INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2025, VOL. 71, NO. 2, PP. 569–582
Manuscript received May 13, 2025; revised May, 2025. doi: 10.24425/ijet.2025.153606

# Swarm Intelligence - conductorless orchestra

Bartłomiej Mastej

***Abstract*—Swarm Intelligence (SI) is a kind of Artificial Intelligence that emerges from local interactions. By principle, systems based on SI are decentralized - they do not have access to the global knowledge. The initial study on SI was on the natural swarms, e.g., ants, flocks of birds, and schools of fish. Swarms have an ability to self-organize, which brings a unique set of features and applications. Despite their origin in biology, SI systems were adopted into technology. At the very beginning, they were used to solve optimization problems, e.g., Ant Colony Optimization and Particle Swarm Optimization; however, they were later adopted in the field of robotics, which is called swarm robotics. Although those are two primary fields of research in SI, other applications, such as in telecommunications, are also presented in this article. Furthermore, the problem of creating SI systems and the current methods used for designing and modeling the swarm systems are presented.**

***Keywords*—Swarm Intelligence, Swarm Robotics, Swarm Design, Swarm Modeling, Swarm Intelligence Applications**

## I. Introduction

SWARM Intelligence (SI) is a special kind of artificial intelligence that utilizes local interactions between independent agents. Based on these interactions, a solution to a problem emerges, without the involvement of global knowledge. The agents are the basic elements of an SI system (a "swarm"). Noticeably, they are relatively simple, hence their unit cost is relatively low. As defines[1] on the one hand, the swarm concept suggests a significant number of agents, randomness or messiness, while the intelligence concept indicates that this way of solving problems is nevertheless successful for some reason.

Swarm Intelligence can be compared to an orchestra that plays a concert without the participation of a conductor—an element of global knowledge that is responsible for the entirety of the piece played, including but not limited to the synchronization of all instrumental sections. Each musician knows the exact course of the entire piece by heart; however, the authority of the conductor in the orchestra remains intact for good reason.

Nonetheless, swarm intelligence engineers are trying to achieve such an effect. The aim is to design a system that can solve a given problem without access to global knowledge. That is due to the unique features of SI systems [2][3]:

- Multitude - as the name suggests, SI consists of numerous agents;
- Self-organization - global behavior emerges directly from local agent interactions;
- Scalability - the more agents in the swarm, the more efficient the SI system;
- Robustness - the system continues to operate even if a certain number of agents are lost;
- Flexibility - the ability of the system to solve tasks unforeseen during design;
- Adaptivity - the system adapts its behavior to changing circumstances;
- Superadditivity - in some cases, the aggregate performance of the system may be greater than the sum of the performance of individual agents [4];

B. Mastej is with Doctoral School, Warsaw University of Technology, Warsaw, Poland (e-mail: bartlomiej.mastej.dokt@pw.edu.pl).

## II. Swarm Intelligence applications

The features of IR systems directly point to their possible applications. By definition, these systems consist of numerous relatively simple agents. Due to an agent's relatively low unit cost, the SI systems are simple to expand. In addition, they can operate independently, thus without access to global knowledge, enabling them to operate in places where access to such knowledge is impossible or uneconomical. For this reason, the primary focus of SI research is robotics systems. However, swarm intelligence research initially evolved from an attempt to describe biological systems. That research led in the late 1990s to the creation of a whole family of optimization algorithms, which also attracted research interest and still have many applications. A certain niche is the use of IR systems in telecommunications, both in routing and other distributed systems.

### A. Swarm Robotics

Swarm intelligence that deals with robotics has become known as swarm robotics. The name is taken from swarms found in nature, a strong source of inspiration for the field's pioneers. Swarm robotics is usually characterized by complete decentralization - agents do not have elements of global knowledge. Unlike optimization algorithms or simulations, which possess the synchronization element, it results from utilizing the hardware. As a result, robotics allows for research into swarm intelligence principles, and it can gain the most from research advances on SI.

*1) Classification of behavior in swarm robotics:* At this point, only certain classes of swarm behaviors are in use. However, in some robotic systems, implementing particular behaviors is enough to increase autonomy or enable possible

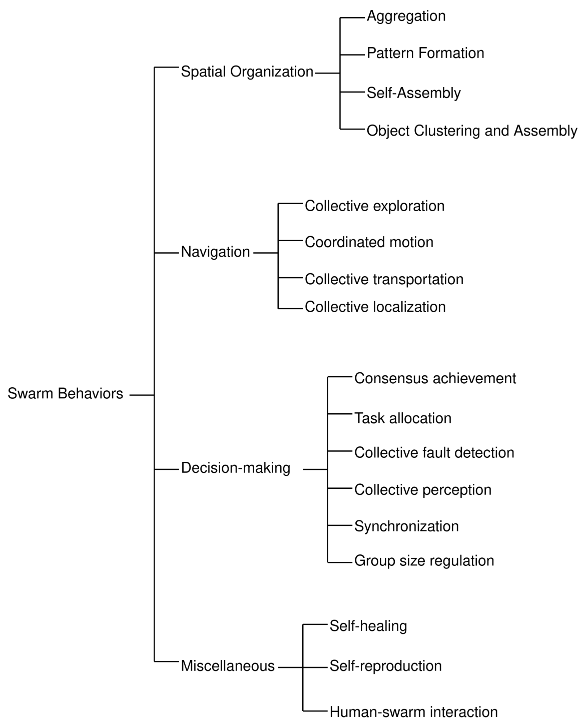



Fig. 1: Classification of swarming behavior proposed by [5].

self-repair. The division into well-studied behavior classes originally proposed by [3] and later extended by [5] can be seen in Fig. 1. The following paragraphs will present the aforementioned classes.

*a) Spatial Organization:* - a class of behaviors responsible for arranging robots and objects in a given environment. Spatial organization includes the following behaviors:

- Aggregation - independent grouping of robots in a given region. Example: aggregation of cockroaches [6].
- Pattern formation - maintaining distances between robots to lay out regular and repeatable patterns. Example: forming a lattice [7].
- Self-assembly - Physical connection of robots with each other. Example: overcoming terrain obstacles [8].
- Object clustering and assembly - manipulation of the objects distributed in the given space. Example: robots assembling an object to match the pattern [9].

*b) Navigation:* - includes all behaviors referring to the position of robots:

- Collective exploration - coverage of as much space as possible, or searching a particular region. Example: searching for disaster victims [10].
- Coordinated motion - collision-free movement of robots in a given space. Example: the use of flying drones in autonomous movement [11].
- Collective transportation - joint transport of cargo, which is most often impossible to transport by a single robot. Example: transportation of cargo by a swarm of molecular machines [12].
- Collective localization - Enabling mutual orientation in the field by swarm members. Example: creating a local coordinate system [13].

*c) Decision-making:* - a class of behaviors relating to broadly understood decision-making within the swarm:

- Consensus achievement - making a joint decision for all members of a swarm or its subgroup. Example: First-Past-the-Post voting (single-winner voting rule) [14], also known from the Ant Colony Optimization (the winner is a the best solution in the certain area).
- Task allocation - independent division of tasks by swarm members. Example: forming letters by simplifying complex shapes into simpler subtasks [15].
- Collective fault detection - omitting defective robots (usually due to hardware failure) from swarm-level decisions. Example: an immune system-inspired behavioral deviation detection algorithm for a swarm of robots [16].
- Collective perception - a view of a task from a swarm's perspective, using data collected by individual robots. Example: in a swarm of flying drones, a common perception of the environment is based on sensors placed only on some of them [17].
- Synchronization - common perception of time by robots, for example, to coordinate the given task. Example: future use of synchronization in harsh environments (e.g., oceans, orbit) has been proposed [18].
- Group size regulation - self-dividing into groups to maintain a certain swarm size. Example: separating an aggregated group into smaller groups of similar size [19]

*d) Miscellaneous:* - behaviors that do not yet have a classified affiliation. Only the most important behaviors are listed below:

- Self-healing - the ability to repair swarm behavior after an error caused by a single agent. In contrast to collective error detection, it is not about the physical error of the device, but, for example, finding a local maximum/minimum and bringing the entire swarm to this solution. Example: stopping a search group by an occurring error on three robots (the robots were standing, thus the target was found), solved by an algorithm inspired by an immune system [20].
- Self-reproduction - the ability to produce new robots in a swarm by the swarm itself, or the ability to copy some of the behaviors of another swarm encountered. That is a step toward evolutionary swarms being able to independently improve the next "generation" from both the hardware and behavioral side. Example: There is currently no physical swarm with such abilities, but the theory of a self-generating automaton was proposed in the 1960s by Von Neumann (the book was published posthumously) [21]. A swarm of material-robots has been proposed, enabling the production of structures by joining together to form larger robots [22].
- Human-swarm interaction - allowing humans to communicate with and influence a collective of robots. Example: changing swarm formation due to a human-made gesture [23].

