible, the requesting of images and the monitoring of the proximity sensors can only be selected one at a time. For the operator to be able to influence the robots, the GUI provides two options:

- *Motion control*: There are five buttons on the right side of the GUI (Fig. 4.5b). Four arrows and one center button with a big "S". With these, the operator can issue basic

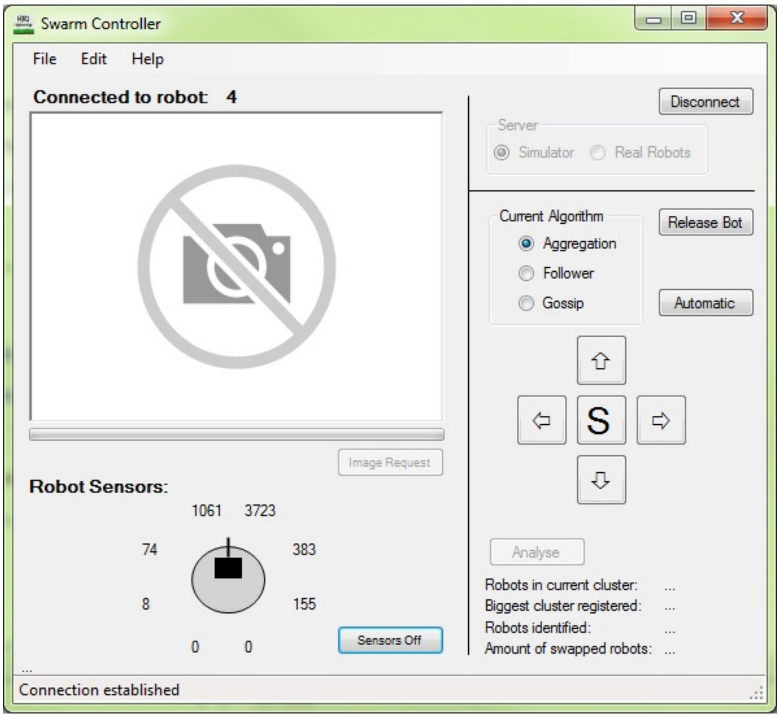


(a)

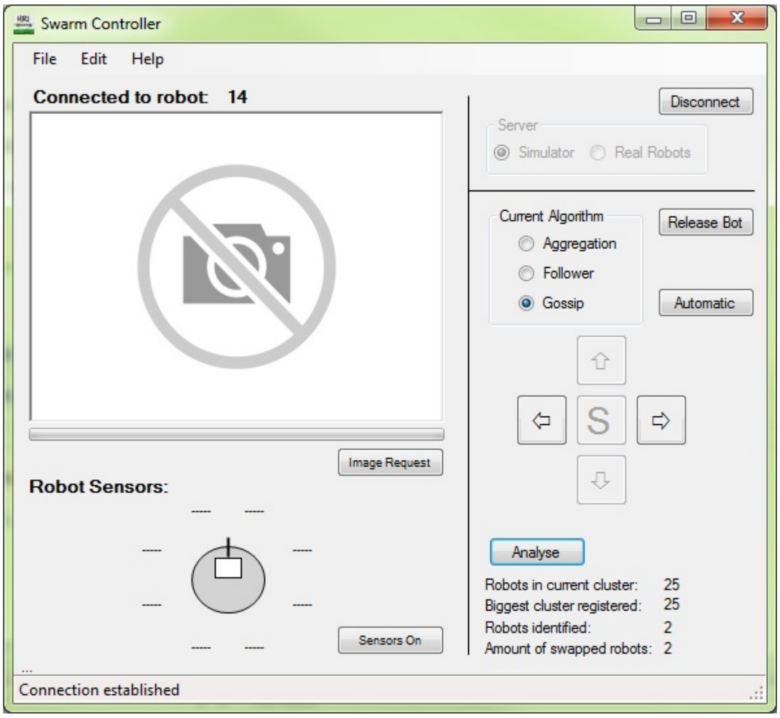


(b)

Fig. 4.6 a) The GUI presenting the loading process of the image requested. b) The GUI presenting the requested image one the loading was completed.



(a)



(b)

Fig. 4.7 a) The GUI presenting the IR sensor readings from the leader robot. b) The GUI presenting the unit count of the local cluster from the leader robot perspective.

motion commands to the currently selected robot. These are forward, backward, rotate left, rotate right and stop. Nevertheless, when in gossip behaviour, the forward, backward and stop buttons are disabled as in Fig. 4.7b.

- *Behaviour control*: The operator can change the behaviour that is being executed on the selected robot. This change is broadcast from the selected robot to all the locally connected robots (as in Fig. 4.4). Therefore all robots in the network change their behaviour as well. The operator has the possibility to swap between *Aggregation*, *Follower* and *Gossip* algorithms. When disconnecting from a robot, the behaviour which is currently executed remains active. However, it is not possible to disconnect from a robot while it is in gossip mode. This was implemented to avoid robots from being left in a static position.

When a robot is in gossip mode, the operator has the possibility to request for further information with the "Analyze" button. Consequently, the operator is shown a count of the robots that were counted in the local cluster. The GUI also keeps track of the biggest cluster encountered so far, the number of distinct robots the operator had interacted with and the number of times the operator had requested a new robot. Fig. 4.7b shows an example of the information been displayed.

4.2 Experimental Setup

In this study[1] we investigated the impact of allowing only local sensory information to be retrieved. During the experiment, the operator did not have a bird's eye view over the arena and was not provided with access to global state information. However, they were shown a map of the environment prior to the start of the experiment.

Figure 4.8 provides an overview of the simulation environment used for this experiment. The robots operate in a bounded rectangular environment, which will be refereed to as the arena. The arena is of dimensions 400×300 cm and contains two 200 cm walls symmetrically arranged. They divide the arena in three equally sized areas joined only at the extremes. The walls are sufficiently tall to prevent robots at opposite sites from perceiving each other. More specific details concerning this experiment are provided in the following sections.

[1] A published version of this study was presented in the conference ANTS 2016.

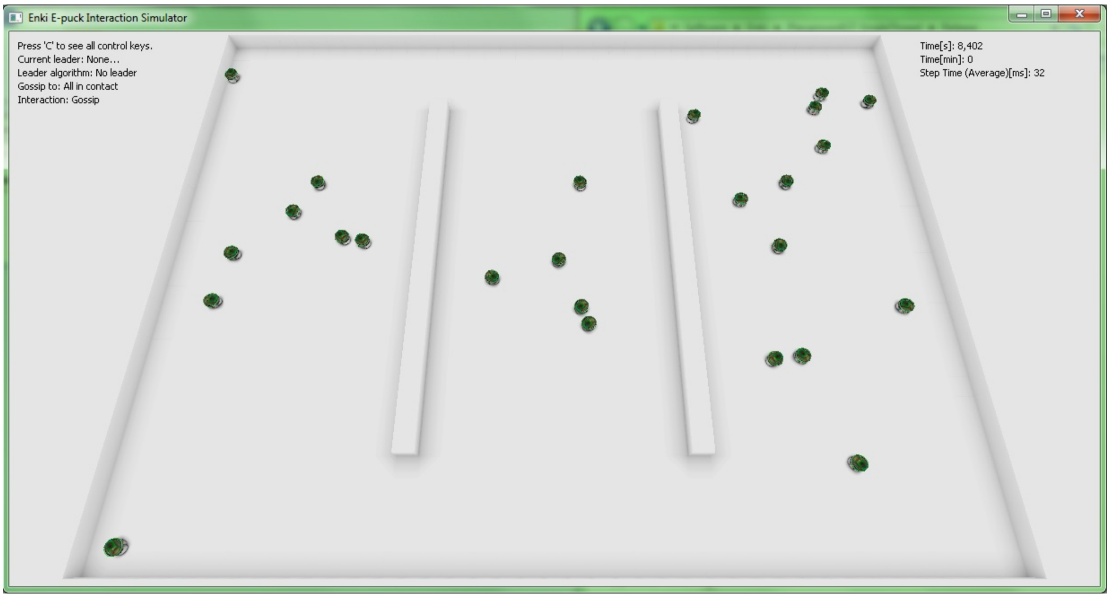


Fig. 4.8 Experimental arena with the robots positioned in specific locations.

The study received ethical approval by The University of Sheffield. All participants were students of the university and their age ranged between 18 and 39.

Experiment Introduction

Participants were given a 10 min presentation in which the mission was explained. It can be seen from Fig. 4.9 that the objective was presented as: *Your mission is to cluster as many robots as possible*. At the same time, participants had the opportunity to observe a snapshot of the simulation environment with an example of the initial randomly distributed position of the robots.

The presentation also introduced the participants to the three available swarm behaviours (see Fig. 4.3) and to the user interface (see Fig. 4.5a). Also, participants were informed that they would conduct three trials and that they would be in control of 25 robots. As a final note, participants were informed that each trial was going to last 10 min (600 s). The used slides are presented in the first section of Appendix C.

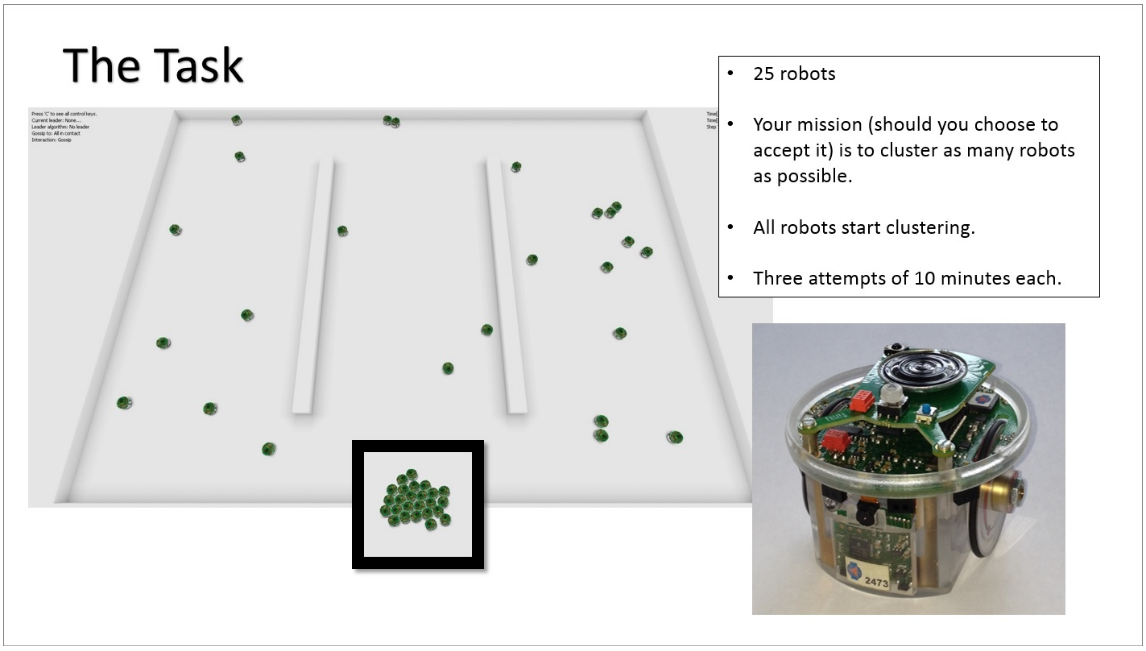


Fig. 4.9 A slide from the introductory presentation given to the participants. It presented the participants with the mission objective.

Participant Classification

The default group of participants is referred to as the *untrained participants*. They were not provided with the opportunity to test the system in advance of the experiment. Overall, data from 38 untrained participants was collected. However, three participants' data was excluded as they did not complete all three trials. Six further participants received training on the system prior to conducting trials. Three of these received 60 min training (five to six trials), these are referred to as *trained participants*. The other three participants were chosen from the developer team and received several hours of training; they are considered as *experts*.

Untrained participants were further assigned to one of two conditions at random:

- Blind-Blind-Blind (BBB): Participants of this group had no access to global state information (i.e., the bird's-eye perspective) during any of their trials. There were 19 participants in this group.
- Visual-Blind-Blind (VBB): Participants of this group had access to global state information for the entire duration of their first trial (referred to as VBB_V), but had no

access to that information during the second and third trials (referred to as VBB_B). There were 16 participants in this group.

Throughout all the trials, various sets of information from the robots and participants was recorded. Regarding the robots, the recorded information included the positions of each robot on average every 30 ms, total amount of clusters and every cluster size every 1 s. From the participants interactions with the interface, every command issued, image taken and cluster size count request was recorded including a time stamp of the event.

4.3 Results

In this sections the interaction actions and strategies taken by the participants are analysed.

4.3.1 Performance Metrics and Baseline Performance

This section provides the baselines to which the performance of untrained, trained and expert operators were compare to. The main performance metric is the number of robots in the largest cluster. A pair of robots is considered in close proximity if the distance between their centres is less than 15 cm. We consider then that two robots that are in close proximity belong to the same cluster. Moreover, if $\{a,b\}$ belong to the same cluster and $\{b,c\}$ belong to the same cluster, then the same holds true for $\{a,c\}$.

Table 4.1 Size of the final biggest cluster in trials with (i) no operator, (ii) no wall obstructions & no operator, and (iii) a virtual operator agent choosing commands at random. Each value corresponds to 10 trials with 25 simulated e-puck robots.

Baselines

	$NoInt.$	$NoWalls$	$RandomInt.$
Average [%]	42	92	48
Average	10.5	23	12
Std. Dev.	0.81	3.8	2.24

To establish a point of comparison, some baselines need to be defined:

- *No Interaction*: This was the performance of the swarm in the absence of any interaction with an operator. In other words, each robot of the swarm executes the aggregation algorithm for the entire duration of the trial.
- *No Walls or Interactions*: This was the performance of the swarm when aggregating in the absence of wall obstructions and interactions with an operator. These represent the ideal environment conditions as for the algorithm presented in Gauci *et al.* [23]. However the limited sensing range limitation of the robots was kept for this baseline.
- *Random Interactions*: This is the performance of the swarm when interacting with a virtual operator agent. The virtual operator was programmed to choose random instructions. These instructions were composed of all possible commands that human operators could apply during the trials. The model of the distribution of these commands was drawn from the record of the participants across all trials.

For each of the baseline performance strategies, 10 trials of 600 s were conducted. Table 4.1 shows the average size of the biggest cluster at the end of the trial. From this table it is possible to observe that random commands resulted in slightly better performance than no interactions but with a larger standard deviation.

4.3.2 Operator Performance

First we validated the efficacy of the swarm controls from the GUI. This was done using untrained operators that had access to real-time global state information (bird's-eye view) of the position of all robots. This was the first trial for the VBB group, which aggregated 90% of the robots. Their performance was as good as the 'no walls or interactions' baseline (as seen in Table 4.1). Through this trial operators were able to use the available controls to mitigate the shortcomings of the aggregation algorithm in the presence of obstacles.

However, when untrained operators became restricted from the real-time global state information (bird's-eye view), their performance was similar to an autonomous agent choosing random actions. All the blind trials of both groups of untrained operators (trials 1, 2 and 3 for BBB and trials 2 and 3 for VBB) did not perform significantly better than the random interaction baseline (two-sided Mann–Whitney test, p-values = 0.985 and 0.481)[2]. In their final trial, untrained participants aggregated 51% (BBB) or 59% (VBB) of robots

[2]Throughout this thesis, a 5% significance level is being considered.

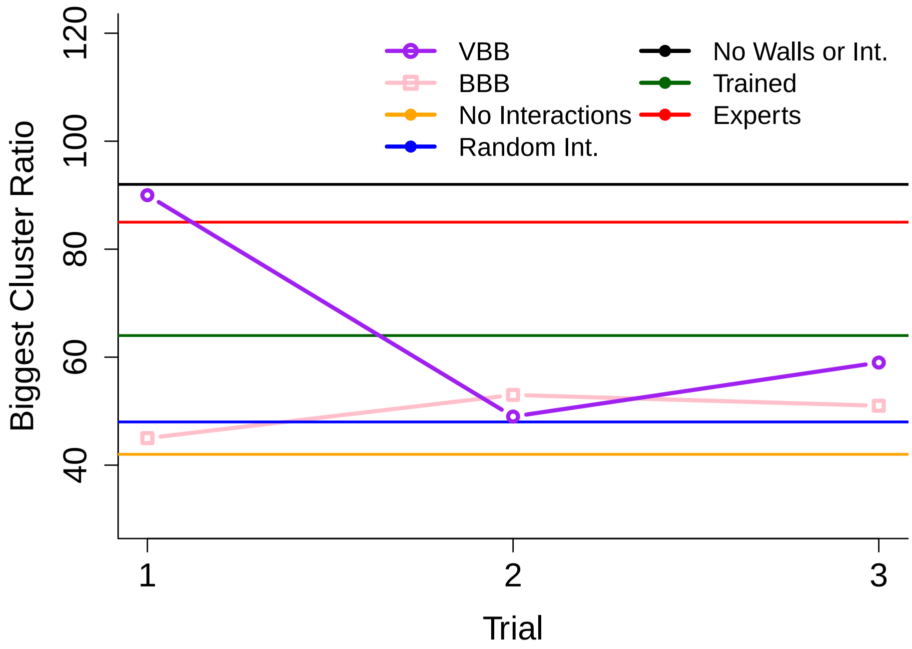


Fig. 4.10 The graph presents the percentage of robots in the biggest cluster at the end of the trial for each group of untrained participants (BBB and VBB). The performance of all baselines and the average performance in the last three trials of trained and expert participants are plotted as lines to provide a reference performance.

into a single cluster, an improvement over the no interactions baseline that aggregated 42% (two-sided Mann–Whitney test, p-values = 0.049 and 0.029). The percentages of each trial for the four groups can be seen in Table 4.2.

It is important to note that, for the virtual operator, the proportion of the types of instructions was identical to an untrained human operator but did not exploit any sensory information. Fig. 4.10 illustrates these comparisons and suggest that untrained operators have similar difficulties in exploiting local sensory information. When realizing a comparison between the blind trials of both groups, (BBB and VBB) it shows no significant differences in their performance (two-sided Mann–Whitney test, p-values = 0.215).

The VBB group served as a test group that suggests that untrained operators had a minimal learning effect from the initial trial with global state information. It further supports the conclusion that operator performance in blind trials was diminished due to a lack of SA rather than lack of planning. If it were due to a lack of planning the trial with global state information would be expected to have facilitated the learning of plans.

Table 4.2 The table presents the percentage of robots in the biggest cluster at the end of the trials with human participants.

Performance Metrics [%]

Trial	*BBB*	*VBB*	*Trained*	*Experts*
First	45	90	69	76
Second	53	49	65	80
Third	51	59	57	99

The trained operators group had the opportunity to perform five to six trials in the one hour of training. The expert group had more experience and understanding of the robot hardware, the task and the development of the experiment. Both groups were able to obtain significantly improved performance in their three test trials over the random interactions baseline (two-sided Mann–Whitney test, p-values = 0.029 and 0.001). They were able to aggregate 57–69% and 76–99% of robots respectively. From their experience, we were able to observe some learning of neglect benevolence patterns, this is further discussed in the next section.

It is important to notice that expert operators had a similar performance to untrained operators from the VBB group in their first trial (where they had access to full global state information). Also, they performed nearly as well as the baseline performance of the autonomous algorithm under ideal conditions, that is, without obstructions. Despite the dramatic drop in performance of untrained operators when removing access to global state information, the recovery of performance for trained and expert operators, with at least one hour of training, shows that learning does occur and suggests that the task is solvable. A closer look, in the following section, into the actions and strategies that were learned suggests that constant training and understanding of the robots' behaviour can help the operator overcome the SA constraints.

4.3.3 Interaction Analysis

A detailed history of the operators' actions was recorded throughout all the test trials. We analysed the distribution of the time that the operator spent interacting with the robots. It included only the activities involved in the three test trials for all participants (untrained, trained and expert operators). Fig. 4.11 shows this data grouped into three categories:

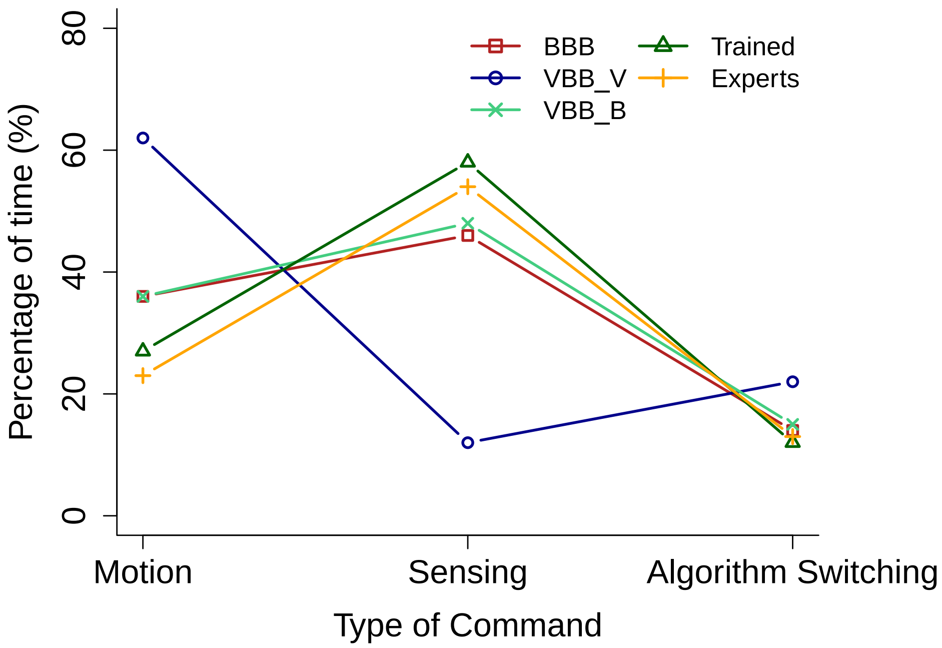


Fig. 4.11 The graph presents the percentage of time that operators spent through all trials executing a certain type of command. The subdivision of VBB_V represents only the first trial of the VBB group, which was the one where they had access to the bird's−eye view. The VBB_B subdivision represents the second and third trial of the VBB group where participants had no access to the bird's−eye view.

- *Motion*: Includes all the commands from the operator involving the motion of the robot: "Forward", "Backward", "Left", "Right" and "Stop".
- *Awareness*: Includes all the commands from the operator involving the use of the sensors of the robot: "Sensors On/Off", "Image Request" and "Analyse" (this last one, requesting a cluster count).
- *Behaviours*: Includes all the commands from the operator involving the switching between algorithms: "Aggregation", "Follower" and "Gossip".

As expected, untrained operators with access to the global state information (trial 1 in group VBB) rarely requested local sensory information. Their motion commands as well as their switching algorithms commands were more frequent as they had direct visual contact with the robots. Because of this, operators had instant feedback of the robots movements and were able to completely focus on their behaviours. However, when in the blind trials, untrained operators spent a larger proportion on obtaining sensor information attempting to recover some SA.

