

Article

Not peer-reviewed version

Implementation of an Artificially Empathetic Robot Swarm

Joanna Siwek , Patryk Żywica , Przemysław Siwek , Adrian Wójcik , Witold Woch , Konrad Pierzyński ,
[Krzysztof Dyczkowski](#) *

Posted Date: 18 December 2023

doi: 10.20944/preprints202312.1190.v1

Keywords: artificial empathy; swarm; fuzzy sets; similarity measure; open simulation; physical experimentation



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Implementation of an Artificially Empathetic Robot Swarm

Joanna Siwek ^{1,†} , Patryk Żywica ^{1,†} , Przemysław Siwek ² , Adrian Wójcik ² ,
Witold Woch ¹, Konrad Pierzyński ¹ and Krzysztof Dyczkowski ^{1,*} 

¹ Department of Artificial Intelligence, Faculty of Mathematics and Computer Science, Adam Mickiewicz University, Uniwersytetu Poznańskiego 4, Poznań, Poland

² Institute of Robotics and Machine Intelligence, Faculty of Automatic Control, Robotics and Electrical Engineering, Poznan University of Technology, Piotrowo 3A, Poznań, Poland

* Correspondence: krzysztof.dyczkowski@amu.edu.pl

† These authors contributed equally to this work.

Abstract: This paper introduces an innovative framework for the integration of artificial empathy into robot swarms to enhance communication and cooperation. The proposed model relies on fuzzy state vectors to represent the knowledge and environment of individual agents, accommodating uncertainties in the real world. Utilizing similarity measures, the model compares states, enabling empathetic reasoning for synchronized swarm behavior. A practical application example demonstrates the model's efficacy in a robot swarm striving for a common goal. The evaluation methodology involves the Open Source Physical-based Experimentation Platform (OPEP), emphasizing empirical validation in real-world scenarios. The paper proposes a transitional environment that allows automated and repeatable execution of experiments on a swarm of robots using physical devices.

Keywords: artificial empathy; swarm; fuzzy sets; similarity measure; open simulation; physical experimentation

1. Introduction

The research described in the paper was motivated by the work on the application of artificial empathy algorithms in a swarm of mobile robots. These include an attempt to transfer the biological mechanisms of the human brain, such as empathy, to computer systems. They make it possible to improve the quality of those, extending them with a cognitive aspect, in the form of learning and adaptation mechanisms.

We present preliminary results of computer simulations which allowed us to identify problems that caused divergence between simulations and reality, e.g. significant resource consumption, limiting the number and range of parameters. That prevented the effective evaluation of the algorithms. Hence, we propose the solution that aims to solve the problem of discrepancies between the simulation environment and the real experiment, allowing for the verification of complex algorithms and offering universality, accessibility, standardization and repeatability of results.

The construction of platform that achieve those objectives required intensive research. The paper will present the assumptions made and solutions used for the construction of two versions of mobile robot prototypes and an experimentation arena. Particular attention will be paid to the possibility of modeling empathetic behavior in the swarm, which had a significant impact on many technical requirements for the devices built.

1.1. Motivation

In the field of robotics, one of the primary challenges is the need for standardization and the reproducibility of research results, especially in the field of AI. Often, research outcomes depend on various factors such as hardware configuration, software version, or environmental factors.

Simulators, such as CoppeliaSim [1], available in the field, are imperfect and may not accurately reflect reality. Another issue is the significant centralization of the research and development process in AI. Major academic institutions possess substantial resources, giving smaller universities and individual researchers significantly fewer opportunities to engage in state of the art research.

Another challenge is the high entry barrier to AI and swarm robotics research. Specialized tools and infrastructure are required for research, which can be costly and challenging for smaller research units to acquire. A proposed solution to these issues is the provision of an affordable and standardized experimental environment for swarm robotics. Remote access to such an environment allows for conducting experiments without physical access to the test platform, thereby enhancing the potential for international and interdisciplinary collaboration. Cloud access in a pay-as-you-go model could reduce initial expenses on research infrastructure [2].

The transition from the research stage to industrial implementation points to the problem of inability to manufacture and test swarms in operational conditions. Additionally, there is a lack of variable and modifiable test environments for swarms, especially those with remote access capabilities. The limited applicability of existing cloud-based testing platforms is also a concern, as seen in the case of DeepRacer [2], which is solely dedicated to robot racing. Furthermore, imperfections in existing robotics simulators, such as CoppeliaSim, may not provide accurate representations of reality. The ability to modify the real test environment, enabling the testing of swarm solutions at an early stage of development, would greatly facilitate the development of effective products. For instance, the DeepRacer platform has an unmodifiable race track, limiting the number of testing scenarios. Remote access to an experimental platform with mobile robots would allow for testing swarm algorithms without the need for physical access to the equipment. This approach could reduce capital costs for companies utilizing AI in swarm robotics solutions and redirect capital towards operational activities.

Addressing this problem involves testing algorithms in a standardized and predictable environment, providing greater versatility and computational power than other commonly used tools, such as Kilobots [3]. Introducing a modifiable experimental platform would allow the exploration of various swarm robot operation scenarios in different environments, ultimately leading to improved efficiency. The development of a highly automated experimental environment based on physical devices could eliminate issues inherent to simulation software, opening new research perspectives for scientists dealing with robotics and artificial intelligence challenges.

In summary, the utilization of advanced technologies and sensors in artificial intelligence research is imperative. These innovations not only simplify and enhance the research process but also broaden the accessibility of AI technologies. The establishment of standardized experimental environments, particularly in the realm of swarm robot experiments, addresses issues related to result reproducibility, fostering international and interdisciplinary collaboration in AI.

The experimental platform's flexibility and universality play a crucial role in algorithm testing across diverse environments, leading to the identification of more efficient solutions. The implementation of a highly automated experimental environment, grounded in physical devices, mitigates challenges associated with simulator imperfections and simulation software limitations.

Moreover, remote access to the experimental platform proves to be a game-changer, allowing researchers to conduct experiments without the need for physical presence. This not only streamlines research efforts but also facilitates educational activities in schools, overcoming barriers such as high costs and challenges associated with engaging in AI research. In essence, the integration of advanced technologies and the establishment of robust experimental frameworks are indispensable for advancing AI research and promoting its broader application.

This article presents the process of developing the concept of such an environment, introduces the proposed model of artificial empathy, and then discusses the methodology and results obtained in simulation-based experiments and work on building two prototype versions of the solution.

1.2. Empathy modeling

The human decision process relies highly on a person's knowledge, intelligence and experience. Emotional intelligence that constitutes a large part of general intelligence [4] plays a significant role in action-taking, especially involving cooperation [5]. In such a case communication is crucial, and a significant part of every message consists of signalling emotions and inner state of the sender [5]. Empathy is the ability to put oneself in the "mental shoes" of another person, to understand their emotions and feelings [6]. It allows one to predict what kind of behaviour can be expected from the target of empathy and to plan own actions accordingly. Since human reasoning is highly imprecise, one does not have complete information about the inner state of the empathy target and can only reason based on imperfect, highly subjective knowledge. When creating artificial intelligence systems, especially cooperative ones like swarms, omitting empathy seems to be wasteful. The attempt of transferring the concept of human empathy to artificial systems is called artificial empathy [7,8].

Human reasoning, especially the part connected with emotions, is highly imprecise. Therefore, recreating decision mechanism that include empathy requires tools that allow the model to grasp the imprecision. Empathy requires comparing emotions and states of others to own knowledge - thus the states need to have an imprecise representation. Since we do not have a complete information about the inner state of the empathy target, we can only deduce based on a not ideal, subjectively produced and subjectively understood signal. Fuzzy sets and linguistic variables provide useful solutions.

In the paper, we will deal with fuzzy agents, i.e. agents that decide with fuzzy knowledge [9]. The decision is whether an egoistic or empathetic action should be taken to achieve a shared goal. Egoistic action is focused on a local goal, compatible with the general goal of the swarm. Empathetic action is cooperative. The problem at hand is whether the swarm performance can be further optimized by integrating socio-psychological aspects of empathy in swarm control.

Agent's actions are taken based on generalized knowledge represented as a table of a fixed number of fuzzy sets, modelling actions-consequence pairs. Fuzzy feature representation converts the collected information into membership degrees [10]. Resulting degrees are taken as a description of the agent's state. Before an agent performs an action, its current state is assigned a reward, based on the similarity to known states and their outcomes. After the performed action, the state is assigned a realized reward, and is then remembered as "knowledge". By replacing each value in the input data with their corresponding degree of membership to parameter realization we obtain a fuzzified set of labelled states [11].

Since empathy requires comparing the states of the agents, it is important to introduce similarity measures for fuzzy sets. From [12,13] we can ascertain that fuzzy sets are the proper tool to describe the similarity in such a situation. An agent will compare known, "experienced" states to a new one, for which the reward is not yet known or stored in any knowledge base available for the agent. The empathetic approach has a great potential of use in swarms, since there are many opportunities where one agent can "help" another, to achieve a shared goal.

1.3. Structure of the paper

Simulation of swarm behaviour is an important step before implementation on a target platform. Yet, the simulations are often inaccurate and unrealistic. This is especially true for the empathetic swarm, due to the key role of communication. In software simulations, it is not possible to reliably reflect problems/errors in signal transmission or hardware failures. In this paper, we propose a new approach to simulating empathetic behaviours in swarms, with the use of an Open source Physical-based Experimentation Platform. OPEP is an intermediate solution between software-based simulation and the target environment. This work presents the architecture and current implementation of the platform.