*2) Swarm robotics applications:* The unique characteristics of swarms, particularly their ability to function autonomously and relatively low cost (the loss of a single, low-cost robot is not particularly significant), make there a demand for swarm robotics wherever a robot is difficult to control. In addition, it is tempting to use them in missions with a high risk of failure or in those requiring a significant number of robots. At the current stage of Swarm Intelligence development, it is inefficient and challenging to implement systems that do not have any global knowledge elements. Nonetheless, when implementing a robotic system to perform a specific task, certain behaviors can be used to increase the performance and reliability of the system. Hence, in Tab. I operating environments, possible applications that are implemented by the presented behaviors, and industries for which swarm robotics is targeted are presented. For each industry, an article is also presented describing an example of the application of swarm robotics for that industry. The interested reader can also find swarms used in laboratories and industry (as of 2020) in the article [5], Furthermore, a broad overview of research and applications can also be found in the article [24] from 2021.

*3) Why don't we see robotic swarms on a daily basis?:* Although we are able to identify specific types of behavior found in swarms, a big problem is linking them together. For this reason, achieving even simple behaviors without the slightest bit of global knowledge is difficult. For example, a swarm of centrally controlled drones can easily arrange into complex shapes.(Fig. 2a). A commonly targeted task in swarm robotics is shape formation because it combines multiple behaviors, such as coordinated motion, pattern formation, decision-making, and many others. Shape formation is not the same as pattern formation - shapes, unlike patterns, can be irregular. However, creating even simple shapes without global knowledge is a significant problem. For example, the problem of arranging letters, which are slightly more complicated shapes, is challenging, and the results are not very spectacular (Fig. 2b). That is why researchers often use mixed systems, for example, in [38] GPS was used to simplify the task slightly.

*4) What is the future of swarm robotics?:* In 2020, an article was presented [39] that discusses the future of swarm robotics development in a rather loose way. Although it is a visionary article, it was co-written by Prof. Marco Dorigo, one of the pioneers of swarm intelligence and the creator of the Ant Colony Optimization algorithm (Alg. 1). The article estimates the development time of swarm robotics and potential applications. According to the authors, by 2030, drone swarms will be used widely in precision agriculture, infrastructure maintenance, and military (for the sake of reconnaissance missions). Subsequently, the development of swarm robotics will be followed in underwater robotics and the entertainment industry. At that time, robot swarms are expected to be found commonly in cities. By the year 2050, robots' swarms will be used in space exploration and even in the precise delivery of medicines inside the human body in the form of swarms of nanorobots. It is worth mentioning that work on molecular machine transportation is ongoing, as one can find in [12], as well as the research on other applications. While the time horizon is hard to predict, the proposed applications seem likely to appear in the future.

Science fiction literature also provides insight into the development of swarm robotics, which is currently strongly ahead of the technical possibilities. The strong connection between literature and swarm robotics was noted in an article titled "Swarm Robotics in Science Fiction" [40] in the journal Science Robotics in 2021.

### B. Optimization

Optimization algorithms based on the principles of Swarm Intelligence have initiated intensive research into creating new SI solutions. Earlier there was a study on the description of biological systems Due to a certain simplicity and surprising efficiency, for instance, in finding solutions to nonlinear systems, as indicated in [41], they are used in: transportation problems, network routing, route planning, robotics, scheduling, energy systems, parameter optimization, image processing, signal processing, and many others. To indicate the wide range of applications, below are listed problems and methods of solving them using these algorithms. In addition, review articles for each problem are included, where the reader can find many examples of applications and solutions to the problem. As can be easily seen, the most common uses of optimization IR algorithms are for task scheduling and route selection in dynamic systems.

- Transportation engineering - optimization algorithms are used to solve vehicle routing and task scheduling problems, both for static and dynamic tasks [42].
- Photovoltaic energy storage systems - optimizing the use of infrastructure [43].
- Path planning - similarly to the above examples, those algorithms perform well not only in static environments but also in dynamic ones [44].
- Network routing - due to swarm characteristics (scalability, adaptability, resilience), optimization algorithms are used to create routing protocols, e.g., in Wireless Sensor Networks [45].
- Processor task scheduling - distributed task scheduling for High-Throughput Computing (HTP) [46].
- Cloud computing - using IR optimization algorithms to improve optimization and task allocation in cloud computing [47].

### C. Telecommunications

A non-obvious application of distributed intelligence is in the domain of telecommunications networks. Due to their characteristics, they have a partially decentralized structure. The more decentralized a given network is, the greater the opportunities for applying SI appear. One of the first applications of IR in telecommunications was, as already mentioned, the optimization of routing in networks [48][49]. SI can also play an important role in the 6G network nowadays. As indicated in [50] the optimization algorithms based on swarm intelligence will be an integral part of the computational intelligence layer, along with the rest of the AI algorithms.

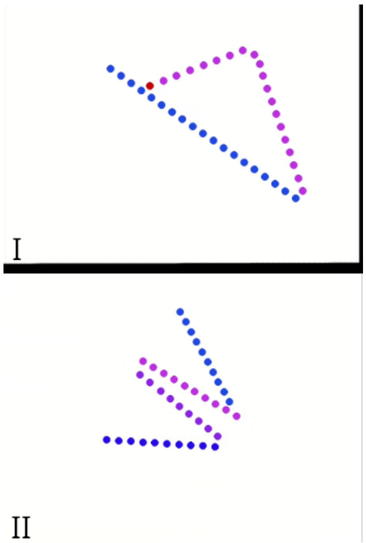
TABLE I: Industries interested in swarm robotics by operational environment, applications, and behaviors.

| Environment | Application | Behaviors | Industry |
|---|---|---|---|
| Land | warehouse work - cargo transportation sorting | collective transportation, coordinated motion, task allocation, group size regulation | logistics [24] |
| | construction | pattern formation, self-assembly, aggregation, self-reproduction | construction [22] |
| | dynamic area coverage, scouting and logistics | collective exploration, collective localization, collective perception | military [25] |
| | crop inspection, seeding, plant care, cereal harvesting | coordinated motion, collective transportation, pattern formation | agriculture [24] |
| Underground | wydobycie, transport ładunku, eksploracja | collective transportation, coordinated motion, task allocation, consensus achievement, aggregation | mining [26] |
| | poszukiwanie osób zaginionych | collective exploration, task allocation, collective perception | emergency services [27] |
| Water | environment monitoring | collective perception, coordinated motion | ecology [28] |
| | underwater exploration, underwater infrastructure monitoring | collective exploration, coordinated motion, collective perception, collective localization | extraction industry [29] |
| | anti-submarine warfare, mine, mine clearing, reconnaissance, guarding | collective exploration, coordinated motion, collective transportation, task allocation, aggregation, pattern formation, self-assembly, collective fault detection | military [30] |
| Air | target finding and tracking, reconnaissance, guarding | collective exploration, coordinated motion, collective transportation, task allocation, aggregation, pattern formation, self-assembly, collective fault detection | military [31] |
| | search for missing persons | collective exploration, task allocation, collective perception, coordinated motion | emergency services [10] |
| | infrastructure inspection | collective perception, coordinated motion, pattern formation | construction [32] |
| | crop inspection | coordinated motion, pattern formation, collective perception | agriculture [33] |
| Space | study of the surface of celestial bodies (e.g., seismographic study), creating spatial maps | collective exploration, coordinated motion, collective perception | space exploration [34] [35] |
| | finding deposits of natural resources, natural resources extraction, transportation | collective exploration, collective transportation, coordinated motion, task allocation, consensus achievement, aggregation | space mining [36] |
| | construction in orbit or on a celestial body | collective transportation, aggregation, pattern formation, self-assembly, collective fault detection, self-reproduction | space construction [37] |

For example, Industrial Internet of Things (IIoT) networks have advanced perceptual capabilities, intelligent information processing capabilities, and the ability to self-organize and self-maintain [51]. For example, Industrial Internet of Things (IIoT) networks have advanced perceptual capabilities, intelligent information processing capabilities, and the ability to self-organize and self-maintain. It is not difficult to see the strong connection between IIoT and SI. Indeed, IIoT often makes use of Wireless Sensor Networks (WSNs) in the communication layer, which utilize SI elements. Because of its ability to perceive its environment and process information, IIoT has great potential to use various SI phenomena to multiply its capabilities. One of the problems of the IIoT is the need to process a significant amount of data collected from sensors. Processing certain information locally could enhance the IIoT's capabilities and even increase its perception capabilities, as in swarm robotics. However, at this point, this is a topic that needs to be further studied.

These WSN flat networks are supposed to be self-maintaining, so they mostly lack global knowledge elements and often operate on an ad-hoc basis [52]. For this reason, their use is most often found in hostile environments [51]. Hence, they can be considered a SI system. Many of the routing algorithms used in WSNs are used in swarm robotics, for example, the gradient communication paradigm [53] was used by [54] to self-distribute tasks among the members of the robotic swarm. Nevertheless, WSNs also utilize other swarm intelligence applications to solve their problems. In particular, mobile WSNs (MWSNs) use IR optimization algorithms to optimize network coverage, sensor distribution, and routing protocols, among other things [51]. Of considerable interest is network coverage, since WSNs intended to operate in hostile


(a) Drones using global knowledge - used in the entertainment industry.




(b) A swarm of drones forming shapes in a simulation (I - letter "P", II - letter "W") [15].

Fig. 2: Comparison of centralized multi-robot system and the decentralized swarm.

environments may, for instance, be dropped from helicopters. It is then necessary to put to sleep some of the nodes being clustered in one place in order to optimize resources [55].