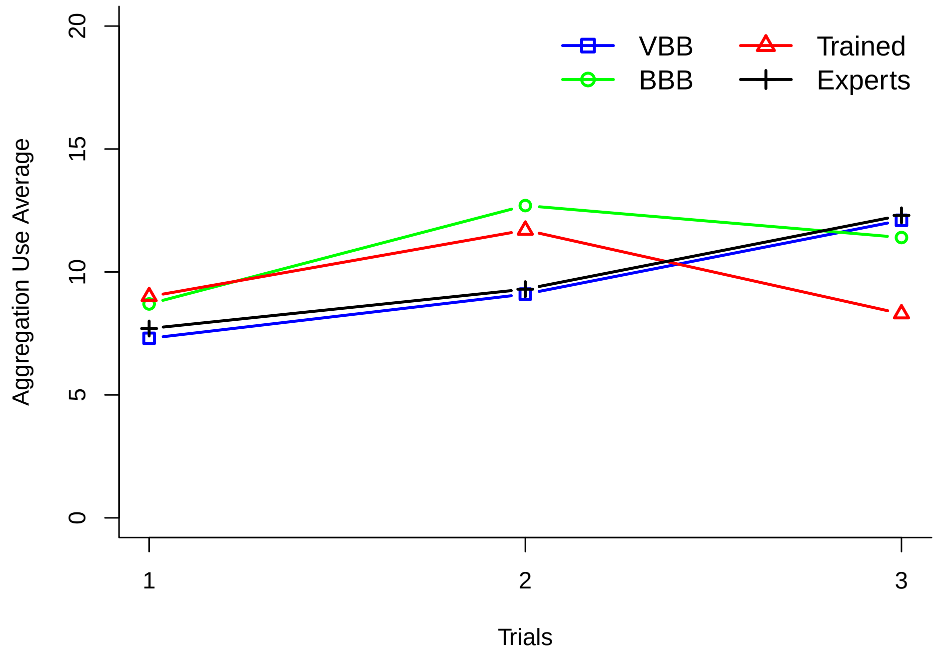
The key observation is found by comparing trained and expert operators to untrained operators. The improved performance of trained and expert operators seems to rely on acquiring more requests from the sensors while reducing the amount of time spent moving the robots. Given that the their time spent on motion commands is significantly less than for untrained operators with global state information, the efficiency of the motion commands for the former group was higher. This is likely where the training effect materializes.

The time spent switching algorithms turned out to be identical between all groups. However, the analysis of the amount of repetitions where participants request a change of algorithm through the trials presents some differences in their use. Figures 4.12 and 4.13 present the average use of each algorithm of every group through all the test trials. In addition, Fig. 4.13b presents a compilation of the use of all the algorithms also through all the test trials. It is important to note that "time spent" and "amount of repetitions" are different measurements. All participants spent the same time switching algorithms (see Fig. 4.11), but they could have requested different amount of switches during the same time. Each figure presents one of the available algorithms:

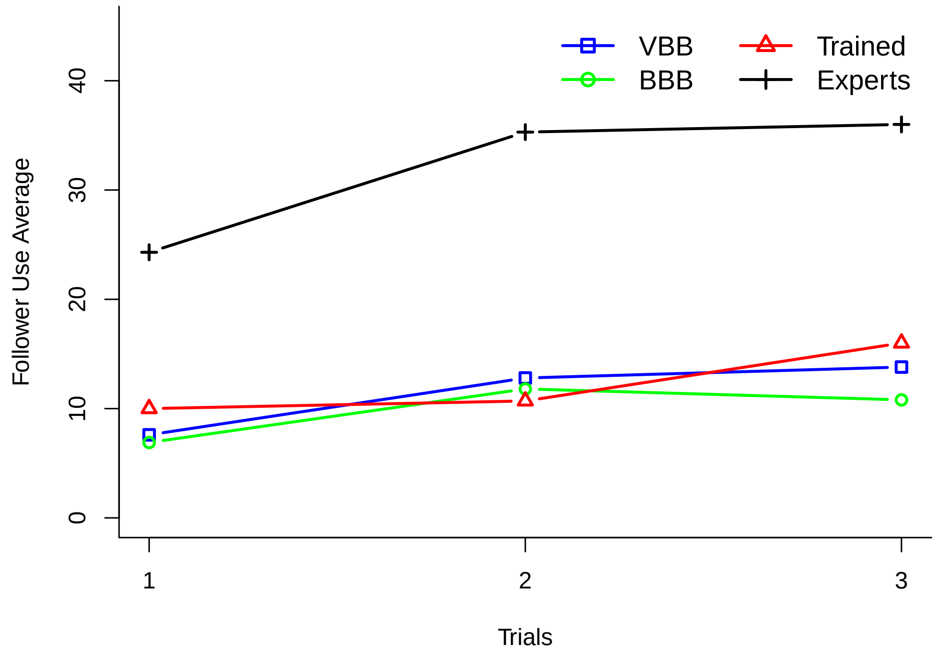
- *Aggregation*: Fig. 4.12a shows the average use of this algorithm through all the trials.
- *Follower*: Fig. 4.12b shows the average use of this algorithm through all the trials. It is evident that experts used this algorithm more frequently than the other groups.
- *Gossip*: Fig. 4.13a shows the average use of this algorithm through all the trials. It suggests that as the operator starts gaining experience, they start understanding the importance of this algorithm and its role in helping them acquire SA.

Finally, in Fig. 4.13b we can observe that, through all the trials, the expert operators used the switching of algorithms more. This suggests that they understood each algorithm better and used it for their advantage to influence more robots at the same time.

Overall, operators had access to swarm controls with which they were able to complete the aggregation task successfully when given global state information. When given only local information, however, untrained operators did not perform significantly better than random interactions. Nor did they exhibit a significant learning effect within three trials. Furthermore, operators that once were given global state information did not demonstrate improved performance on subsequent trials when being restricted to local information. This suggests no learning benefit from having observed the global dynamics once. However,

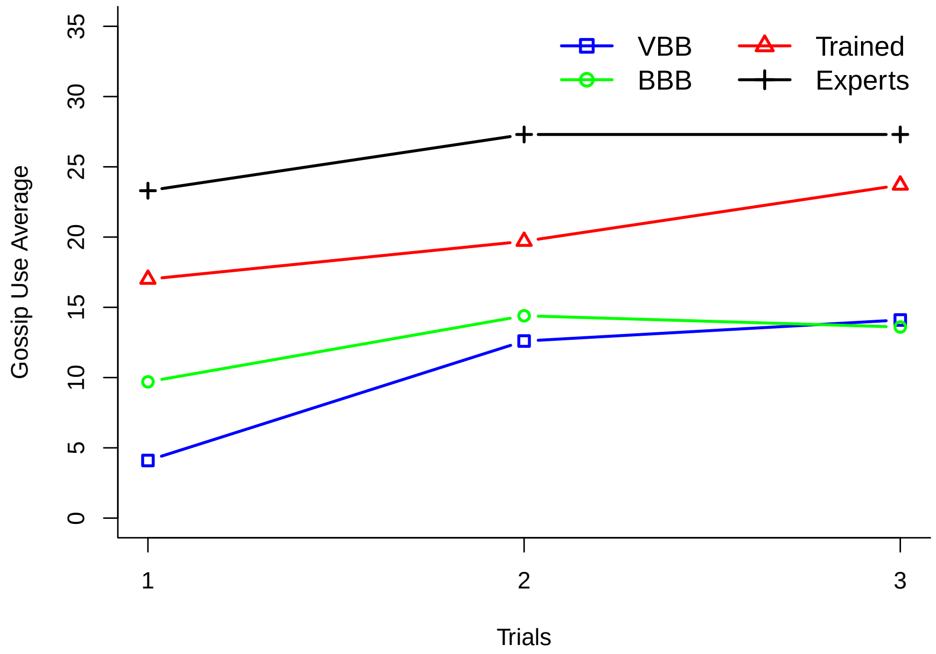


(a)

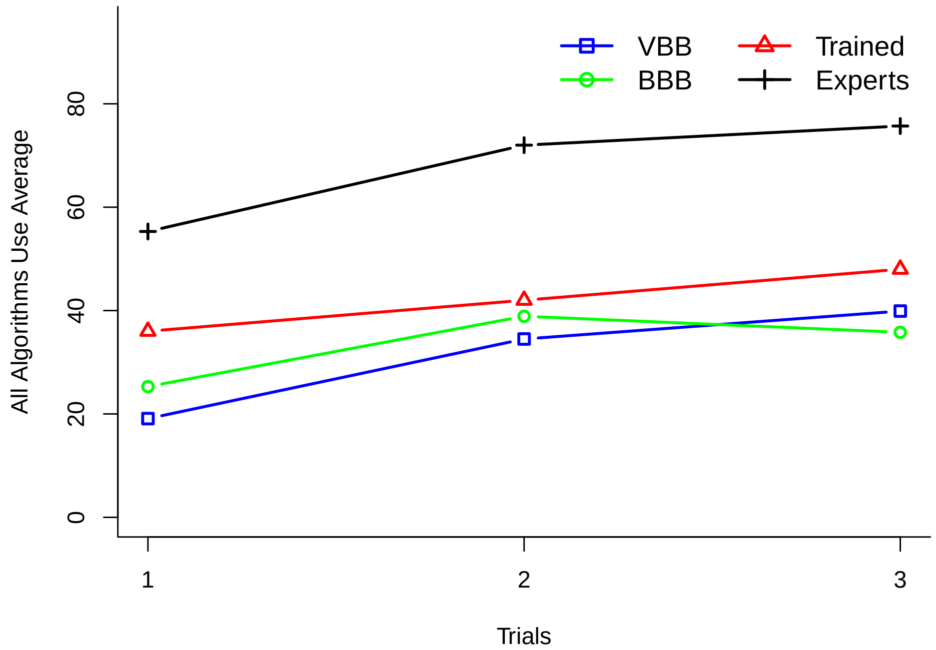


(b)

Fig. 4.12 These graphs present the average amount of times an algorithm switch was used in every trial, divided by group: a) Aggregation b) Follower



(a)



(b)

Fig. 4.13 These graphs present the average amount of times an algorithm switch was used in every trial, divided by group: a) Gossip b) All Algorithms together (Aggregation, Follower and Gossip) trained and expert operators were able to compensate the lack of global SA with increased requests for local sensory information while reducing the number of motion commands.

4.3.4 Demonstration Trials

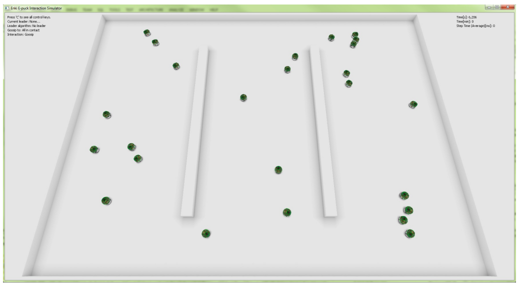
Through this study we investigated a distributed human-swarm interaction scheme in which operators have access to only local information while aiding a swarm in an aggregation task. To illustrate this more qualitatively, Fig. 4.14 and Fig. 4.15 show two different sets of example snapshots collections. They show two different trials taken from two different expert participants. They are a collection of snapshots taken in specific moments of the trials.

The first sequence is from Fig. 4.14 (a-g), it starts with the initial positions of the robots when they are randomly distributed through the arena (a). Because of the aggregation algorithm, the robots start grouping and forming three clusters (b). The operator then starts moving the right cluster to the center area (c). The operator finds the third cluster and guides it to the center area (d). Again, when the robots are in visual range, they attempt to group together (e). Finally, the operator is monitoring the process until the swarm reports a complete aggregation of the swarm (f).

The second sequence is in Fig. 4.15 (a-h). This sequence initial positions are also randomly distributed through the arena (a). Because of the aggregation algorithm, the robots start grouping and form three clusters (b). The circle formations appear when the operator switches the clustered robots to the follower algorithm. The operator then starts moving the right cluster to the center area (c). The operator finds the central cluster and leaves the guided robots to group in the center area (d). The operator finds the third cluster (e). The operator then attempt to lead the last group of robots to the center area (f). Again, when the robots are in visual range, they group together (g). Finally, the operator is monitoring the process until the swarm reports a complete aggregation of the swarm (h).

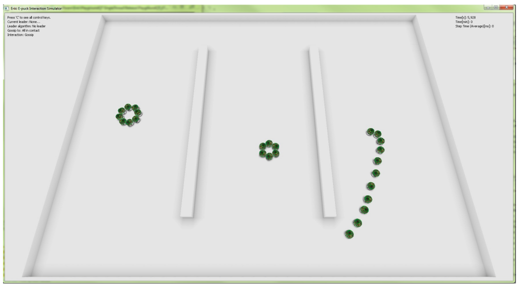
4.3.5 Operators Neglect Benevolence

In addition to varying the time spent on certain activities we observed a difference in the time the operators would wait to start interacting with the swarm. It was the time the operators waited at the beginning of the trials before performing the first interaction that was of interest. Fig. 4.16 shows the average times of that operators divided by group.

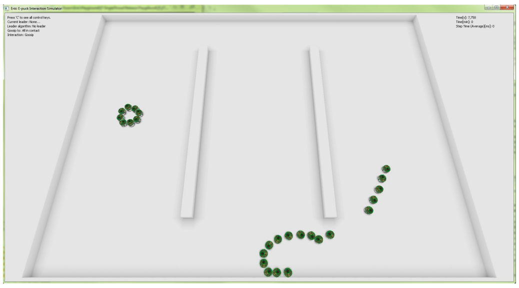


(a) Initial Setup

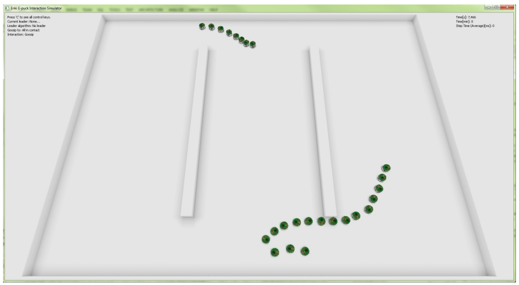
(b) After 240s



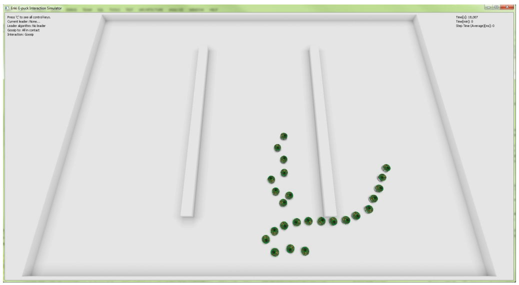
(c) After 270s



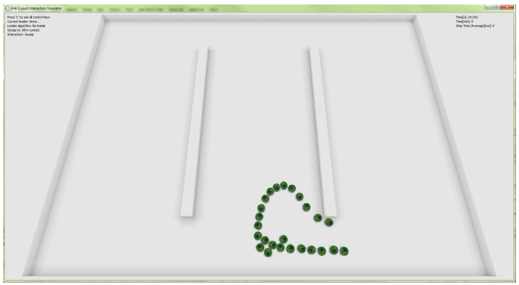
(d) After 300s



(e) After 360s

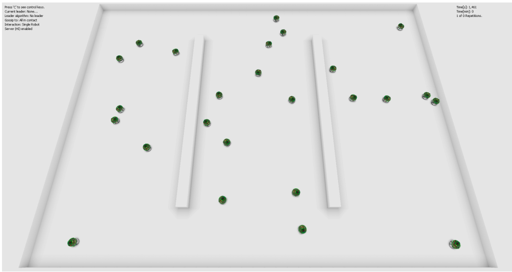


(f) After 480s

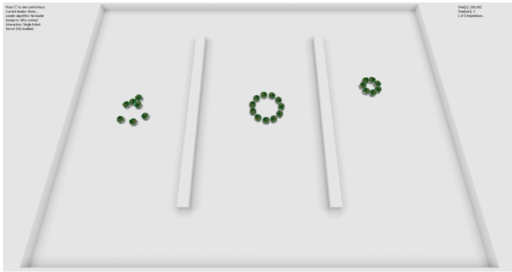


(g) After 480s

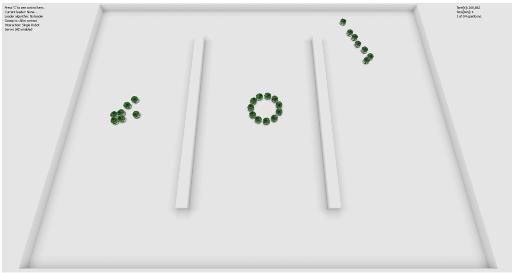
Fig. 4.14 Sequence of snapshots taken during a trial with an expert participant. The expert was not provided with the birds-eye view of the scene, which is depicted here. a) Initial positions of the robots. b) The operator waited fot the robots to aggregate and modified the behaviour of the left cluster to follower. c) The operator modified the behaviour of the middle and right clusters to follower and started moving the right cluster to the south. d) The middle cluster starts mixing with the right cluster. e) The operator guides the left cluster to the north gap. f) The left cluster starts mixing with the other cluster. g) All the robots are now locally connected and therfore clsutered.



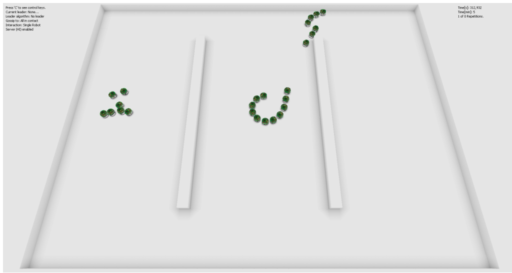
(a) Initial Setup



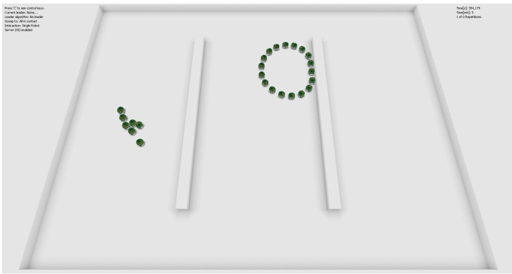
(b) After 200s



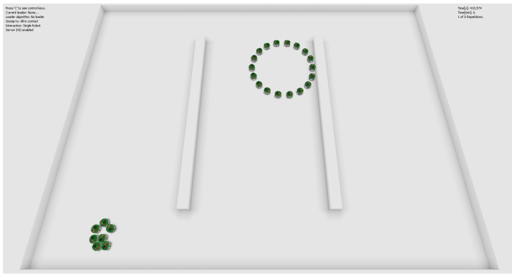
(c) After 270s



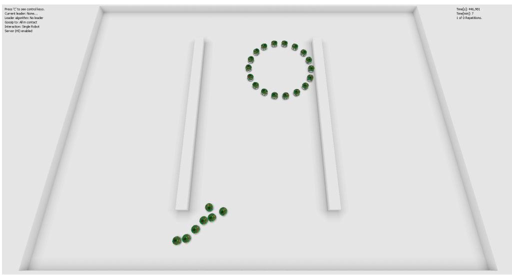
(d) After 310s



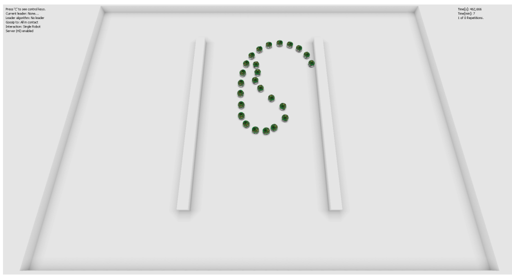
(e) After 380s



(f) After 440s



(g) After 490s



(h) After 540s

Fig. 4.15 Sequence of snapshots taken during a trial with an expert participant. The expert was not provided with the birds-eye view of the scene, which is depicted here. a) Initial positions of the robots. b) The operator waited fot the robots to aggregate and modified the behaviour of the middle and right clusters to follower. c) The operator guides the right cluster to the north gap. d) The middle cluster starts mixing with the right cluster. e) The middle cluster remains in follower behaviour. f) The operator guides the left cluster to the south gap. g) The left cluster starts mixing with the middle cluster. h) All the robots are now locally connected and therfore clsutered.

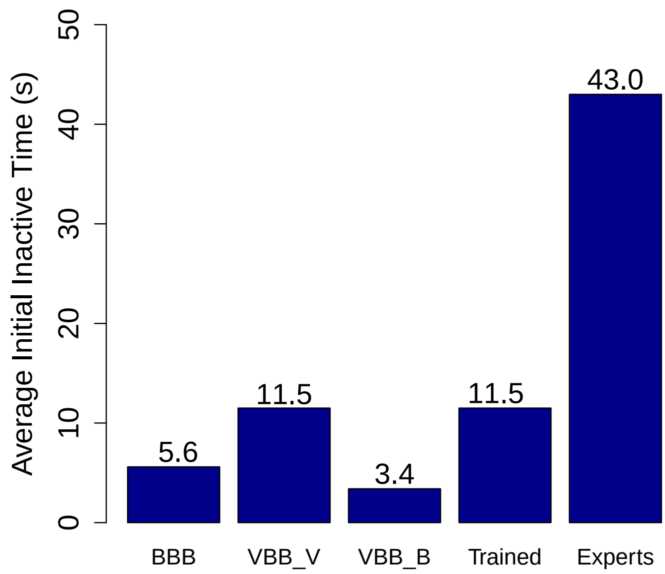


Fig. 4.16 The graph presents the average initial inactive time that operators waited at the beginning of the trials. The subdivision of VBB_V represents only the first trial of the VBB group, which was the one where they had access to the birds-eye view. The VBB_B subdivision represents the second and third trial of the VBB group where participants had no access to the birds-eye view.

This period of inactivity allowed the swarm to exhibit local aggregation behaviour and form small clusters within parts of the environment. When the robots are clustered, they are controlled more effectively than when the robots are dispersed. Operators that attempt to interact with the swarm too early were disturbing this process and had less effective subsequent interactions. This suggests that we have evidence to believe that the concept of neglect benevolence is being learned and exploited by trained and expert operators. It is worth noting that untrained participants with access to global state information also increased their initial period of inactivity. While they were observing the swarm global dynamic, yet in their next trials (subsequent blind trials) they do not repeat this waiting.

Our findings suggest that exposure to global swarm dynamics does not necessarily accelerate learning, neither for improving situational awareness nor for understanding swarm dynamics to accommodate for neglect benevolence. In addition, learning to interact with a swarm through a distributed interaction scheme that relies on local information requires training times even for simple tasks and interfaces. With trained and expert operators we were able to observe evidence for neglect benevolence. These operators learned to wait at the beginning of the trial for the swarm to converge. With the emergent local clusters, interactions were more beneficial as these clusters could be changed into leader-follower formations more easily. On the other hand, untrained operators only disturbed the swarm while interacting with it prior to its settling and formation of the local clusters.