The main purpose of this paper is twofold. The first one is to introduce a novel model of swarm control with the use of artificial empathy. The second is to establish the Open source Physical-based

Experimentation Platform, a cost-effective and reliable way to evaluate various models and scenarios in robotics, focused on modelling emphatic behaviours.

The paper is organized as follows: the second section gives materials and methods that include definition of artificial empathy and its theoretical models. It also briefly reviews the available methods for conducting experiments in a swarm of robots. The third section contains main results and introduces the idea of an empathetic swarm, presents simulation results of the proposed swarm model and proposes the Open source Physical-based Experimentation Platform. The last section concludes the obtained results and gives some ideas for further research.

2. Materials and methods

2.1. Empathy theory

Artificial empathy is the ability of computational systems to understand and respond to the thoughts, feelings and emotions of humans [8]. Most definitions of artificial empathy describe it as the artificial version of human empathy [14]. Human empathy is said to consist of three components: emotional, cognitive and motivational. Emotional and motivational empathy is more biological and allows for "automatic" responses to emotions elicited by internal or external factors. Cognitive empathy is more inductive - it allows an agent to understand the inner state of another agent, based on the signals they broadcast (e.g. expressions), and the situation they are in.

It is easy to notice that in swarms emotional empathy has no significant applications (for now, since robots do not yet commonly present or feel emotions [15]). Yet, there is a vast scope of possible applications of cognitive empathy - since cognitive empathy is connected with learning and deducing about behaviours in a certain environment. Knowledge sharing by cognitive empathy can help a swarm of robots to learn effective behaviours faster.

In the literature we can find three main models of artificial empathy, each of them created for different use cases.

1. Emotional and cognitive empathy model – the model was developed from medical and neuroscientific research of human brain and has its justification in the brain structure. It assumes that empathy can be divided into parts: 1). responsible for recognizing and reacting to emotions; 2). a part responsible for cognitive, more logical, and deductive mechanisms of understanding the inner states of others [16].
2. Russian doll model – the model assumes that empathy is learned during human life - it resembles a Russian doll, with layers of different levels of understanding others. The first, most inner layers are mimicry and automatic emotional reactions, the next layers are understanding others' feelings and the outer layers are taking the perspective of others, sympathizing, and experiencing *schadenfreude* [17].
3. Multi-dimensional model - This model assumes that we have four dimensions of empathy - antecedents, processes, interpersonal outcomes, and intrapersonal outcomes. Antecedents encompass the agent's characteristics: biological capacities, learning history, and situation. Processes produce empathetic behaviours: non-cooperative mechanisms, simple cognitive mechanisms, and advanced cognitive mechanisms. Intrapersonal outcomes are to resonate or not with the empathy target, and interpersonal outcomes are relationship related [18].

The artificial empathy finds its applications in a growing number of fields and practical problems.

- medical, e.g. detection of autism or Parkinson's disease, depression treatment [19–22]
- psychology e.g. ethics, artificial companions [23,24]
- marketing e.g. personalized adverts [25]
- entertainment e.g. games with high immersion or biofeedback [26–28]
- education e.g. empathetic tutors [29,30]

- car industry [31].

Swarm applications also seem to open an opportunity to effectively utilize empathy.

The idea of swarm comes from nature - insects, animals and people work together in cooperative groups to achieve goals that would be impossible to reach for a single agent. Examples are ants, bees, fish, etc.

The main characteristics of a swarm are a large number of agents, lack of external control, and simple behaviours exhibited by a single agent. The more advanced and complicated swarm behaviours are to emerge automatically, based on the environmental conditions and agent capabilities.

One of the most important research areas of swarms is collective decision-making [32]. The problem has also an application in empathetic swarms, where the decision of which agent to help can be crucial in achieving the goal. An example can be found in [33].

Current research of empathetic swarm behaviours emerging automatically from broadcast and received signals include [34,35]. Also, theory of mind is used to model collective behaviours of artificially modelled rescue teams [36,37]. The study of Huang et. al. [38] considers a simulated swarm of a few caribou agents to escape from a superior wolf agent. The introduction of empathy in the form of an additional parameter (the distance from the chased caribou to the wolf), allowed for a significant increase in the number of learned successful escape strategies. This kind of approach is very limited – the decision process is automatic and based only on one parameter. It resembles more of an emotional contagion (one of the primitive levels of empathy), rather than a cognitive empathetic process. In our model, we would like to emphasize the role of learning, deduction and experience in empathizing with other agents.

2.2. Available experimental environments

An analysis of various approaches and tools for conducting robot swarm experiments was conducted. The 3 main categories of available experimental environments are described below.

1. Stand-alone robots, allowing the construction and modeling of swarm behavior.

- Kilobot [3]: is a swarm-adapted robot with a diameter of 3.3 cm, developed in 2010 at Harvard University. It operates in a swarm of up to a thousand copies, carrying out user-programmed commands. The total cost of Kilobot parts was less than \$15. Kilobots move in a vibration-based manner. In addition, they are capable of recognizing light intensity, communicating and measuring distance to nearby units. Currently, the project is not being actively developed, but it is still popular among researchers.
- e-puck2 [39]: is a 7 cm diameter mini mobile robot developed in 2018 at the Swiss Federal Institute of Technology in Lausanne. It supports Wi-Fi and USB connectivity. It has numerous sensors including IR proximity, sound, IMU, distance sensor, camera. The project is being developed in open source and open hardware.
- MONA [40]: is an open-hardware/open source swarm research robotic platform developed in 2017 at the University of Manchester. MONA is a small, round robot with a diameter of 8 cm, equipped with 5 IR transmitters, based on Arduino architecture.
- Colias [41]: is an inexpensive 4 cm diameter micro-robot for swarm simulation, developed in 2012 at the University of Lincoln. Long-range infrared modules with adjustable output power allow the robot to communicate with its immediate neighbors at a range of 0.5 cm to 2 m. The robot has two boards - an upper board responsible for high-level functions (such as communication), and a lower board for low-level functions such as power management and motion control.
- SwarmUS [42]: is a project that helps create swarms of mobile robots using existing devices. It is a generic software platform that allows researchers and robotics enthusiasts to easily deploy code in their robots. SwarmUS provides the basic infrastructure needed for robots to form a swarm: a decentralized communication stack and a localization module that helps

robots locate each other without the need for a common reference. The project is not in development as of 2021.

2. Robot simulation software

- AWS Robomaker: is a cloud-based simulation service released in 2018 by Amazon, with which robotics developers can run, scale and automate simulations without managing any infrastructure, and create user-defined, random 3D environments. Using the simulation service, you can speed up application testing and create hundreds of new worlds based on templates you define.
- CoppeliaSim [1]: is a robotics simulator with an integrated development environment, it is based on the concept of distributed control: each object/model can be individually controlled using a built-in script, plug-in, ROS node, remote API client or other custom solution. This makes it versatile and ideal for multi-robot modeling applications. It is used for rapid algorithm development, simulation automation of complex processes, rapid prototyping and verification, and robotics-related education.
- EyeSim [43]: is a virtual reality mobile robot simulator based on the Unity engine, which is able to simulate all the main functions of RoBIOS-7. Users can build custom 3D simulation environments, place any number of robots, and add custom objects to the simulation. Thanks to Unity's physics engine, robot motion simulations are highly realistic. Users can also add bugs to the simulation, using built-in simulated bug functions.

3. Comprehensive services including simulator and hardware platform.

- AWS DeepRacer: is a 1/18 scale fully autonomous racing car designed in 2017 by Amazon and controlled by Reinforcement Learning algorithms. It offers a graphical user interface that can be used to train the model and to evaluate its performance in a simulator. AWS DeepRacer, on the other hand, is a Wi-Fi enabled physical vehicle that can drive autonomously on a physical track using a model created in simulations.
- Kilogrid [44]: is an open-source Kilobot robot virtualization and tracking environment. It was designed in 2016 at the Free University of Brussels to extend Kilobot's sensorimotor capabilities, simplify the task of collecting data during experiments, and provide researchers with a tool to precisely control the experiment's configuration and parameters. Kilogrid leverages the robot's infrared communication capabilities to provide a reconfigurable environment. In addition, Kilogrid enables researchers to automatically collect data during an experiment, simplifying the design of collective behavior and its analysis.

3. Results

3.1. Artificial empathy of a swarm

The general idea of artificial empathy in swarm applications comes from observing cooperative behaviour in group of agents (human, animals) and realising what kind of knowledge and experience is needed, to create successful behaviours and strategies.

In human cooperation, it is easy to see that taking perspective of another person and trying to understand their point of view highly improves the cooperation and leads to better results. The two mentioned actions are actually the definition of cognitive empathy, i.e. driving own conclusions from the state broadcasted by another agent and the environment, therefore anticipating target's behaviours based on inferred knowledge. The proposed model of an artificially empathetic swarm is based on how humans cooperate. People collect experience and knowledge and decide what action to take based on it. While cooperating, they take into account the experience and knowledge of others. Yet, it is impossible to access other peoples' minds directly - one has to interpret signals sent from cooperating partners. These signals constitute largely of emotions [5]. After receiving a signal, the empathizer imagines what knowledge the signal represents, and what could be the possible consequences [6].

Based on the person's own knowledge, one envisions the state of the sender. Finally, an action choice is made - whether to use own knowledge and capabilities or to combine efforts with others in order to improve performance.

The proposed model consists of six parts: module for evaluating egoistic behaviours, module for evaluating artificially empathetic behaviours, memory module for storing knowledge, decision module for choosing the action type, and egoistic and artificially empathetic behaviour controllers for choosing and executing particular action. Modules are described in the following subsections and interactions between them are depicted in Figure 1.

