drones

Review

A Review of Swarm Robotics in a NutShell

Muhammad Muzamal Shahzad [1,2], Zubair Saeed [3], Asima Akhtar [1], Hammad Munawar [2], Muhammad Haroon Yousaf [1,3,*], Naveed Khan Baloach [1,3] and Fawad Hussain [3]

1 Swarm Robotics Lab (SRL), National Center of Robotics and Automation (NCRA), Taxila 47050, Pakistan; mmuzamil.shahzad@uettaxila.edu.pk (M.M.S.); asima.akhtar@uettaxila.edu.pk (A.A.); naveed.khan@uettaxila.edu.pk (N.K.B.)
2 Department of Avionics Engineering, College of Aeronautical Engineering, National University of Science and Technology (NUST), Islamabad 44000, Pakistan; h.munawar@cae.nust.edu.pk
3 Department of Computer Engineering, University of Engineering and Technology (UET), Taxila 47050, Pakistan; zubair.saeed@students.uettaxila.edu.pk (Z.S.); fawad.hussain@uettaxila.edu.pk (F.H.)
* Correspondence: haroon.yousaf@uettaxila.edu.pk

bygyuftx

Abstract: A swarm of robots is the coordination of multiple robots that can perform a collective task and solve a problem more efficiently than a single robot. Over the last decade, this area of research has received significant interest from scientists due to its large field of applications in military or civil, including area exploration, target search and rescue, security and surveillance, agriculture, air defense, area coverage and real-time monitoring, providing wireless services, and delivery of goods. This research domain of collective behaviour draws inspiration from self-organizing systems in nature, such as honey bees, fish schools, social insects, bird flocks, and other social animals. By replicating the same set of interaction rules observed in these natural swarm systems, robot swarms can be created. The deployment of robot swarm or group of intelligent robots in a real-world scenario that can collectively perform a task or solve a problem is still a substantial research challenge. Swarm robots are differentiated from multi-agent robots by specific qualifying criteria, including the presence of at least three agents and the sharing of relative information such as altitude, position, and velocity among all agents. Each agent should be intelligent and follow the same set of interaction rules over the whole network. Also, the system's stability should not be affected by leaving or disconnecting an agent from a swarm. This survey illustrates swarm systems' basics and draws some projections from its history to its future. It discusses the important features of swarm robots, simulators, real-world applications, and future ideas.

Keywords: swarm intelligence; swarm behaviors; swarm robotics; industrial swarm; swarm robotics applications

Citation: Shahzad, M.M.; Saeed, Z.; Akhtar, A.; Munawar, H.; Yousaf, M.H.; Baloach, N.K.; Hussain, F. A Review of Swarm Robotics in a NutShell. *Drones* **2023**, *7*, 269. https://doi.org/10.3390/drones7040269

Academic Editors: Xiwang Dong, Mou Chen, Xiangke Wang and Fei Gao

Received: 10 March 2023
Revised: 9 April 2023
Accepted: 10 April 2023
Published: 14 April 2023

1. Introduction

A swarm of robots refers to the coordination of multiple individual entities, which traditionally operate without centralized control and instead rely on simple local behaviors to cooperate. Robot technology, particularly Unmanned Aerial Systems (UAS), is becoming more affordable, efficient, and is boosting the transmission capacity of robots as solutions to problems ranging from disaster relief to research mapping. Independent robots can perform tasks that need simple, ready to go solutions and a consistent real time approach. The autonomous robot can be a part of a robot swarm, if it fulfills at least three significant characteristics. These characteristics include the following: the minimum number of individual entities must be three or more, minimal or no human control, and cooperation between these robots based on a simple set of rules as depicted in Figure 1. Swarm robotics include a group of independent robots working collaboratively to complete a shared task without relying on any external infrastructure or a centralized control system/robot. Figure 2 illustrates how the fundamental concept of the swarm may be comprehended. In Figure 2a, the system is a robot swarm which consists of three autonomous agents that cooperate in response to the orders received from a single ground control station. Figure 2b indicates that the system is a sensor network rather than a swarm of robots. Each sensor is neither a robot nor an intelligent agent, and is solely responsible for providing data through readings without the capability of taking any actions. Figure 2c does not depict a robot swarm since a swarm necessitates more than two agents. Despite the robots working together towards a shared objective, each one has its own designated tasks to accomplish, which are directed by a separate operator. Figure 2d depicts a software system comprising multiple agents, which cannot be classified as a robot swarm as the agents are not autonomous robots, despite their collaboration on a shared hardware platform.

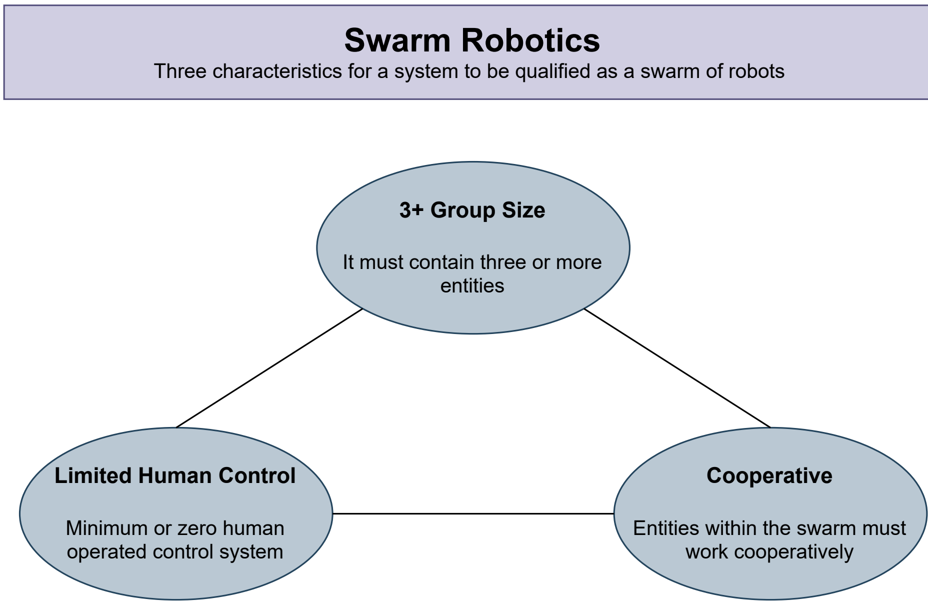

Figure 1. Basic Characteristics of Swarm Robotics.

Swarm robotics involves a group of robots that collaborate to address problems through the development of advantageous structures and behaviours that resemble those observed in nature, such as birds, fish, and bees. These robots, which can be either homogeneous or heterogeneous, form an intelligent network of a swarm, enabling individual robots to interact autonomously with each other and their environment by leveraging onboard communication, processing, and sensing capabilities. Such behaviours can be classified into four categories, namely *navigation* , *spatial organization*, *intelligent and precise decision-making*, and *miscellaneous* [1]. This study offers an in-depth analysis and mathematical comprehension of swarm intelligence algorithms. It also provides a comprehensive review of the evolution of swarm robotics from its inception to the present day and highlights the future ambitions of this field. Our aim is to present a broad overview of swarm robotics by exploring its history, current research, and future directions. The main contributions are as follows:

- To understand the fundamental difference between multi-agent and swarm of robots, along with the natural behaviours of a swarm.
- Multiple swarm intelligence algorithms derived from the natural set of rules and constraints for their transformation on multi-agent robots.
- Industrial and academic utilization of swarm robotics keeping in view the history and future perspectives.
- The objective is to address the research gap that exists between theoretical and industrial research in the field of swarm robotics. Theoretical research mainly involves simulating swarm behaviours using algorithms, while research in industrial settings are primarily focused on designing and developing hardware capable of executing

swarm behaviour. Therefore, it is imperative to deploy swarm algorithms using specific hardware that can accommodate swarm behaviour functionality.