4.4 Summary

This chapter presented a distributed human-swarm interaction scheme in which operators had access to only local information while aiding a swarm in an aggregation task. Human operators had access to swarm controls with which they were able to complete the aggregation task successfully when given global state information. When given only local information, however, untrained operators did not perform significantly better than random interactions. Nor did they exhibit a significant learning effect within three trials. Furthermore, human operators that once were given global state information did not demonstrate improved performance on subsequent trials when being restricted to local information. This suggests that no learning benefit was obtained from having observed the global dynamics once.

On the other hand, trained and expert operators, with at least one hour of training, showed significantly improved performance suggesting that the task was solvable. These operators compensated the lack in global situational awareness with increased requests for local sensory information while reducing the number of motion commands. Expert operators performed nearly as well as the baseline performance of the autonomous algorithm under ideal conditions, that is, without obstructions.

Finally, we observed evidence for neglect benevolence for trained and expert operators. These operators waited at the beginning of the trial for the swarm to converge to the emergent local clusters. From this configuration, interactions with the swarm were more beneficial as emerging clusters could be changed into leader-follower formations more easily. Untrained operators disturbed and interacted with the swarm prior to it settling into local clusters.

Chapter 5

Human-Robot Swarm Interaction with Limited Situational and Task Awareness

In this chapter we present a study that adds some restrictions to the ones presented in Chapter 4. Besides having limited situational awareness while interacting with a swarm robot, the human operator is now restricted to specific information regarding the mission objective. Different to SA (that limits the ability of the operator to understand the state of the environment and the surroundings), the limitation of task awareness (TA) is achieved by providing the operator with a main objective that requires a set of minor tasks not explicitly defined. We attempt to achieve better understanding of the consequences that detailed information, regarding the main objective, could have over the human operators decisions. Furthermore, if the limitation of this information could have some negative effect over the operators' interaction strategies. By keeping present the limited SA constraint, the limitation of TA emulates better a real world scenario.

In addition, if the swarm robot has some redundant agents, the human operator could perform additional tasks (exploration, complex data analysis, task support, etc) while maintaining the original task performance. Furthermore, requesting information to acquire situational awareness should not have an impact on the execution time of the task, especially if the human operator is interacting with a single robot of the swarm at a time. This means that the human operator could act like a super-agent in the swarm when needed.

The structure of this chapter is as follows: Section 5.1 presents the formulation of the problem (5.1.1), the updated simulation environment (5.1.2), the used swarm behaviour

(5.1.3) and the updates that the user interface had in comparison to the last study (5.1.4). Section 5.2 presents the experimental setup, including the participant training (5.2.1) and classification (5.2.2). Next, section 5.3 presents the results, including the validation metrics (5.3.1) and the validation of the training session (5.3.2), followed by an analysis of the participants performance (5.3.3) and perception (5.3.4). Section 5.3.5 presents an example of one of the experiment trials. Finally, section 5.4 has a summary of the chapter.

5.1 Methodology

5.1.1 Problem Formulation

We now focus on the impact that restricted task awareness (TA) has over the performance of a human operator, particularly while the situational awareness (SA) constraint from the last chapter remains. We attempt to measure the consequences that information regarding the task could have over the human operators' decisions and if this information could affect the operators' actions. By default, the robots execute the *Object Clustering* algorithm presented in [27]. This was the only behaviour left available for the robots, preventing the operator from changing the behaviour of the swarm.

For this study, we consider an environment with a working area connected to a hidden area through a small entrance. Inside the hidden area there are extra objects located. The robots' initial position is in the working area, preventing them to detect those objects located inside the hidden area.

As the robots are not programmed to explore the environment, we hypothesize that the human operator, when provided with additional information, will attempt to perform some exploration for the swarm. We attempt to evaluate if the operator is able to identify alternative tasks to achieve the objective without the need of them being made explicit. Our objective is to identify the amount of additional information and how explicit it should be so that the human operator attempts to further explore the environment and therefore, find the objects within the hidden area.

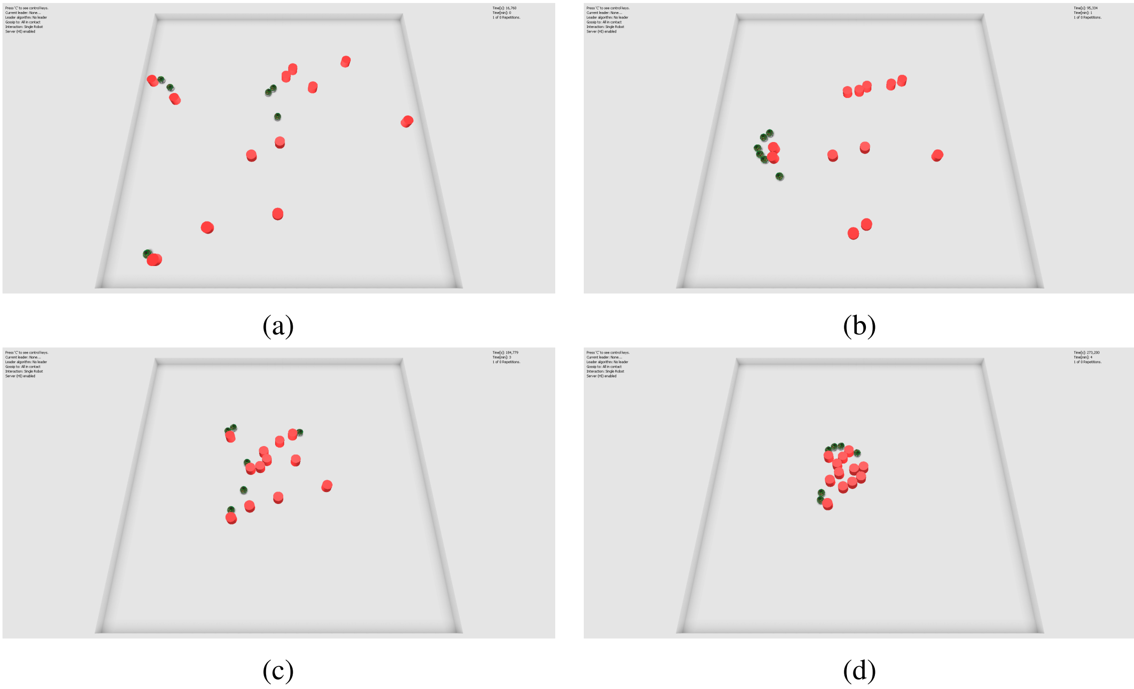


Fig. 5.1 Some snapshots from an example of a swarm of six robots (green cylinders) performing the object (red cylinders) clustering algorithm: a) Initial position of the robots and the objects. b-c) The robots keep pushing the objects to a random location. d) Finally, the robots cluster all the objects in a random location of the arena.

5.1.2 Robot and Simulation Platform

The virtual robot platform and the simulation environment were the same as the ones explained in Section 4.1.2. The simulators' "general menu" was re-designed and upgraded in a way that none of the changes would affect the robot simulations. These upgrades were mainly implemented to help the experimenters configure the simulation environments better and have more control over the results data.

5.1.3 Swarm Behaviour

For this study, the robots followed one unique behaviour: the object clustering. In contrast to the study from Chapter 4, in this case the operator had no need of swapping the swarm behaviour at any time. For this reason, the operator was not allowed to use the gossip algorithm or any other algorithm, yet was able to take control of the motion of random robots and their sensors.

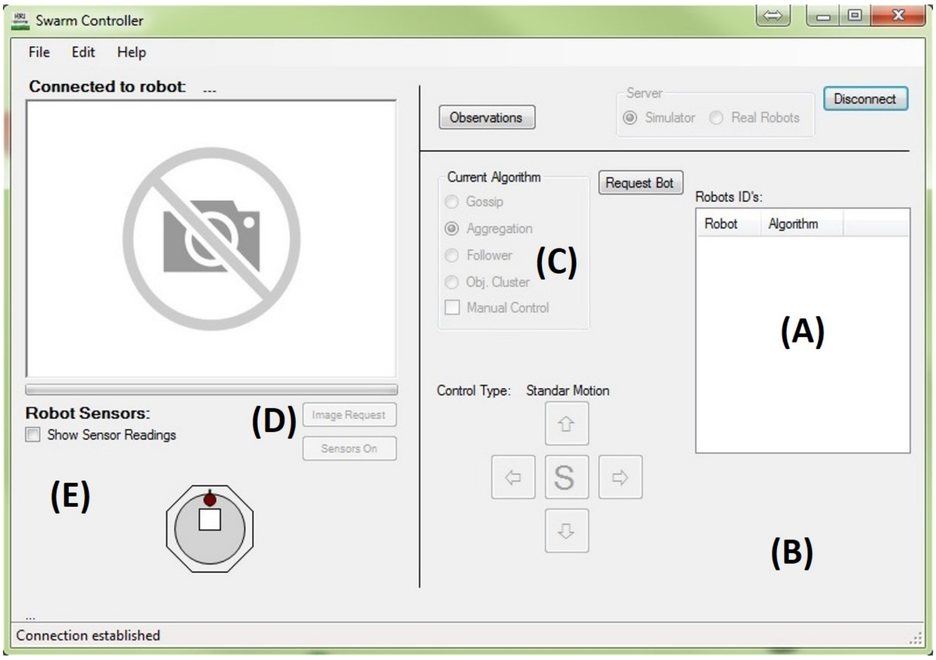


Fig. 5.2 The updated GUI that the participants used in the 2nd human-robot swarm interaction study.

The *Object Cluster Algorithm* (see Fig. 5.1) uses a similar line-of-sight sensor as the one used in aforementioned algorithms (section 4.1.3). However, in this case, the sensor can differentiate between three possible states. The states depend on it detecting an object, another robot or nothing. Because this sensor is limited by a maximum detection distance of 150 cm, the wall or an out-of-range reading are taken as a “nothing” measurement.

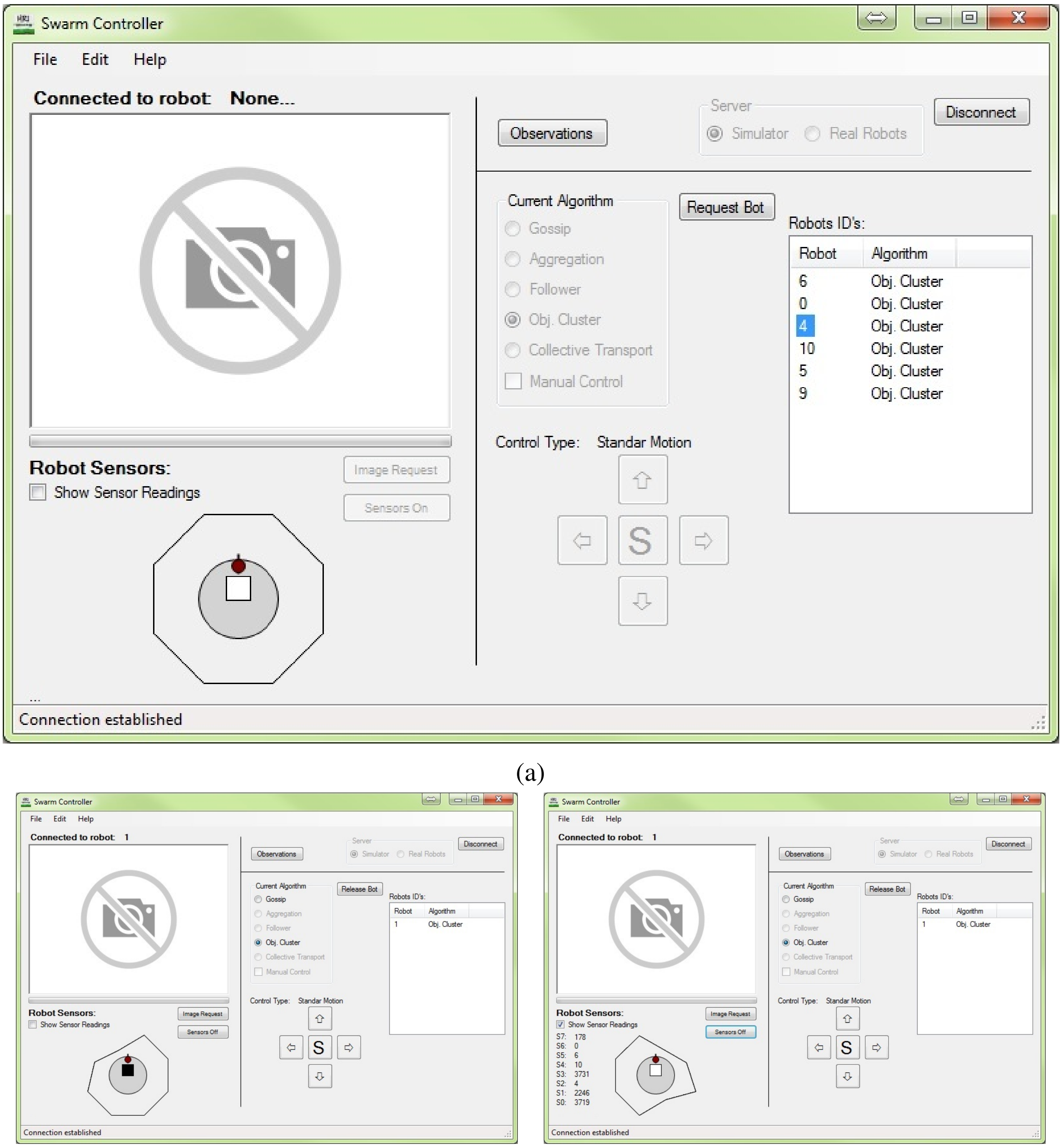
As shown in [27], this algorithm leads to the overall cluster of the objects (Fig. 5.1d), provided the sensing range is sufficiently large and no obstacles are present in the environment. When the robot sensor detects “nothing”, the robot moves forwards in a circular manner (the velocities of each wheel are scaled to values ranged between $[0.5, 1]$). When a robot detects an object, the velocity pair becomes $[1, 0.5]$ and if another robot is detected, the velocity pair becomes $[0.5, 0.8]$.

5.1.4 Updated User Interface

As in Chapter 4, the interaction between the human operator and the robot swarm occurs through a graphical user interface. This GUI is similar to the one used in Chapter 4 but with some significant upgrades. One of the implemented upgrades was to provide the experimenters with the possibility of configuring the GUI depending on the experiment needs. This means that the GUI can have certain sections deactivated to prevent the operator from accessing them.

Fig. 5.2 shows an example screenshot of the upgraded GUI. We refer to this as the "upgraded GUI" and to the one used in chapter 4 as the "old GUI". The positions of the changes/upgrades done to it are marked with letters. These changes are:

- A) *List of detected robots*: This is a new feature in comparison with the old GUI. In it, a history of the robots that the operator has connected to is stored. When a robot has been registered in this list, the operator can always attempt to reconnect to it at any moment. This is to give the operator the option to avoid the random selection. An example of the list when in use can be seen in Fig. 5.3a where six robots are already registered.
- B) *Disabled Gossip Algorithm*: The old GUI had an "Analyse" button and a set of parameters always visible to the operator that provided some extra information about the swarm when the *Gossip Algorithm* was executed (as seen in Fig. 4.7b). With the upgraded GUI, when the experimenter disables the use of the *Gossip Algorithm*, the "Analyse" button as well as the set of parameters are automatically hidden from the operator.
- C) *Manual control box*: The old GUI had a button with the text "Automatic" on it. Its functionality was to retrieve the manual control from the operator and return the robot to its original behaviour. The upgraded GUI changed this button for a check box. This new control provides better feedback for the operator to understand if the robot is in manual control or executing its normal behaviour.
- D) *Automatic switch between feedbacks*: In the old GUI, the operator had to stop the sensors' feedback before being able to request an image. With the upgraded GUI the operator can request an image despite the sensors' feedback being on, as the upgraded GUI manages the "stop sensors feedback" request, waits for the image to be sent and re-starts the sensors' feedback.
- E) *Graphical interpretation of the sensors' data*: In the old GUI the operator had to interpret the distance of objects from the robot from the raw data of the sensors. With the upgraded GUI, the operator can see a visual representation of this data (as seen in Fig. 5.3b) and only if desired, the operator can request to see the raw values of the sensors (as seen in Fig. 5.3c).



(a)

(b) (c)

Fig. 5.3 a) GUI with a list showing the IDs of some previously connected robots. b) Example snapshot of the GUI when the sensors' feedback is activated. c) An example with the sensors' feedback activated and the raw data visible.

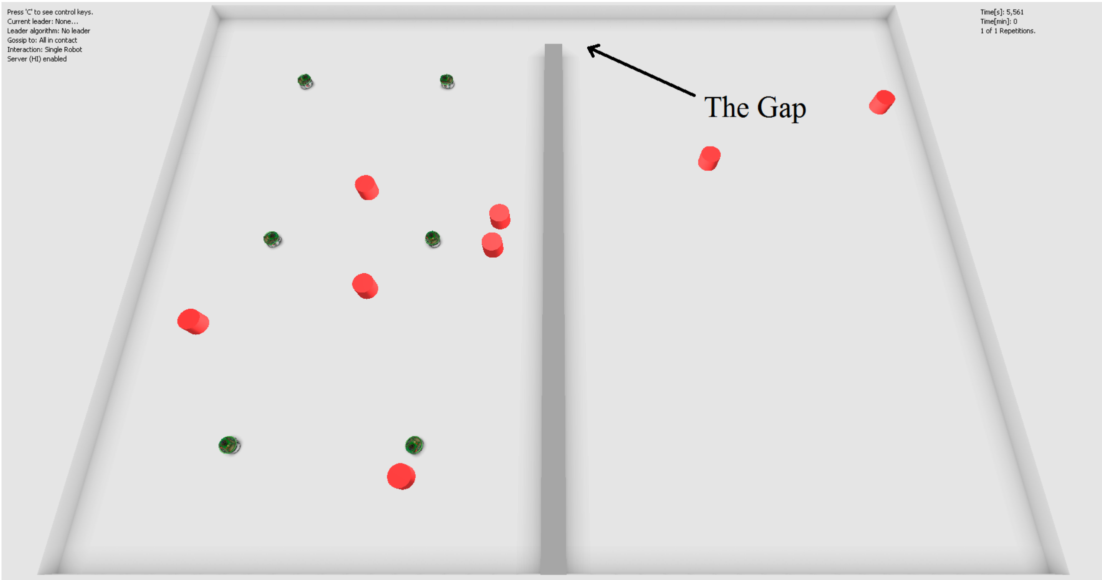


Fig. 5.4 Experiment arena with the robots initial positions and random initial localization of the objects.

Despite the possibility of configuring the upgraded GUI, the only section that was configured differently to the study of Chapter 4 was the algorithms section. The upgraded GUI kept all the old GUI's abilities (connect to the simulator environment, request a random robot, request an image from the robot, monitor the robot's sensors, control the robot's motion and —if available— behaviour, etc.), and added some others (memory of connected robots, graphic sensors' data representation, improved controls, etc.) to provide the operator with better control over a single robot and therefore better influence over the swarm.

5.2 Experimental Setup

In this study we investigate the consequences that limited situational awareness mixed with limited information regarding the task has over the operators' performance. During the experiment, the operators received the same task but with different types of complementary information about it.

Fig. 5.4 provides an overview of the simulation environment used for this experiment. A group of six robots operate in a rectangular arena of size 400×300 cm. A wall in the middle divides the arena in half, creating two smaller areas of 200×300 cm each. This wall is coloured with a darker gray than the rest of the environment. Both areas are connected through an entry point of 30 cm located at the upper extreme. We refer to this entry point as *the gap*. The wall is sufficiently tall to prevent the robots from perceiving other robots or objects located on the other side. The left side is referred to as the "working area" and the right side is referred to as the "hidden area". Overall, eight red cylindrical objects are randomly scattered through the arena. Six of them located in the working area and the other two in the hidden area. Fig. 5.4 also presents an example of the objects' random initial positions as well as the initial fixed positions of the robots for every trial.

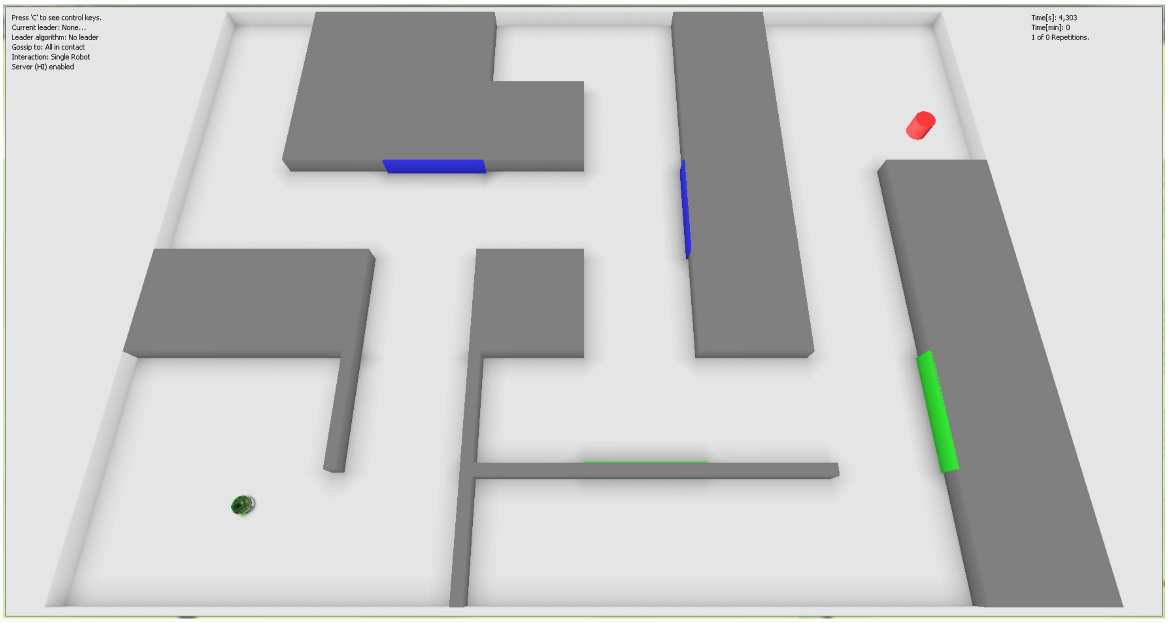
Throughout all trials, different sets of information from the robots and participants were recorded. Regarding the robots, the simulator recorded their positions and orientations on average every 30 ms. The simulator also checked every second if a robot had crossed from the "working area" to the "hidden area". Regarding the GUI, all the interactions from the participants were also recorded (commands issued, images taken, changes of robot leader, etc), with a time stamp of every interaction event.

At the beginning of the experiment, participants were given a 10 minutes presentation comprised of two parts. The first part explained the mission objective and introduced the participants to the object clustering behaviour. The second part of the presentation introduced the participants to the user interface (as shown in Fig. 5.2) and its features. After the presentation, participants had a training session followed by the experiment trials. Finally, participants were asked to answer a short questionnaire. More details about the training, the experiment and the questionnaire are provided in the following sections. The used slides are presented in the second section of Appendix C.