Although distributed intelligence is currently not commonly found in telecommunications networks, the potential for applications of certain SI elements is significant.

### D. Distributed systems

Orchestration is a process used to automate the coordination and management of distributed systems. Its name originates from the coordination of the instruments in an orchestra, which leads to the creation of music [56]. What is noticeable for orchestration is the presence of at least one “conductor” - a central/hierarchical center of central knowledge. The concept of orchestration is most often found in the context of telecommunications. For example, in mobile networks (5G/5G+), orchestration is commonly used to deal with network functions running in the cloud or to manage virtualized infrastructure. Orchestration is responsible for the continuous maintenance of proper system operation and self-repair. This is done by properly managing individual functions/subsystems and their lifecycle, etc.

The use of a mobile network system using SI was proposed in master’s thesis [57], in which there was proposed a system which used mobile network terminals to create a consistent computing infrastructure operating on an ad-hoc basis. The proposed system aimed to operate in emergencies without access to a central network. Therefore, the resources available on the terminals were limited, and their mobility caused rapid changes in the topology. By using elements of Swarm Intelligence, including gradient communication, the system was able to maintain a constant level of computing resources for some time.

Swarm Intelligence systems could potentially support, and in the future, even replace current orchestrators in some applications. Orchestration is expensive - it is the hidden cost of information exchange and processing. By designing a function using SI, this cost could be significantly reduced. One of the goals of swarm intelligence research is to create a logical swarm - a distributed system in which there is no central management, and the resources of the whole system are treated as a consistent infrastructure. The capabilities of such a swarm are to make efficient use of all the resources that belong to it, regardless of how many they have at any given time. In telecommunications, such an infrastructure comes under the term Cloud-Continuum, but currently proposed solutions use only global knowledge. Nevertheless, SI research can also contribute to the efficiency of similar solutions by, for example, partially offloading the orchestrator.

### E. Biological systems

The original focus of research on Swarm Intelligence was the study of the swarm behavior of relatively simple insects such as ants, termites, and cockroaches. Although a single agent (in this case, an insect) does not know the whole operation of the system, the swarm exhibits various emergent behaviors that enable them to achieve effects that are not observable to a single agent, but are relevant to the system as a whole. An example is the construction of termite mounds - they can reach up to 30 meters in diameter and 6 meters in height. These structures are designed to provide adequate ventilation to regulate temperature and maintain sufficient humidity to provide suitable conditions for the swarm development.

Slightly more advanced creatures that also exhibit collective behavior include schools of fish and flocks of birds. The study of biological systems and, more specifically, attempts to describe the mathematical relationships in a swarm have led to the development of optimization algorithms. The article [58] provides further information concerning the description of biological systems and their evolution into Swarm Intelligence algorithms.

Another interesting use of Swarm Intelligence principles is to describe decision-making and some of the behavior of groups of people. This research aims to model some collective behaviors so as to better understand why they appear and how they affect group behavior. The idea is to support the decision-making process in the future. Article [59] describes the possible use of SI to describe human collective behavior, while in article [60] one can find a survey on state of the art in this domain. In article [61] there were shown similarities between the neural cognition and collective cognition. Hence, one can observe that indeed the SI is returning to its roots, that is to describe biological systems; however, in a different way than it used to.

## III. Problems with Swarm Intelligence

The analogy of a conductorless orchestra can once again be used to point out the fundamental obstacles to creating IR systems. A well-prepared orchestra certainly had to practise to play a given piece for a long time. Preparation must have consisted of both independent and group practice. Furthermore, such practice required an unchanging composition of the orchestra. One can easily imagine that a well-prepared but late-arriving musician would be able to join the orchestra and, after a moment's listening, play his part. The problem begins to get more serious when the musicians in the orchestra have never played with each other before and join or leave the orchestra at random times. One can suppose that such an orchestra could grow so large that certain sections of the instruments would be inaudible to others. Furthermore, if a musician had a whole set of pieces that could be played during such an unusual concert, there is a certain chance that they would start playing the wrong piece. Another musician who would join the orchestra, hearing different tunes, could play something different - the resulting music could significantly differ from the intended one. In swarm intelligence, we are dealing with such an unusual orchestra. Therefore, the goal of swarm intelligence engineers is to compose such a concert for each section of instruments that the resulting music meets certain expectations. Standard complex systems have an element of global knowledge - a conductor, who can make such a system play the whole piece more or less correctly; however, it can be overloaded when the system grows enough.

On the basis of the aforementioned example, one can easily observe the problems with the Swarm Intelligence systems. The main problem is to find a solution to combine the behavior of a single agent (micro level) with the emergent behavior of the whole collective (macro level) without any centralized synchronization element as shown in Fig. 3. As pointed out by [62] currently there are no universal methods which would allow for join of the micro level behavior with the macro level behavior. SI systems engineers are frequently inspired by the behaviors observed in the nature [3]. Nevertheless, they have significant limitations of applications.

Swarm intelligence systems can be divided into layers of abstraction that interact with each other. Such a decomposition idea for swarm design was proposed by [63], where a division was divided into four layers for unmanned aerial vehicle: decision-making layer, path planning layer, trajectory generation layer, and redundancy management layer Continuing with the idea, but focusing more on the characteristics of the micro-macro swarm there is proposed another decomposition in Fig. 4. The micro level (physical layer) directly describes the capabilities of a single agent, while the macro level (intelligence layer) describes the operation of the entire system. All layers are involved in linking the individual-collective behavior, and each layer has different tasks (Tab. **??**). An additional problem is that the layers directly influence and constrain each other - for instance, the physical layer capabilities of an individual agent determine its perception and communication capabilities, which also influence the decision. The collective decision, on the other hand, influences the behavior of a single agent, indirectly influencing other agents and therefore subsequent decisions. Although it is possible to decompose IR systems into layers and design each layer separately, combining them into a coherent system remains difficult. However, decomposition makes it possible to look at specific types of behavior in more detail, thereby sorting out the SI system in some way.

The process of creating swarm intelligence systems is divided into two stages: design and modeling. SI system design is responsible for the overall planning process of a given system, while modeling is responsible for the mathematical representation of the designed system.

## IV. SI systems design

Returning to the concert analogy, the design of SI systems can be compared to the general idea of a musical piece. More precisely, it is about the goal the piece is supposed to achieve - the composer has to determine whether it should be an opera telling a story or a Mazurka intended to evoke certain emotions and sensations in the audience. The composer determines what should happen and in what parts. That is how the SI system design looks like - the SI systems engineer at this level determines what goals the swarm is supposed to achieve, what the stages of actions are supposed to be, and what the expected behavior and interactions are.

When discussing the design of SI systems, one usually assumes the design of swarms of robots. It results from the fact that they represent independent agents, and unlike biological systems, it is possible to design new behaviors. Biological or optimization behavior models are briefly described in the section V, as they most often specify a single behavior, rather than a set of behaviors, as it happens at the design stage.

Designing swarm intelligence systems is a difficult task due to the need to combine micro and macro levels. For this reason, swarm design methods for solving specific tasks have been developed over the years, of which three main categories can be distinguished [3][64]:

- behavior-based design;
- supported design;
- automatic design;

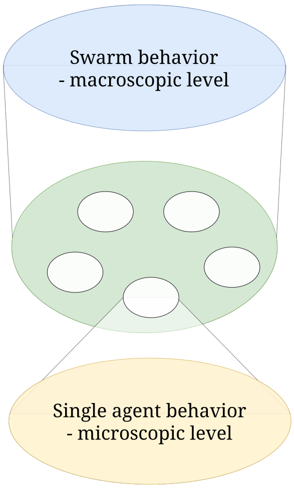



Fig. 3: Micro-macro perspective on Swarm Intelligence.

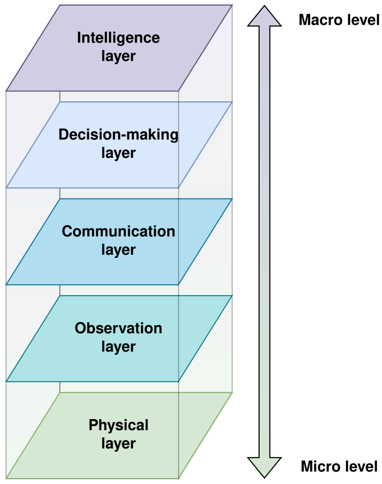



Fig. 4: Division of th SI systems into layers.

Fig. 5: Two types of SI system decomposition.

TABLE II: Tasks in different layers of SI systems.

| Layer | Description | Task |
|---|---|---|
| Physical | the agent's activity on its environment (physical or virtual) | - controlling the agent<br>- the agent's impact on the environment |
| Observation | agent and swarm perception capabilities [5] | - agent's perception capabilities<br>- collective perception of the whole swarm |
| Communication | communication between agents | - information exchange between agents (e.g., type of the communication, communication scheme)<br>- information exchange in the group of agents (e.g., message routing) [15] |
| Decision-making | decision-making agent $\leftrightarrow$ swarm | - collctive decision making (e.g. voting, majority rule) [64]<br>- local decision-making (e.g., leader-follower [65], observation based [66], stigmergy based [67]) |
| Intelligence | observed high level behavior | - collective task solving by joining and changing the behaviors of lower layers. |

### A. *Behavior-based design*

Behavior-based design was proposed by [3] as an iterative process of designing, testing, and improving the behavior of a single robot to achieve a complex global effect. Although a design process assumes a lot of trial and error, it remains the most popular swarm design method. Most often, designing with this method uses observation of multiple behaviors found in nature, thus improving the process (e.g., the cockroach aggregation presented in Fig. 6).