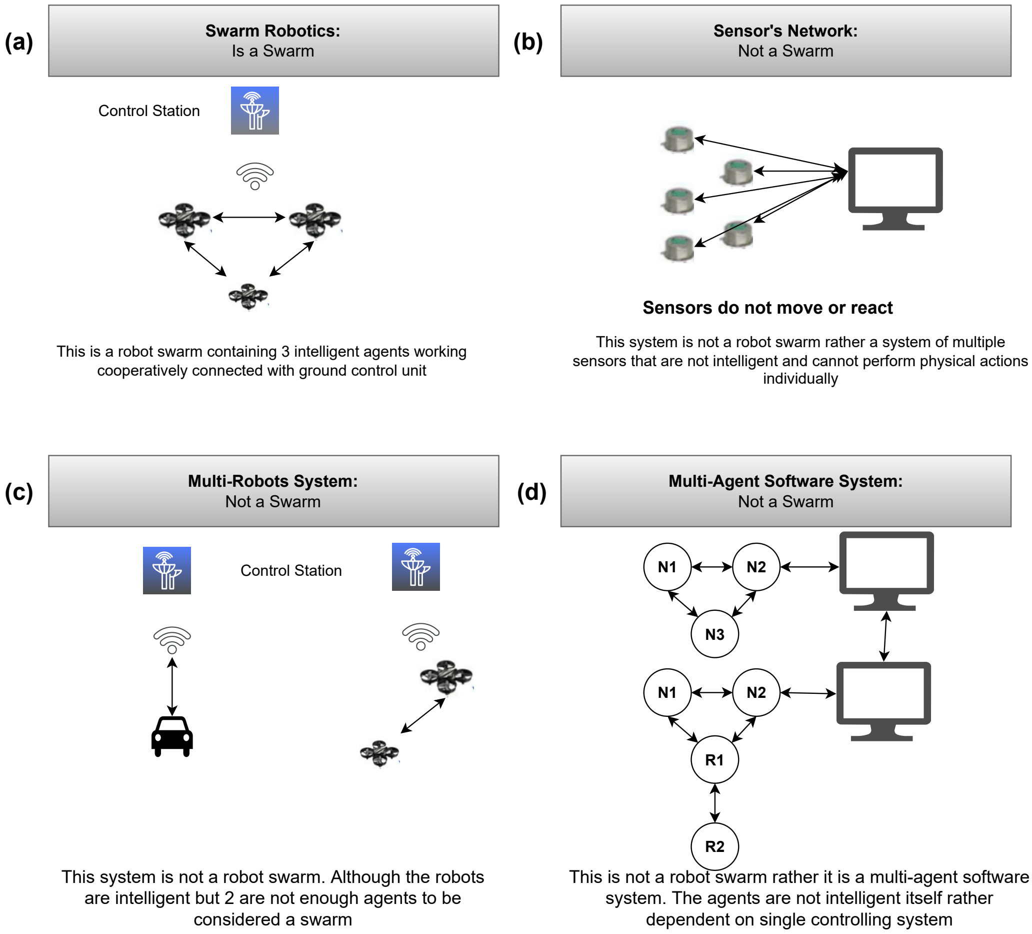

Figure 2. Comparison of Multi-agent Systems and Robot Swarm [2]. (**a**) depicts swarm robotics system, while (**b**–**d**) show non-swarm systems.

Figure 3 depicts the deployment of swarm behaviours in simulation and hardware, which is thoroughly explored in Section 2 of this article. The behaviours are simulated using existing and state-of-the-art swarm intelligence algorithms, as explained in Section 3 with mathematical reasoning. The simulation results demonstrate high accuracy in replicating natural animal behaviours. For the past two decades, the main research challenge in swarm robotics has been to develop multi-robot systems that are robust, flexible, fault-tolerant, and capable of incorporating self-organizing behaviours dynamically and by design. The swarm robotics field has evolved from algorithmic studies to mature academic, laboratory, and industrial-based solutions since the early 2000s. A comprehensive review of swarm robotics and its applications is presented in Table 1 and Section 4, respectively. Despite significant progress, cooperation and coordination in deploying the developed swarming algorithms among swarm robots remain limited [3]. Section 5 provides a brief overview of the era of swarm robotics, and the article concludes in the final section.

Table 1. Era of Swarm Robotics: Past, Present, and Future Perspectives.

1990–2000	The first robot tests show self-organization through indirect and local interactions, clearly inspired by swarm intelligence.	SW
2000–2005	The ability to generate swarms of robots that work together has now been expanded to a variety of additional tasks, including object handling, task allocation, and occupations that require significant teamwork to achieve.	SW
2002–2006	Swarm-bots is a project that shows how robot swarms self-assemble. Robots can construct pulling chains and massive constructions capable of transporting large loads and dealing with tough terrain.	HW and SW
2004–2008	The evolving swarm robotics technique was devised after the first demonstrations of autonomous assembly of robot swarms using evolutionary algorithms.	SW
2005–2009	For swarm robotics research, the first attempts at building standard swarm robotics platforms and small robots.	HW
2006–2010	Swarmanoid showed heterogeneous robot swarms made up of three different types of robots: flying, climbing, and ground-based robots for the first time.	HW and SW
2010–2015	Advanced autonomous design methods such as AutoMoDe, novelty search, design patterns, mean-field models, and optimal stochastic approaches are all employed in the creation of robot swarms.	SW
2016–2020	Decentralized solutions have been investigated and deployed as swarms of flying drones become available for investigation.	HW and SW
2020–2025	The first example of robot swarms that may self-learn suitable swarm behaviour in response to a specific set of challenges.	SW
2025–2030	Marine and deep-sea robotic swarms will be utilized for ecological monitoring, surveillance, and fishing,among other things.	HW
2030–2040	Small rover swarms will be utilized for the first mission to the Moon and Mars to expand the exploration area and showcase on-site construction capabilities.	HW
2040–2045	Soft-bodied robot swarms measuring in millimeters will be deployed to explore agricultural fields and aquatic areas to identify plastic usage and assist with pest control.	HW and SW
2035–2050	Clinical research with human volunteers will begin after nanoscale robot swarms have been shown for therapeutic objectives such as customized medication delivery.	HW and SW

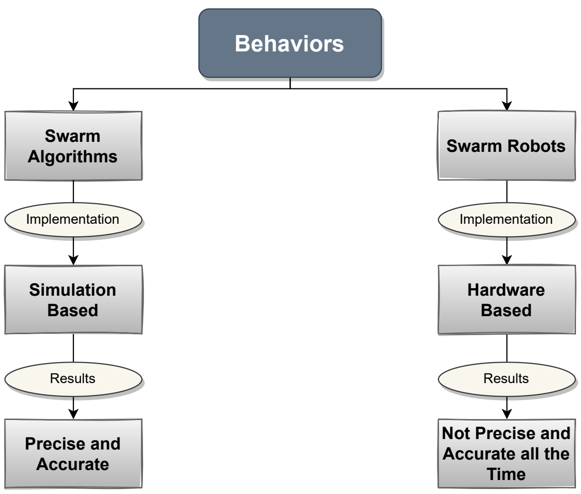

Figure 3. Swarming Behaviours' Deployment in Simulation and Hardware.