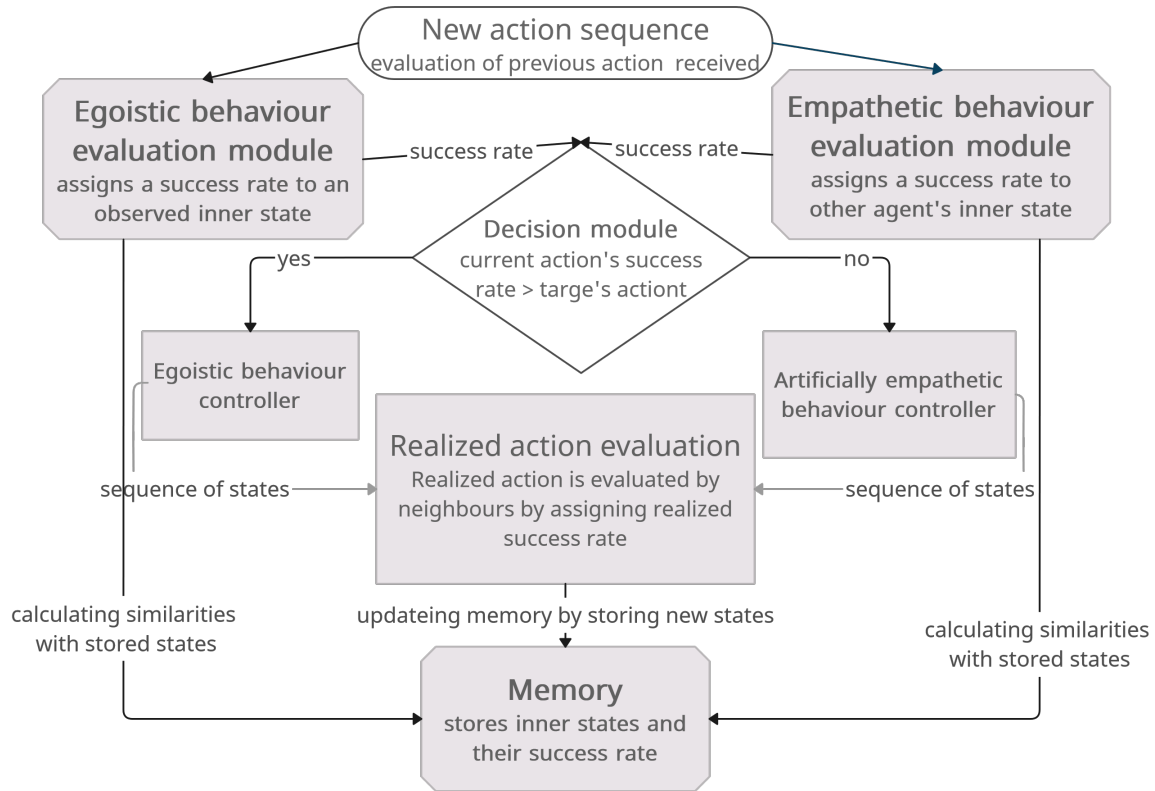


Figure 1. Swarm behaviour control system with artificial empathy. Schema presents component modules: for egoistic and empathetic control and behaviour evaluation, decision and memory.

3.1.1. Egoistic behaviour evaluation module

The egoistic behaviour evaluation module assigns a reward to an egoistic action, based on the state vector representing the state of the agent and perceived environment, and the knowledge of the agent. This state is defined as a fuzzy set:

$$\mathbf{x} = (x_1, x_2, \dots, x_n), \quad (1)$$

where $x_i \in [0, 1]$, $i = 1, \dots, n$ are membership values, representing the satisfaction level of a state X_i . All X_i should not be correlated. The state should be updated in time.

Each of the states is evaluated to decide whether it can bring the agent closer to the goal. The state $\mathbf{x}^i$ is assigned a reward $r(\mathbf{x}^i)$ based on the similarity to the known states and their outcomes, stored in memory and representing the agent's knowledge. The agent has an initial set of states and their rewards, to be able to generalize in an unknown environment. Those first states can be understood as instincts or basic knowledge. The reward from the evaluated state $\mathbf{x}^i$ is then calculated as

$$r(\mathbf{x}^i) = \frac{\sum_{j=1}^m s(\mathbf{x}^j, \mathbf{x}^i) \cdot r(\mathbf{x}^j)}{m}, \quad (2)$$

where m is the number of states and their rewards $(\mathbf{x}^j, r_j)$ stored in memory and s is a fuzzy similarity measure.

3.1.2. Artificially empathetic behaviour evaluation module

The target of empathy broadcasts its state, which the empathizing agent interprets based on their own knowledge. In our model, the empathizing agent (A) evaluates the target's (B) state by comparing its broadcast state to the agent's own knowledge. Reward $r(\mathbf{x}^B)$ is calculated using the same knowledge as the agent's rewards according to (2). The rewards calculated for both empathizing agent and the empathy target are used as decision parameters in choosing agent's next action.

3.1.3. Memory module

The memory module is responsible for storing the agent's knowledge, obtained by performing actions. It is composed of state-reward pairs $(\mathbf{x}^i, r_i)$. At the beginning agent only has initial states. With time and performed actions, new knowledge is added to this base. Namely, each action taken by the agent is evaluated and assigned a realized reward. The state representing the action and corresponding reward are then added to the base and can be compared with other, new states to calculate the similarity and assign rewards. Since the platforms that are used to implement the model have finite resources, we assume that the memory module stores only a finite number of states. In order to reduce redundancy, only clusters of state-reward pairs are stored. Any clustering strategy may be used, including k-means [45]. Since the decision process is based on similarity, the method does not lose its generality and allows to omit the problem of "perfect knowledge", i.e. remembering every detail of an action (since instead of all, only action representatives are being stored).

This method of storing information also allows for updating the state/reward pairs. If a state was remembered and it appears again with different reward, the clustering algorithm can recalculate the representative of the action, with different reward.

3.1.4. Decision making

The decision module is responsible for choosing between the controllers that will provide next action: the egoistic or empathetic behaviour controller. First provides egoistic actions that bring the agent closer to the local goal. Second provides cooperative actions with the same goal but possibly synergistic results. The choice is made based on the reward assigned to the current state of the agent in the egoistic behaviour evaluation and, if received a signal, the reward of the empathy target B, calculated by the artificially empathetic behaviour evaluation module. The module performs the following comparison:

$$r(\mathbf{x}^A) \geq r(\mathbf{x}^B) \quad (3)$$

where $\mathbf{x}^A$ is the agent's current state and $\mathbf{x}^B$ is the the empathy target's broadcast state. If the $r(\mathbf{x}^A)$ value is higher than $r(\mathbf{x}^B)$, the egoistic behaviour of the agent has a greater chance of success in achieving the goal than stopping the current action and helping the neighbouring agent. If the contrary is true, acting with empathy may result in a greater chance of success, so the current course of action should be dropped. In case of multiple incoming signals, only the first one is considered.

3.1.5. Learning

After a full action sequence is performed (that is a set of actions that result in a particular change in environment), the realized action is evaluated. Each agent's actions are evaluated by signals of its neighbours. The sequence is defined as a series of atomic actions that begin with a starting action (the first one without a realized reward) and ends with a last action before an evaluation signal is received:

$$seq_i = (\mathbf{x}_1^i, \mathbf{x}_2^i, \dots, \mathbf{x}_k^i) \quad (4)$$

where $\mathbf{x}_j^i, j = 1, \dots, k$ is the state vector and k is the number of actions performed before the evaluation signal was received. The realized reward $\hat{r}_i$ is the first evaluation signal received by the agent, weighted accordingly to the set goal:

$$\hat{r}_i = \hat{\mathbf{x}} \cdot \mathbf{W} \quad (5)$$

and where $\hat{\mathbf{x}}$ is the received evaluation signal and the weights $\mathbf{W}$ are assigned accordingly to the defined agent's goal. The reward is then assigned to the action process vector, forming new knowledge:

$$\mathbf{ev}_{\text{seq}_i} = (\mathbf{x}_1^i, \mathbf{x}_2^i, \dots, \mathbf{x}_k^i, \hat{r}_i) \quad (6)$$

The aggregated action and the realized reward are then stored in the memory module as a newly learned behaviour.

3.2. Simulation results

3.2.1. Problem description

Let's consider the warehouse that stores grain. The stacks of grain often change place due to normal warehouse operation. The warehouse is infested with rats. We introduce a swarm of empathetic guarding robots that collaborate in order to detect rats and find their nests. The robots operate without a central control system but can communicate between themselves. They patrol the dynamic environment. When a rat is spotted, the robot broadcasts a signal about a spotted target, its own chance of success in chasing it and other information, like battery level. Robots that received that signal calculate whether it is better to continue the current action (e.g. patrolling another part of the warehouse, going to the charging station) or to approach the broadcasting robot and help it in chasing the rat. The described environment is an enclosed space with a static obstacle in the form of grain, defined walls, and mobile hostile objects (rats). The robot's task is to detect and follow the rat, engaging in group encirclement. The rat's goal is to reach the grain. Different views on virtual experimentation environment are given in Figure 2.

The analysis of robot behavior regarding the influence of artificial empathy was conducted only on chasing robots, as they had the most possible actions to perform and could process the most information among all the robots. In physical experiments, rats were also represented by robots, but they were not equipped with empathetic modules.