*a) (Probabilistic) Finite-State Machine:* As [3] points out, the most popular method for designing robot swarms is PFSM (Probabilistic Finite-State Machine). That is primarily due to its relative intuitiveness - when designing swarms based on PFSM, one focuses on the expected behavior of the swarm. The approach is usually based on a top-down approach - the engineer focuses on achieving the goal with specific states and transitions from them for the entire system. Subsequently, one designs the lower level of abstraction (a single agent behavior for most cases) to close the gaps as much as possible. Finite-state machines allow both micro-level and macro-level design, however, according to [64] there is a fundamental problem of combining both levels.

The exemplary application of finite automata is the aggregation of a swarm of robots, motivated by the behavior of cockroaches [6]. In nature, the more cockroaches in a given cockroach's closest neighborhood, the greater the chance of stopping it [68]. Simplifying the algorithm, the researchers in the article [6] proposed several simple aggregation strategies; two proposed automata implementing them are seen in Fig. 6. The first strategy, based on finite automata (a robot wanders in a random direction; if it approaches a neighboring robot for a certain distance, it stops), leads to the formation of many small clusters of robots. In contrast, the second one based on probabilistic finite automata - the robot, after approaching its neighbor, also stops, but still draws a random value of *rand*, which if it is greater than the given threshold of $P_{exit}$ then it enters the waiting state. Similarly, the robot moves from the waiting state to the wandering state. The second strategy leads to robots merging into larger groups, which is the desired behavior for this task.

In the case of swarms, finite-state machines often lead to the dominance of one of the states [64]. - primarily at the level of collective decision-making. The problem deals with choosing which state - micro or macro - is to dominate the behavior of

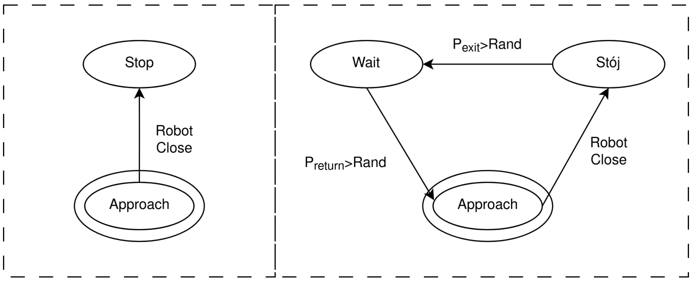



Fig. 6: Exemplary SI system design (robot aggregation) with the use of: a) Finite-state machine (FSM), b) Probabilistic finite-state machine (PFSM) [6].

the individual/collective. Therefore, the probability of exiting a state is enables the negative feedback loop so that the system can become self-organizing [64].

*b) Virtual Physics:* This is a wide range of IR system design methods that is inspired directly from physics. It was proposed by [3]; however, methods themselves have been used since the 1980s and are quite popular. A common application is to treat agents as particles that can interact with each other [69], or to use virtual forces that affect agent behavior (e.g., an artificial potential field [70]). It is hard to identify a specific design process using these methods as they differ significantly between each other, but given their popularity and effectiveness in certain behaviors, they cannot be ignored.

*c) Hierarchical Distributed Clusters:* This is the design method presented by [15] (Fig. 7), in which a complex behavior ($n$-super cluster) is broken down into a set of simplified behaviors ($n-1$-super clusters). The simplified behaviors are broken down again into another, even simpler behavior. This process is repeated until the very basic behaviors of a single robot are obtained. The method makes it possible to choose different design methods for each cluster and at each level of abstraction, thus simplifying the entire design process. For instance, one cluster can use PFSM and another can utilize virtual physics.

### B. Supported Design

It is a set of methods that aim to predict the final behavior of a swarm even at the design stage. An example of such a design is the property-driven design method proposed by [71], which involves prescriptive modeling and checking the resulting model. In this case, the designer first creates a prescriptive model of the swarm, which describes the target behavior but does not implement the whole thing. What is more, its properties are checked even before implementation. Thus, some system features can be verified before the time-consuming implementation. The method consists of four phases, and each phase consists of layers (Fig. 8 - blocks represent the layers). In the first phase, the expected properties of the swarm are defined. In the next phase, a prescriptive model is created based on them. Then, based on the model, a simulation of the expected swarm is created. In the last phase, the behaviors are implemented on robots. As highlighted [71] in each phase, the layers defined in previous phases are updated to maintain consistency.

Another method supported design method was proposed by [72]. It used the concept of design patterns known from software engineering. The method aims to develop a model for the macro level and then the micro level for a specific class of problems. Further, it describes the relationship between the micro and macro levels. The designer then implements specific behaviors on a given agent, following the rules defined by the obtained model.

### C. Automatic design

Automated design methods aim to streamline the creation of robot swarms by automatically generating robot controllers. Unlike supported design methods, they do not aim to predict the final result by supporting the designer, but to eliminate the need for a swarm engineer. Automatic methods may use the trial-and-error principle, but due to the lack of manual correction, they can perform many trials in a relatively short time. In the following paragraphs, the two most popular design methods will be presented: optimization-based design and reinforcement learning based desing.

*a) Optimization-based design:* As noted in [73], automatic design is frequently the problem of selecting the optimal controller from a pool of controllers in a given design space. Such a controller should maximize the selected performance metrics of the overall system. In such a system, automatic design actually solves an optimization problem, hence the name optimization-based design.

This approach often combines the use of neural networks that form the controller of an individual robot and evolutionary algorithms that select the parameters of these networks and their topology [74]. Due to the usage of evolutionary algorithms, these design methods are often also referred to as evolutionary swarm robotics. As the researchers noted in the article [73], most work on evolutionary swarms shares three important characteristics:

1) All robots in the swarm have an exact copy of the same controller (homogeneous swarm).
2) The objective function is defined globally, and it is centrally evaluated.
3) For the sake of optimization, the evolutionary algorithm is used.

This design approach is surprisingly similar to optimization algorithms that use swarm intelligence. However, unlike commonly used SI optimization algorithms, optimization-based design has not yielded spectacular results at this point.

*b) Reinforcement Learning based desing:* Novelist Stanisław Lem proposed an alternative approach to evolutionary swarms in his 1964 novel "The Invincible" [75]. The swarms of robots depicted in the work had the ability to self-replicate and improve the next generations. That phenomenon was called necroevolution, the evolution of non-living matter. While the exact use of necroevolution seems difficult to implement, such an approach that would improve successive generations has inspired researchers to design swarms using reinforcement learning.

In reinforcement learning, the environment gives feedback (reward) to the agent, while the agent's task is to maximize

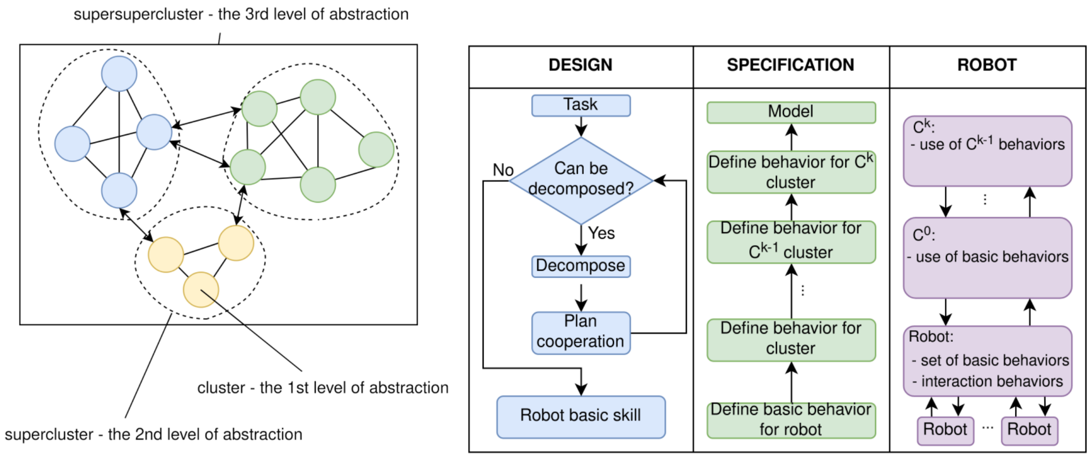



Fig. 7: Hierarchical Distributed Clusters method for SI system design [15].