2. Swarm Robotics Fundamental Behaviours

Swarm algorithms are characterized by individual entities following local rules, resulting in the emergence of overall behaviour through swarm interactions. In swarm robotics, robots exhibit local behaviours based on a set of rules ranging from basic reactive mapping to complex local algorithms. These behaviours often involve interactions with the physical environment, such as other robots and surroundings [4]. The interaction process involves retrieving environmental values and subsequently processing them to drive the actuators in accordance with a set of instructions. This recurring process is referred to as the fundamental activity and persists until the desired state is attained. Figure 4 illustrates a summary of several naturally occurring behaviours that are further elaborated in the subsequent subsection.

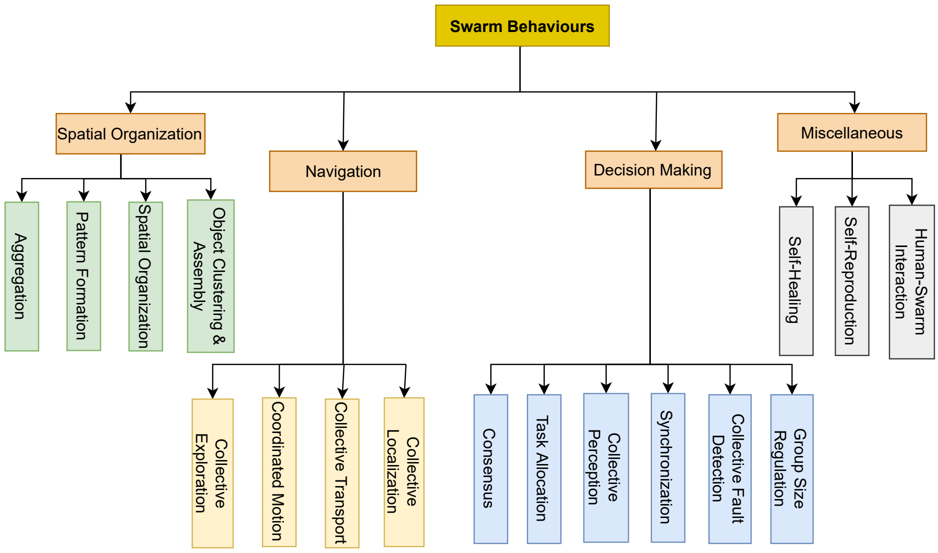

Figure 4. Swarm Behaviours [1].

2.1. Spatial Organization

These behaviours allow robots in a swarm to move around the environment and spatially arrange themselves around things.

Object Clustering and Assembly allow a swarm of robots to control geographically dispersed things. These are critical for construction processes. *Pattern Formation* organizes the robot swarm into a precise form. *Chain Formation* is a specific instance where robots construct a line to establish multi-hop communication between two places [1,5]. *Self-assembly* links robots to form structures. They can be connected physically or remotely via communication lines [1,6]. *Morphogenesis* is a specific instance in which the swarm grows into a predetermined form [1,7,8]. *Aggregation* pushes the individual robots to gather spatially in a certain location of the environment. This permits swarm members to get geographically near to one another for further interaction [1,9,10].

2.2. Navigation

These characteristics enable a swarm of numerous robots in the environment to move in unison. Thus, a group of robots move in harmony from one location to another or from a source to a final destination [1,11].

Collective Localization allows the swarm's robots to determine their location and orientation relative to one another by establishing a local coordinate system across the swarm [1]. In *Collective Transport*, a swarm of robots may collectively move things that are too heavy or massive for individual robots [1]. *Coordinated Motion* moves the swarm in a configuration that must have a well-defined shape or structure, such as a line, triangle, or arbitrary formation of robots, as in flocking [1]. *Collective Exploration* navigates the environment to examine things, monitor the environment, or create a robot-to-robot communication network [1,12].

2.3. Decision Making

This characteristic of swarm robotics facilitates collective decision-making for accomplishing specific tasks collaboratively. The *Group Size Regulation* feature empowers the swarm's robots to create groups of the required size, and if the swarm's size exceeds the required group size, it automatically divides into multiple groups or sub-swarms [1,13]. Additionally, the *Collective Fault Detection* feature detects individual robot shortcomings inside the swarm, enabling the identification of robots that deviate from the expected behaviour due to hardware or some algorithmic issues [1,14]. Furthermore, *Synchronization* aligns the frequency and phase of the swarm's oscillators, enabling the robots to share a common perception of time and execute tasks in synchrony. The *Collective Perception* feature aggregates the locally collected data from the swarm's robots into a comprehensive image. It allows the swarm to make collective decisions, such as accurately classifying objects, allocating a suitable percentage of robots to a given task, or determining the best solution to a global problem [1]. Moreover, the *Task Allocation* feature dynamically assigns emergent tasks to individual robots, aiming to maximize the overall performance of the swarm system. In cases where the robots possess diverse skill sets, the work can be assigned differently to further enhance the system's performance [1,15]. Finally, the *Consensus* feature allows the swarm of robots to converge on a single common point from multiple available options [1,16].

2.4. Miscellaneous

The swarm robots exhibit additional behaviours beyond the previously discussed categories. *Self-healing* behaviour allows the swarm to recover from individual robot failures, improving the swarm's reliability, resilience, and overall performance [1,17]. *Self-reproduction* enables a swarm of robots to add new robots/agents or replicate the patterns created by several individuals, thereby increasing the swarm's autonomy by eliminating the need for human intervention in the construction of additional robots. *Human-swarm Interaction* facilitates communication between humans and the swarm of robots, either remotely via a computer terminal or in a shared area using visual or auditory cues [1].

3. Swarm Intelligence Algorithms

Swarm Intelligence (SI) is a collective intelligence employed in various applications, including self-organized and decentralized systems [18]. Some examples are collective sorting, cooperative transportation, group foraging, and clustering. Self-organization and division of work are two essential notions in SI. The ability of robots to evolve into a proper pattern without external assistance is referred as *self-organization*. In contrast, division of labor refers to the simultaneous execution of multiple tasks by individual robots. It enables the swarm to execute a challenging task that requires individuals to collaborate. Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particles Swarm Optimization (PSO), Differential Evolution (DE), Artificial Bee Colony (ABC), Glowworm Swarm Optimization (GSO), and Cuckoo Search Algorithm (CSA), are all examples of famous and currently used swarm intelligence algorithms.