Patrolling robots could perform their task individually or through communication with other robots. Each robot could signal its own state through an LED strip, displaying information such as the robot class or the currently performed action. This included:

- Call for help
- Encircling the rat
- Helping
- Another robot nearby
- Rat nearby

In contrast to the control group, where robots were not equipped with empathetic modules, experiments on empathetic robots indicate that robots have much more information to process before taking specific actions. Similar to the control group, the rat is searched for in the camera image. The empathetic model difference lies in the fact that each patrolling robot additionally signals information about its state and surroundings on the LED strip. It also has the ability to analyze this information from other robots.

This enables robots to make decisions based not only on their own observations but also on those collected from the surrounding environment. Before taking any action, the robot calculates the reward for performing a specific action, i.e., how much it contributes to achieving the global goal. Rewards are

calculated for both the currently performed action and the planned action. If the reward for the new action is greater than the currently performed action, the robot interrupts it and starts a new one. Using the artificial empathy module, robots could make decisions that were optimal for the entire group.

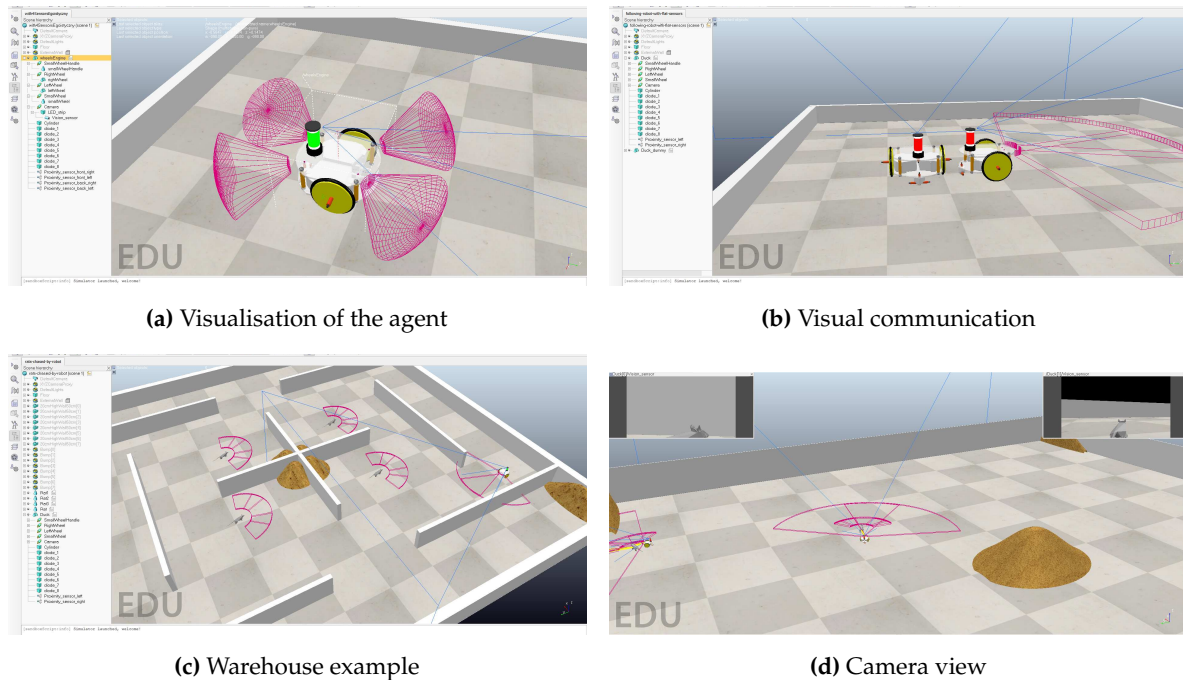


Figure 2. Empathetic swarm simulation in CoppeliaSim.

Performed experiments considered the following list of scenarios:

- Detection of a rat in the warehouse – solitary pursuit
 - Robot 1 patrols the warehouse
 - Robot 1 notices a rat
 - Robot 1 starts chasing the rat
 - Robot 1 catches the rat, meaning it approaches the rat to a certain distance
- Detection of a rat in the warehouse – pursuit handover
 - Robot 1 patrols the warehouse
 - Robot 1 notices a rat in the adjacent area
 - Robot 1 lights up the appropriate color on the LED tower to inform Robot 2 that there is a rat in Robot 2's area
 - Robot 2, noticing the appropriate LED color, starts chasing the rat
 - Robot 2 catches the rat, meaning it approaches the rat to a certain distance
- Detection of a rat in the warehouse – collaboration
 - Robot 1 patrols the warehouse
 - Robot 1 notices a rat
 - Robot 1 starts chasing the rat
 - The rat goes beyond Robot 1's patrol area
 - Robot 1 lights up the appropriate color on the LED tower to inform Robot 2 that the rat entered its area
 - Robot 2, noticing the appropriate LED color, continues chasing the rat
 - Robot 2 catches the rat, meaning it approaches the rat to a certain distance
- Change of grain color
 - Robot 1 patrols the warehouse
 - Robot 1 notices that the grain color is different than it should be
 - Robot 1 records the event in a report

- Robot 1 continues patrolling
- Change of grain color - uncertainty
 - Robot 1 patrols the warehouse
 - Robot 1 notices that the grain color is possibly different than it should be – uncertain information
 - Robot 1 lights up the appropriate color on the LED tower
 - Robot 2, noticing the appropriate LED color, expresses a willingness to help and approaches Robot 1
 - Robot 2 from the adjacent area checks the grain color and confirms or denies Robot 1's decision
 - Robot 1 records the event in a report if confirmed by Robot 2
 - Robot 2 from the adjacent area returns and continues patrolling
 - Robot 1 also continues patrolling
- Weak battery
 - Robot 1 has a weak battery
 - Robot 1 lights up the appropriate color on the LED tower, expressing a desire to recharge its battery
 - Robot 2, noticing the appropriate LED color, agrees to let Robot 1 recharge the battery
 - Robot 1 goes to recharge
 - Robot 2 additionally takes over Robot 1's area for patrolling
- Exchange of patrol zones
 - Robot 1 has passed through its patrol area several times without any events.
 - Robot 1 lights up the appropriate color on the LED tower, expressing a desire to exchange the patrol area
 - Robot 2, noticing the appropriate LED color, expresses a desire to exchange the patrol area
 - Robot 1 and Robot 2 exchange patrol areas

3.2.2. Implementation

The simulations were performed in CoppeliaSim (V4.4.0). Up to ten robots, equipped with virtual cameras, LED communication and touch sensors were to detect and chase four rats in a synthetic warehouse environment. Robots broadcast state signals to other agents, which receive them via camera and decide whether to take egoistic or empathetic action.

The YOLOv2 real-time object detection system [46] was used to detect objects in the camera images, and the VGG16 [47] convolutional neural network to determine the status of other robots (sent via LED strip). All simulation scripts were implemented in Lua (internal CoppeliaSim scripting) and Python (external backend service).

All the fuzzy descriptions like "far" or "long" are modelled with linguistic variables and terms - the value of a parameter is actually the value of a membership function for each of the considered terms.

Vectors of parameters in Tables 1 and 2 are used as initial knowledge. Each new state that arises is compared to those, and the similarity is calculated to decide if the new state has a chance of success, or not. Here we use the similarity measure:

$$s(\mathbf{x}^j, \mathbf{x}^i) = 1 - \frac{\sqrt{\sum_{k=1}^n |x_k^j - x_k^i|^2}}{n}. \quad (7)$$

Table 1. Parameters describing the state of the agent.

Name	Sym	Description of boundary values
others close	a	1 many other agents in the vicinity, 0 for none
in touch	n	1 for long contact time, 0 for none
long search	t	1 for long duration of current search, 0 for not searching
calling for help	c	1 for calling for a long time, 0 for not calling
neutralized	e	1 if "I am inactive" signal was received from newly inactive agent; 0 if not
close to neighbour	d	1 if the distance to neighbour is 0; 0 if distance to neighbour is far
target at right	p	1 for agent at the immediate right, 0 for agent not in sight
target at left	l	1 for agent at the immediate left, 0 for agent not in sight
fully charged	f	1 for robot fully charged, 0 for not charged
helping	h	1 for long duration of helping, 0 for not helping
reward	s_i	describes the chance of success of the current action sequence

Table 2. Agent’s initial knowledge - states and rewards.

parameter	x_1	x_2	x_3	x_4	x_5	x_6
a	0.5	1.0	0.5	0.5	0.0	0.1
n	0.5	0.5	0.5	0.0	1.0	1.0
t	0.5	0.5	0.5	0.5	0.5	0.5
c	1.0	1.0	1.0	0.5	1.0	0.5
d	0.5	1.0	0.5	0.5	0.0	0.1
p	0.5	0.5	0.5	0.5	0.5	0.5
l	0.5	0.5	0.5	0.5	0.5	0.5
f	0.5	0.5	0.5	0.5	0.5	0.5
h	0.5	0.5	0.5	1.0	0.5	1.0
r_i	1.0	1.0	1.0	1.0	0.0	0.0

3.2.3. Results

Robots cooperate in order to detect and chase the rats, and empathetic behaviours are visible. Due to the limitations of CoppeliaSim, mainly lack of repeatability and poor performance when using virtual cameras, simulations did not allow for a reliable comparison between egoistic and empathetic behaviour. These problems lead directly to the OPEP project. OPEP allows for the inclusion of such factors as acceleration, friction, light intensity and reflections, while maintaining high control over the environment and experiment course.