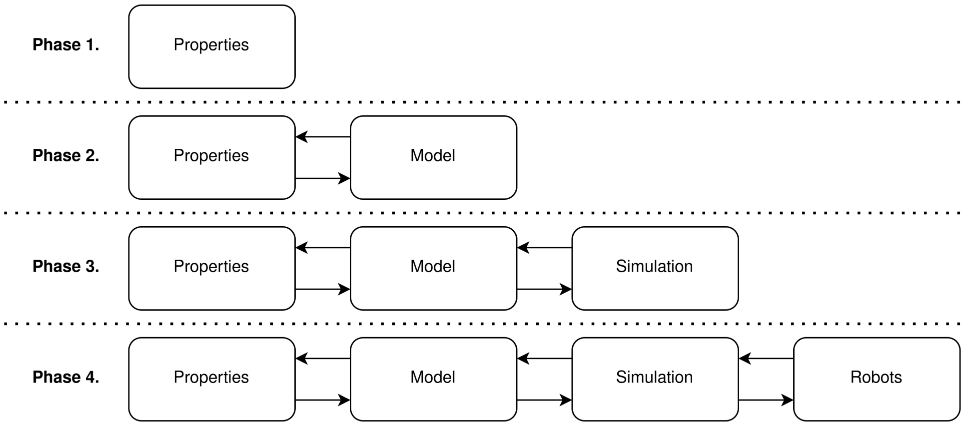



Fig. 8: Property-driven design for robotic swarm proposed by [71].

the reward. In the case of swarm robotics, the agent is a single robot with a particular skill set. The two most popular methods for teaching such SI systems are joint action learners (JAL) and multi-agent reinforcement learning (MARL) methods. Both methods can be seen in Fig. 9. The JAL method is characterized by perceiving the state of the environment as a whole and also rewarding robots for acting in combination. In contrast, the MARL method evaluates each robot separately. It can be clearly seen that a micro-macro problem occurs here as well.

As the researchers point out in the article [76], the performance of the systems designed using reinforcement learning methods was initially unsatisfactory. The development of reinforcement learning methods and combining them with classical principles of collective behavior (such as collision avoidance) began to yield better results. Numerous examples of the use of reinforcement learning to design swarm systems can be found in the aforementioned article [76].

*c) Future developement of automatic design methods:* In the article [73], the authors suggested that nowadays there is still too little empirical experience in designing swarms to make the automatic methods successful. As in the case of reinforcement learning, much better results will likely be achieved when automatic methods are built on the achievements of classical methods. For this to happen, it is necessary to develop design methods further and combine them with modeling methods to increase their versatility.

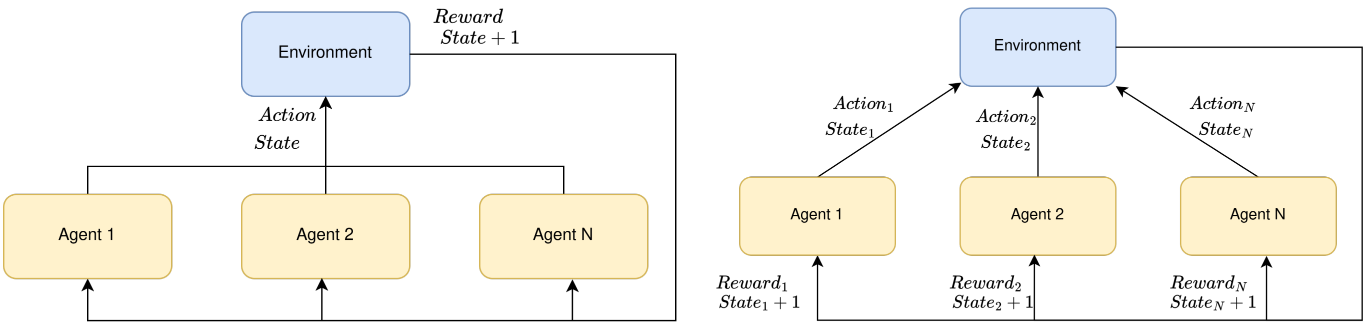
## V. Swarm Intelligence systems modeling

SI system modeling is like a music note sheet for each instrument section and each musician - an accurate and formal (mathematical) description of a specific behavior for a single agent or a particular group (or a whole swarm). In other words, the SI system model complements the previously created design that indicates expected behavior with a description that tells how to do it.

### A. Inherent elements of SI models

Despite the multiplicity of approaches to SI systems and their applications, certain elements present in any swarm intelligence system can be identified. This section will briefly discuss these.

*1) Randomness:* A common practice in modeling SI systems is to use an element of randomness, or as [78] points out, "craziness." Swarm intelligence systems tend to fall into local minima [78] or stall at the target state (as in Fig. 6 a)). In order to break the target state and allow the system to self-organize again, adding a negative feedback [64] is necessary. The most common way to do this is to add an element of "craziness" that makes each agent have a chance to get out of the current state, and thus the whole system, as indicated in Fig. 6 b).




(a) Joint Action Learners (JAL) [77].

(b) Multi-agent Reinforcement Learning (MARL) [77].

Fig. 9: Reinforcement learning methods used in swarm robotics.

*2) Neighborhood:* One of the basic terms used with all SI systems is neighborhood, denoting the other agents in an agent's immediate vicinity, with whom the agent can mutually interact. In fact, the neighborhood can be defined in terms of the three SI layers (out of five defined in section III): observation, communication, and decision. Most often, the neighborhood is defined in one of them and taken as the definition for the whole system. As indicated [79] in swarms of robots, the narrower the neighborhood (in this case, communication), the more efficient the system. On the other hand, [80] indicates that for optimization tasks, wider neighborhoods in the PSO algorithm (Alg. 2) perform better for simpler problems, while narrower neighborhoods for more difficult problems. Consequently, the neighborhood is one of the inherent components of all SI models.

## B. Optimization models

They derive from behavior-based design - one of the first optimization techniques using SI (ACO) was inspired by observing ant colonies [81]. Many other optimization techniques using SI have been similarly developed; an overview and description of many techniques can be found in [82]. The two most important optimization algorithms using SI: ACO and PSO will be briefly described.

*a) ACO:* - Ant Colony Optimization algorithm is a meta-heuristic algorithm, inspired by ants searching for food [83]. While searching for food, the ants wander in a random direction. If they find it, they release pheromones that depend on the quantity and quality of the food. As the ant carries the food to the anthill, it creates a trail of pheromones. Other ants can smell the pheromones and are more likely to choose paths with a strong scent.

ACO was originally developed for discrete optimization tasks, but it has also found application in continuous optimization [80]. The ACO algorithm proposed by [81] aimed to solve static problems (i.e., discrete optimization, in which the characteristics of the problem do not change during its solution). The algorithm is shown in Alg. 1. The principle of its operation is as follows: in the first step, *ConstructAntsSolutions()* each ant moves in its neighborhood (problem space), making a decision to move based on the level of pheromones and a heuristic function. After each ant moves, they solves the problem for a given value (their location). Once these are solved, a pheromone update *UpdatePheromones()* is performed, in which the information from the best (or all) solutions is used to reinforce the corresponding paths, making them more attractive to the other ants. In addition, optional "daemon" actions can be performed (*DaemonActions()*) - such that require global knowledge of the problem (e.g., local search (*ApplyLocalSearch()*) or selective update of pheromones). The whole process is repeated until the termination conditions are met.

**Algorithm 1** Ant Colony Optimization (ACO) [81]

**while** not terminated **do**
    ConstructAntsSolutions()
    UpdatePheromones()
    DaemonActions() %optional
**end while**

*b) PSO:* - Particle Swarm Optimization was proposed by [78] as an optimization method for continuous nonlinear functions. Like ACO, it was inspired by nature - the algorithm was originally designed to simulate a flock of birds with collision avoidance. Over time, it began to resemble the behavior of a particle swarm rather than a flock of birds. As a consequence, the term "particles" was chosen as the authors did not use mass or volume, but they did use velocity and acceleration. Hence, the term particle swarm optimization.

A simplified description of the canonical PSO algorithm [1] (its most popular version) was proposed by [80] and can be seen at Alg. 2. By definition, the algorithm is a swarm and therefore can execute on all particles simultaneously, while iterativity is a simplification of the algorithm. The principle of the algorithm is as follows: at runtime, particles move through the searched space in search of an optimal (or sufficient) solution. A particle communicates with other particles in their neighborhood (the definition of neighborhood can vary depending on the application; it can be very narrow or wide) and passes information about their best position to each other. Based on the information they have, each particle calculates a new velocity. The velocity is affected by three factors: 1) the previous velocity, 2) the direction to the best own position, and 3) the direction to the best position in the neighborhood. Each time randomized, the velocity estimate is influenced by a factor $R$, which is a negative feedback to increase the searched space and reduce the risk of falling into a local minimum. In addition, the influence of the velocity factors can also be adjusted. After calculating the velocity, each particle updates its position with it and propagates to its neighbors. The procedure is repeated until a satisfactory solution is found.

**Algorithm 2** Particle Swarm Optimization (PSO) [80]

```
while not terminated do
    for each particle i do
        if f(x_i) < p(x_i) then
            p_i ← x_i
        end if
        p_g = min(p_neighbors)
        v_i ← v_i + c_i R_1 ⊙ (p_i − x_i) + c_2 R_2 ⊙ (p_g − x_i) %
Update velocity
        x_i ← x_i + v_i %Update position
    end for
end while
```

For each particle $i$: $x_i$ - particle's position, $v_i$ - particle's velocity, $p_i$ - particle's best position, $p_g$ - The best position in the local neighborhood of the particle, $f()$ - Function for which the minimum/maximum is being searched, $R_1$, $R_2$ - Coefficients obtained from uniform distribution $[0, 1]$, $c_1$, $c_2$ - Coefficients that define the influence of the given factors. Additionally, $\odot$ is the Hadamard product - the particle's position can be defined in $d$ dimensions.