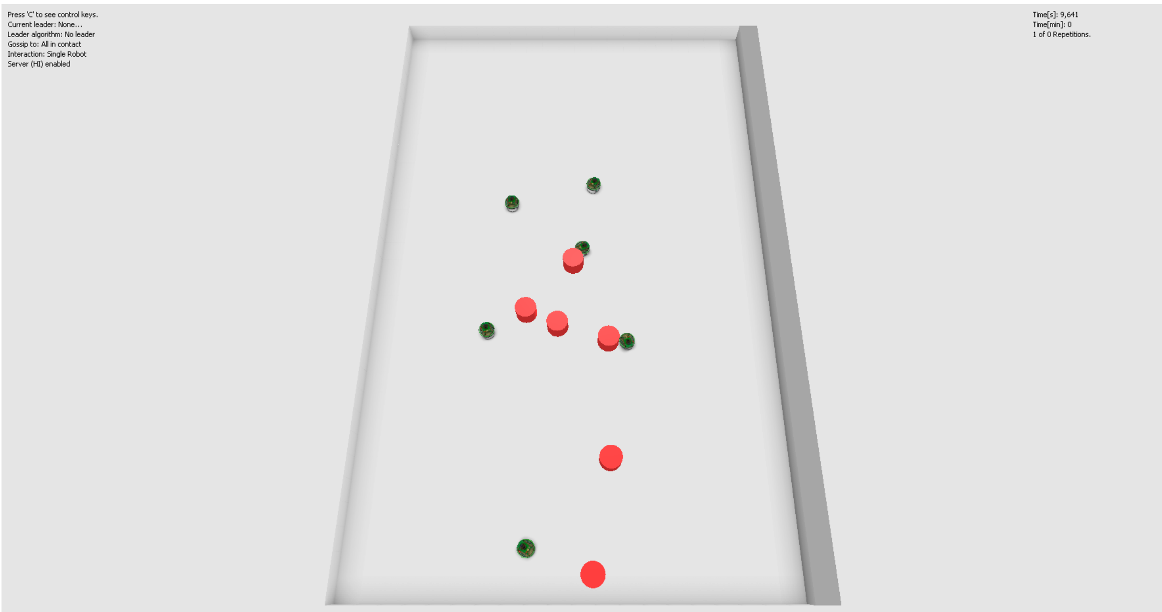
The study received ethical approval by The University of Sheffield. All participants were students of the university and their ages ranged between 18 and 39.

5.2.1 Participant Training

The training session was divided in two parts: Exploration of a maze and the observation and understanding of the swarm behaviour.



(a)



(b)

Fig. 5.5 Example snapshots of the two different training arenas (Green cylinder = Robot, Red cylinder = Object): a) Maze arena used for the GUI training. b) Training arena for understanding of the object clustering behaviour with only the “working area” and the gray wall emulating the center wall of the experimental arena.

- *Maze exploration*: This exercise was designed to train the participants in the use of the GUI. Participants were asked to guide a single robot through a maze (see Fig. 5.5a). The maze had coloured walls as hints for the participants to know in which direction to move the robot. In the introductory presentation, they were told that the "blue" walls indicated that the robot had to turn to the right, while the "green" walls indicated a left turn. Participants were asked to move the robot through the maze and take an image of a red object located at the end of the maze. All participants had a maximum allowed time of 10 minutes.

- *Swarm behaviour understanding*: This exercise was designed to help participants understand the object clustering behaviour. We presented them with a two minute video where six robots were performing the object clustering behaviour with six objects. In this video, the robots' initial positions were the same as the ones used in the experiment trials. The arena used was a rectangle of 200×300 cm with one of the long walls coloured in a darker gray than the others. This emulates the wall dividing the "working area" from the "hidden area" from the experimental setup (as seen in Fig. 5.5b). Participants were only told that this darker gray wall was there to help them orientate better.

5.2.2 Classification of Participants

Overall, data of 50 participants was collected. The experiment was divided in two trials, each lasting five minutes. In each trial, participants were told to monitor and support the robots while carrying out a task. For trial 1, all participants had the same objective; it was presented to them as: "*Cluster all red objects*". For trial 2, participants were assigned to one of three conditions at random. Depending on the assigned condition, different information was added to the original objective in trial 2:

- **Group A**: Participants of this group had extra task details. The information presented to them was "*There are 8 objects in total*". There were 17 participants in this group.

- **Group B**: Participants of this group had extra environment details. The information presented to them was "*Be aware that the arena might have changed*". There were 16 participants in this group.

- **Group C**: Participants of this group had an explicit second objective added. The information presented to them was "*Find the missing objects inside the hidden area*". There were 17 participants in this group.

5.2.3 Questionnaire

After the two experimental trials were completed, each participant was asked to answer a short questionnaire (Appendix D). The questionnaire asked for their opinion about their perception of three different aspects of their experience: Their curiosity level, their understanding of the robot control and their understanding of the environment. All questions were measured with a Likert scale ranging from 1 to 10, where 1 meant "None" and 10 meant "A lot". The questions presented to the participants were:

- Do you consider yourself a curious person?
- On a scale from 1 to 10, how hard was it for you to control the robot?
- On a scale from 1 to 10, how hard was it for you to understand the robots' environment?

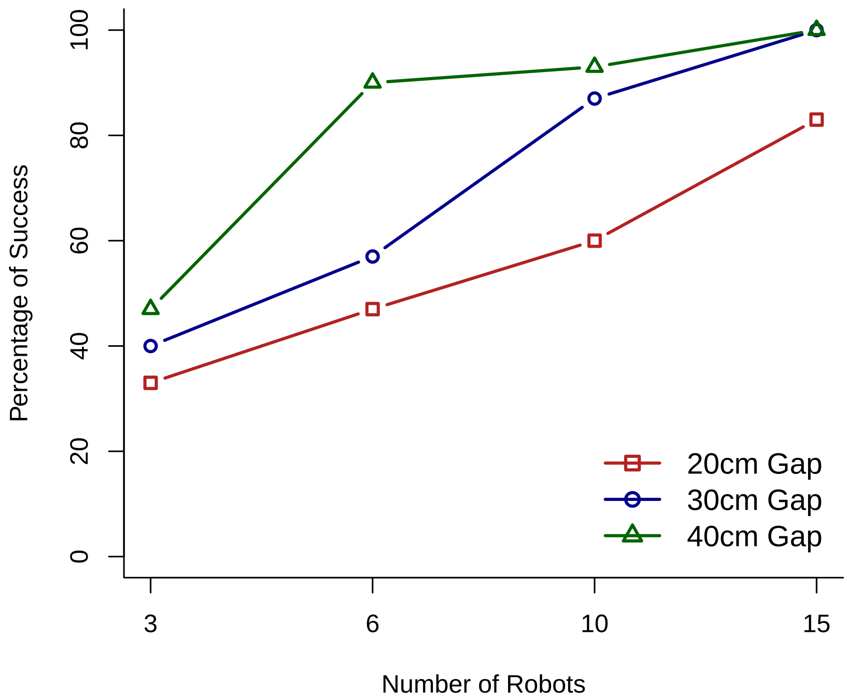
5.3 Results

In this section we compare the performance of the participants (the operators) with the baselines and provide an analysis of their actions. We also analyse the data from the questionnaires that provide the perception that participants had of their own performance.

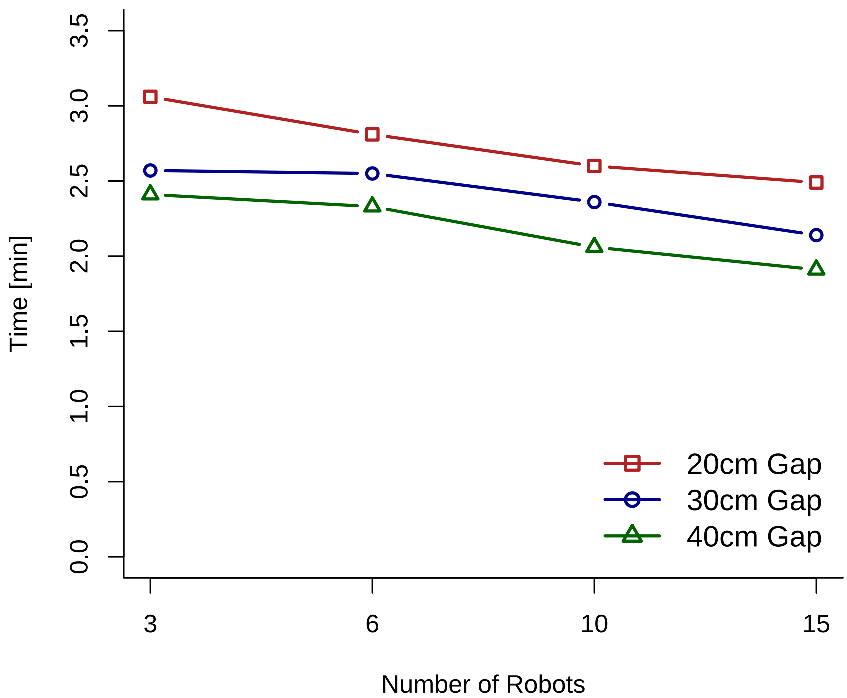
5.3.1 Performance Metrics and Baseline Performance

A set of exercises with a 6 robot swarm were used to determine baseline parameters and to define a performance metric. For each baseline, 30 trials of 300 s were conducted. All tests had the robots' initial positions and the test environment as the same used for the experiment trials.

For the first baseline, the gap connecting the working area from the hidden area was modified of size. Three different gap sizes (20cm, 30cm and 40cm) were tested against 4



(a)



(b)

Fig. 5.6 a) Percentage of successful trials where a member of the swarm, employing a random movement behaviour, discovered the hidden area. b) Time for the first robot to discover the hidden area (all the unsuccessful trials were discarded).

different swarm sizes (3, 6, 10 and 15 robots). However, the behaviour of the swarm was changed to random movement[1] and without any kind of external interaction.

The swarm performance was determined by the success rate of it discovering the hidden area. Fig. 5.6a illustrates the success rates that the different swarms had discovering the hidden area. For each case we measured the time it took for a robot of the swarm to discover the hidden area. Table 5.1 shows the average discovery times for the different sizes of the swarm. These times are the same values illustrated in Fig. 5.6b. As expected, the larger the swarm, the faster and more often it finds the hidden area.

Table 5.1 Average time and the standard deviation that different sized swarms took to discover the hidden area. Presents only the successful cases.

Average Discovery Time [min]

$GapSize$ $[cm]$	3 $Robots$	6 $Robots$	10 $Robots$	15 $Robots$
20	3.06 ± 0.62	2.81 ± 1.08	2.60 ± 0.90	2.49 ± 1.20
30	2.57 ± 1.33	2.55 ± 1.28	2.36 ± 1.00	2.14 ± 1.06
40	2.41 ± 1.12	2.33 ± 1.11	2.06 ± 1.24	1.91 ± 0.83

The concept of scalability is meant to be one of the main attributes of a robot swarm. Following this, if a human operator takes control of one robot of the swarm, the impact on its performance should not be significantly affected. The second baseline tests if the amount of robots that conform our swarm were enough to execute the task (clustering all the objects) and leave a spare unit for the human operator to control without considerably affecting the swarm performance. We did a comparison of the overall performance between a swarm of five robots against a swarm of six robots. The performance was measured by the total distance between all the clustered objects. In both cases the swarms were clustering six objects. The test environment was the same used for the training of the participants. Fig. 5.7 presents the average distance between clustered objects of 30 trials for each case.

Five minutes were long enough for both swarms to cluster all the objects within a constant range. The greatest impact of having one less robot was on the time it took the swarm to achieve the minimum geometric median. While a swarm of five robots took 212 seconds, one more robot decreased that time to 173 seconds. These times are represented in Fig. 5.7

[1]The implementation of random movement behaviour involved every robot travelling in a straight line for a random time (between 1 to 5 seconds), then rotating to the right over its own axis for another random amount of time (between 1 to 5 seconds). If an obstacle was in front of the robot, a simple object avoidance reaction was executed.

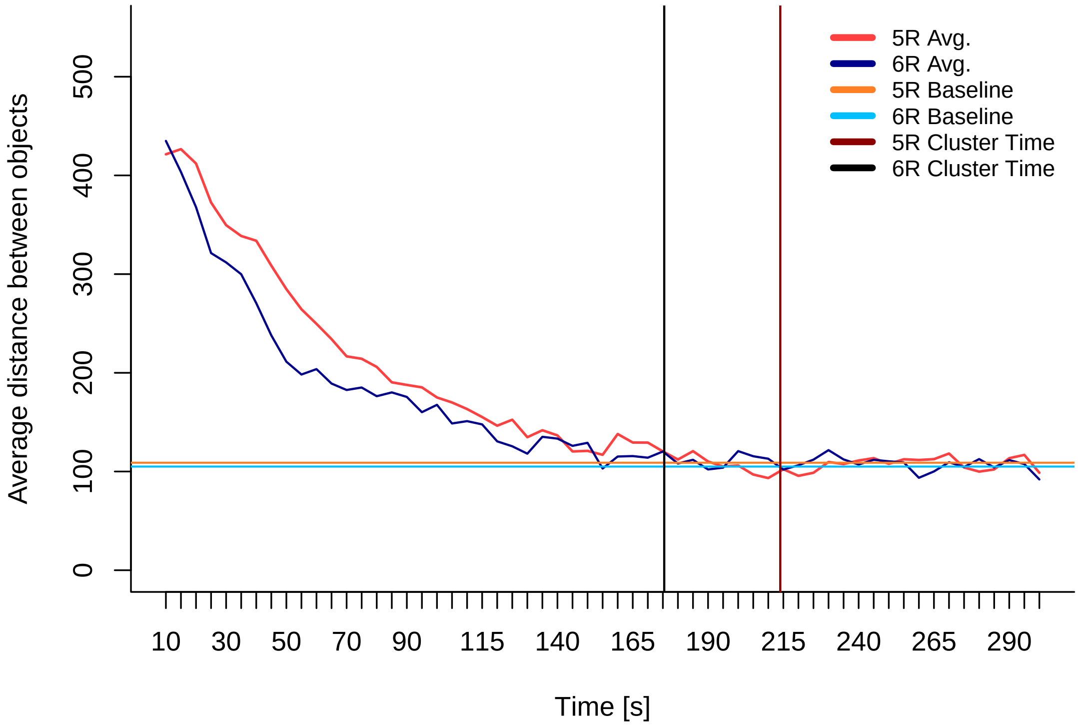


Fig. 5.7 Comparison of the average performance between a swarm of five robots and a swarm of six robots in clustering six objects. All trials were in an environment with no obstructions through 30 trials.

by the "5R Cluster Time" and "6R Cluster Time" lines. However, despite the time difference, the geometric median was not significantly different (two-sided Mann-Whitney test, p-values = 0.2426). Also in Fig. 5.7 the average of the geometric median of the objects is represented by the "5R Baseline" and "6R Baseline" lines. While a five robot swarm achieved an average geometric median of 108.9 cm, with one more robot in the swarm the average geometric median was of 105 cm after 5 minutes. This means that if a human operator takes control of one robot of the swarm, the rest will still be able to complete the task without significant decay of the performance.

5.3.2 Training Validation

To corroborate that the training sessions were useful for the participants, we compared the actions during trials from all the recorded actions of their training sessions. Table 5.2

contains the averaged amount of repetitions of the most significant activities. The difference between averages of both trials was small in all cases and not significant. The similarity in the amount of actions between trials suggests that the operators were adapting to the task new information, but without further learning of the robot's control.

Table 5.2 Average amount of specific actions performed by the participants. The last column presents if the two-sided Mann-Whitney test indicated that the comparison between Trial 1 and Trial 2 was significantly different based on a 0.05 p-value.

Training Record

$Action$	$Trial\ 1$	$Trial\ 2$	$P-Value(0.05)$
$Changes\ of\ leader$	7.16	7.06	0.361
$Image\ Requests$	31.98	33.18	0.499
$Motion\ Commands$	452.26	543.28	0.438

5.3.3 Operator Performance

We compared the performance of the participants to the baseline performance. The main metric was the success rate of the swarm discovering the hidden area. Its discovery was established to be as soon as any of the robots moved inside it. For a robot to go inside the hidden area, it had to cross through the gap (as seen in Fig. 5.4) linking the "hidden area" to the "working area".

Another important aspect was whether the operator had visual evidence of the gap. This evidence was defined as the "visual proof quality" (VPQ) and relied on three factors at the moment that an image was taken. The first factor relied on the orientation of the robots' camera towards the gap. The second factor was related to the position of the robot in the hidden area, specifically in reference to the *EP Horizon* line (as seen in Fig. 5.8). The last factor was the *alpha* (α) angle defined as the one existing between two imaginary lines drawn from the robot center to the two extreme points of the gap (*G1* and *G2* in Fig. 5.8). With these factors we were able to define three levels of VPQ; Strong evidence, weak evidence and no evidence.

- *Strong evidence* - At least one image was taken while the robot camera was oriented towards the gap. When taken, the gap was not obstructed by any other robot or object and the robot was located above the *EP Horizon* line or the α value was bigger than 0.15 radians.

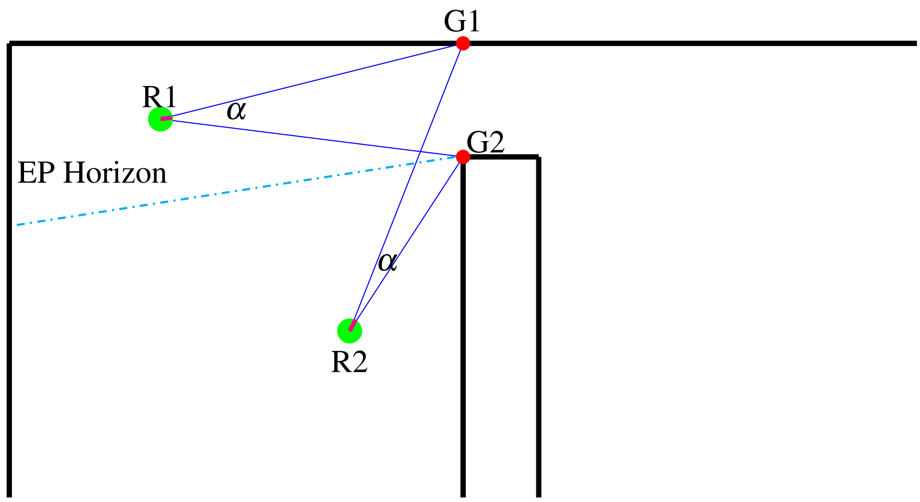


Fig. 5.8 Representation of a section of the environment with two robots pointing to the gap.

- *Weak evidence* - At least one image was taken while the robot camera was oriented towards the gap. When taken, the gap was not obstructed by any other robot or object and the robot was located under the *EP Horizon* line and the α value was smaller than 0.15 radians.
- *No evidence* - From all the images taken by the operator, none of them were oriented towards the gap or were obstructed by another robot or object.

Depending on the success rate and the achieved VPQ, participants were classified in one of five performance groups:

- No evidence / Not found (NE/NF) - Participants in this group had *No evidence* VPQ level and were unable to find the hidden area.
- Weak evidence / Not found (WE/NF) - Participants in this group had *Weak evidence* VPQ level and were unable to find the hidden area.
- Weak evidence / Found (WE/F) - Participants in this group had *Weak evidence* VPQ level and were able to find the hidden area.
- Strong evidence / Not found (SE/NF) - Participants in this group had *Strong evidence* VPQ level but were unable to find the hidden area.
- Strong evidence / Found (SE/F) - Participants in this group had *Strong evidence* VPQ level and were able to find the hidden area.

- No evidence / Found (NE/F) - Participants in this group had *No evidence* VPQ level but they discovered the hidden area. Nevertheless, this participants finished the trials unaware of their achievement. This was possible to be discovered as this participants neither had SE or WE but controlled one robot inside the hidden area.

Despite the “Weak evidence / Found" group being a possible combination, there were no participants who fell in this classification. It was highly likely that all aware participants searched for some kind of evident proof while discovering the hidden area.

From the 17 participants belonging to group “A”, three were able to find the hidden area (two-sided Mann–Whitney test, p-values = 0.07977). Similarly, from the 16 participants of group “B”, three were able to find it (two-sided Mann–Whitney test, p-values = 0.5757). Finally, from the 17 participants of group “C”, eight were able to find the hidden area (two-sided Mann–Whitney test, p-values = 0.001561). This suggests that making the objective explicit in the second trial gave the participants of group “C” significant information to influence their strategy and performance. Fig. 5.9 presents the results of the performance from all participants by group division. Similarly, Table 5.3 presents the performance in percentage value of all participants in general.

Table 5.3 All 50 participants divided by performance subgroup.

All Participants

Classification	*Count*	*Percentage*
NE/NF	15	30%
WE/NF	11	22%
SE/NF	10	20%
NE/F	2	4%
SE/F	12	24%

Using the group classification described in Section 5.2.2 we refer to the participants of groups “A” and “B” as the non-informed (NIP) participants and those of group “C” are as the informed (IP) participants. This differentiation was based on the explicit mention of the existence of a hidden area in the operators task objective.

There were 33 NIP within groups “A” and “B” together. The overall success rate of the NIP was of 18.2%. This was the combination of the SE/F participants (12.1%) and the NE/F participants (6.1%). In general, NIP who found the hidden area were slower at finding it than a three robots swarm in the same environment. Also, compared to the discovery

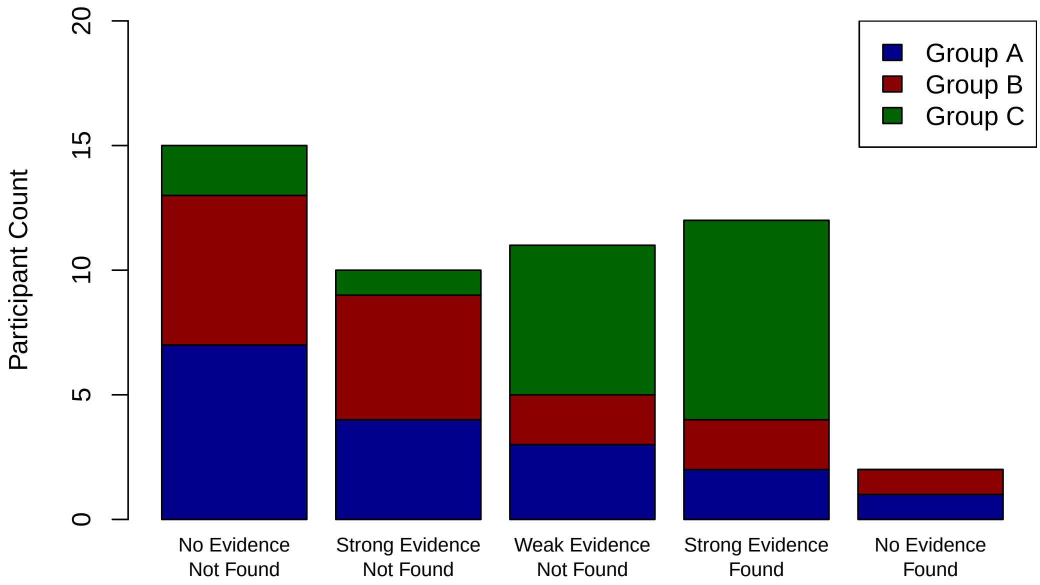
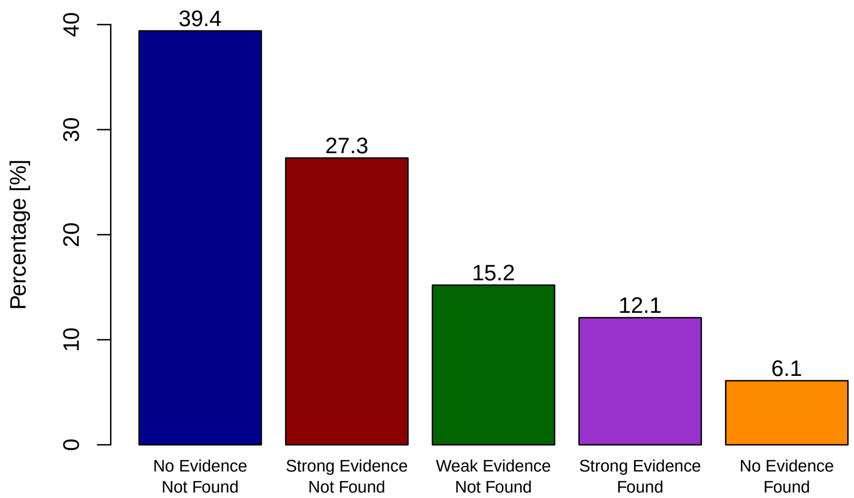


Fig. 5.9 Performance of all 50 participants divided by trial group.