During the simulation, the time in which the robots achieve the global goal, i.e., detecting and catching all rats, was measured. For each case, empathic and non-empathic models, 10 experiments were conducted, measuring the time to achieve the global goal. An important aspect was that objects in the arena were randomly distributed each time to ensure diversity in the observed behaviors.

After conducting experiments on a swarm of 5 robots, it was decided to double the number of objects in the scene. This change introduced more opportunities for interactions between individual units, and the simulation could proceed differently. Additionally, the larger the group of patrolling robots, the more the positive impact of empathic behaviors could be observed, allowing for a focus on the analysis of behaviors between neighboring objects.

As in previous experiments, objects before each simulation were randomly distributed, and the simulation ended when the global goal was achieved, i.e., when all rats were detected and surrounded.

In this way, a total of 40 experiments were conducted in 4 variants. This material was further analyzed, with a primary focus on the analysis of models’ behaviors using artificial empathy and those without it. In many cases, the empathic model recorded lower times to achieve the global goal. However, the differences are small, with the effectiveness of empathy being most visible in larger groups. In such situations, the true power of unit cooperation, forming the entire swarm, can be observed.

An interesting phenomenon was the significant differences in times between individual simulations. The shortest simulation time for 5 patrolling robots was only 58 seconds, while the longest was as much as 116 seconds, nearly a twofold difference. The average simulation time in the egocentric model was 84.1 seconds, and in the empathy-utilizing model, it was 81.6 seconds. Summing up all experiments in this section, the empathic swarm of robots performed its tasks, on average, 2.5 seconds faster. For 10 patrolling robots, the fastest achievement of the goal occurred after 88 seconds, while the longest took 205 seconds. In this case, there were many more possible interaction scenarios for 10 robots, influencing the disparities in simulation times. The average neutralization time for all viruses in the egocentric model was 148.5 seconds, while for the empathic model, it was 137.3 seconds. The significant positive impact of using the artificial empathy module is evident, with a difference of 11.2 seconds, confirming the effectiveness of the empathic model.

We prepared few visualisations of proposed empathetic model, along with comparison with egoistic one.¹

1. Egoistic, two rats. Shortly after starting a patrol, both robots spot the same rat and start chasing it. Meanwhile, the second rat destroys the grain located in the middle of the arena. After neutralizing the first rat, one of the robots begins chasing the second pest.
2. Empathetic, two rats. The robot on the right spots a rat and signals it with an LED strip. The second robot, noticing this, continues to patrol the surroundings in search of other pests. After a while, it detects the second rat and starts following it. As a result, both rats are neutralized and grain loss is reduced.
3. Egoistic, robots run out of battery. Robots detect the same rat. During the chase, the robots interfere with each other, making it difficult to follow and neutralize the rat. Eventually, the rat is neutralized, but before the robots can spot and begin their pursuit of the other pest, both of them run out of battery and the second rat escapes.
4. Empathetic, low battery help. The robot on the right starts chasing the detected rat. During this action, the agent signals with an LED strip that it needs assistance, due to a low battery level. The other robot notices this and decides to help to catch the weaker rat. After neutralizing it, the second robot starts searching for other pests.

3.3. Open-source Physical-based Experimentation Platform

In the field of robotics, experiments are crucial for the development and verification of new algorithms and technologies. However, conducting experiments in this area is challenging, costly, and comes with a range of problems.

Simulations, one of the ways to conduct experiments, are often simplified and inaccurate due to the multitude of parameters that need to be considered. On the other hand, accurate simulations are very time-consuming. It is difficult to include all parameters in the simulation, as some may be unidentified or challenging to model. CoppeliaSim is one tool used for simulating the motion of robot swarms, but it has significant limitations and does not consider all parameters of the real environment, leading to unrealistic simulation results. Time-consuming robot swarm simulations also pose a problem, making it difficult to test various scenarios and restricting frequent algorithm changes.

Real-world experiments, on the other hand, are costly, requiring the purchase of components and the creation of physical experimental platforms, which is time-consuming. However, experiments using dedicated hardware lack high repeatability and reproducibility of results, making it challenging to compare models experimentally. Additionally, there is a lack of standards and a unified approach to the design and implementation of mobile robots, requiring individualized approaches for each experiment and resulting in significant time delays and increased costs. These issues exacerbate discrepancies between simulations and experiments.

¹ Videos available on <https://github.com/open-pep/coppelia-simulations>

Therefore, in this work we propose the concept of Open-source Physical-based Experimentation Platform (OPEP), which is an intermediate environment between simulators and full experiments on dedicated equipment. In the following part of the work, we will present general assumptions towards the offered functionalities, the architecture, and the implementation of two versions of prototypes of the postulated solution.

3.3.1. Proposed platform features and architecture

The Open-source Physical-based Experimentation Platform (OPEP) consists of several parts - a swarm of autonomous, mobile robots, an arena for controlled experiments, charging stations, an overhead controller (camera and mini PC) and a web application for remote experiment control as depicted on Figure 3. The platform allows for performing experiments on physical robots remotely, in a controlled environment. It stands as a middle step between fallible and imperfect simulations and costly physical implementation.

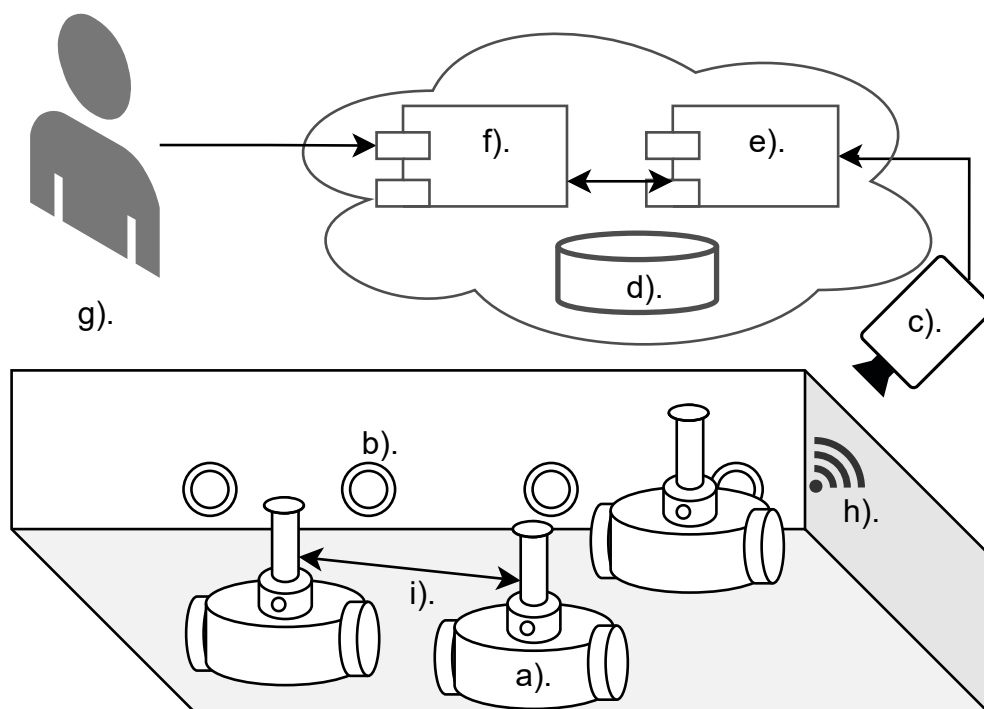


Figure 3. Context architectural diagram for OPEP. a). independent robots that form a swarm; b). automatic charging stations; c). overhead controller (camera + miniPC); d). data store; e). experiment control API module; f). simulations web API; g). researcher interacts with the system via a web browser; h). wireless communication with overhead (maintenance only); i). visual communication between robots.

The proposed experimentation platform contribute to the creation of a completely new product — a universal, remote service for experimenting and testing artificial intelligence algorithms on a hardware swarm of robots, provided in a cloud computing model. Its main features are given in the following paragraphs.

1. Comprehensive Support for Swarm Design Process using Hardware Platform This feature corresponds to the need to verify AI algorithms in a hardware environment, including early-stage development, consideration of environment parameters unavailable in simulations, and the ability to study algorithms considering variable environments and interactions. In this area, there are two alternatives: comprehensive algorithm evaluation (Kilogrid + Kilobots, DeepRacer) and simulation software (CoppeliaSim, DynaVizXMR, EyeSim, Microsoft Robotics). Alternative

solutions only support the design process in simulated environments or require significant financial investment for prototyping, limiting accessibility in early development stages.

2. **Low Cost of Building and Size of Swarm Robots** This feature corresponds to the need for evaluating complex behaviors and the latest AI algorithms in a large swarm of robots, considering the requirement for low cost and easy availability of solutions. Alternatives includes miniature robots like Kilobots and minisumo robots. Those solutions are expensive, with costs often including additional resources and services. Additionally, computational power drastically decreases with the robot's size, limiting capabilities such as running a vision system.
3. **Remote Programming of Robots** This feature addresses the need for sharing research/educational infrastructure without physical access, fostering interdisciplinary and international research collaborations. Alternatives include cloud-based robot simulators like AWS Robomaker and DeepRacer. In competitive solutions, this functionality is only available in simulations or limited environments and specific research areas.
4. **Standardization and Scalability of Experimental Environment** This feature corresponds to the need to adapt and expand the experimental platform to different projects while maintaining standardization for experiment repeatability and reproducibility, facilitating comparison across research centers. Alternatives include open-source software and hardware projects like SwarmUS, Kilobots, and colias.robot, as well as simulation software. They lack the ability to expand robot software in any way using high-level languages. Moreover, existing solutions are not designed for result repeatability (e.g., randomness in Kilobots' movements).
5. **Open Specification and Hardware** This feature corresponds to the need for independently building a complete experimental platform. Current solutions include open-source software and hardware projects such as SwarmUS, Kilobots, and colias.robot. Most competitive solutions are closed, and open solutions often have limited computational resources.