### C. Modeling of the robotic swarm

Currently, there is no universal method for modeling robot swarms, but the set of tools proposed for creating models is rather extensive. Most often, modeling a swarm is divided into two levels—micro and macro. So far, there are few methods that connect the micro level with the macro level directly. In most cases, one models the two levels separately and then tries to combine them. This chapter will briefly describe the most popular methods for designing swarms at the micro-macro levels, and a layer-based modeling method will be proposed.

*1) Micro level:* Micro-level models describe the internal state of the robot and its interactions with the environment and with other robots (layers: physical, observation, communication). Consequently, they strongly depend on the capabilities of a particular robot and require a description that directly targets a specific application. For this reason, the problem of describing the state of a single robot, and therefore describing the state of the entire system, arises.

*a) Langevin equation:* - is used to model an intermediate state between micro and macro. It is a stochastic differential equation used in physics to describe Brownian motion. In robotics, the deterministic part is used to model the motion of a single robot, while the stochastic part describes interactions with other robots and the environment [3]. It is sometimes combined with the Fokker-Planka equation, which describes the macro level, to combine the micro and macro [84] levels. That is one of the few examples of combining the two levels.

*2) Macro level:* Models describing the macro level focus on expressing the state of the entire system (decision and intelligence layers) without focusing on individual robots. Because models created at the macro level are more universal, there are significantly more solutions. This section will briefly present the most popular methods.

*a) Kinetic equations:* - methods popular in the early 2000s. Kinetic equations were borrowed from chemistry, where they are used to describe chemical reactions (more precisely, to describe the dependence of the rate of a chemical reaction on the concentration of reactants). In swarm robotics, they are used to create a macroscopic description of changes in robot states (a description of the time dependence of the number of robots in a given state). This modeling method often "fills in" the design of PFSMs. One of the first works using this method was [85], in which a swarm of robots taking out poles was described. A detailed description of the method can be found in [64]. The main limitation of this method is the need to determine the position of the robots over time.

*b) Differential equations:* - The most widely used is the Fokker-Planck equation, borrowed from physics. In physics, it describes the time evolution of the probability density function. In swarm robotics, on the other hand, it describes the probability density function of the states of all robots in the environment. In principle, any collective behavior can be described using the Fokker-Planck equation. This method has two major drawbacks - it is difficult to solve analytically (numerical solutions are computationally demanding), and secondly, some aspects, e.g., communication, are difficult to model [3].

*3) Layer-based modeling:* An alternative method of modeling robot swarms can be to use layers 4. This method allows certain aspects to be modeled separately, but again, there is the problem of combining them to get a consistent model. However, this method has the ability to focus on layers, allowing one to develop proven solutions to certain issues. For example, the problem occurring in the decision-making layer is heavily researched (in [64] there is an entire chapter devoted to collective decision-making. Furthermore, other disciplines also research the topic). Things are similar regarding the communication layer (for example, gradient communication [54]). By creating universal models in each layer, creating basic swarm functionality seems to be simplified. Nonetheless, connecting all layers together using a common model remains a pending problem.

### D. Novel approaches towards swarm modeling

A team from the University of Palermo [86] presented a rather unusual approach to modeling swarms using Quantum Information Technologies. The goal of the robot swarm was to find a target in a given space. The paper presented a model of the entire system as an interaction matrix of pairs of robots (Fig. 10). Each robot had a defined quantum state that determined the robot's position and the target's position. The superposition of the two expressed the reward. The system modeled this way was then implemented using quantum gates in the IBM Quantum Composer simulator. The simulation was able to achieve the expected effect for a swarm of 10 robots.

$$S_n^{(t)} = \begin{pmatrix} R_1 & R_1 * R_2 & \dots & R_1 * R_{n-1} & R_1 * R_n \\ R_2 * R_1 & R_2 & \dots & R_2 * R_{n-1} & R_2 * R_n \\ & & \vdots & & \\ R_{n-1} * R_1 & R_{n-1} * R_2 & \dots & R_{n-1} & R_{n-1} \\ R_n * R_1 & R_n * R_2 & \dots & R_n * R_{n-1} & R_n \end{pmatrix}$$

Fig. 10: Robotic swarm state expressed as the pairwise interaction matrix [86].

Currently, swarm modeling methods still have the fundamental problems of universality and combining different levels or layers. This area is, therefore, very interesting for further research.

## VI. Conclusions and Future Prospects

This article outlined the issues of Swarm Intelligence and introduced the reader to the current achievements in its field. It also presented the applications of IR systems. Then, it presented the current methods of designing Swarm Intelligence systems and outlined the most popular modeling methods.

An experienced musician can play a vista - without prior preparation. The musician receives the notes just before playing the piece, along with information on the key, tempo, and melody line. In particular, the melody is important, as it connects each instrument (micro level) to the effect produced by the whole orchestra (macro level). Returning to the unusual concert of Swarm Intelligence consisting of all agents playing a vista, one can see that at this point Swarm Intelligence still needs to define its "melody" to enable it to play much more complex "pieces." In addition, the "melody" of the swarm would aim to combine elements of projection with modeling, combining micro and macro levels.

Although Swarm Intelligence is still at a relatively early stage of development, this remarkable subset of artificial intelligence has much to offer in many industries. The aim of the SI work is to create the General Swarm Intelligence proposed by [87] in the future. - to combine a logical swarm (e.g., a coherent infrastructure of distributed computing resources) with a physical swarm (e.g., robots). Such a procedure aims to increase the autonomy and capabilities of relatively simple physical agents.

Given the numerous applications and opportunities of swarm intelligence, its further development is undoubtedly worth watching.

## References

[1] J. Kennedy, *Swarm Intelligence*. Boston, MA: Springer US, 2006, pp. 187–219. [Online]. Available: https://doi.org/10.1007/0-387-27705-6_6

[2] M. Dorigo, G. Theraulaz, and V. Trianni, “Swarm robotics: Past, present, and future,” *Proceedings of the IEEE*, vol. 109, pp. 1152–1165, 07 2021. [Online]. Available: https://doi.org/10.1109/JPROC.2021.3072740

[3] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, “Swarm robotics: a review from the swarm engineering perspective,” pp. 1–41, Mar 2013. [Online]. Available: https://doi.org/10.1007/s11721-012-0075-2

[4] T. Kuyucu, I. Tanev, and K. Shimohara, “Superadditive effect of multi-robot coordination in the exploration of unknown environments via stigmergy,” *Neurocomputing*, vol. 148, pp. 83–90, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231214009242

[5] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, “Swarm robotic behaviors and current applications,” *Frontiers in Robotics and AI*, vol. 7, 2020. [Online]. Available: https://doi.org/10.3389/frobt.2020.00036

[6] O. Soysal, E. Bahçeci, and E. Sahin, “Aggregation in swarm robotic systems: Evolution and probabilistic control,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 15, pp. 199–225, 2007. [Online]. Available: https://api.semanticscholar.org/CorpusID:16198241

[7] W. Spears, D. Spears, J. Hamann, and R. Heil, “Distributed, physics-based control of swarms of vehicles,” *Auton. Robots*, vol. 2, 06 2004. [Online]. Available: https://doi.org/10.1023/B:AURO.0000033970.96785.f2

[8] A. Christensen, R. O’Grady, and M. Dorigo, “Swarmorph-script: A language for arbitrary morphology generation in self-assembling robots,” *Swarm Intelligence*, vol. 2, pp. 143–165, 12 2008. [Online]. Available: https://doi.org/10.1007/s11721-008-0012-6

[9] G. H. Gebhardt, K. Daun, M. Schnaubelt, A. Hendrich, D. Kauth, and G. Neumann, “Learning to assemble objects with a robot swarm,” in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS ’17. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2017, p. 1547–1549.