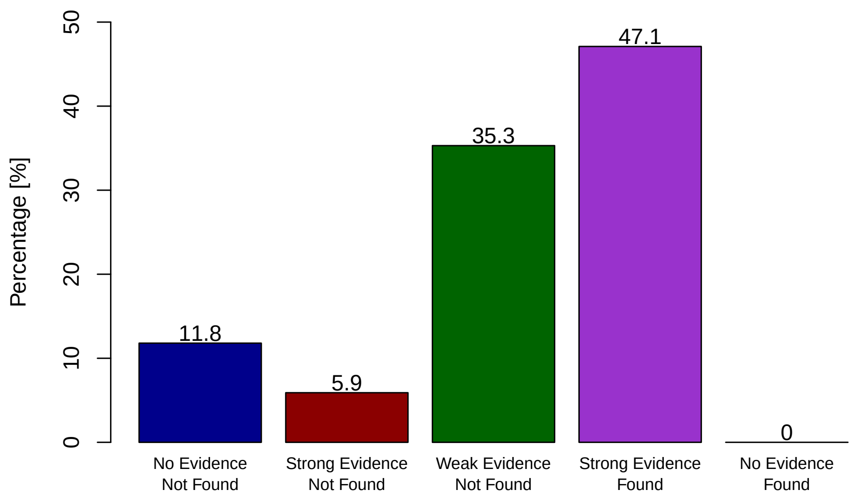
success baseline (Fig. 5.6a), their efficiency was even lower than that of a three robots swarm (three robots swarm with 40% success rate). Fig. 5.10a illustrates the performance of all NIP participants through all trials, including those who were unsuccessful in finding the hidden area.

Regarding group “C”, it had 17 IP. They were able to achieve a 47.1% success rate. Compared to the discovery success rate baseline (Fig. 5.6a), their efficiency is between a three and six robot swarm (three robots swarm with 40% success rate & 6 robots swarm with 57% success rate). However, they were able to find the hidden area faster than a swarm of 15 robots. Their performance improved in comparison to the NIP, yet lacks of considerable improvement to argue that a human operator is significantly beneficial for the swarm. Fig. 5.10b illustrates the performance of these participants through all trials.

Regarding the comparison between the IP and NIP, some interesting differences can be noticed. There were no participants from the NE/F group within the IP because, by definition, all of them were told of the existence of a hidden area. The overall performance of IP was significantly better than the NIP (two-sided Mann–Whitney test, p-values = 0.03405). The



(a)



(b)

Fig. 5.10 Performance in percentage of participants from: a) All non-informed participants. b) All informed participants.

amount of WE/NF participants in the IP was doubled in comparison to the NIP. Based on this results, some participants could have been searching for the hidden area, but were not able to achieve enough SA. The SE/NF participants decreased in the IP group. As all of them were explicitly searching for the gap (or an entrance to the hidden area), these participants found it but had problems to guide a robot to get through it. Finally, the amount of NE/NF participants decreased in the IP. This could be because they were consciously searching for the hidden area and were giving less priority to the overall clustering mission.

Lastly, a comparison of the average time that successful participants from both groups (NIP and IP) took to discover the hidden area suggests that the explicitness of the secondary objective improved the operator performance. Table 5.4 presents these times by group division.

Table 5.4 Presents the time (average and standard deviation) that operators took to discover the hidden area. This only counts the successful cases where the operators discovered the hidden area.

Average Discovery Time [min]

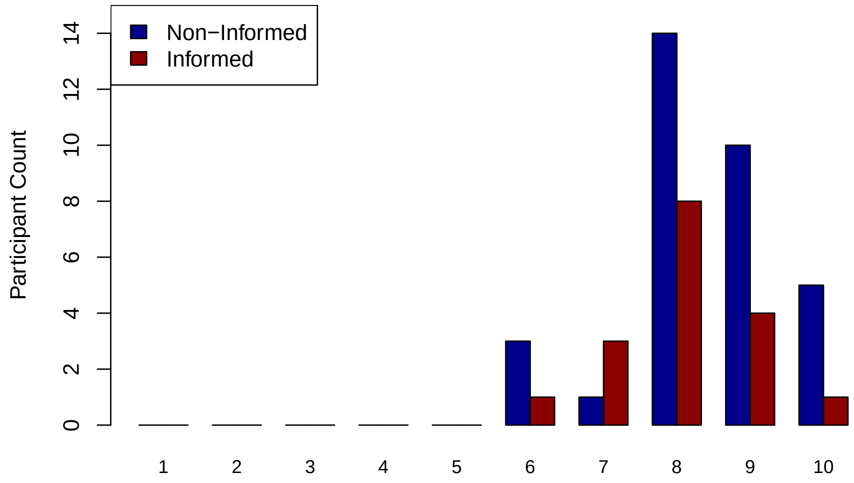
Group	*Time*
A	2.76 ± 0.79
B	2.76 ± 1.82
C	1.91 ± 0.59

5.3.4 Operator Perception

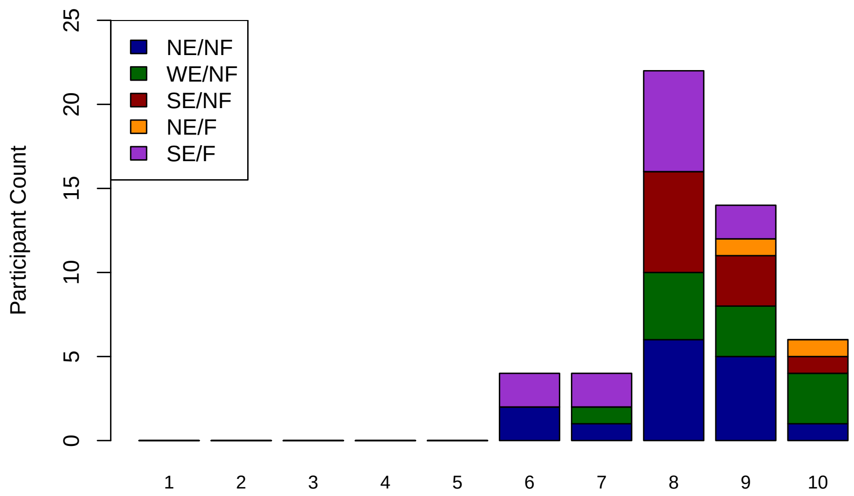
As mentioned in Section 5.2.3 the answers from the questionnaire had a scale from 1 to 10. We classify the answers again between 1 and 5 as “low” or “easy” and the answer between 6 and 10 as “high” or “hard”.

The first question (*Do you consider yourself a curious person?*) addressed the participants’ self-rated level of curiosity. Fig. 5.11a presents the answers of all participants divided in IP and NIP, while Fig. 5.11b presents the same answers but classified by performance group. Their responses had no significant difference between the NIP and the IP (two-sided Mann-Whitney test, p-values = 0.2121).

From the 33 NIP, 54.5% of the participants (27.3% from SE/NF, 15.2% from WE/NF and 12.1% from SE/F) were able to see the gap within at least one of their taken images. Despite the quality of the gap in the SE/NF group, most of them did not enter or identify the hidden



(a)



(b)

Fig. 5.11 Responses about how curious participants think they are (1 = Not Curious, 10 = Very Curious). a) All participants classified by response and group. b) All participants classified by response and performance group.

area. However, 12.1% from the SE/F (NIP) subgroup was successful, suggesting that finding the hidden area was possible. This means that some participants from the SE/NF group gave no priority to the gap, presumably as it was not mentioned in the given objective. Yet, all of them classified themselves as highly curious. For the IP group, as the name describes, they were informed of the existence of the hidden area and for this reason the curiosity question had no importance for the analysis of the study.

The classification of the answers by the performance groups (Fig. 5.11b) shows similar curiosity ratings between all participants and with no particular tendency. An interesting observation is that despite the answers being all in the "highly curious" range, many of the participants were not curious about a gap appearing in their taken images. The main example of this were the participants from the SE/NF and the WE/NF groups, where despite the images showing the existence of a gap in the environment, they did not perform further exploration of that area.

The second question (*On a scale from 1 to 10, how hard was it for you to control the robot?*) addressed the robot control perception of the participants. Fig. 5.12a illustrates the answers of the participants by groups (NIP and IP) and Fig. 5.12b by performance groups. A comparison between the answers from the NIP and the IP reveals that the robot control seemed to be unrelated to the information delivered to them (two-sided Mann–Whitney test, p-values = 0.9509).

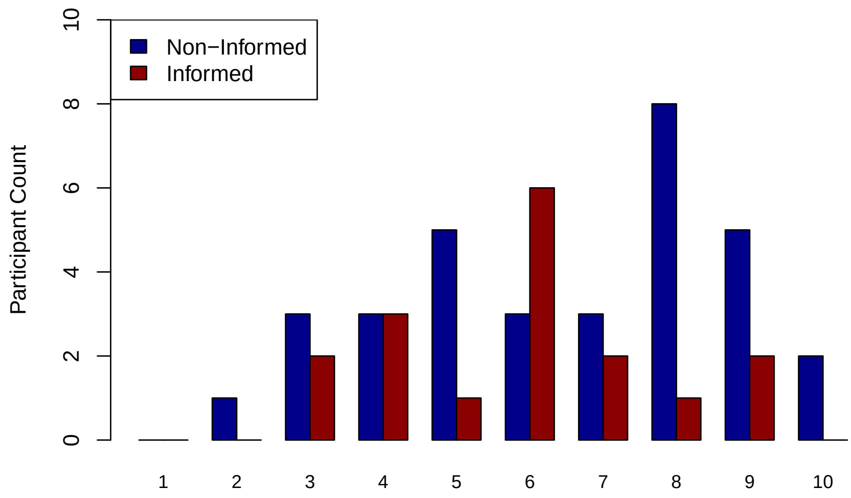
Analysis from the groups shows that 64% of the NIP participants declared it was hard to control the robots. Similarly, the IP participants had 65% of them declaring the same. This suggests that overall, the robot control and interface understanding were independent of the task and objectives given to the operators.

Table 5.5 Presents the percentage values of the participants responses regarding the robot control question from four of the subgroups.

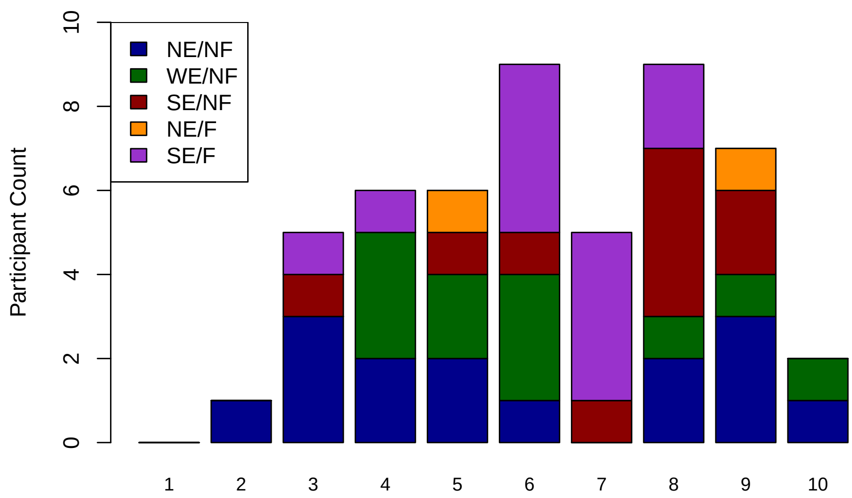
Subgroup Percentage Classification

$Subgroup$	$Global$	$Easy$	$Hard$	$P-Value(0.05)$
SE/F	24%	16.7%	83.3%	0.001
SE/NF	20%	20%	80%	0.010
WE/NF	22%	45.5%	54.5%	0.705
NE/NF	30%	53.3%	46.7%	0.738

The analysis of the performance groups (Fig. 5.12b) suggests that those participants who were able to acquire strong evidence (SE) were more aware of the difficulty of the robot



(a)



(b)

Fig. 5.12 Responses about how participants felt with the control of the robot (1 = Very Easy, 10 = Very Hard). a) All participants classified by response and group. b) All participants classified by response and performance group.

control. From Table 5.5 it is possible to see that a significant amount of participants from the SE groups rated the robot controls as hard. On the other hand, the other two groups responses (WE/NF and NE/NF) were somehow balanced. This encourages the conclusion that the more understanding of the robot interface the operator possesses, the more conscious they are of the control limitations and understand better the challenge to acquire SA.

Finally, the last question (*On a scale from 1 to 10, how hard was it for you to understand the robots' environment?*) addressed the perception that participants had in acquiring SA. Different to the last question, a comparison between the answers from the NIP and the IP reveals that the understanding of the environment might be affected by the information delivered to them within the mission objective. Fig. 5.13a illustrates the answers of the participants by groups (NIP and IP) and Fig. 5.13b by performance groups.

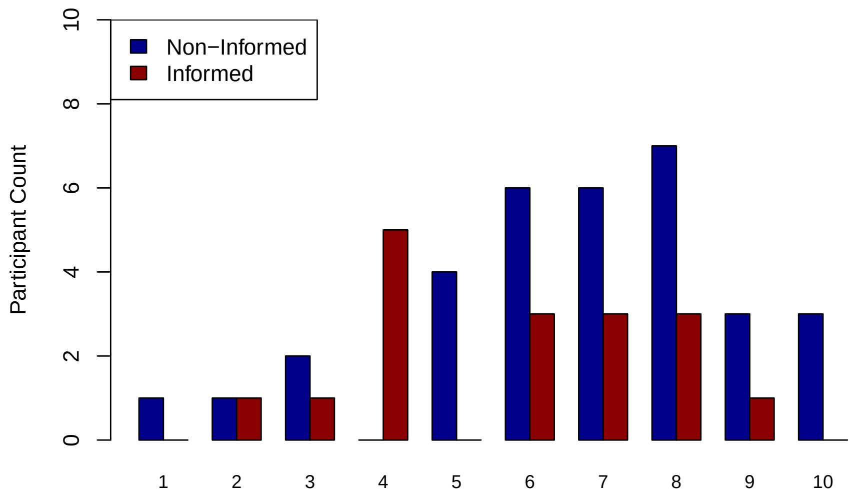
Overall, participants found the understanding of the environment hard (global response: easy 30%, hard 70%). Nevertheless, there was a difference of opinion between the NIP and the IP participants. While 76% of the NIP said it was hard for them to understand the environment, only 59% of the IP expressed the same opinion. This could be caused by the extra information regarding the hidden area, affecting the participants perception of their own understanding and therefore, their opinion of their own SA.

Analysis of the performance subgroups (Fig. 5.13b) gave no significant differences except for the NE/NF subgroup (see Table 5.6). The NE/NF group was composed of 30% of all participants, where 73.3% declared that their understanding of the environment was hard and only 26.7% said it was easy (two-sided Mann-Whitney test, p-value = 0.01281). This suggests that this participants possibly felt lost and lacking of SA, adding difficulty for them to understand the environment.

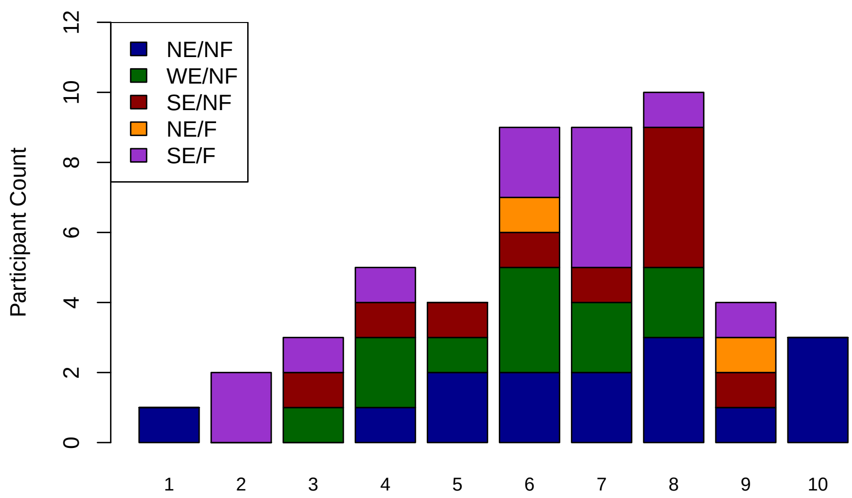
Table 5.6 Presents the percentage values of the participants responses regarding the environment understanding question from four of the performance groups.

Subgroup Percentage Classification

$Subgroup$	$Global$	$Easy$	$Hard$	$P-Value(0.05)$
SE/F	24%	33.3%	66.7%	0.118
SE/NF	20%	30%	70%	0.089
WE/NF	22%	36.4%	63.6%	0.225
NE/NF	30%	26.7%	73.3%	0.013



(a)



(b)

Fig. 5.13 Responses about how participants rated their understanding of the environment (1 = Very Easy, 10 = Very Hard). a) All participants classified by response and group. b) All participants classified by response and performance group.

5.3.5 Example Trial

Fig. 5.14 presents a sequence of snapshots from a trial taken from one participant of group "C". This participant was successful in finding the hidden area. The collection of snapshots present the positions of the robots, their orientations and the positions of the objects in specific moments of the trial.

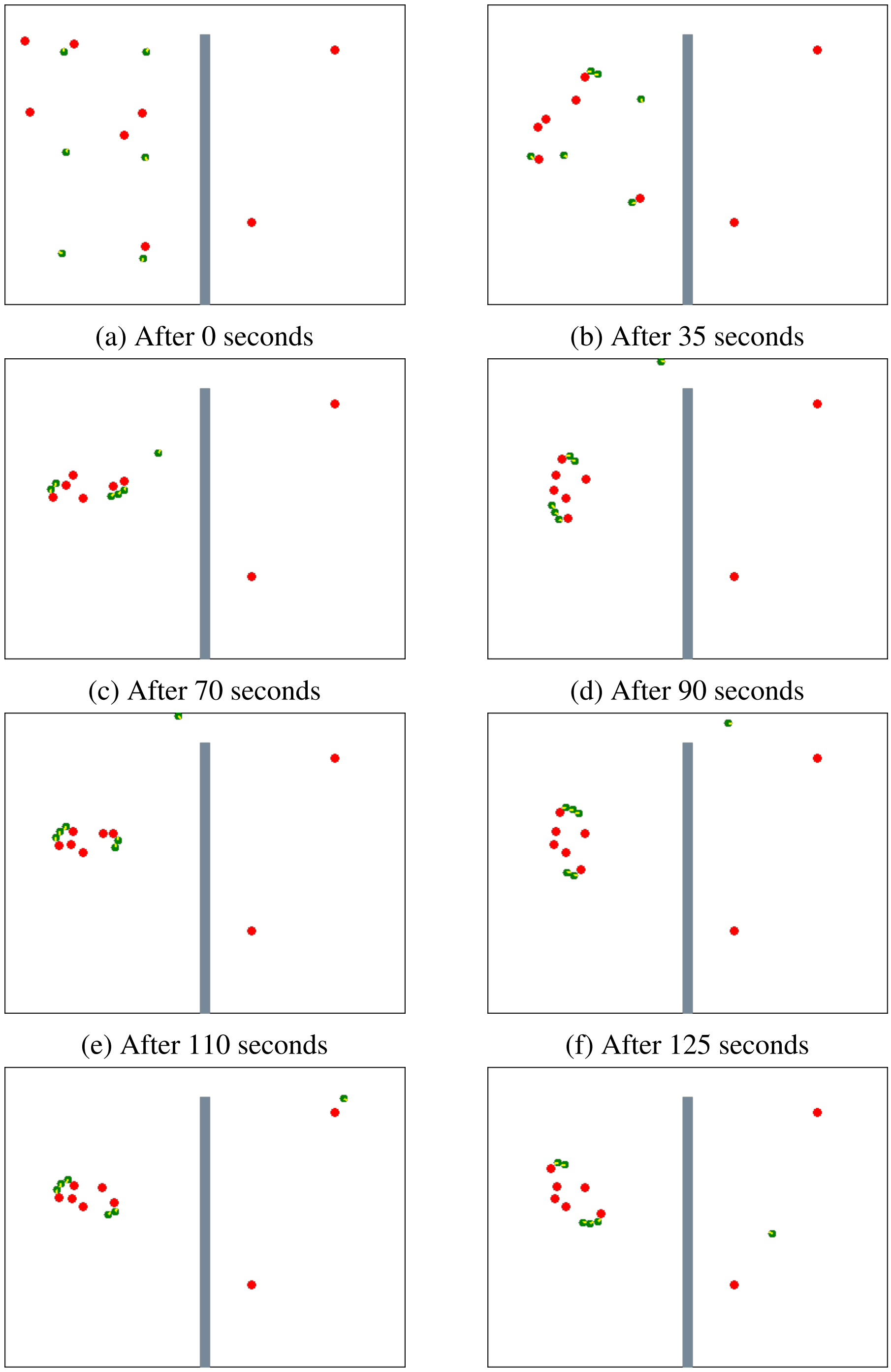
The sequence starts with the initial positions of the robots and the eight objects randomly distributed through the arena (a). Note that from the eight objects, always two of them are located in random positions inside the hidden area. Because of the object clustering algorithm, the robots start grouping the objects into a common cluster (b). The operator then starts searching for the hidden area by controlling one of the robots (c). The operator finds the gap to the hidden area (d). The operator attempts to orientate the robots' front in the correct direction (e). The operator leads the robot into the hidden area (f). Finally, the operator explores the newly discovered area (g-h).