3.3.2. Platform implementation

First prototype

The authors conducted a series of analyses and experiments, creating a swarm of 8 early prototypes of robots equipped with mobility, vision, and communication systems, enabling the realization of simple empathetic behaviors. The prototype robots had a diameter of 14.9cm, a height of 15cm, and were equipped with a 4000mAh Li-Poly battery, allowing about 40 minutes of continuous operation (Figures 4 and 5). Based on team members' experiences during the creation of the prototype robot swarm, several phenomena and issues crucial for proper robot operation in a real environment were observed. These would likely be overlooked in computer simulations, including mirror reflections of robots in the arena walls, slipping robot wheels, uneven traction, imperfections in mechanical components (e.g., motors, gears), and the impact of the environment on the performance of the vision system (too strong or weak room lighting).

The first version of robots were built with the use of Raspberry Pi Zero 2 W microcomputers that take care of robot control, vision and all decision-making. Agents use two motors to power the wheels. The communication is performed via a custom 360° RGB LED strip, placed on a rotating turret, that also holds the OV5647 5MPx camera which gives an effective 240° angle of view. The use of 8 RGB LED lights allows to express about 2M distinguishable inner states. The communication is imperfect since robots have to detect the signal that can be disturbed by light level, reflection, other robots in view, etc. The outer case is 3D-printed and was designed for minimal collision damage.

Each robot in the swarm runs on 64-bit Raspberry Pi OS Lite (5.15 kernel). Vision system is implemented using *picamera2* (0.1.1) and *opencv-python* (4.6.0.66) libraries. Intra-process communication is handled using Redis (7.0.5) pub/sub feature. Web-based user interface is still being actively developed. It uses Python 3 and Go 1.19 for API implementation. The overhead controller uses

both visual monitoring (experiment recording) as well as a wireless network (software upload, swarm maintenance). The ongoing research and development of the project can be tracked on GitHub [48]².

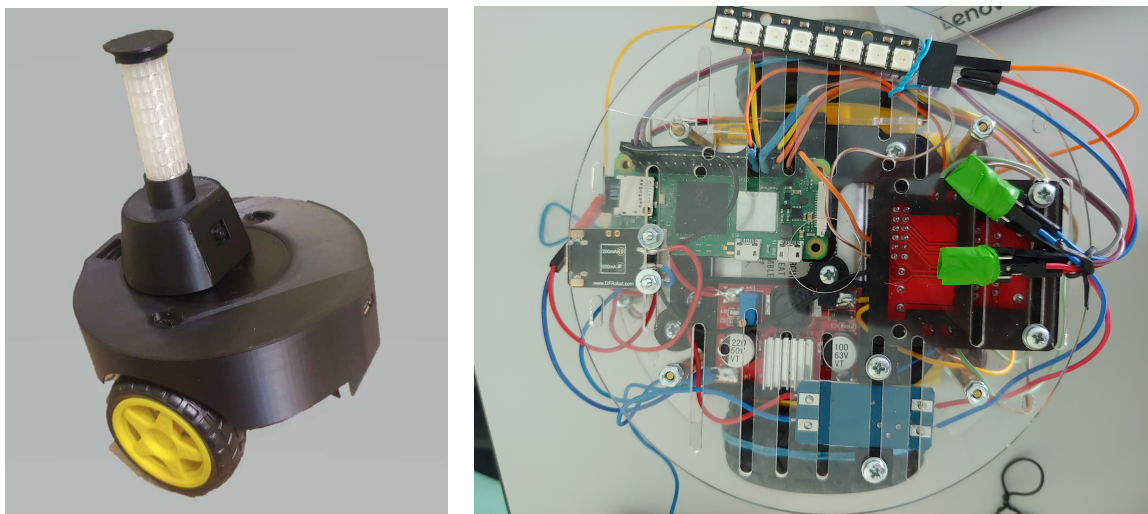


Figure 4. Hardware implementation and view on internal components of first prototype.

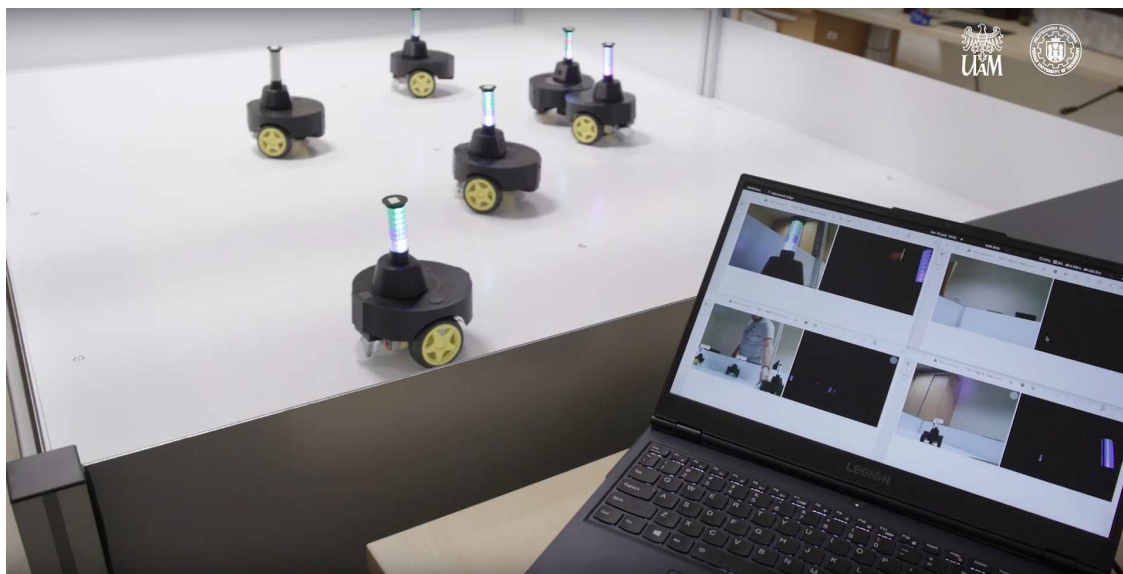


Figure 5. First version prototypes moving on the arena with live monitoring of cameras.

Second prototype

The first prototype has some problems that need to be solved so that more effective research can be conducted:

- The robots need to be stopped and physically plugged in for charging when the batteries run out. This causes delays in conducting experiments.
- Current robots are characterized by large dimensions compared to the work area. This minimizes the simultaneous number of robots that can move around the arena.
- The presence of an experimenter is required to activate the robots. This makes it impossible to conduct remote experiments.

² The project is in the process of migrating from an internal repository to GitHub and not all components are available yet

For the development of the next, more efficient and miniaturized iteration of the platform, it is necessary to solve the above problems by creating a custom, compact version of the PCB. This requires conducting a thorough analysis of available circuit boards and selecting the best solutions to use in designing the electrical as well as mechanical structure of the robots that make up the swarm. This will allow the robots to respond to their environment in the best possible way, and allow researchers to conduct experiments on artificial empathy remotely.

Due to the minimization of the robot's dimensions, its design envisions two circuit boards connected above each other. The lower board will be responsible for interaction with the environment, and the upper board, for information processing and decision-making. The second version of the prototype is less than 8cm in diameter, and has a much smaller height (Figure 6).

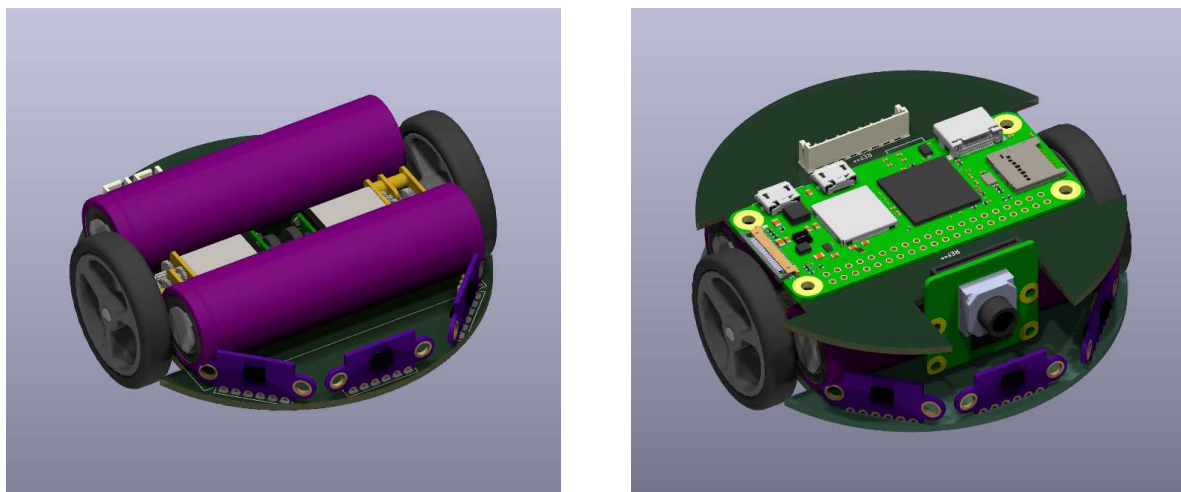


Figure 6. Visualization of second prototype, with (left) and without (right) upper board.