3.1. Genetic Algorithm

Genetic Algorithms (GA) were introduced in 1975 by John Holland [19,20]. This type of algorithm mimics natural existing biological behaviours in order to evaluate the survival of the fittest. In a genetic algorithm (GA), a specified number of individuals, also known as members, comprise the population. Mathematical operators such as crossover, reproduction, and mutation are used to manipulate the genetic makeup of individuals. Based on these operators, the fitness value of each member is calculated and ranked accordingly. The previous population's traits, represented by chromosomes (or strings), are combined with new traits to generate a new population [21–24]. A GA algorithm with five basic steps is shown in Figure 5. The fitness function evaluates population members, which begins with an initial population that can be generated randomly or through a heuristic search. After the population members are assessed, the lowest-ranked chromosome is eliminated, and the remaining members are used for reproduction. The final step is mutation, in which the mutation operator modifies genes on a chromosome to ensure that every part of the problem space is explored. This process of evaluating and generating new populations continues until the best solution is found.

It has a vast area as an application, which includes, navigation and formation control [25], path planning [26], scheduling [27,28], machine learning [29], robotics [30,31], signal processing [32], business [33], mathematics [34], manufacturing [35] and routing [36].

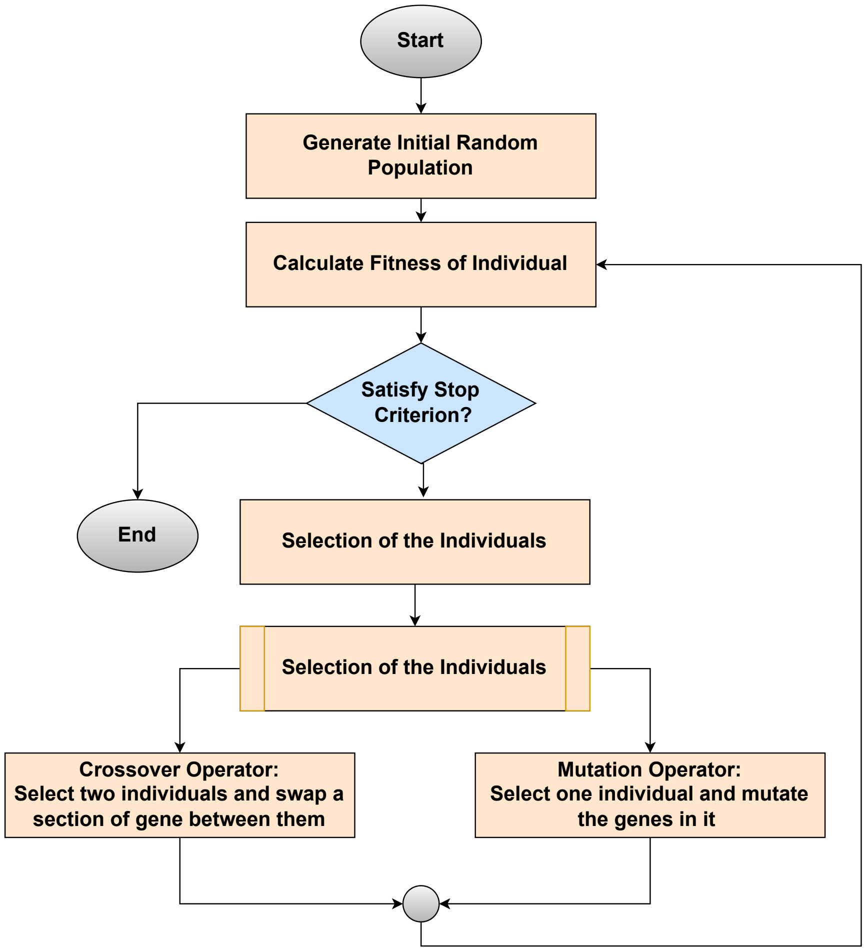

Figure 5. Flow Chart of Genetic Algorithm [37] .

3.2. Ant Colony Optimization

Ant Colony Optimization (ACO) is a heuristic search-based algorithm that uses the ant colony system to solve problems. It was proposed by Marco Dorigo as part of his Ph.D. study in 1992 [38]. The four fundamental components of the ant-inspired foraging algorithm are the ant, pheromone, daemon action, and decentralized control. The *ant* acts as an imaginary agent which mimics the behaviour of exploitation and exploration processes in a search space and produces a chemical substance called *pheromones*. Its intensity varies with the passage of time due to the evaporation process and serves as a global memory for the ant's path of travel. *Daemon activity* is used to gather global data whereas, the decentralized control is used for the robustness of the ACO algorithm and to maintain flexibility within a dynamic environment. The Figure 6a–c show the initial, mid-range, and final outcomes of the ACO algorithm, respectively [38,39]. Figure 6a shows the initial random environment in which the agent (or ant) from the nest begins the process. When ants discover numerous viable paths from the nest to the source, they go through many iterations of execution, as shown in Figure 6b. The ant has chosen the shortest possible path, which contributes to the pheromone trail's high intensity. Equation (1) below is used as an initial step in determining the optimal solution to select the best node from the current search space.

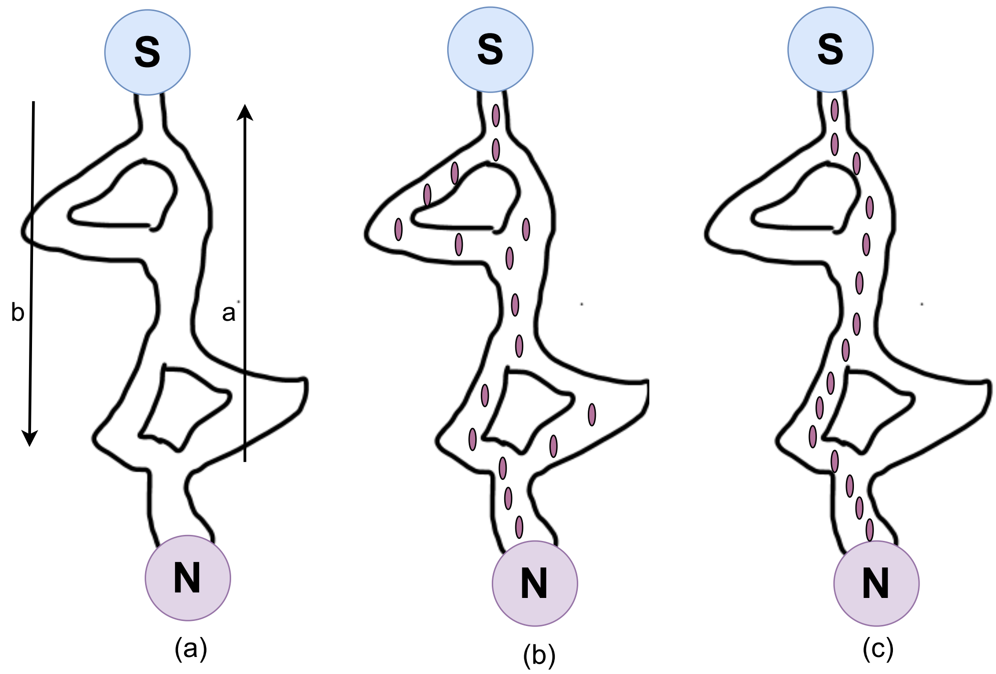

Figure 6. Nest and Food-Source have been shown by letters N and S, respectively. (**a**) depicts the early stages of the process, in which ants start to discover a passage between the nest and the source and lay their pheromones. (**b**) depicts the intermediate phase, in which the ants took all available pathways. (**c**) demonstrates that the majority of ants chose the road with the highest pheromone concentration [36] .