5.4 Summary

This chapter presented a distributed human-swarm interaction scheme in which operators had the same lack of global situational awareness as in the previous study, and limited access to information regarding the swarm task and the mission objective while aiding a swarm. The operators had access to upgraded swarm controls with which they were able to monitor the swarm and control one robot at a time. When given the first objective with basic information, operators did not explore any better than the baseline swarm that lacked of any external interaction.

Also, when given more information regarding the swarm task, the operators had no significant improvement in their performance. However, we observed evidence suggesting that when explicitly given a second objective, operators were able to explore the environment without interfering with the swarm dynamic. This suggests that the effectiveness of the human operator is mission dependant.

Finally, analysis of the participants answers and comments to the questionnaire suggests that the explicitness of the information not only affects the operators' performance, but also part of their perception. Some NIP comments expressed that they felt that their contributions to the swarm were not needed and had no impact at all. The perception of their own curiosity level showed that despite them classifying themselves as highly curious, within the trials, only those who were presented with explicit information implemented some exploration strategies. Similarly, their own perception of how they understood the environment seemed to be linked to the mission objective information. The operators' perception regarding the robot control turned out to be not related to any other factor leaving it totally dependent to each of the operators own perspective.



(a) After 0 seconds

(b) After 35 seconds

(c) After 70 seconds

(d) After 90 seconds

(e) After 110 seconds

(f) After 125 seconds

(g) After 200 seconds

(h) After 250 seconds

Fig. 5.14 Sequence of snapshots taken during a trial of a participant from group “C”. The red dots represent the objects, the green dots represent the robots and the middle wall is represented by the gray line in the middle.

Chapter 6

Conclusions

Swarm robotic systems are still not ready to interact with humans in real world environments. This thesis approached and explored a crucial concept to aid in bringing these systems to such point. It presents the effect that three different awareness levels have over the interaction process that a human operator exerts over a swarm robot. This as an attempt to develop the interaction techniques to be prepared for more realistic constraints that will be present in the real world. The awareness constrains tested the operators' performance through full awareness, limited situational awareness and limited situational and task awareness. In addition, this thesis presents the development and implementation of a GUI that had the capability to overcome the interaction and awareness problems, however dependant of the operators' ability. Overall, we show that humans have the potential to learn how to interact with a swarm robot, how to manage and manipulate multiple robots and how to work as a team with a swarm system, with certain limitations. Furthermore, through all the chapters we present the strengths and modifications that the GUI and the simulation environment had.

Chapter 3 presented a human operator with full awareness that gained control of a swarm of robots. The interaction was applied to a cooperative transport task with physical robots. The introduction of a fully aware operator to the swarm system made possible to dynamically choose the goal location to which an object was going to be transported. Allowing the operator to gain control over the swarm provided with required information so that it could negotiate obstacles. The operator was able to compare between the usage of a hands-free device (the Google Glass) and a portable phone (the Cubot C9+). As the robots had very low communication requirements (no need to explicitly communicate with each other), the operator was able to guide multiple robots of the swarm using simple commands. The operator guidance took effect via the portable device with a single leader robot. Yet, the operator had enough influence over the entire swarm to provide positive global feedback and direct the collective force such that the object would move in the desired direction.

Chapter 4 introduced the situational awareness constraint problem. It showed how the human ability to process data, like global state information, better than the robots helped the swarm to overcome problems through the interaction. With the proposed distributed human-swarm interaction scheme, human operators had access to only local information while aiding the swarm with an aggregation task in an environment with obstructions.

Operators who had access to global state information were able to help the robots to complete the aggregation task successfully. However, when deprived of such global state information, untrained operators did not perform significantly better than random interactions. Nor did they exhibit a significant learning effect within three trials. Furthermore, operators that once were given global state information did not demonstrate improved performance on subsequent trials when being restricted to local information. This suggests no learning benefit from having observed the global dynamics once. However, trained and expert operators showed significantly improved performance. This improvement suggests not only that training of operators had a positive impact, but that the task was solvable with the available controls. These operators compensated the lack of global situational awareness with increased requests for local sensory information while reducing the number of motion commands. Expert operators performed nearly as well as the baseline performance of the autonomous algorithm under ideal conditions, that is, without obstructions.

Finally, Chapter 5 introduced the task awareness constraint along with the situational awareness constraint. This study showed the importance of providing the human operator with explicit information regarding the mission objective and related tasks. Furthermore, it identified a possible problem that might directly affect the motivation of HSI.

When extra information regarding the task or the environment was given to the operators, they still did not perform better. This suggests that the operators were unable to identify needed actions that were required to successfully complete the mission. It raises the question of human interaction becoming futile if the operator is not able to identify required extra actions that are needed to successfully complete the mission. Only when those extra actions were explicitly mentioned as a second mission, were operators able to execute them and understand their role in the team. Only then was the whole system (human and robots) able to benefit from the interaction.

In addition, there were some indications of the existence of neglect benevolence in all studies. First, when the operator waited for the robots to push the object (Chapter 3) and react to the different positions of the robot leader. Then, when trained operators (Chapter 4) waited at the beginning of a trial for the swarm to converge and group into clusters. In this case, interactions with the swarm were more beneficial as emerging clusters could be changed into leader-follower formations more easily. Finally, when the informed operators (Chapter 5) allowed the swarm to work while they performed other tasks, neglecting the object clustering task. In this last case, teamwork arised between the human and the robots by covering more tasks and ground simultaneously.

Overall, our findings suggest that exposure to global swarm dynamics does not necessarily accelerate learning, neither for improving situational awareness nor for understanding the swarm dynamics. In addition, learning to interact with a swarm through a distributed interaction scheme that relies on local information requires training, even for simple tasks and interfaces. We believe that the obtained results will contribute to the continuation of development of better human-swarm robot interaction methodologies. However, the results also reveal that there is still a lot of room for improvement, in order to develop interfaces capable of dealing with real world applications. If these limitations are overcome, swarm systems could be used in real disaster zones as rescue robots, as exploration teams (space exploration, underwater, underground) in dangerous environments, or even as security (surveillance or reconnaissance) distributed systems. Such interfaces should allow the human operator not only to achieve better understanding of the robots environment, but of its own duty as part of a team.

6.1 Future Work

This thesis has generated some new questions: What is the best way to combat the lack of situational and/or task awareness? How should an operator be trained against task awareness? How can the operator deal with the lack of explicitness? Which type of interaction is the most reliable in non stable communication environments? Do human-swarm teams perform better when limited to one operator or with multiple operators? Is the performance of the interaction affected if shared with multiple operators? How will limited task awareness affect a multi-human interaction scheme? The potential of human-swarm robot interaction can still be expanded, nevertheless multiple challenges have to be tackled first.

These questions are just a high-level glance that attempts to approach the human-swarm robot interaction problem so that swarm systems can be used in real world applications. A concrete next step for this thesis would be to explore the presented interaction techniques with real robots. Moreover, the interaction technique could be validated with other platforms, potentially with robots that are already used in the real world.

Finally, the next sections present some possible mayor upgrades to start addressing other questions.

6.1.1 A Psychological Approach

The main focus of this thesis was on the interaction technique requirements and effects. However, study of the impact that limited situational awareness and task awareness have on a psychological level could provide with more insight on proper interface development. Better understanding of the human side could improve further understanding of the awareness effects on the operator self-perception and reactions to different problems and scenarios.

This type of reactions could be important, specially when a human operator in involved with a robotic system in charge of rescue missions with high levels of stress and human lives involved.

6.1.2 Speech Interaction with Limited SA

In Chapter 4 the limitation to the situational awareness of the human operator introduced extra difficulties for them to interact with the swarm. The implementation of speech recognition to the interaction scheme with the swarm while the operator lacks of the bird's eye view could benefit both sides. There are two major benefits that we can identify so far:

- Improve the performance of the interaction commands allowing the operator to worry less about the interaction interface.
- Give more time for the operator to gain situational awareness.

Further work could focus on the interaction impact rather than on the hardware limitations of the implementation. It would prove beneficial to understand and measure the real benefit of speech interaction with limited SA and if it improves the performance of the operator. Specially if this could help the operator to achieve SA easier and faster than with other interface designs.

6.1.3 Multi-user Interaction

Finally, in Chapter 5 the presented study involved multiple awareness constraints (situational and task awareness). The lack of SA proved to be a problem that added complexity to the TA problem. We have reason to believe that the addition of extra operators cooperating simultaneously could lead to an improvement in the task understanding and management of the swarm. As some operators gained SA faster than others, teamwork between operators could lead to a faster understanding on the environment, simplifying the TA challenge.

Future work could focus on the implications of having multi-operators either with full communication, partial communication or even no communication between them. This could lead to further understanding of generic swarm interaction strategies.

6.1.4 Audio as Main Feedback Source

In Chapter 3 command of the swarm was allowed through speech instructions. Speech proved to be a reliable method for the operator to deliver instructions to the swarm, but the feedback was left untouched. Further exploration to use the audio channel as the main feedback source could help the human operator improve the understanding of the robots' perception of the environment.

The implementation of auditive responses as feedback from the swarm would emulate more closely how normal human-to-human interaction works. If these audio feedback could also be mapped to 3D-audio, the feedback could not only give distance translated as intensity, but also position in reference to the robot(s) providing these feedback. In theory, allowing the human operator to not learn how to interpret the robots' feedback through new interfaces and instead focus on the swarm behaviour and task development could benefit the human-swarm teamwork. Furthermore, it could improve the human fan-out and allow swarm systems to increase in number of agents and increase the amount of potential applications.

References

[1] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a survey," *Foundations and trends in human-computer interaction*, vol. 1, no. 3, pp. 203–275, 2007.

[2] K. Dautenhahn, "Socially intelligent robots: dimensions of human–robot interaction," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 362, no. 1480, pp. 679–704, 2007.

[3] M. Dorigo, "Swarm intelligence and swarm robotics-the swarm-bot experiment.," in *ICINCO-ICSO*, pp. 29–29, 2008.

[4] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*, vol. 4. Oxford University Press New York, 1999.

[5] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[6] J. Krause, G. D. Ruxton, and S. Krause, "Swarm intelligence in animals and humans," *Trends in ecology & evolution*, vol. 25, no. 1, pp. 28–34, 2010.

[7] C. Vasile, A. Pavel, and C. Buiu, "Integrating human swarm interaction in a distributed robotic control system," in *Automation Science and Engineering (CASE), 2011 IEEE Conference on*, pp. 743–748, IEEE, 2011.

[8] T. Dewi, N. Uchiyama, S. Sano, and H. Takahashi, "Swarm robot control for human services and moving rehabilitation by sensor fusion," *Journal of Robotics*, vol. 2014, 2014.

[9] S. Doctor, G. K. Venayagamoorthy, and V. G. Gudise, "Optimal PSO for collective robotic search applications," in *Evolutionary Computation, 2004. CEC2004. Congress on*, vol. 2, pp. 1390–1395, IEEE, 2004.

[10] I. R. Nourbakhsh, K. Sycara, M. Koes, M. Yong, M. Lewis, and S. Burion, "Human-robot teaming for search and rescue," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 72–79, 2005.

[11] M. S. Couceiro, R. P. Rocha, and N. M. Ferreira, "A novel multi-robot exploration approach based on particle swarm optimization algorithms," in *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*, pp. 327–332, IEEE, 2011.

[12] R. R. Murphy, “Human-robot interaction in rescue robotics,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, no. 2, pp. 138–153, 2004.

[13] S. Bashyal and G. K. Venayagamoorthy, “Human swarm interaction for radiation source search and localization,” in *Swarm Intelligence Symposium, 2008. SIS 2008. IEEE*, pp. 1–8, IEEE, 2008.

[14] M. Cummings, “Human supervisory control of swarming networks,” in *2nd Annual Swarming: Autonomous Intelligent Networked Systems Conference*, pp. 1–9, Citeseer, 2004.

[15] J. Guzzi, A. Giusti, L. M. Gambardella, G. Theraulaz, and G. A. Di Caro, “Human-friendly robot navigation in dynamic environments,” in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 423–430, IEEE, 2013.

[16] N. Miyata, J. Ota, T. Arai, and H. Asama, “Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment,” *Transactions on Robotics and Automation, IEEE*, vol. 18, no. 5, pp. 769–780, 2002.

[17] H. Ding and H. Hamann, “Sorting in swarm robots using communication-based cluster size estimation,” in *International Conference on Swarm Intelligence*, pp. 262–269, Springer, 2014.

[18] S. Magnenat and Laboratory of Intelligent Systems, EPFL, Lausanne, “Enki reference documentation.” http://lis2.epfl.ch/resources/download/doc1.0/libenki/, 2005. [Online; accessed 25-August-2015].

[19] G. Kapellmann-Zafra, “Upgraded enki simulator.” https://github.com/gkapellmann/epuckEnkiSimulator, 2014–2017.

[20] P. Amstutz and A. H. Fagg, “Real time visualization of robot state with mobile virtual reality,” in *Robotics and Automation, 2002. Proceedings. ICRA’02. IEEE International Conference on*, vol. 1, pp. 241–247, IEEE, 2002.

[21] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Gross, “Occlusion-based cooperative transport with a swarm of miniature mobile robots,” *Transactions on Robotics, IEEE*, vol. 31, no. 2, pp. 307–321, 2015.

[22] Z.-D. Wang, Y. Hirata, Y. Takano, and K. Kosuge, “From human to pushing leader robot: Leading a decentralized multirobot system for object handling,” in *International Conference on Robotics and Biomimetics, 2004. ROBIO 2004. IEEE*, pp. 441–446, IEEE, 2004.

[23] M. Gauci, , J. Chen, W. Li, T. J. Dodd, and R. Groß, “Self-organized aggregation without computation,” *The International Journal of Robotics Research*, vol. 33, no. 8, pp. 1145–1161, 2014.

[24] D. S. Brown and M. A. Goodrich, “Limited bandwidth recognition of collective behaviors in bio-inspired swarms,” in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pp. 405–412, International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[25] D. S. Brown, S. C. Kerman, and M. A. Goodrich, "Human-swarm interactions based on managing attractors," in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pp. 90–97, ACM, 2014.

[26] P. Walker, S. Amirpour Amraii, N. Chakraborty, M. Lewis, and K. Sycara, "Human control of robot swarms with dynamic leaders," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pp. 1108–1113, IEEE, Springer-Verlag, 2014.

[27] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Gross, "Clustering objects with robots that do not compute," in *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pp. 421–428, International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[28] J. C. Scholtz, "Human-robot interactions: Creating synergistic cyber forces," in *Multi-Robot Systems: From Swarms to Intelligent Automata*, pp. 177–184, Springer, 2002.

[29] A. Couture-Beil, R. T. Vaughan, and G. Mori, "Selecting and commanding individual robots in a multi-robot system," in *Computer and Robot Vision (CRV), 2010 Canadian Conference on*, pp. 159–166, IEEE, 2010.

[30] T. B. Sheridan, "Human–robot interaction status and challenges," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 58, no. 4, pp. 525–532, 2016.

[31] J. C. Barca and Y. A. Sekercioglu, "Swarm robotics reviewed," *Robotica*, vol. 31, no. 03, pp. 345–359, 2013.

[32] G. Kapellmann-Zafra, J. Chen, and R. Groß, "Using google glass in human–robot swarm interaction," in *Proc. 17th Annual Conference Towards Autonomous Robotic Systems (TAROS 2016)*, vol. 9716 of *Lecture Notes in Artificial Intelligence*, Springer-Verlag, 2016.

[33] G. Kapellmann-Zafra, N. Salomons, A. Kolling, and R. Groß, "Human-robot swarm interaction with limited situational awareness," in *International Conference on Swarm Intelligence*, pp. 125–136, Springer, 2016.

[34] I. Navarro and F. Matía, "An introduction to swarm robotics," *ISRN Robotics*, vol. 2013, 2012.

[35] E. Şahin, "Swarm robotics: From sources of inspiration to domains of application," in *International workshop on swarm robotics*, pp. 10–20, Springer, 2004.

[36] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, "Swarm-bot: A new distributed robotic concept," *Autonomous robots*, vol. 17, no. 2-3, pp. 193–221, 2004.

[37] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.

[38] G. Beni, "From swarm intelligence to swarm robotics," in *International Workshop on Swarm Robotics*, pp. 1–9, Springer, 2004.

[39] E. Şahin and A. Winfield, "Special issue on swarm robotics," *Swarm Intelligence*, vol. 2, pp. 69–72, Dec 2008.

[40] L. E. Parker, "Multiple mobile robot systems," *Springer Handbook of Robotics*, pp. 921–941, 2008.

[41] J. Nagi, H. Ngo, L. Gambardella, and G. A. Di Caro, "Wisdom of the swarm for cooperative decision-making in human-swarm interaction," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1802–1808, IEEE, 2015.

[42] C. E. Harriott, A. E. Seiffert, S. T. Hayes, and J. A. Adams, "Biologically-inspired human-swarm interaction metrics," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 58, pp. 1471–1475, SAGE Publications, 2014.

[43] H. A. Yanco and J. L. Drury, "A taxonomy for human-robot interaction," in *Proceedings of the AAAI Fall Symposium on Human-Robot Interaction*, pp. 111–119, 2002.

[44] G. Hoffman, G. E. Birnbaum, K. Vanunu, O. Sass, and H. T. Reis, "Robot responsiveness to human disclosure affects social impression and appeal," in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pp. 1–8, ACM, 2014.

[45] M. A. Salichs, R. Barber, A. M. Khamis, M. Malfaz, J. F. Gorostiza, R. Pacheco, R. Rivas, A. Corrales, E. Delgado, and D. García, "Maggie: A robotic platform for human-robot social interaction," in *Robotics, Automation and Mechatronics, 2006 IEEE Conference on*, pp. 1–7, IEEE, 2006.

[46] K. Dautenhahn, S. Woods, C. Kaouri, M. L. Walters, K. L. Koay, and I. Werry, "What is a robot companion-friend, assistant or butler?," in *Intelligent Robots and Systems.(IROS 2005). IEEE/RSJ International Conference on*, pp. 1192–1197, IEEE, 2005.

[47] H. Fang, S. Ong, and A. Nee, "Interactive robot trajectory planning and simulation using augmented reality," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 227–237, 2012.

[48] S. Squyres, R. Arvidson, J. F. Bell, J. Brückner, N. Cabrol, W. Calvin, M. Carr, P. Christensen, B. Clark, L. Crumpler, *et al.*, "The opportunity rover's athena science investigation at meridiani planum, mars," *Science*, vol. 306, no. 5702, pp. 1698–1703, 2004.

[49] A.-B. Karami, L. Jeanpierre, and A.-I. Mouaddib, "Human-robot collaboration for a shared mission," in *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pp. 155–156, IEEE, 2010.

[50] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

[51] D. Michel, A. Qammaz, and A. A. Argyros, “Markerless 3d human pose estimation and tracking based on rgbd cameras: an experimental evaluation,” in *Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 115–122, ACM, 2017.

[52] M. Katzenmaier, R. Stiefelhagen, and T. Schultz, “Identifying the addressee in human-human-robot interactions based on head pose and speech,” in *Proceedings of the 6th international conference on Multimodal interfaces*, pp. 144–151, ACM, 2004.

[53] R. Stiefelhagen, C. Fugen, R. Gieselmann, H. Holzapfel, K. Nickel, and A. Waibel, “Natural human-robot interaction using speech, head pose and gestures,” in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3, pp. 2422–2427, IEEE, 2004.

[54] J. Scholtz, “Theory and evaluation of human robot interactions,” in *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pp. 10–pp, IEEE, 2003.

[55] A. Steinfeld, “Interface lessons for fully and semi-autonomous mobile robots,” in *Robotics and Automation, 2004. Proceedings. ICRA’04. 2004 IEEE International Conference on*, vol. 3, pp. 2752–2757, IEEE, 2004.

[56] C. Lee and Y. Xu, “Online, interactive learning of gestures for human/robot interfaces,” in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 4, pp. 2982–2987, IEEE, 1996.

[57] A. D. Jacq, S. Lemaignan, F. Garcia, P. Dillenbourg, and A. Paiva, “Building successful long child-robot interactions in a learning context,” in *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, pp. 239–246, IEEE Press, 2016.

[58] C. Finn, I. Goodfellow, and S. Levine, “Unsupervised learning for physical interaction through video prediction,” in *Advances in Neural Information Processing Systems*, pp. 64–72, 2016.

[59] D. J. Atkinson and M. H. Clark, “Methodology for study of human-robot social interaction in dangerous situations,” in *Proceedings of the second international conference on Human-agent interaction*, pp. 371–376, ACM, 2014.

[60] A. Kolling, K. Sycara, S. Nunnally, and M. Lewis, “Human swarm interaction: An experimental study of two types of interaction with foraging swarms,” *Journal of Human-Robot Interaction*, vol. 2, no. 2, pp. 103–128, 2013.

[61] N. Nishikawa, R. Suzuki, and T. Arita, “Coordination control design of heterogeneous swarm robots by means of task-oriented optimization,” *Artificial Life and Robotics*, pp. 1–12, 2016.

[62] H. A. Yanco and J. Drury, “" where am i?" acquiring situation awareness using a remote robot platform,” in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 3, pp. 2835–2840, IEEE, 2004.

[63] M. Yu, Y. Lin, D. Schmidt, X. Wang, and Y. Wang, "Human-robot interaction based on gaze gestures for the drone teleoperation," *Journal of Eye Movement Research*, vol. 7, no. 4, pp. 1–14, 2014.

[64] J. Crossman, R. Marinier, and E. B. Olson, "A hands-off, multi-robot display for communicating situation awareness to operators," in *Collaboration Technologies and Systems (CTS), 2012 International Conference on*, pp. 109–116, IEEE, 2012.

[65] D. R. Olsen Jr and S. B. Wood, "Fan-out: measuring human control of multiple robots," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 231–238, ACM, 2004.

[66] S. Pourmehr, V. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "A robust integrated system for selecting and commanding multiple mobile robots," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 2874–2879, IEEE, 2013.

[67] J. A. Frank, S. P. Krishnamoorthy, and V. Kapila, "Towards mobile mixed-reality interaction with multi-robot systems," *IEEE Robotics and Automation Letters*, 2017.

[68] F. Ghiringhelli, J. Guzzi, G. A. Di Caro, V. Caglioti, L. M. Gambardella, and A. Giusti, "Interactive augmented reality for understanding and analyzing multi-robot systems," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pp. 1195–1201, IEEE, 2014.

[69] J. Banfi, N. Basilico, and F. Amigoni, "Minimizing communication latency in multirobot situation-aware patrolling," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 616–622, IEEE, 2015.

[70] E. Sklar, E. Schneider, A. Ozgelen, and M. Azhar, "Toward human/multi-robot systems to support emergency services agencies," in *2014 AAAI Fall Symposium Series*, 2014.

[71] T. Setter, A. Fouraker, H. Kawashima, and M. Egerstedt, "Haptic interactions with multi-robot swarms using manipulability," *Journal of Human-Robot Interaction*, vol. 4, no. 1, pp. 60–74, 2015.