Robot movement will be accomplished using two miniature geared motors manufactured by Pololu (Figure 7). Each of them is equipped with a magnetic encoder based on TLE4946-2K Hall effect sensors, specially designed for this scenario. The rotation of the motors will additionally be monitored by current sensors, one for each motor, to achieve accurate speed control. To eliminate losses on typical shunt resistor current sensors, a circuit that measures current based on the Hall effect can be used. Examples of such sensors are the ACS712 or ACS724. To power the motors, the DRV8833 will be used as an executive circuit. This controller allows the operation of two motors, and is characterized by the fact that it requires only two control lines per motor to regulate speed and direction of rotation, and is easy to operate using hardware Timers in the microcontroller.

To provide the robot with precise orientation in the field, it will be equipped with an integrated BNO055 chip, consisting of a Bosch accelerometer, gyroscope and magnetometer. This chip will support the robot's motion algorithms to achieve maximum precision in its maneuvers. In addition, the VL53L0X sensors will provide information about the distance of obstacles in front of the robot. The use of these sensors introduces increased complexity in both the board design and control algorithms compared to traditional pushbuttons and limiters. Nevertheless, this change contributes to reducing mechanical contact with the environment, which translates into minimizing the risk of damage to the robot and increasing its reliability.

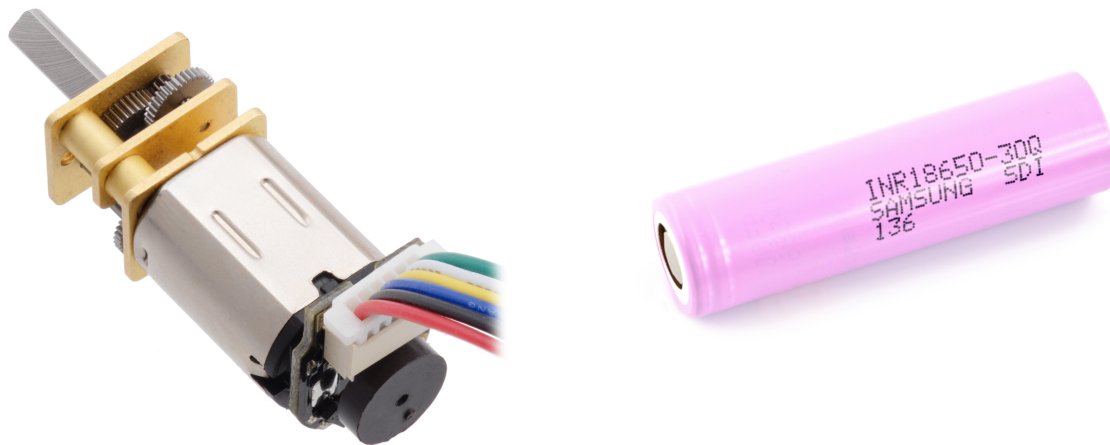


Figure 7. Engine and battery used in second prototype.

Two Samsung INR18650-35E lithium-ion cells with a total capacity of 7Ah will be used to power the robot (Figure 7). Due to the robot's small size, the design will use cylindrical cells instead of flat lithium-polymer batteries, necessitating the addition of protection circuits (Figure 6). In order to ensure safe power management, a DW01 battery protection circuit will be used in conjunction with the executive transistors. This circuit is designed to monitor the battery voltage so that it does not exceed the permissible range, and to disconnect the voltage if a short circuit occurs. The charging process will be supervised by the TP5100 chip, which will not only provide up to two amps of current, but also ensure that the cell charging process will follow the constant current constant voltage (CC CV) charging specification. One of the goals of the project is to be able to charge the robot without human intervention. To achieve this, two contact pads will be placed on the bottom of the lower circuit board. Using them, the robot will be able to charge batteries after hovering over a special charging station. This solution is extremely simple, takes up little space and does not introduce losses in energy transfer.

The robot, despite its small size, will be equipped with systems that require high power consumption. To meet their expectations, the project will use two inverters. One with an output voltage of 5V - U3V70A, which is capable of supplying up to 10A at peak demand even for a few seconds. The second inverter, this time for circuits powered by 3.3V - U7V8F3 will provide current for all sensors and the microcontroller. It is worth noting that the 3.3V inverter will operate in two modes: step-down when the battery is charged and step-up when the battery is closer to being discharged.

The bottom board will also house a 32-bit microcontroller from the STM32 family. Its high computing power, flexibility and popularity will allow the implementation of almost arbitrarily complex algorithms. It will control all the above circuits and communicate with the robot's main processor.

The role of the device's brain will be played by the Raspberry Pi Zero 2W, it is a quad-core, single-board computer on a top circuit board, which will react to its environment and other robots with the help of the Raspberry Pi Cam V3. All artificial intelligence and empathy algorithms will be implemented right on this microcomputer. Signaling of its own internal state will be achieved by individually addressable RGB LEDs - WS2812B, arranged in 8 rows, each with 3 LEDs. This will make it possible to display at least one and a half million different states. A detailed diagram showing the internal layout of the proposed device is shown in Figure 8.

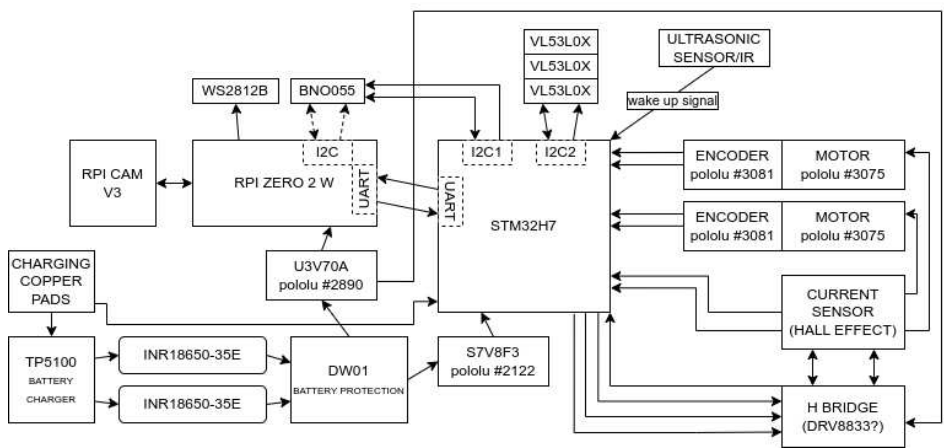


Figure 8. Block diagram showing internal layout of second prototype.

Experimentation arena

One of the challenges was to create a suitable arena that serves as an environment for controlled experiments. Another challenge is the inability to modify and introduce variability to experimental platforms, limiting the number of test scenarios and hindering research in different applications. This arena was designed to simulate various environmental conditions and allow the study of swarm behavior in different situations, added to the arena as modular modifications (e.g., obstacles).

Variability in the experimental environment is a distinctive feature of the constructed experimental platform. Research efforts will be necessary to identify available and feasible strategies to achieve this goal. Further research will consider elements of a variable environment, such as changing parameters (e.g., lighting) and altering the structure of the arena (obstacles, cooperative logical puzzles). A key assumption is the automation of the process of introducing modifications to the arena, crucial for both achieving repeatability of experiment results and enabling remote access to the platform.

This will allow testing whether and how (with what energy expenditure and difficulties) the robot can perform simple tasks in the presence of designed variable elements in the environment. Conducting these experiments will provide insights into the requirements for the experimental arena and robot prototypes, allowing for further considerations in subsequent prototyping iterations.



Figure 9. Visualization and realisation of the controlled environment for the empathetic robot swarm – arena.

4. Conclusions

In this paper, we propose a pioneering model for swarm behavior control through artificial empathy, employing fuzzy set theory and similarity measures. The model emulates human empathetic decision processes to optimize behavior toward achieving a common goal. The main contribution of the authors in that field was to redefine the theoretical cognitive empathy model into a particular swarm control model, the most significant part being the knowledge representation and decision making process.

Unlike existing swarm models of artificial empathy, such as [33,38], which utilize a simple parameter to describe the empathy target's state, our model stands out by incorporating empathetic communication, knowledge representation, and the use of similarity measures to convey empathy mechanisms. This approach, inspired by neural science's emotional/cognitive empathy model, filters agent-broadcasted states based on similarity to known situations.

The presented example showcases only a fraction of the model's applicability, emphasizing its versatility across platforms. The model's independence from a specific application allows for the separate development of empathetic decision-making modules and platform controllers. A notable consideration is the model's potential application in dynamically changing environments, where agents can "borrow" knowledge from others. Incorporating perceived rewards of other agents into the decision-making process could enable neighboring agents to learn from experience, comparing differences in perceived and realized rewards. This paper lays the groundwork for further exploration and development of empathetic swarm control models in diverse environments.

The integrated experimental environment proposed in the article is an important step in the development of research on swarm robot control algorithms. This, in turn, will increase the possibility of implementing the results of scientific research in practice. The proposed experimental platform, its scope and architecture are based on the experience that has been developed while working on two of its earlier prototypes. As a result, it was possible to optimize many technical and operational parameters of the developed solution. It is also not insignificant that the work on the hardware environment is strongly connected with research on empathetic robotic swarm, which imposes real requirements on the direction of the project's development. The most significant example of this, is the use of innovative vision communication using LED towers.

In further research we want to upgrade the model to even further recreate the human empathy - the egoistic behaviour evaluation module to be implemented with a neural network, and the similarity of other agent's state to be decided based on internal knowledge represented by the net. Also, reinforcement learning is to be used to teach the net new data, and adapt to the environment and behaviour of other agents.