$$p^{u}_{(n,m)}(t_o) = \frac{\left([\tau_{nm}(t_o)]^{\alpha} \cdot [\eta_{nm}]^{\beta}\right)}{\left(\sum_{u \in I_u} [\tau_{nm}(t_o)]^{\alpha} \cdot [\eta_{nm}]^{\beta}\right)} \tag{1}$$

The probability of travelling from node n to node m is $p_{(n,m)}$, I_u are the nodes to which the ant is permitted to go from node n, whereas $\eta_(nm)$ adds to visibility between nodes n and m and it indicates the quantity of un-evaporated pheromone between nodes at a time t_o. α and β in Equation (1) regulate the impact of $\tau_{nm}(t_o)$ and η_{nm}, where, if α is larger, the ant's searching behaviour is more pheromone-dependent, and if β is higher, then the ant's searching depends on its visibility or knowledge.

In order to deposit a pheromone, the following equation is used:

$$\Delta\tau^{u}_{nm}(t) = \begin{cases} \frac{Q}{L_u}(t) \\ 0 \end{cases} \tag{2}$$

Q is a constant, L is the cost of the ant's tour that represents the length of the created path, t is the iteration number and u shows a specific ant. Another key factor is pheromone evaporation rate, which shows exploration and exploitation behaviour of the ant. In Equation (3), s is the number of ants in the system and p is the pheromone evaporation rate or decay factor.

$$\tau_{nm}(t+1) = (1-p).\tau_{(n,m)(t)} + \sum_{k=1}^{s}[\Delta\tau_{n,m}^{u}(t)] \tag{3}$$

Compared to other heuristic-based approaches, ACO guarantees to converge, but the time required for it is uncertain and for better performance, the search space should be small [40,41]. Its applications include vehicle routing [42,43], network modelling problem [44,45], machine learning [46], path planning robots [47], path planning for Unmanned Aerial Vehicles (UAVs) [48], project management [49] and so on.

3.3. Particle Swarm Optimization

Kennedy and Eberhart invented Particle Swarm Optimization (PSO) in 1995, and it uses a simple method to encourage particles to explore optimal solutions [50]. It is based on flocking bird and schooling fish behaviours [51], by exhibiting three simple behaviours: separation, alignment, and cohesiveness. *Separation* is used to avoid congested local flock-mates, *alignment* is the travelling of one flock-mate in the same average direction of the other flock-mates, and *cohesiveness* is the movement of flock-mates toward the average position. The PSO algorithm is as follows [50,52,53]:

$$\begin{gathered} v_{id}^{t+1} = v_{id}^{t} + c_1 \cdot \mathrm{rand}(0,1) \cdot (p_{id}^{t} - x_{id}^{t}) + c_2 \cdot \mathrm{rand}(0,1) \cdot \left(p_{gd}^{t} - x_{id}^{t}\right) \\ x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \end{gathered} \tag{4}$$

where v_{id}^{t} and x_{id}^{t} are particle velocity and position, whereas d is search space dimension, i represents particle index and t shows the iteration number. c_1 and c_2 depict the speed and regulating length of the swarm when it travels towards the optimal particle position. The optimal position attained by particle i is p_i and the best position found by neighbouring particles of i is p_g. The process of exploration ensues if either or both of the differences between the best of particle p_{id}^{t} and the previous position of particle x_{id}^{t} and between the population all-time best p_{gd}^{t} and the previous particle's position x_{id}^{t} are large. Similarly, the process of exploitation happens when both of these values are small. PSO has been demonstrated as an effective, robust, and stochastic optimization algorithm for high-dimensional spaces. The key parameters of PSO include the position of the agent in space, the number of particles, velocity, and the agent's neighbourhood [54–56].

The PSO algorithm begins by initializing the population, and the second step is to calculate the fitness of each particle. Whereas, the third step is followed by updating the individual and global best. In the fourth step velocity and neighbourhood of the particles are updated. Steps two to four keep repeating until the terminating condition is satisfied [51,54,57,58].

Figure 7 shows the working of the PSO method, where the particles are spread out in the first iteration to discover the best exploration. The best solution is identified in terms of neighbourhood topology, and each member's personal and global best particles are updated. As indicated in the figure, the convergence would be determined by attracting all particles towards the particle with the best solution.

PSO is simple to configure for efficient global search, has few parameters to set, is scale-insensitive, and parallelism for concurrent processing is also easy. Population size is one of the key factors that ensures precise and fast convergence for large population sizes [51,59]. Networking [60], power systems [61], signal processing [62], control systems [63], machine learning [64], and image processing [65–67] are some of the applications.

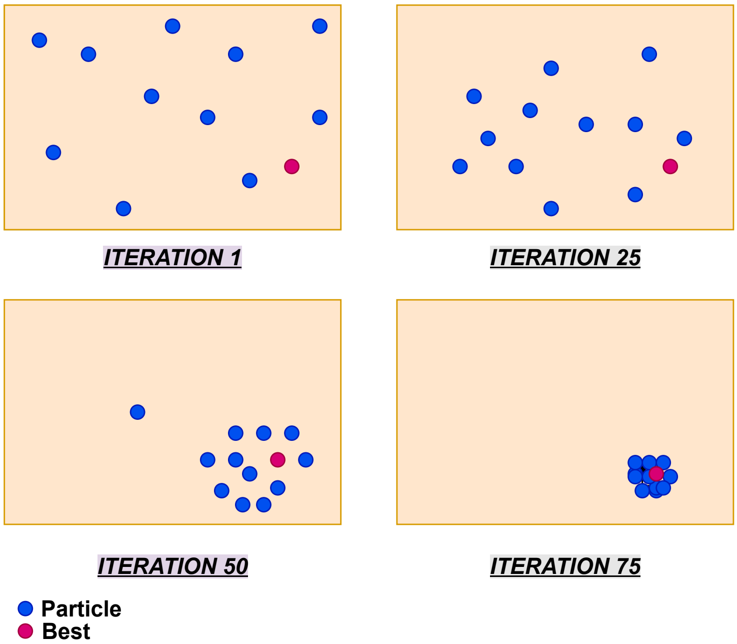

Figure 7. The operation of the PSO algorithm and its progress towards global optima as measured by iteration numbers [47].

3.4. Differential Evolution

Differential Evolution (DE) is similar to GA, using the same crossover, mutation, and selection operators. The fundamental difference between the two algorithms is that the DE utilizes the mutation operator while GA uses the crossover operator to produce a superior solution. Price and Storn first introduced it in 1997 [68]. DE repeatedly generated new populations using three properties: mutant vector, target vector, and trail vector explained in Figure 8. A crossover process between the target and mutant vectors produces the trailing vector. The mutant vector represents the mutation of the target vector, whereas the target vector represents the vector holding the search space solution [69,70]. The DE algorithm starts with population initialization and then evaluates the population to find the fittest members. The weighted difference between the two population vectors is added to the third vector to create new parameter vectors and this process is known as *mutation*. The vector is blended within the crossover to perform a final selection.