[10] M. Bakhshipour, M. J. Ghadi, and F. Namdari, “Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach,” *Applied Soft Computing*, vol. 57, pp. 708–726, 2017. [Online]. Available: https://doi.org/10.1016/j.asoc.2017.02.028

[11] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, “A survey on aerial swarm robotics,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018. [Online]. Available: https://doi.org/10.1109/TRO.2018.2857475

[12] M. Akter, J. J. Keya, K. Kayano, A. M. R. Kabir, D. Inoue, H. Hess, K. Sada, A. Kuzuya, H. Asanuma, and A. Kakugo, “Cooperative cargo transportation by a swarm of molecular machines,” *Science Robotics*, vol. 7, no. 65, p. eabm0677, 2022. [Online]. Available: https://doi.org/10.1126/scirobotics.abm0677

[13] M. Rubenstein, A. Cornejo, and R. Nagpal, “Programmable self-assembly in a thousand-robot swarm,” *Science*, vol. 345, no. 6198, pp. 795–799, 2014. [Online]. Available: https://doi.org/10.1126/science.1254295

[14] Q. Shan, A. Heck, and S. Mostaghim, “Discrete collective estimation in swarm robotics with ranked voting systems,” 12 2021, pp. 1–8. [Online]. Available: https://doi.org/10.1109/SSCI50451.2021.9659868

[15] B. Mastej and M. Figat, “Hierarchical distributed cluster-based method for robotic swarms.” Wydawnictwo Politechnik Lodzkiej. [Online]. Available: http://hdl.handle.net/11652/4852

[16] D. Tarapore, J. Timmis, and A. L. Christensen, “Fault detection in a swarm of physical robots based on behavioral outlier detection,” *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1516–1522, 2019. [Online]. Available: https://doi.org/10.1109/TRO.2019.2929015

[17] T. A. Karagüzel, A. E. Turgut, A. E. Eiben, and E. Ferrante, “Collective gradient perception with a flying robot swarm,” *Swarm Intelligence*, vol. 17, no. 1, pp. 117–146, Jun 2023. [Online]. Available: https://doi.org/10.1007/s11721-022-00220-1

[18] J. Markdahl, D. Proverbio, and J. Goncalves, “Robust synchronization of heterogeneous robot swarms on the sphere,” in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 5798–5803. [Online]. Available: https://doi.org/10.1109/CDC42340.2020.9304268

[19] N. Cambier, V. Frémont, and E. Ferrante, “Group-size regulation in self-organised aggregation through the naming game,” in *International Symposium on Swarm Behavior and Bio-Inspired Robotics (SWARM 2017)*. [Online]. Available: https://hal.science/hal-01679600

[20] J. Timmis, A. Ismail, J. Bjerknes, and A. Winfield, “An immune-inspired swarm aggregation algorithm for self-healing swarm robotic systems,” *Biosystems*, vol. 146, pp. 60–76, 2016, information Processing in Cells and Tissues. [Online]. Available: https://doi.org/10.1016/j.biosystems.2016.04.001

[21] J. von Neumann, *Theory of Self-Reproducing Automata*, A. W. Burks, Ed. Urbana, IL: University of Illinois Press, 1966, posthumously compiled from von Neumann’s lectures and notes.

[22] A. Abdel-Rahman, C. Cameron, B. Jenett, M. Smith, and N. Gershenfeld, “Self-replicating hierarchical modular robotic swarms,” *Communications Engineering*, vol. 1, no. 1, p. 35, Nov 2022. [Online]. Available: https://doi.org/10.1038/s44172-022-00034-3

[23] 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 *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5948–5953. [Online]. Available: https://doi.org/10.1109/ICRA.2015.7140033

[24] P. G. F. Dias, M. C. Silva, G. P. Rocha Filho, P. A. Vargas, L. P. Cota, and G. Pessin, "Swarm robotics: A perspective on the latest reviewed concepts and applications," vol. 21, no. 6, p. 2062. [Online]. Available: https://doi.org/10.3390/s21062062

[25] T. Chung and R. Daniel, "Darpa offset: A vision for advanced swarm systems through agile technology development and experimentation," *Field Robotics*, vol. 3, pp. 97–124, 01 2023. [Online]. Available: https://doi.org/10.55417/fr.2023003

[26] J. Tan, N. Melkoumian, D. Harvey, and R. Akmeliawati, "Evaluating swarm robotics for mining environments: Insights into model performance and application," *Applied Sciences*, vol. 14, no. 19, 2024.

[27] S. Ratnayake and M. Figat, "Beacon-based swarm search and rescue." Wydawnictwo Politechniki Łódzkiej. [Online]. Available: http://hdl.handle.net/11652/4840

[28] I. Lončar, A. Babić, B. Arbanas, G. Vasiljević, T. Petrović, S. Bogdan, and N. Mišković, "A heterogeneous robotic swarm for long-term monitoring of marine environments," *Applied Sciences*, vol. 9, no. 7, 2019.

[29] N. D. Griffiths Sànchez, P. A. Vargas, and M. S. Couceiro, "A darwinian swarm robotics strategy applied to underwater exploration," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018, pp. 1–6. [Online]. Available: https://doi.org/10.1109/CEC.2018.8477738

[30] G. Liu, L. Chen, K. Liu, and Y. Luo, "A swarm of unmanned vehicles in the shallow ocean: A survey," *Neurocomputing*, vol. 531, pp. 74–86, 2023. [Online]. Available: https://doi.org/10.1016/j.neucom.2023.02.020

[31] Z. Xiaoning, "Analysis of military application of uav swarm technology," in *2020 3rd International Conference on Unmanned Systems (ICUS)*, 2020, pp. 1200–1204. [Online]. Available: https://doi.org/10.1109/ICUS50048.2020.9274974

[32] S. Halder and K. Afsari, "Robots in inspection and monitoring of buildings and infrastructure: A systematic review," *Applied Sciences*, vol. 13, no. 4, 2023. [Online]. Available: https://doi.org/10.3390/app13042304

[33] G. S. Berger, M. Teixeira, A. Cantieri, J. Lima, A. I. Pereira, A. Valente, G. G. R. d. Castro, and M. F. Pinto, "Cooperative heterogeneous robots for autonomous insects trap monitoring system in a precision agriculture scenario," *Agriculture*, vol. 13, no. 2, 2023. [Online]. Available: https://doi.org/10.3390/agriculture13020239

[34] E. Staudinger, D. Shutin, C. Manss, A. Viseras, and S. Zhang, "Swarm technologies for future space exploration missions," in *ISAIRAS '18: FOURTEENTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE, ROBOTICS AND AUTOMATION IN SPACE*, June 2018. [Online]. Available: https://elib.dlr.de/120345/

[35] D. St-Onge, M. Kaufmann, J. Panerati, B. Ramtoula, Y. Cao, E. B. Coffey, and G. Beltrame, "Planetary exploration with robot teams: Implementing higher autonomy with swarm intelligence," *IEEE Robotics & Automation Magazine*, vol. 27, no. 2, pp. 159–168, 2020. [Online]. Available: https://doi.org/10.1109/MRA.2019.2940413

[36] J. Tan, N. Melkoumian, R. Akmeliawati, and D. Harvey, "Design and application of swarm robotics system using abco method for off-earth mining," in *Proceedings of the Fifth International Future Mining Conference*, 2021.

[37] A. Vardy, "Orbital construction: Swarms of simple robots building enclosures," in *2018 IEEE 3rd International Workshops on Foundations and Applications of Self* Systems (FAS*W)*, 2018, pp. 147–153. [Online]. Available: https://doi.org/10.1109/FAS-W.2018.00040

[38] H. Wang and M. Rubenstein, "Shape formation in homogeneous swarms using local task swapping," *IEEE Transactions on Robotics*, vol. 36, no. 3, pp. 597–612, 2020. [Online]. Available: https://doi.org/10.1109/TRO.2020.2967656

[39] M. Dorigo, G. Theraulaz, and V. Trianni, "Reflections on the future of swarm robotics," *Science Robotics*, vol. 5, no. 49, 2020. [Online]. Available: https://doi.org/10.1126/scirobotics.abe4385

[40] R. R. Murphy, "Swarm robots in science fiction," *Science Robotics*, vol. 6, no. 56, p. eabk0451, 2021.

[41] J. Tang, G. Liu, and Q. Pan, "A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends," vol. 8, no. 10, pp. 1627–1643. [Online]. Available: https://doi.org/10.1109/JAS.2021.1004129

[42] D. Teodorović, "Swarm intelligence systems for transportation engineering: Principles and applications," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 6, pp. 651–667, 2008. [Online]. Available: https://doi.org/https://doi.org/10.1016/j.trc.2008.03.002

[43] S. Wang, Y. Yue, S. Cai, X. Li, C. Chen, H. Zhao, and T. Li, "A comprehensive survey of the application of swarm intelligent optimization algorithm in photovoltaic energy storage systems," *Scientific Reports*, vol. 14, no. 1, p. 17958, Aug 2024. [Online]. Available: https://doi.org/10.1038/s41598-024-68964-w

[44] C. Y. Jie, M. B. Jasser, S.-S. M. Ajibade, H. N. Chua, R. T. Wong, A. S. Rafsanjani, and A. P. A. Majeed, "A survey on swarm intelligence algorithms for optimizing path planning," in *2025 21st IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*, 2025, pp. 283–288. [Online]. Available: https://doi.org/10.1109/CSPA64953.2025.10933006

[45] M. Saleem, G. A. Di Caro, and M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions," *Information Sciences*, vol. 181, no. 20, pp. 4597–4624, 2011, special Issue on Interpretable Fuzzy Systems. [Online]. Available: https://doi.org/https://doi.org/10.1016/j.ins.2010.07.005

[46] E. Pacini, C. Mateos, and C. G. Garino, "Distributed job scheduling based on swarm intelligence: A survey," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 252–269, 2014, 40th-year commemorative issue. [Online]. Available: https://doi.org/https://doi.org/10.1016/j.compeleceng.2013.11.023

[47] Y. Qawqzeh, M. Alharbi, A. Jaradat, and K. Abdus Sattar, "A review of swarm intelligence algorithms deployment for scheduling and optimization in cloud computing environments," *PeerJ Computer Science*, vol. 7, p. e696, 08 2021. [Online]. Available: https://doi.org/10.7717/peerj-cs.696

[48] I. Kassabalidis, M. El-Sharkawi, R. Marks, P. Arabshahi, and A. Gray, "Swarm intelligence for routing in communication networks," in *GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No.01CH37270)*, vol. 6, 2001, pp. 3613–3617 vol.6. [Online]. Available: https://doi.org/10.1109/GLOCOM.2001.966355