[72] J. Alonso-Mora, S. Haegeli Lohaus, P. Leemann, R. Siegwart, and P. Beardsley, "Gesture based human-multi-robot swarm interaction and its application to an interactive display," in *International Conference on Robotics and Automation (ICRA), 2015 IEEE*, pp. 5948–5953, IEEE, 2015.

[73] A. Kolling, P. Walker, N. Chakraborty, K. Sycara, and M. Lewis, "Human interaction with robot swarms: A survey," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 9–26, 2016.

[74] Z.-Q. Mi and Y. Yang, "Human-robot interaction in uvs swarming: A survey," *IJCSI International Journal of Computer Science Issues*, vol. 10, 2013.

[75] B. Pendleton and M. Goodrich, "Scalable human interaction with robotic swarms," in *proceedings of the AIAA Infotech@ Aerospace Conference*, 2013.

[76] G. Podevijn, R. O'Grady, N. Mathews, A. Gilles, C. Fantini-Hauwel, and M. Dorigo, "Investigating the effect of increasing robot group sizes on the human psychophysiological state in the context of human–swarm interaction," *Swarm Intelligence*, vol. 10, no. 3, pp. 193–210, 2016.

[77] A. Rossi, M. Staffa, A. Origlia, and S. Rossi, "An analysis of perceptual cues in robot group selection tasks," in *Robot and Human Interactive Communication (RO-MAN), 2015 24th IEEE International Symposium on*, pp. 588–593, IEEE, 2015.

[78] W. Hönig, C. Milanes, L. Scaria, T. Phan, M. Bolas, and N. Ayanian, "Mixed reality for robotics," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 5382–5387, IEEE, 2015.

[79] C. E. Harriott, T. Zhang, G. L. Buford, and J. A. Adams, "Measuring human workload in a collaborative human-robot team," *Journal of Human-Robot Interaction*, vol. 4, no. 2, pp. 61–96, 2015.

[80] M. A. Goodrich, B. Pendleton, P. Sujit, and J. Pinto, "Toward human interaction with bio-inspired robot teams," in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, pp. 2859–2864, IEEE, 2011.

[81] B. S. Perelman, S. T. Mueller, and K. E. Schaefer, "Evaluating path planning in human-robot teams: Quantifying path agreement and mental model congruency," in *2017 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA)*, pp. 1–7, March 2017.

[82] S. Nunnally, P. Walker, M. Lewis, A. Kolling, N. Chakraborty, and K. Sycara, "Robotic swarm connectivity with human operation and bandwidth limitations," in *2012 AAAI Fall Symposium Series*, 2012.

[83] J. McLurkin, J. Smith, J. Frankel, D. Sotkowitz, D. Blau, and B. Schmidt, "Speaking swarmish: Human-robot interface design for large swarms of autonomous mobile robots.," in *AAAI Spring Symposium: To Boldly Go Where No Human-Robot Team Has Gone Before*, pp. 72–75, 2006.

[84] G. K. Karavas and P. Artemiadis, "On the effect of swarm collective behavior on human perception: Towards brain-swarm interfaces," in *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2015 IEEE International Conference on*, pp. 172–177, IEEE, 2015.

[85] V. M. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori, "HRI in the sky: Creating and commanding teams of UAV's with a vision-mediated gestural interface," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 617–623, IEEE, 2013.

[86] S. Hauri, J. Alonso-Mora, A. Breitenmoser, R. Siegwart, and P. Beardsley, "Multi-robot formation control via a real-time drawing interface," in *Field and Service Robotics*, pp. 175–189, Springer, 2014.

[87] G. Podevijn, R. O'Grady, and M. Dorigo, "Self-organised feedback in human swarm interaction," in *Proceedings of the workshop on robot feedback in human-robot interaction: how to make a robot readable for a human interaction partner (Ro-Man 2012)*, 2012.

[88] M. Daily, Y. Cho, K. Martin, and D. Payton, "World embedded interfaces for human-robot interaction," in *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pp. 6–pp, IEEE, 2003.

[89] J.-P. de la Croix and M. Egerstedt, "A control lyapunov function approach to human-swarm interactions," in *American Control Conference (ACC), 2015*, pp. 4368–4373, IEEE, 2015.

[90] M. A. Goodrich, B. Pendleton, S. Kerman, and P. Sujit, "What types of interactions do bio-inspired robot swarms and flocks afford a human?," in *Proc. 2013 Robotics: Science and Systems (RSS VIII)*, pp. 105–112, MIT Press, 2013.

[91] A. E. Turgut, F. Gokce, H. Celikkanat, L. Bayindir, and E. Sahin, "Kobot: A mobile robot designed specifically for swarm robotics research," *Middle East Technical University, Ankara, Turkey, METUCENG-TR Tech. Rep*, vol. 5, p. 2007, 2007.

[92] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," in *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, pp. 59–65, 2009.

[93] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 3293–3298, IEEE, 2012.

[94] M. Le Goc, L. H. Kim, A. Parsaei, J.-D. Fekete, P. Dragicevic, and S. Follmer, "Zooids: Building blocks for swarm user interfaces," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 97–109, ACM, 2016.

[95] Y. Diaz-Mercado, S. G. Lee, and M. Egerstedt, "Distributed dynamic density coverage for human-swarm interactions," in *American Control Conference (ACC), 2015*, pp. 353–358, IEEE, 2015.

[96] A. Kolling, S. Nunnally, and M. Lewis, "Towards human control of robot swarms," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pp. 89–96, ACM, 2012.

[97] M. A. Goodrich, B. Pendleton, S. Kerman, and P. Sujit, "What types of interactions do bio-inspired robot swarms and flocks afford a human?," *Robotics: Science and Systems VIII, Sydney*, 2012.

[98] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective memory and spatial sorting in animal groups," *Journal of Theoretical Biology*, vol. 218, no. 1, pp. 1–11, 2002.

[99] H. Hexmoor, B. McLaughlan, and M. Baker, "Swarm control in unmanned aerial vehicles," in *Proceedings of the International Conference on Artificial Intelligence*, pp. 911–917, 2005.

[100] L. Christopher, W. Boler, C. Wieczorek, J. Crespo, P. Witcher, S. A. Hawkins, and J. Stewart, "Asset allocation with swarm/human blended intelligence," in *Swarm/Human Blended Intelligence Workshop (SHBI), 2016*, pp. 1–5, IEEE, 2016.

[101] S. Hauert and S. N. Bhatia, "Mechanisms of cooperation in cancer nanomedicine: towards systems nanotechnology," *Trends in biotechnology*, vol. 32, no. 9, pp. 448–455, 2014.

[102] P. S. S. Kim, A. Becker, Y. Ou, A. A. Julius, and M. J. Kim, "Swarm control of cell-based microrobots using a single global magnetic field," in *Ubiquitous Robots and Ambient Intelligence (URAI), 2013 10th International Conference on*, pp. 21–26, IEEE, 2013.

[103] S. Shahrokhi and A. T. Becker, "Object manipulation and position control using a swarm with global inputs," in *Automation Science and Engineering (CASE), 2016 IEEE International Conference on*, pp. 561–566, IEEE, 2016.

[104] P. Walker, S. A. Amraii, M. Lewis, N. Chakraborty, and K. Sycara, "Control of swarms with multiple leader agents," in *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pp. 3567–3572, IEEE, 2014.

[105] V. M. Trifa, C. M. Cianci, and D. Guinard, "Dynamic control of a robotic swarm using a service-oriented architecture," in *13th International Symposium on Artificial Life and Robotics (AROB 2008)*, 2008.

[106] D. B. Kaber, E. Onal, and M. R. Endsley, "Design of automation for telerobots and the effect on performance, operator situation awareness, and subjective workload," *Human Factors and Ergonomics in Manufacturing*, vol. 10, no. 4, pp. 409–430, 2000.

[107] Z. Kira and M. A. Potter, "Exerting human control over decentralized robot swarms," in *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on*, pp. 566–571, IEEE, 2009.

[108] J. Y. Chen, E. C. Haas, and M. J. Barnes, "Human performance issues and user interface design for teleoperated robots," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1231–1245, 2007.

[109] S. T. Hayes and J. A. Adams, "Human-swarm interaction: sources of uncertainty," in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pp. 170–171, ACM, 2014.

[110] S. Nunnally, P. Walker, A. Kolling, N. Chakraborty, M. Lewis, K. Sycara, and M. Goodrich, "Human influence of robotic swarms with bandwidth and localization issues," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 333–338, IEEE, 2012.

[111] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[112] H. Hu, X. Gao, H. Sun, Q. Jia, and Y. Zhang, "Design and implementation of the teleoperation platform based on augmented reality," in *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2013 12th IEEE International Conference on*, pp. 126–132, IEEE, 2013.

[113] R. T. Azuma *et al.*, "A survey of augmented reality," *Presence*, vol. 6, no. 4, pp. 355–385, 1997.

[114] P. Walker, S. Nunnally, M. Lewis, A. Kolling, N. Chakraborty, and K. Sycara, "Neglect benevolence in human control of swarms in the presence of latency," in *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, pp. 3009–3014, IEEE, 2012.

[115] D. Kedar and S. Arnon, "Urban optical wireless communication networks: the main challenges and possible solutions," *IEEE Communications Magazine*, vol. 42, no. 5, pp. S2–S7, 2004.

[116] L. Lanbo, Z. Shengli, and C. Jun-Hong, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 8, pp. 977–994, 2008.

[117] N. Ahmed, S. S. Kanhere, and S. Jha, "The holes problem in wireless sensor networks: a survey," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 2, pp. 4–18, 2005.

[118] A. M. Naghsh, J. Gancet, A. Tanoto, J. Penders, C. R. Roast, and M. Ilzkovitz, "Human robot interaction in guardians," in *Proc. of the EURON/IARP International Workshop on Robotics for Risky Interventions and Surveillance of the Environment. Benicassim, Spain*, 2008.

[119] S. Nagavalli, S.-Y. Chien, M. Lewis, N. Chakraborty, and K. Sycara, "Bounds of neglect benevolence in input timing for human interaction with robotic swarms," in *Proc. 10th Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI 2015)*, pp. 197–204, ACM, 2015.

[120] D. R. Olsen and M. A. Goodrich, "Metrics for evaluating human-robot interactions," in *Proc. 2003 Workshop on Performance Metrics for Intelligent Systems (PerMIS 2013)*, vol. 2003, p. 5, 2003.

[121] F. E. Ritter, D. Van Rooy, R. S. Amant, and K. Simpson, "Providing user models direct access to interfaces: An exploratory study of a simple interface with implications for hri and hci," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 36, no. 3, pp. 592–601, 2006.

[122] E. C. Ferrer, "A wearable general-purpose solution for human-swarm interaction," *arXiv preprint arXiv:1704.08393*, 2017.

[123] S. Nunnally, P. Walker, M. Lewis, N. Chakraborty, and K. Sycara, “Using haptic feedback in human robotic swarms interaction,” in *Proc. 2013 Annual Meeting Human Factors and Ergonomics Society*, vol. 57, pp. 1047–1051, SAGE Publications, 2013.

[124] F. Annaz, “Real time robot navigation in virtually created environments,” *International Journal of Electronics, Computer and Communications Technologies*, vol. 3, no. 4, pp. 7–13, 2013.

[125] R. Grieder, J. Alonso-Mora, C. Bloechlinger, R. Siegwart, and P. Beardsley, “Multi-robot control and interaction with a hand-held tablet,” in *Workshop Proc. Int. Conf. Robotics and Automation*, vol. 131, 2014.

[126] K. Kawamura, P. Nilas, K. Muguruma, J. A. Adams, and C. Zhou, “An agent-based architecture for an adaptive human-robot interface,” in *System sciences, 2003. proceedings of the 36th annual hawaii international conference on*, pp. 8–pp, IEEE, 2003.

[127] S. Pourmehr, V. M. Monajjemi, R. Vaughan, and G. Mori, ““you two! take off!”: Creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands,” in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 137–142, IEEE, 2013.

[128] A. J. Silva, O. A. D. Ramirez, V. P. Vega, and J. P. O. Oliver, “Phantom omni haptic device: Kinematic and manipulability,” in *Electronics, Robotics and Automotive Mechanics Conference, 2009. CERMA’09.*, pp. 193–198, IEEE, 2009.

[129] D. Sieber, S. Musić, and S. Hirche, “Multi-robot manipulation controlled by a human with haptic feedback,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 2440–2446, IEEE, 2015.

[130] T. L. Sanders, T. Wixon, K. E. Schafer, J. Y. Chen, and P. Hancock, “The influence of modality and transparency on trust in human-robot interaction,” in *Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2014 IEEE International Inter-Disciplinary Conference on*, pp. 156–159, IEEE, 2014.

[131] M. Jacobsson, Y. Fernaeus, and E. Holmquist, “Glowbots: designing and implementing engaging human-robot interaction,” *Journal of Physical Agents*, vol. 2, no. 2, pp. 51–60, 2008.

[132] M. Fields, E. Haas, S. Hill, C. Stachowiak, and L. Barnes, “Effective robot team control methodologies for battlefield applications,” in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5862–5867, IEEE, 2009.

[133] A. M. Naghsh, J. Gancet, A. Tanoto, and C. Roast, “Analysis and design of human-robot swarm interaction in firefighting,” in *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, pp. 255–260, IEEE, 2008.

[134] “Cambridge advanced learner’s dictionary & thesaurus,” 2017.

[135] J. L. Drury, J. Scholtz, and H. A. Yanco, "Awareness in human-robot interactions," in *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, vol. 1, pp. 912–918, IEEE, 2003.

[136] J. E. Mercado, M. A. Rupp, J. Y. Chen, M. J. Barnes, D. Barber, and K. Procci, "Intelligent agent transparency in human–agent teaming for multi-uxv management," *Human factors*, vol. 58, no. 3, pp. 401–415, 2016.

[137] M. R. Endsley, "Situation awareness global assessment technique (SAGAT)," in *Aerospace and Electronics Conference, 1988. NAECON 1988., Proceedings of the IEEE 1988 National*, pp. 789–795, IEEE, 1988.

[138] N. Stanton, P. Salmon, G. Walker, E. Salas, and P. Hancock, "State-of-science: Situation awareness in individuals, teams and systems," *Ergonomics*, no. just-accepted, pp. 1–33, 2017.

[139] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 37, no. 1, pp. 32–64, 1995.

[140] M. R. Endsley and D. Garland, "Theoretical underpinnings of situation awareness: A critical review," *Situation awareness analysis and measurement*, pp. 3–32, 2000.

[141] C. T. Recchiuto, A. Sgorbissa, and R. Zaccaria, "Visual feedback with multiple cameras in a uavs human–swarm interface," *Robotics and Autonomous Systems*, vol. 80, pp. 43–54, 2016.

[142] J. Scholtz, B. Antonishek, and J. Young, "Evaluation of a human-robot interface: Development of a situational awareness methodology," in *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the*, pp. 9–pp, Citeseer, 2004.

[143] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, "Common metrics for human-robot interaction," in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pp. 33–40, ACM, 2006.

[144] C. Gutwin, G. Stark, and S. Greenberg, "Support for workspace awareness in educational groupware," in *The first international conference on Computer support for collaborative learning*, pp. 147–156, L. Erlbaum Associates Inc., 1995.

[145] J. Fransen, P. A. Kirschner, and G. Erkens, "Mediating team effectiveness in the context of collaborative learning: The importance of team and task awareness," *Computers in Human Behavior*, vol. 27, no. 3, pp. 1103–1113, 2011.

[146] G. L. M. Schellings and H. Broekkamp, "Signaling task awareness in think-aloud protocols from students selecting relevant information from text," *Metacognition and Learning*, vol. 6, pp. 65–82, Apr 2011.

[147] R. M. Ratwani, J. M. McCurry, and J. G. Trafton, "Single operator, multiple robots: an eye movement based theoretic model of operator situation awareness," in *Proceedings of the 5th ACM/IEEE International Conference on Human-robot Interaction*, pp. 235–242, IEEE Press, 2010.

[148] I. Rae, B. Mutlu, and L. Takayama, "Bodies in motion: mobility, presence, and task awareness in telepresence," in *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pp. 2153–2162, ACM, 2014.

[149] J. Beer, A. D. Fisk, and W. A. Rogers, "Toward a framework for levels of robot autonomy in human-robot interaction," *Journal of Human-Robot Interaction*, vol. 3, no. 2, p. 74, 2014.

[150] C. R. Kube and H. Zhang, "Task modelling in collective robotics," *Autonomous Robots*, vol. 4, no. 1, pp. 53–72, 1997.

[151] G. A. S. Pereira, M. F. M. Campos, and V. Kumar, "Decentralized algorithms for multi-robot manipulation via caging," *The International Journal of Robotics Research*, vol. 23, no. 7–8, pp. 783–795, 2004.

[152] E. Tuci, R. Groß, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo, "Cooperation through self-assembly in multi-robot systems," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 1, no. 2, pp. 115–150, 2006.

[153] R. Großand M. Dorigo, "Evolution of solitary and group transport behaviors for autonomous robots capable of self-assembling," *Adaptive Behavior*, vol. 16, no. 5, pp. 285–305, 2008.

[154] G. Kapellmann-Zafra, J. Chen, and R. Groß, *Using Google Glass in Human-Robot Swarm Interaction Video*, 2016.

[155] N. Salomons, G. Kapellmann-Zafra, and R. Groß, "Human management of a robotic swarm," in *Proc. 17th Annual Conference Towards Autonomous Robotic Systems (TAROS 2016)*, Lecture Notes in Artificial Intelligence, Springer-Verlag, 2016.

Appendix A

Android Software

The developed interface for the portable devices had as objective to facilitate the interaction between the operator and the leader robot. The complete code from the Google Glass interface as well as from the mobile phone were developed in the Java environment. The "Eclipse" compiler was used in both cases as developing interface. The next section presents the main structure of the code where both software were based.

A.1 Code Structure

Both projects had the "MainActivity" class and the "MyBroadcastReceiver" class. The first one being the main program manager and the second one in charge of managing the bluetooth connections.

```
public class MainActivity extends Activity {
   ...
   private class ConnectThread extends Thread {...}
   private class ConnectedThread extends Thread {...}
   ...
}

public class MyBroadcastReceiver extends BroadcastReceiver{
   MainActivity mainclass;
   ...
```

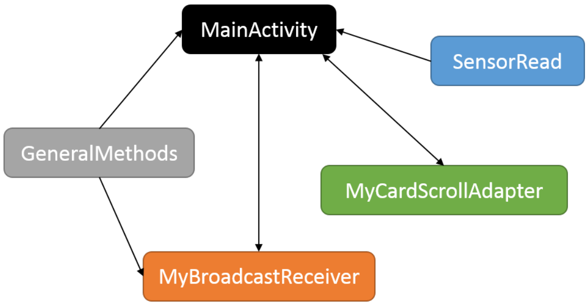


Fig. A.1 Google Glass software architecture.

In addition, the Google Glass interface had another three classes. These additional classes were needed as the interface was more complicated than the mobile phone one. It included a set of extra capabilities, such as voice recognition and a specific graphical design.

A.2 Robot Commands

Once the operator established a connection with a robot, the operator is able to ask for a "status report" by sending a simple byte command to it. The interface then sends a single character command 'r' to the robot. It then answers back a set of characters in an specific order. The current configuration is defined by a starting 'X', followed by a set of char byte characters, each with a different meaning. The complete configuration word is composed of 4 bytes, being the first one always 'X', followed by the other three depending on the status of the robot (see Table A.1):

Status Word Options

Position	*Option* 1	*Option* 2
Second	'L' - Glass Remote Task	'.' - Collective Transport Task
Third	't' - Active Task mode	'.' - Stand-by Mode
Fourth	'm' - Overdrive On	'.' - Overdrive Off

Table A.1 Possible characters combinations and meanings representing the status of the controlled robot.

To illustrate this, an example word that the robot would answer after a "status report" request from an operator could be: "XL.m"

Where 'X' is the initial char, meaning that the robot has sent a status word. Followed by a 'L' meaning that the current task of the robot is the "Google Glass Remote Control Task".

The ‘.’ means that the robot is currently in “Stand-by mode”. Finally, the ‘m’ meaning that the override is active.

Depending on the status of the robot, the operator has the possibility to command a set of instructions. For example, when in Stand-by mode:

- Begin override mode: Ignore all IR remote input, therefore only serial input is effective.
- Stop override mode: Respond to IR remote input as well as serial input.
- Start task mode: Quit Stand-by mode and proceed to execute the current task.
- Stand-by mode: Return to stand-by mode and wait for instructions.
- Change task: Set the “Cooperative Transport” as current task to this.
- Change Task: Set the “Google Glass Remote Control” as the current task.

When executing the *Cooperative Transport* task:

- ‘n’: Stand-by mode - The robot will return to stand-by mode and wait for instructions.

When executing the *Google Glass Remote Control* task:

- Move forwards
- Rotate left
- Move backwards
- Rotate right
- Stop moving
- Turn on front LED
- Turn off front LED
- Send the feedback of the infrared proximity sensors data (#0 and #7)

Appendix B

Gossip Algorithm Theory and Mechanics

The original algorithm was designed as the *Management Algorithm*. Its main purpose was to provide the operator with advanced capabilities to organize a robotic swarm. The *Management* algorithm allows the operator to reassign subgroups of robots to different tasks during run-time, and to verify the status of the robot swarm without requiring direct visual contact with the swarm. The *Gossip* algorithm is the sub-program in charge of the verification of the status of the robot swarm.

The *Gossip* algorithm allows the operator to verify the size of the current cluster to what each robot belongs without requiring direct visual contact with the swarm. It also allows the operator to read the leader robot sensors without the nearby robots moving. The commands are given by the operator to a leader robot through wireless communication. The following commands are available for the operator to use while controlling the *Gossip* algorithm:

- **Change Host Orientation**: The operator can rotate the robot orientation in any direction.
- **Request Sensor Data**: The operator can obtain data from the available sensors of the robot.
- **Request Count**: The operator can request a count of robots belonging to the same cluster group as the leader robot.
- **Release Robots**: The robots relay a “resume” message and perform their newly assigned task.

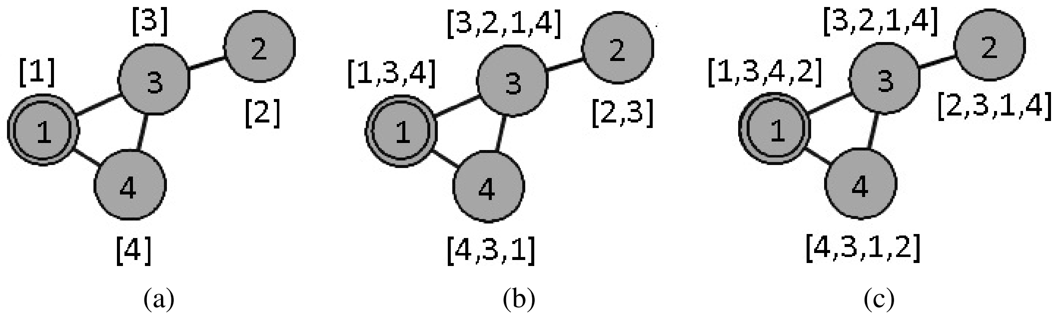


Fig. B.1 Example of robots counting (lines denote pairs of robots within communication range). (a) Initially, only their own ID is in the list. (b) The robots include the IDs received by their neighbors. (c) After a further iteration, each robot contains everyones' IDs in their list. Note that perfect communication is assumed. In practice, the e-pucks may be unable to receive multiple messages simultaneously.