N parameter vector mutation is generated by using the following equation:

$$v_{j,N+1} = x_{l1,N} + F(x_{l2,N} - x_{l3,N}). \tag{5}$$

i shows the index of the 2D vector. x_{l1}, x_{l2}, and x_{l3}, are solution vectors selected randomly and the values of $l1$, $l2$ $l3$ and i should not be equal to each other. F is the scaling factor $\in$ [0,1], while, a crossover procedure is employed to improve the variety of the disconcerted parameter vectors. The parent and mutant vectors are combined in the following method to create a trial vector:

$$u_{i,G+1} = \begin{cases} v_{i,G+1} \text{ if } R_j \leq CR \\ x_{i,G} \text{ if } R_j > CR \end{cases} \tag{6}$$

where CR denotes the crossover constant. R_j denotes a random real number ϵ [0,1] while j depicts the resultant array's j^{th} component.

The primary distinction between DE and GA operations is that in DE, the probability of being selected as a parent is not based on fitness value. Increasing the population size can significantly improve DE performance.

DE can be found in a variety of fields, including, robot path planning [71,72] engineering [73], image processing domain [74], machine learning [75], and economics [76].

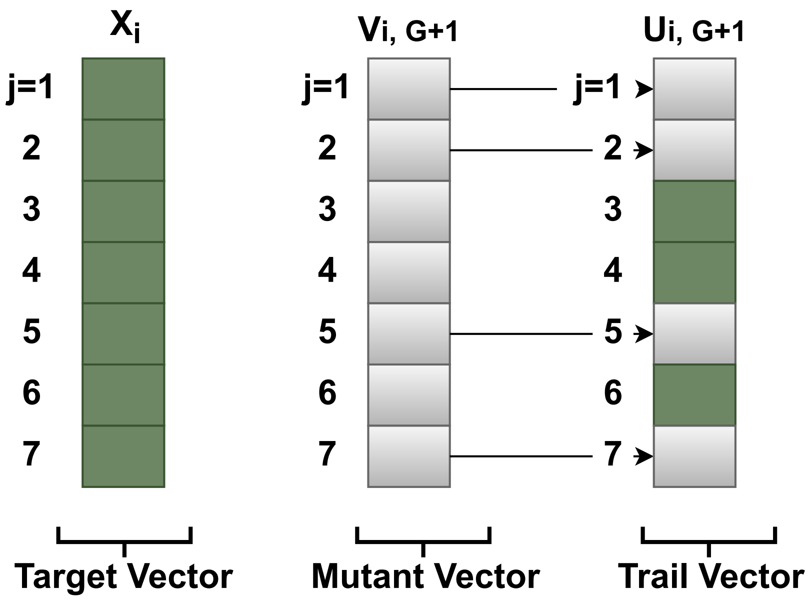

Figure 8. Demonstration of DE with a seven-vector dimension j. A target vector is a current approach; however, a mutant vector is also an alternative. After the crossover operation, the trailing vector is a new solution [55].

3.5. Artificial Bee Colony

Dervis Karaboga presented Artificial Bee Colony (ABC) as an important SI algorithm in 2005 [77]. Its performance is thoroughly examined in [78], which concluded that ABC outperforms other techniques. It is based on honey bees' intelligent behaviour in locating food and communicating information about that food with other bees. ABC is as straightforward as PSO, and DE [78], which divides artificial agents into three types: *employed*, *observer*, and *scout* bees. Each agent bee is given a particular task to finish the algorithm process. The employed bee concentrates and memorizes the food supply. The employed bee provides the observer bee with the information about the hive's food supply. The scout bee is on the lookout for new nectar and its sources. Figure 9 presents the algorithmic flow of the ABC. The ABC method's overall procedure and specifications of each step are explained below [77–79]:

Step 1. Initialization: Food sources, x_i, are initialized with $i = 1 \ldots N$, where N is the number of scout bees in the population. l_i and u_i are the control parameters represent lower and upper limits, respectively. The following Equation (7) represents the initialization phase:

$$x_i = l_i + \text{rand}(0, 1) * (u_i - l_i) \tag{7}$$

Step 2. Employed Bees: The search capacity for finding new neighbour food source v_i increases to accumulate more nectar around the neighbour food source x_i. Once they identify a nearby new food source supply, its profitability and fitness value are assessed. The following formula is used to define the new nearby food source:

$$v_i = x_i + \phi_i(x_i - x_j) \tag{8}$$

where x_j is a randomly selected food source. ϕ_i has random numbers of range between [−a, a]. After the profitability of the new source v_i is determined, a greedy selection is used between $\overrightarrow{x_i}$ and $\overrightarrow{v_i}$. The process of exploration occurs if $x_i - x_j$ is greater, otherwise exploitation happens. The fitness value $fit_i(\overrightarrow{x_i})$ is computed by the following Equation (9) and objective function with solution value x_i is $f_i(\overrightarrow{x_i})$.

$$fit_i(\overrightarrow{x_i}) = \begin{cases} \frac{1}{1+f_i(\overrightarrow{x_i})} & \text{if } f_i(\overrightarrow{x_i}) \geq 0 \\ 1 + abs(f_i(\overrightarrow{x_i})) & \text{if } f_i(\overrightarrow{x_i}) < 0 \end{cases} \tag{9}$$

Step 3. Onlooker Bees: After calculating the fitness value and by obtaining information from employed bees, a probability value p_i is computed by using Equation (10), and this value is then shared with the waiting bees in the hives for selecting food sources. These bees are known as onlooker bees.

$$p_i = \frac{fit_i(\overrightarrow{x_i})}{\sum_{i=1}^{SN} fit_i(\overrightarrow{x_i})} \tag{10}$$

Step 4. Scout Bees: Employed bees that cannot raise their fitness values after multiple repetitions become scout bees. These unemployed bees choose sources at random.

Step 5. Best Fitness: The best fitness value and the exact position with an associated value are memorized.

Step 6. Termination Checking Phase: The program terminates upon meeting the termination condition. If the termination condition may not be reached, the program goes back to step 2 and repeats the process until it is.

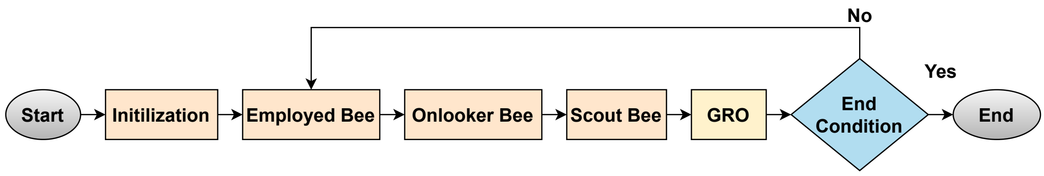

Figure 9. Flow Chat of ABC Algorithm.