[49] "Swarm intelligence for next-generation networks: Recent advances and applications," *Journal of Network and Computer Applications*, vol. 191, p. 103141, 2021. [Online]. Available: https://doi.org/10.1016/j.jnca.2021.103141

[50] B. Ji, Y. Wang, K. Song, C. Li, H. Wen, V. G. Menon, and S. Mumtaz, "A survey of computational intelligence for 6g: Key technologies, applications and trends," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 10, pp. 7145–7154, 2021. [Online]. Available: https://doi.org/10.1109/TII.2021.3052531

[51] L. Cao, Y. Cai, and Y. Yue, "Swarm intelligence-based performance optimization for mobile wireless sensor networks: Survey, challenges, and future directions," vol. 7, pp. 161 524–161 553. [Online]. Available: https://doi.org/10.1109/ACCESS.2019.2951370

[52] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008. [Online]. Available: https://doi.org/10.1016/j.comnet.2008.04.002

[53] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00. New York, NY, USA: Association for Computing Machinery, 2000, p. 56–67. [Online]. Available: https://doi.org/10.1145/345910.345920

[54] R. Haghighi and C. Cheah, "Multi-group coordination control for robot swarms," *Automatica*, vol. 48, no. 10, pp. 2526–2534, 2012. [Online]. Available: https://doi.org/10.1016/j.automatica.2012.03.028

[55] I. Khoufi, P. Minet, A. Laouiti, and S. Mahfoudh, "Survey of deployment algorithms in wireless sensor networks: Coverage and connectivity issues and challenges," *International Journal of Autonomous and Adaptive Communications Systems*, vol. 10, p. 341, 01 2017. [Online]. Available: https://doi.org/10.1504/IJAACS.2017.088774

[56] N. F. Saraiva de Sousa, D. A. Lachos Perez, R. V. Rosa, M. A. Santos, and C. Esteve Rothenberg, "Network service orchestration: A survey," *Computer Communications*, vol. 142-143, pp. 69–94, 2019. [Online]. Available: https://doi.org/10.1016/j.comcom.2019.04.008

[57] B. Mastej, "Exploiting of mobile compute capabilities of mobile network terminals," 2024, promotor: dr inż. Sławomir Kukliński.

[58] S. Garnier, J. Gautrais, and G. Theraulaz, "The biological principles of swarm intelligence," *Swarm Intelligence*, vol. 1, no. 1, pp. 3–31, Jun 2007. [Online]. Available: https://doi.org/10.1007/s11721-007-0004-y

[59] 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, Jan 2010. [Online]. Available: https://doi.org/10.1016/j.tree.2009.06.016

[60] L. O'Bryan, M. Beier, and E. Salas, "How approaches to animal swarm intelligence can improve the study of collective intelligence in human teams," *Journal of Intelligence*, vol. 8, no. 1, 2020. [Online]. Available: HTTPS://DOI.ORG/10.3390/jintelligence8010009

[61] V. Trianni, E. Tuci, K. M. Passino, and J. A. R. Marshall, "Swarm cognition: an interdisciplinary approach to the study of self-organising biological collectives," *Swarm Intelligence*, vol. 5, no. 1, pp. 3–18, Mar 2011. [Online]. Available: https://doi.org/10.1007/s11721-010-0050-8

[62] E. Sahin and A. Winfield, "Special issue on swarm robotics," *Swarm Intelligence*, vol. 2, pp. 69–72, 12 2008. [Online]. Available: https://doi.org/10.1007/s11721-008-0020-6
[63] J. Boskovic, R. Prasanth, and R. Mehra, "A multilayer control architecture for unmanned aerial vehicles," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, vol. 3, 2002, pp. 1825–1830 vol.3. [Online]. Available: https://doi.org/10.1109/ACC.2002.1023832
[64] H. Hamann, *Swarm Robotics: A Formal Approach*, 01 2018. [Online]. Available: https://doi.org/10.1007/978-3-319-74528-2
[65] Y. Jia and L. Wang, "Leader–follower flocking of multiple robotic fish," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1372–1383, 2015. [Online]. Available: https://doi.org/10.1109/TMECH.2014.2337375
[66] N. Correll and A. Martinoli, "Modeling and designing self-organized aggregation in a swarm of miniature robots," *The International Journal of Robotics Research*, vol. 30, no. 5, pp. 615–626, 2011. [Online]. Available: https://doi.org/10.1177/0278364911403017
[67] R. Beckers, O. Holland, and J.-L. Deneubourg, "Fom local actions to global tasks: Stigmergy and collective robotics," *Proceedings of the Workshop on Artificial Life*, vol. 4, 01 1994. [Online]. Available: https://doi.org/10.1007/978-94-010-0870-9_63
[68] R. Jeanson, C. Rivault, J.-L. Deneubourg, S. Blanco, R. Fournier, C. Jost, and G. Theraulaz, "Self-organized aggregation in cockroaches," *Animal Behaviour*, vol. 69, no. 1, pp. 169–180, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003347204002428
[69] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Phys. Rev. Lett.*, vol. 75, pp. 1226–1229, Aug 1995. [Online]. Available: https://doi.org/10.1103/PhysRevLett.75.1226
[70] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 500–505. [Online]. Available: https://doi.org/10.1109/ROBOT.1985.1087247
[71] M. BRAMBILLA, "Formal methods for the design and analysis of robot swarms," PhD thesis, Université Libre de Bruxelles & Ecole Polytechnique de Bruxelles, 2014, available at https://iridia.ulb.ac.be/~mdorigo/HomePageDorigo/thesis/phd/BrambillaPhDThesis.pdf.
[72] A. Reina, G. Valentini, C. Fernandez-Oto, M. Dorigo, and V. Trianni, "A design pattern for decentralised decision making," *PloS one*, vol. 10, p. e0140950, 10 2015. [Online]. Available: https://doi.org/10.1371/journal.pone.0140950
[73] G. Francesca and M. Birattari, "Automatic Design of Robot Swarms: Achievements and Challenges," vol. 3. [Online]. Available: https://doi.org/10.3389/frobt.2016.00029
[74] A. Ligot, A. Cotorruelo, E. Garone, and M. Birattari, "Toward an empirical practice in offline fully automatic design of robot swarms," *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 6, pp. 1236–1245, 2022. [Online]. Available: https://doi.org/10.1109/TEVC.2022.3144848
[75] S. Lem, *The Invincible*. The MIT Press, 2020.
[76] M.-A. Blais and M. A. Akhloufi, "Reinforcement learning for swarm robotics: An overview of applications, algorithms and simulators," *Cognitive Robotics*, vol. 3, pp. 226–256, 2023. [Online]. Available: https://doi.org/https://doi.org/10.1016/j.cogr.2023.07.004
[77] F. Bahrpeyma and D. Reichelt, "A review of the applications of multi-agent reinforcement learning in smart factories," vol. 9. [Online]. Available: https://doi.org/10.3389/frobt.2022.1027340
[78] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948 vol.4. [Online]. Available: https://doi.org/10.1109/ICNN.1995.488968
[79] M. S. Talamali, A. Saha, J. A. R. Marshall, and A. Reina, "When less is more: Robot swarms adapt better to changes with constrained communication," *Science Robotics*, vol. 6, no. 56, 2021. [Online]. Available: https://doi.org/10.1126/scirobotics.abf1416
[80] C. Blum and R. Groß, *Swarm Intelligence in Optimization and Robotics*. Springer Berlin Heidelberg, 2015, pp. 1291–1309. [Online]. Available: https://doi.org/10.1007/978-3-662-43505-2_66
[81] M. Dorigo, "Optimization, learning and natural algorithms," PhD thesis, Politecnico di Milano, 1992.
[82] A. Chakraborty and A. K. Kar, *Swarm Intelligence: A Review of Algorithms*. Springer International Publishing, 2017, pp. 475–494. [Online]. Available: https://doi.org/10.1007/978-3-319-50920-4_19
[83] T. Stützle and M. Dorigo, *Ant Colony Optimization*. The MIT Press, 01 2004. [Online]. Available: https://doi.org/10.7551/mitpress/1290.001.0001
[84] H. Hamann, *Space-Time Continuous Models of Swarm Robotic Systems*, 01 2010, vol. 9. [Online]. Available: https://doi.org/10.1007/978-3-642-13377-0
[85] K. Lerman, A. Galstyan, A. Martinoli, and A. Ijspeert, "A macroscopic analytical model of collaboration in distributed robotic systems," *Artificial life*, vol. 7, pp. 375–93, 02 2001. [Online]. Available: https://doi.org/10.1162/106454601317297013
[86] M. Mannone, V. Seidita, and A. Chella, "Modeling and designing a robotic swarm: A quantum computing approach," *Swarm and Evolutionary Computation*, vol. 79, p. 101297, 2023. [Online]. Available: https://doi.org/10.1016/j.swevo.2023.101297
[87] V. G. Ivancevic, "Entangled swarm intelligence: Quantum computation for swarm robotics." *Mathematics in Engineering, Science & Aerospace (MESA)*, vol. 7, no. 3, 2016.