As soon as the *Gossip* algorithm is triggered, the leader robot stops its movement and starts relaying the request to the local neighbour robots to stop as well. When the near robots receive the request they also stop and relay the message as well. The same message is relayed periodically to other robots that might enter into the communication range. By suspending the movement within the group of connected robots, their connectivity can be maintained for as long as the operator requires it.

B.1 Robots Count

The counting works if each robot has a unique ID. When the counting is triggered, the robot creates a list of the IDs. Upon initialization, a robot's list contains only its own ID. This list of IDs represents the neighbours which are in a robot's local cluster. The robot iterates through its list and broadcasts each ID. Once reaching the end of the list, the robot starts over again. Whenever receiving an ID, the robot adds it to its list if it is not already registered.

Figure B.1 illustrates the counting process for a group of four robots. At the beginning, each robot has only its own ID. The list is updated when receiving messages from neighbouring robots. Notice how robots 1 and 2 are not in communication range of each other, but are both in range of robot 3. Robot 3 will pass the ID of robot 1 to robot 2 and vice versa, and thus robot 1 and robot 2 will, after two iterations, also have each other's IDs in their lists.

B.2 Simulation Software

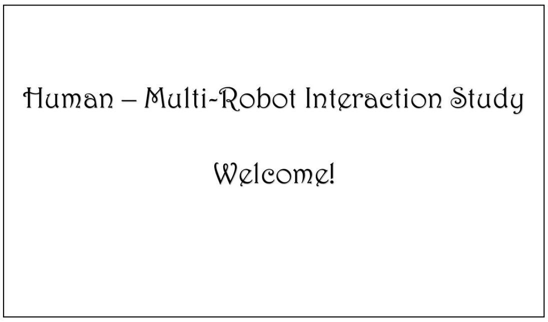
In a simulation platform, the *Gossip* algorithm can be simulated. Following, we present the implementation of the counting function in C++:

```
int countLocalMembers(MyEPuck *EPucks[], int nRobots, int leaderID, bool
    Periodic){
  int p=0, count=1;
  int counted[NumberOfRobots];
  fill_n(counted, NumberOfRobots, -1);
  counted[p] = leaderID;
  do{
    for(int i=0; i<nRobots; i++){ // Check within all initialized
        robots...
        if(distBetweenBots(EPucks[counted[p]]->pos, EPucks[i]->pos) <=
            MaxDist){   // Check for close robots within a range...
            if(find(begin(counted), end(counted), EPucks[i]->getID()) ==
                 end(counted)){ // If doens't exist in focused array...
                counted[j] = EPucks[i]->getID();    // Add for further
                    focus...
                if(Periodic){
                    count++;
                }
            }
        }
    }
    p++;      // Next focused robot...
  }while(p!=count);
  return count;
}
```

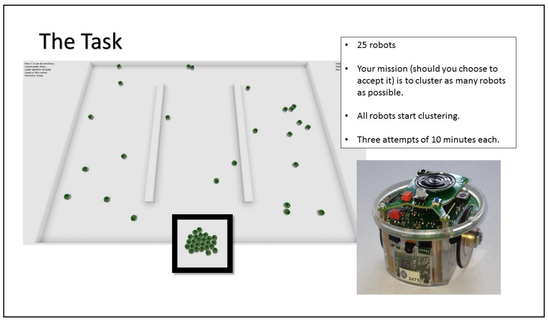
Appendix C

Experiments Presentations

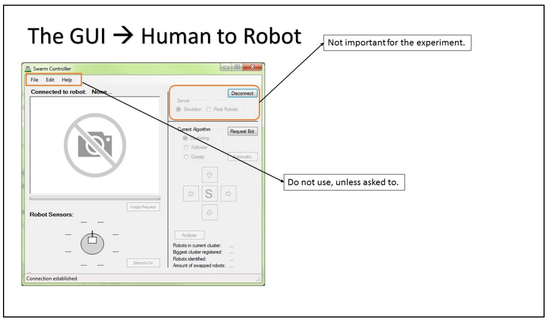
C.1 Experiment 1



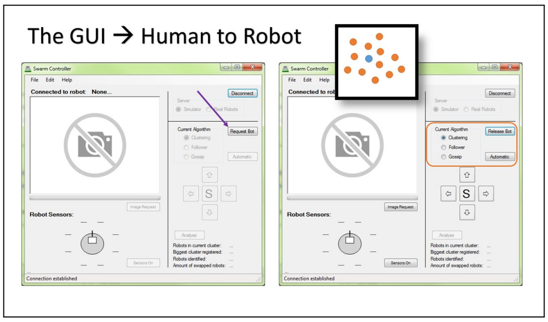
(a)



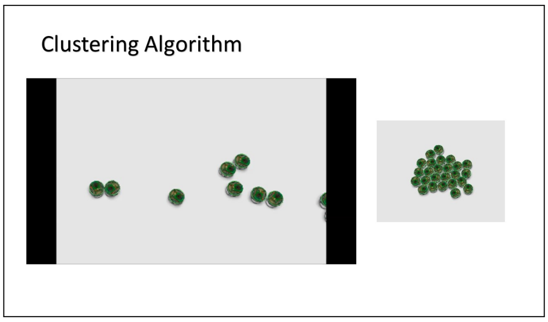
(b)



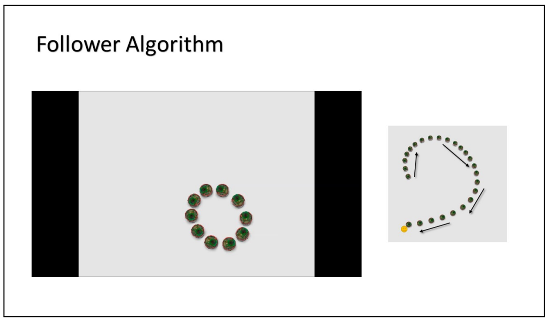
(c)



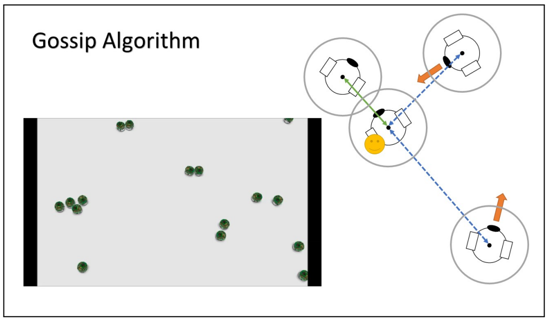
(d)


Clustering Algorithm

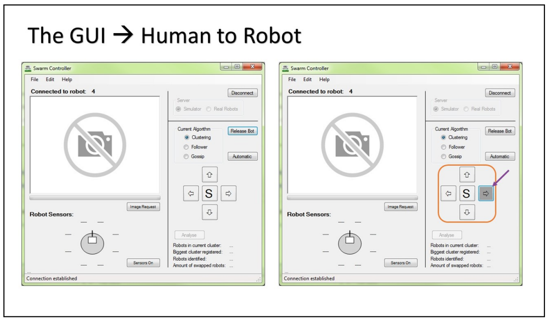
(e)


Follower Algorithm

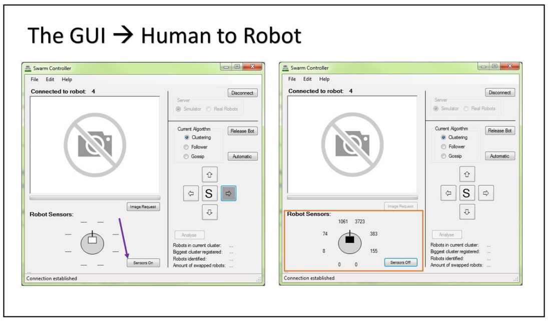
(f)


Gossip Algorithm

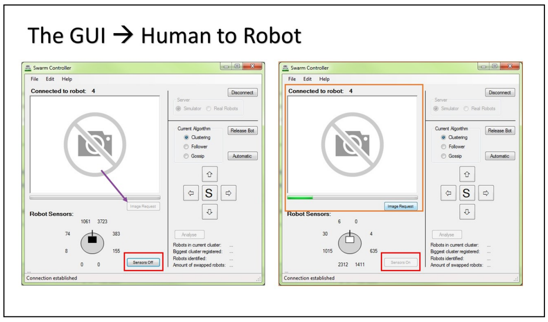
(g)


The GUI → Human to Robot

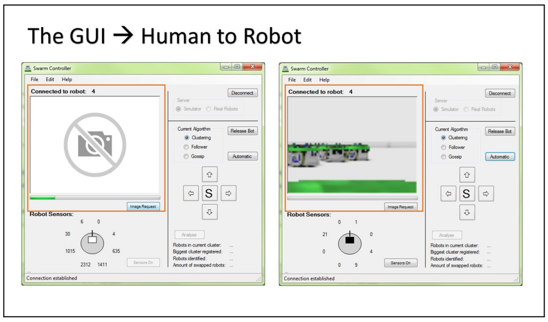
(h)


The GUI → Human to Robot

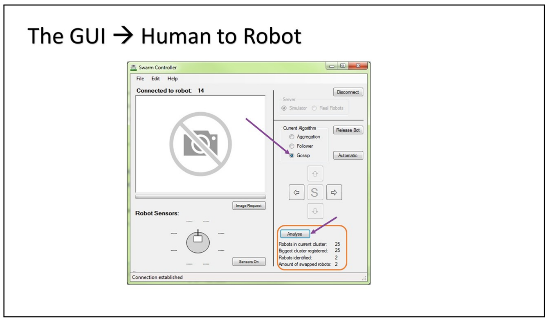
(i)


The GUI → Human to Robot

(j)


The GUI → Human to Robot

(k)


The GUI → Human to Robot

(l)

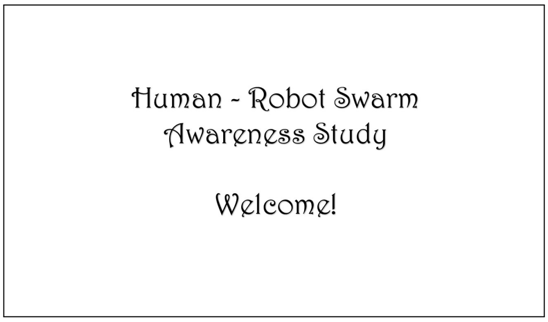

ANY QUESTIONS?
Experiment Setup:
• Trial 1 – 10 minutes
• 5 minute break
• Trial 2 – 10 minutes
• 5 minute break
• Trial 3 – 10 minutes
• Hand in Questionnaires
Thank you for your help!

(m)

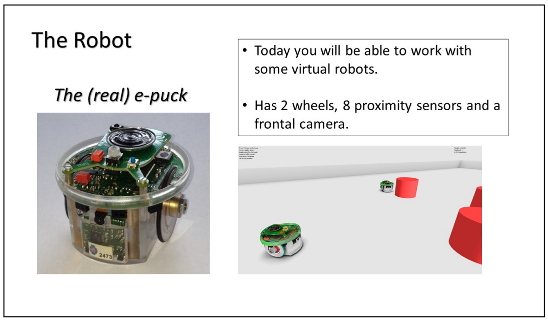

Are you ready?
Have fun!

(n)

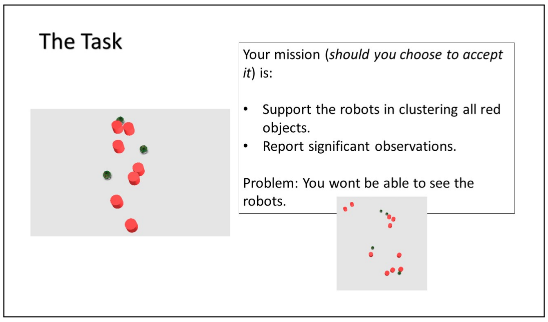
C.2 Experiment 2



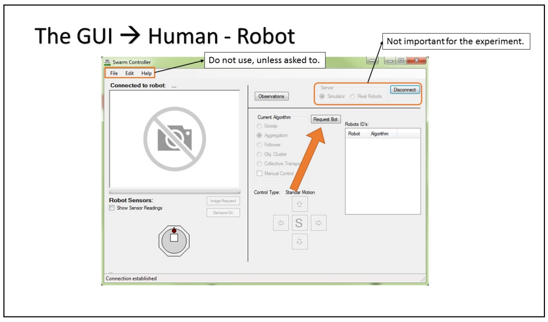
(a)



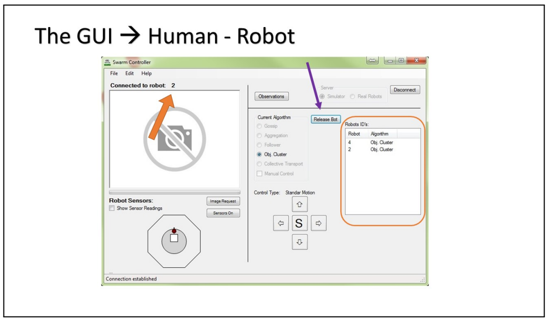
(b)



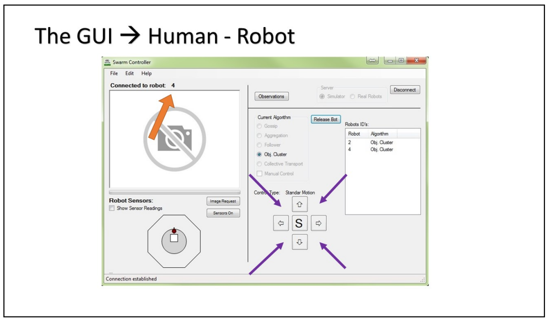
(c)



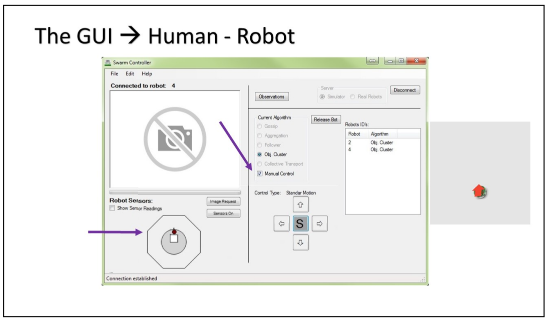
(d)



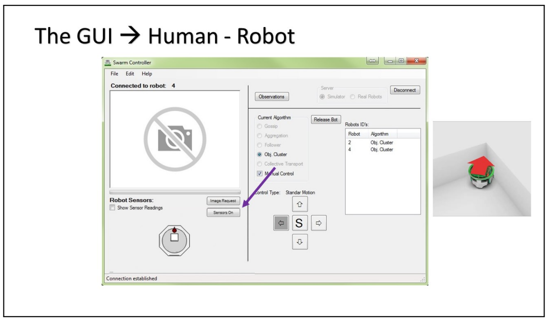
(e)



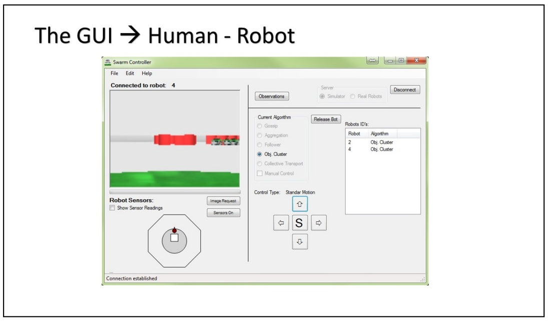
(f)


The GUI → Human - Robot

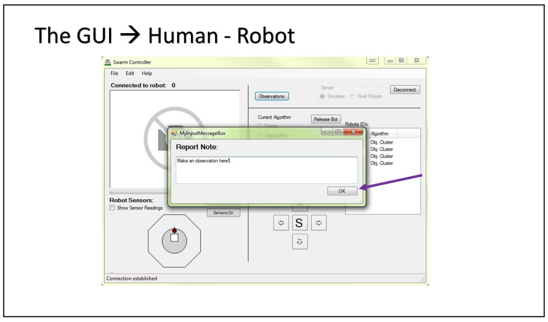
(g)


The GUI → Human - Robot

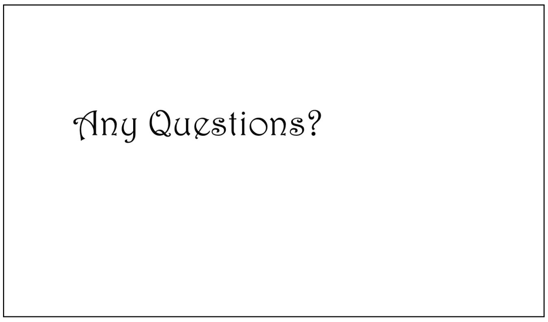
(h)


The GUI → Human - Robot

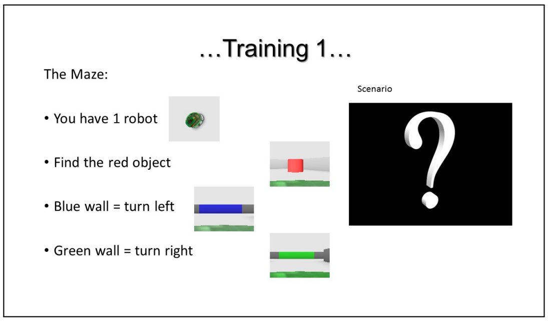
(i)


The GUI → Human - Robot

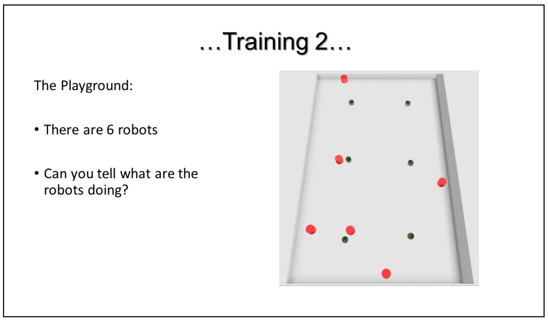
(j)


Any Questions?

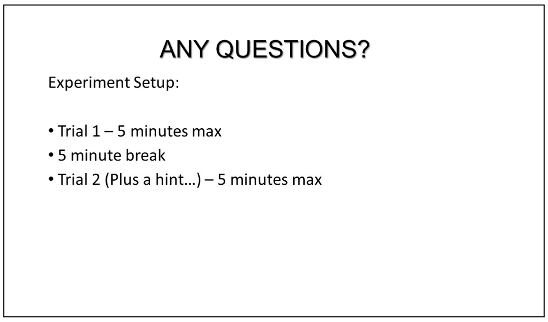
(k)


…Training 1…
The Maze:
• You have 1 robot
• Find the red object
• Blue wall = turn left
• Green wall = turn right
Scenario
?

(l)

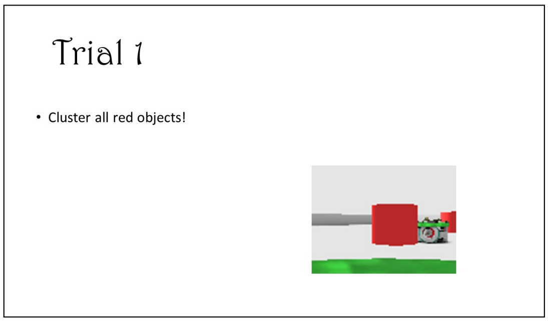

…Training 2…
The Playground:
• There are 6 robots
• Can you tell what are the robots doing?

(m)

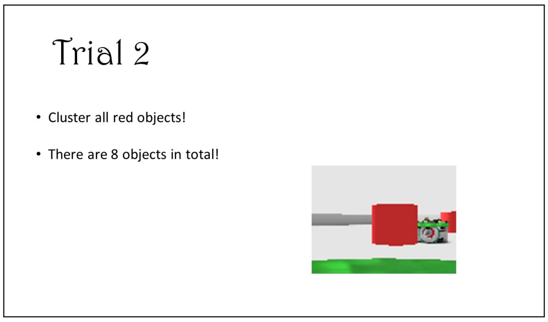

ANY QUESTIONS?
Experiment Setup:
• Trial 1 – 5 minutes max
• 5 minute break
• Trial 2 (Plus a hint…) – 5 minutes max

(n)

(o)

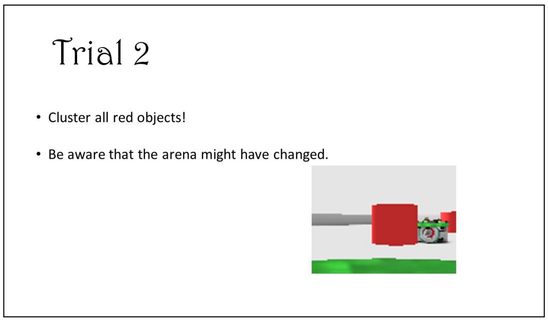


(p)



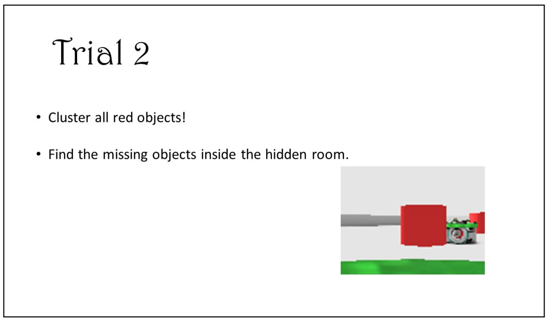
(q)

Group A with 17 participants .



(r)

Group B with 16 participants.



(s)

Group C with 17 participants.

Appendix D

Experiments Questionnaire

University of Sheffield

Title of Research Project: Human - Robot Swarm Awareness Study
Name of Researcher: Gabriel Kapellmann

Participant Identification Number for this project: ____________________

Instructions: Circle the correct answer.

1. Indicate your age range:

   18 – 29   30-39   40-49   50+   Prefer not to say

2. Which is your highest achieved level of education:

   A-Levels / High School   Undergraduate   Master   PhD   Prefer not to say

3. Do you consider yourself a curious person?

   Not at all   1   2   3   4   5   6   7   8   9   10   Very Curious

4. In a scale from 1 to 10, how hard was for you to control the robot?

   Very Easy   1   2   3   4   5   6   7   8   9   10   Very Hard

5. In a scale from 1 to 10, how hard was for you to understand the robots environment?

   Very Easy   1   2   3   4   5   6   7   8   9   10   Very Hard

6. Which of the two trials was the easiest for you?

   Trial 1   Trial 2

7. Is there any particular reason for your answer of question 6?

   ____
   ____
   ____

8. Did you like the activities?

   Yes   No

If you have any additional comments, please write them here: ____________________
____
____
____
____
____

Fig. D.1 Questionnaire used in the experiments from chapters 3 and 4.