Since ABC has only two control factors, colony size and maximum cycle number, it is straightforward to set up, robust and customize-able. It is also possible to add and remove bees without re-initializing the algorithm [80,81]. The disadvantage of ABC is that additional fitness tests for new parameters are required to increase the algorithm's overall performance. It is also slow when a large number of objective function evaluations are required [82]. Path planning for multi-UAVs [83], engineering design difficulties [84,85], networking [86], electronics [87], scheduling [87], and image processing [87] are some of the disciplines where it is used.

3.6. Glowworm Swarm Optimization

Glowworm Swarm Optimization (GSO) is a new SI based approach presented by Krishnanad and Ghose in 2005 [88,89] to optimize multimodal functions. In GSO, glowworms are real-life tangible creatures. There are three key parameters in a glowworm m condition at time t: a search space position $x_m(t)$, a luciferin level $l_m(t)$, and a neighbourhood range $r_m(t)$ [88–90]. These variables change over time, whereas the glowworms are distributed throughout the work area at random initially, and then the other settings are set using predetermined constants. It is similar to earlier algorithms, where three phases are continued until the termination condition is reached. The three steps of [88] are luciferin level update, glowworm migration, and neighbourhood range update. The fitness value of glowworm m's current position of luciferin level is updated by using the following equation:

$$l_m(t) = (1-p) \cdot l_m(t-1) + \gamma J(x_m(t)) \tag{11}$$

where p is the luciferin evaporation factor and J represents the objective function. For position update in the search space, the following equation is used:

$$x_m(t) = x_m(t-1) + s\frac{(x_n(t-1) - x_m(t-1)}{||(x_n(t-1) - x_m(t-1)||} \tag{12}$$

where s is the step size, and | | . | | is euclidean norm operator. Exploration and exploitation behaviours occur on the basis of x_n and x_m difference. Greater difference leads to exploration and smaller to exploitation behaviour.

If a glowworm has several neighbours to choose from, one is selected using the following probability equation and the glowworm m is the neighbour of glowworm n only if the distance between them is shorter than the neighbourhood range $r_m(t)$:

$$p_\mathrm{m}(\mathrm{t}) = \frac{l_\mathrm{m}(\mathrm{t}) - l_\mathrm{n}(\mathrm{t})}{\sum_{\mathrm{k}\in \mathrm{Ni(t)}} l_\mathrm{k}(\mathrm{t}) - l_\mathrm{n}(\mathrm{t})} \tag{13}$$

The following equation is used to compute the neighbourhood range:

$$r_m(t+1) = \min\{r_s, \max[0, r_m(t) + \beta(n_d - |n_m(t)|)]\} \tag{14}$$

r_s represents sensor range, n_d is the desired number of neighbours, $|n_m(t)|$ is several neighbours of the glowworm m at time t, and β is a model constant. The diagram below demonstrates two hypothetical scenarios in which agents developing methods result in distinct behaviours depending on the agents' placement in the search space and the accessible nearby agents. The glowworm's agents are represented by i, j, and k. Figure 10a signifies agent j's sensor range, whereas r_d^j denotes agent j's local-decision range. The same is true for i and k, where r_s^i and r_d^i, r_s^k and r_d^k respectively denote sensor range and local-decision range. It is applied in the circumstances where agent i is in the sensor range of agent j and k. Only agent j uses the input from agent i because the agents have different local decision domains. Glowworm agents are a, b, c, d, and e in Figure 10b. The glowworm agents are ranked 1, 2, 3, 4, and 5, depending on their luciferin values.

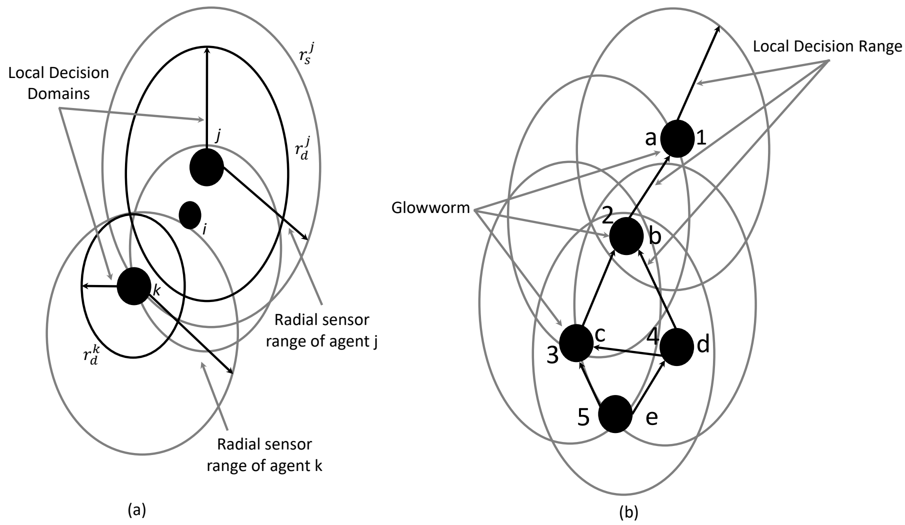

Figure 10. GSO in two different scenarios. The glowworm agents are a, b, c, d, e, f, i, j, and k. Three agents with varied sensor ranges and local-decision ranges are shown in (**a**). It demonstrates how agents gravitate towards agents with higher luciferin values when they are in the same local decision as another agent. Glowworm agent's rating is according to their luciferin levels, as shown in (**b**). Lower numbers indicate greater luciferin values and vice versa [67].

The following modifications can be considered to improve the performance of GSO. (i) To include all agents in the solution, consider increasing the number of neighbourhoods. When the best solution has been identified, all the agents can travel in the direction of the agent which has the best solution, because more agents will be within the optimal solution range and it will also increase the efficiency of exploitation; (ii) In the neighbourhood range, smallest possible number should be selected to increase the convergence rate of GSO. Since there are fewer calculations needed to estimate the probability and direction of the GSO's movement, this action may decrease the GSO's processing time.

GSO is useful in situations when only a small sensor range is required. It can detect many sources and can be used to resolve problems of numerical optimization [88–90]. It is also inaccurate and has a slow convergence rate [91,92]. 3-Dimensional path planning [93], self-organization based clustering scheme for UAVs [94], routing [95], swarm robotics [96], image processing [97], and localization [98,99] difficulties have all been solved using GSO.