References

1. Rohmer, E.; Singh, S.P.N.; Freese, M. CoppeliaSim (formerly V-REP): a Versatile and Scalable Robot Simulation Framework. *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2013.
2. Balaji, B.; Mallya, S.; Genc, S.; Gupta, S.; Dirac, L.; Khare, V.; Roy, G.; Sun, T.; Tao, Y.; Townsend, B.; Calleja, E.; Muralidhara, S.; Karuppasamy, D. DeepRacer: Educational Autonomous Racing Platform for Experimentation with Sim2Real Reinforcement Learning, 2019, [arXiv:cs.LG/1911.01562].
3. Rubenstein, M.; Ahler, C.; Nagpal, R. Kilobot: A low cost scalable robot system for collective behaviors. 2012 IEEE International Conference on Robotics and Automation, 2012, pp. 3293–3298. doi:10.1109/ICRA.2012.6224638.
4. Drigas, A.S.; Papoutsis, C. A new layered model on emotional intelligence. *Behavioral Sciences* **2018**, *8*, 45.
5. Barbey, A.K.; Colom, R.; Grafman, J. Distributed neural system for emotional intelligence revealed by lesion mapping. *Social cognitive and affective neuroscience* **2014**, *9*, 265–272.
6. Decety, J.; Lamm, C. Human empathy through the lens of social neuroscience. *The Scientific World Journal* **2006**, *6*, 1146–1163.

7. Xiao, L.; Kim, H.j.; Ding, M. An introduction to audio and visual research and applications in marketing. *Review of Marketing Research* **2013**.
8. Yalçın, Ö.N.; DiPaola, S. Modeling empathy: building a link between affective and cognitive processes. *Artificial Intelligence Review* **2020**, *53*, 2983–3006.
9. Fougères, A.J. A modelling approach based on fuzzy agents. *arXiv preprint arXiv:1302.6442* **2013**.
10. Yulita, I.N.; Fanany, M.I.; Arymuthy, A.M. Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification. *Procedia computer science* **2017**, *116*, 530–538.
11. Mohmed, G.; Lotfi, A.; Pourabdollah, A. Enhanced fuzzy finite state machine for human activity modelling and recognition. *Journal of Ambient Intelligence and Humanized Computing* **2020**, *11*, 6077–6091.
12. Dubois, D.; Prade, H. The three semantics of fuzzy sets. *Fuzzy sets and systems* **1997**, *90*, 141–150.
13. Żywica, P.; Baczyński, M. An effective similarity measurement under epistemic uncertainty. *Fuzzy sets and systems* **2022**, *431*, 160–177.
14. Asada, M. Development of artificial empathy. *Neuroscience research* **2015**, *90*, 41–50.
15. Suga, Y.; Ikuma, Y.; Nagao, D.; Sugano, S.; Ogata, T. Interactive evolution of human-robot communication in real world. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2005, pp. 1438–1443.
16. Asada, M. Towards artificial empathy. *International Journal of Social Robotics* **2015**, *7*, 19–33.
17. De Waal, F.B. The ‘Russian doll’ model of empathy and imitation. *On being moved: From mirror neurons to empathy* **2007**, pp. 35–48.
18. Yalçın, Ö.N. Empathy framework for embodied conversational agents. *Cognitive Systems Research* **2020**, *59*, 123–132.
19. Morris, R.R.; Kouddous, K.; Kshirsagar, R.; Schueller, S.M. Towards an artificially empathic conversational agent for mental health applications: system design and user perceptions. *Journal of medical Internet research* **2018**, *20*, e10148.
20. Vargas Martin, M.; Pérez Valle, E.; Horsburgh, S. Artificial Empathy for Clinical Companion Robots with Privacy-By-Design. International Conference on Wireless Mobile Communication and Healthcare. Springer, 2020, pp. 351–361.
21. Montemayor, C.; Halpern, J.; Fairweather, A. In principle obstacles for empathic AI: why we can’t replace human empathy in healthcare. *Ai & Society* **2021**, pp. 1–7.
22. Fiske, A.; Henningsen, P.; Buyx, A.; others. Your robot therapist will see you now: ethical implications of embodied artificial intelligence in psychiatry, psychology, and psychotherapy. *Journal of medical Internet research* **2019**, *21*, e13216.
23. Leite, I.; Pereira, A.; Castellano, G.; Mascarenhas, S.; Martinho, C.; Paiva, A. Modelling empathy in social robotic companions. International conference on user modeling, adaptation, and personalization. Springer, 2011, pp. 135–147.
24. Possati, L.M. Psychoanalyzing artificial intelligence: the case of Replika. *AI & SOCIETY* **2022**, pp. 1–14.
25. Affectiva Inc.. Media Analytics. Accessed Feb 2023.
26. Leite, I.; Mascarenhas, S.; others. Why can’t we be friends? An empathic game companion for long-term interaction. International Conference on Intelligent Virtual Agents. Springer, 2010, pp. 315–321.
27. Sierra Rativa, A.; Postma, M.; Van Zaanen, M. The influence of game character appearance on empathy and immersion: Virtual non-robotic versus robotic animals. *Simulation & Gaming* **2020**, *51*, 685–711.
28. Blanchard, L. Creating empathy in video games. *The University of Dublin, Dublin* **2016**.
29. Aylett, R.; Barendregt, W.; Castellano, G.; Kappas, A.; Menezes, N.; Paiva, A. An embodied empathic tutor. 2014 AAAI Fall Symposium Series, 2014.
30. Obaid, M.; Aylett, R.; others. Endowing a robotic tutor with empathic qualities: design and pilot evaluation. *International Journal of Humanoid Robotics* **2018**, *15*, 1850025.
31. Affectiva Inc.. Interior Sensing. Accessed Feb 2023.
32. Ebert, J.T.; Gauci, M.; Nagpal, R. Multi-feature collective decision making in robot swarms. Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, 2018, pp. 1711–1719.
33. Huang, F.W.; Takahara, M.; Tanev, I.; Shimohara, K. Effects of Empathy, Swarming, and the Dilemma between Reactiveness and Proactiveness Incorporated in Caribou Agents on Evolution of their Escaping Behavior in the Wolf-Caribou Problem. *SICE Journal of Control, Measurement, and System Integration* **2018**, *11*, 230–238.

34. Witkowski, O.; Ikegami, T. Swarm Ethics: Evolution of Cooperation in a Multi-Agent Foraging Model. *Proceedings of the First International Symposium on Swarm Behavior and Bio-Inspired Robotics* **2015**.
35. Chen, J.; Zhang, D.; Qu, Z.; Wang, C. Artificial Empathy: A New Perspective for Analyzing and Designing Multi-Agent Systems. *IEEE Access* **2020**, *8*, 183649–183664.
36. Li, H.; Oguntola, I.; Hughes, D.; Lewis, M.; Sycara, K. Theory of Mind Modeling in Search and Rescue Teams. *IEEE International Conference on Robot and Human Interactive Communication*, 2022, pp. 483–489. doi:10.1109/RO-MAN53752.2022.9900572.
37. Li, H.; Zheng, K.; Lewis, M.; Hughes, D.; Sycara, K. Human theory of mind inference in search and rescue tasks. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. SAGE Publications Sage CA: Los Angeles, CA, 2021, Vol. 65, pp. 648–652.
38. Huang, F.; Takahara, M.; Tanev, I.; Shimohara, K. Emergence of collective escaping strategies of various sized teams of empathic caribou agents in the wolf-caribou predator-prey problem. *IEEE Transactions on Electronics, Information and Systems* **2018**, *138*, 619–626.
39. Mondada, F.; Bonani, M.; Raemy, X.; Pugh, J.; Cianci, C.; Klapotcz, A.; Magnenat, S.; Zufferey, J.C.; Floreano, D.; Martinoli, A. The e-puck, a robot designed for education in engineering. *Proceedings of the 9th conference on autonomous robot systems and competitions*. IPCB: Instituto Politécnico de Castelo Branco, 2009, Vol. 1, pp. 59–65.
40. Arvin, F.; Espinosa, J.; Bird, B.; West, A.; Watson, S.; Lennox, B. Mona: an affordable open-source mobile robot for education and research. *Journal of Intelligent & Robotic Systems* **2019**, *94*, 761–775.
41. Arvin, F.; Murray, J.; Zhang, C.; Yue, S. Colias: An autonomous micro robot for swarm robotic applications. *International Journal of Advanced Robotic Systems* **2014**, *11*, 113.
42. Villemure, É.; Arsenault, P.; Lessard, G.; Constantin, T.; Dubé, H.; Gaulin, L.D.; Groleau, X.; Laperrière, S.; Quesnel, C.; Ferland, F. SwarmUS: An open hardware and software on-board platform for swarm robotics development. *arXiv preprint arXiv:2203.02643* **2022**.
43. Bräunl, T. The EyeSim Mobile Robot Simulator. Technical report, CITR, The University of Auckland, New Zealand, 2000.
44. Valentini, G.; Antoun, A.; Trabattoni, M.; Wiandt, B.; Tamura, Y.; Hocquard, E.; Trianni, V.; Dorigo, M. Kilogrid: a novel experimental environment for the Kilobot robot. *Swarm Intelligence* **2018**, *12*, 245–266.
45. MacQueen, J.; others. Some methods for classification and analysis of multivariate observations. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Oakland, CA, USA, 1967, Vol. 1, pp. 281–297.
46. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525. doi:10.1109/CVPR.2017.690.
47. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition, 2014. doi:10.48550/ARXIV.1409.1556.
48. Żywica, P.; Wójcik, A.; Siwek, P. Open-source Physical-based Experimentation Platform source code repository, 2023.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.