3.7. Cuckoo Search Algorithm

Yang and Deb in 2009 proposed Cuckoo Search Algorithm (CSA) as one of the most current meta-heuristic techniques. The behavior of cuckoos, i.e., brood parasites, and the properties of Levy flights [100] inspired this algorithm. Three steps are followed throughout the implementation of this approach. First, in each repetition, each cuckoo lays one egg, and the nest in which the cuckoo lays its egg is chosen at random by the cuckoo. Quality eggs and nests are passed down from generation to generation in the second step. In the third step, the number of possible host nests are fixed, and a host bird uses probability $p_a \epsilon$ [0, 1] to find a cuckoo egg. In other words, the host can either reject the egg or depart the nest and start over. These three major criteria are used to present the specifications of the acts taken in CSA. The following Levy flight equation is used to construct a new solution, $u(i+1)$ [100,101]:

$$u_m(i+1) = u_m(i) + \partial \oplus \mathrm{Levy}(\beta) \tag{15}$$

$$Levy \sim s = t^{-1-\beta}(0 < \beta < 2) \tag{16}$$

The product $\oplus$ is an indication of multiplication, follows the same rules as entry-wise matrix multiplication, and ∂ is the step size and, in most circumstances, $\partial = 1$. The step size begins with a large value and gradually decreases until the last generation, allowing the population to converge on a solution, similar to the processes involved in reducing PSO linearly. Yang [102] introduces the additional component as follows:

$$u_m(i+1) = u_m(i) + \partial \oplus \mathrm{Levy}(\beta) \sim 0.01 \frac{s}{|v|^{1/\beta}}(u_n(i) - u_m(i)) \tag{17}$$

where s and v are selected using the normal distribution, which is defined as follows:

$$s \sim N\left(0, \sigma_s^2\right), v \sim N\left(0, \sigma_s^2\right) \tag{18}$$

where;

$$\sigma_u = \left\{ \frac{\left(\gamma(1+\beta)\sin\left(\frac{\pi\beta}{2}\right)\right)}{(\gamma[1+\beta)/2]\beta 2^{\frac{\beta-1}{2}}} \right\}^{1/\beta}, \sigma_v = 1 \tag{19}$$

γ is the standard gamma function [102]. Exploration happens when the difference between u_n and u_m is high, while exploitation occurs when the difference is minor.

Compared to other approaches, CSA offers the advantage of multi-model objectives and requires fewer parameters to fine-tune them. It is used in a variety of settings, including path planning for UAVs [103], neural networks [104], embedded systems [105], electromagnetics [106], economics [107], business [108], and the Traveling Salesman Problem (TSP) issue [109].

4. Applications of Swarm Robotics

Swarm robotics is an emerging area of research and development that has yet to gain significant industrial adoption. Still, academics have created a variety of platforms to test and analyze the algorithm. In [110], the authors mentioned that they are researching for future industrial platforms. Swarm robotics research (see Figure 11), and industrial efforts & products (see Figure 12), are the two areas of the survey which will be discussed later. Industrial projects and products are examples of deployment in a real-time scenario. The swarm robotics research platform assists researchers in demonstrating, verifying, and experimenting with swarming algorithms in a laboratory setting. The four categories for both platforms are terrestrial, aerial, aquatic, and extraterrestrial. Robotic vehicles include Unmanned Submarine Vehicles (UUV), Unmanned Aerial Vehicles (UAVs), Unmanned Surface Vehicles (USVs), and Unmanned Ground Vehicles (UGVs).

4.1. Research Platforms

This section includes the application from swarm algorithms to swarm robots. Advanced robotics research platforms, such as the *balboa* robot and others, exist but are not included in Figure 11 because, they are not designed to use in swarm applications.

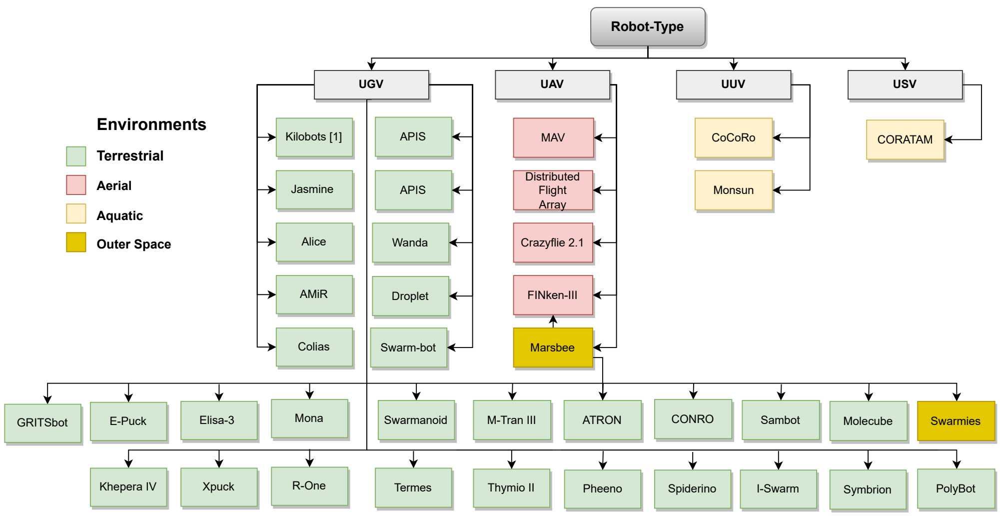

Figure 11. Classification of different research platforms for swarm robotics .

4.1.1. Terrestrial

The *kilobot* swarm is widely considered the best swarm of robots ever produced for educational and research purposes. They are little, measuring 33 mm in diameter. For propulsion, vibration motors are employed, and for communication, infrared light reflected from the ground is used. For swarming, 1024 robots are used and they are well-known for their capacity to self-assemble into various forms [111]. It is open-source and commercially accessible through K-Teams. *Jasmin*, an open-source platform, was created with a large-scale swarm investigation that required touch, proximity, distance, and color sensors. *Alice* [112] is another platform, with additional sensors, including a linear camera, increases the functionalities of swarming. Similarly, *AMiR* [113] and *Colias* [114] are open-source and commercially available swarm robots that provide a foundation for a number of research platforms. *Mona* is a commercial product as well as an open-source initiative. However, *R-One* may be used as a swarm robotics platform since it comes with a camera for ground-truth localization and software to connect all devices. The swarming platform *Elisa-3* incorporates an Arduino with eight infrared sensors, three accelerometers, and four ground sensors, all of which can be charged by a charging station and communicate through infrared or radio waves. The *Khepera IV* [115] was created for indoor use. K-Team is a tiny and unique swarming research platform with a linux core, color camera, WLAN, bluetooth, USB, accelerometer, loudspeakers, gyroscope, three RGB

LEDs, and it is also commercially available. The *GRITSbot* [116] is an open-source robot found at Georgia Tech's Robotarium in Atlanta. Researchers can utilize the resources by uploading code, performing experiments, and gathering data using Robotarium's remote access. As the size and quantity of these robots increase, more maintenance and usability aspects become crucial.

The *e-puck* and its successor, the *e-puck2*, are designed to make programming and controlling robot behavior simple for research and education. It includes an infrared proximity sensor, a CMOS camera, and a microphone. Both commercial and open-source versions are available. Its new edition, *Xpuck*, introduces new features, including aggregation of raw processing power, which is used in current mobile system-on-chip (SoC) devices with roughly two teraflops of processing power.

ArUco marker tracking in image processing computations is another example [117]. Similarly, *Thymio II* [118] swarm robots offer a range of sensors, including temperature, infrared distance, microphone, and accelerometer. Visual and text-based programming are also available. *Thymio II* is open-source and commercially available at Thymio, whereas *Pheeno* [119] is also a free and open-source swarm robotics platform for teaching and research. Custom modules with three degrees of freedom may be employed, and an IR sensor is used to communicate with the outside world. The open-source and locomotion-capable *Spiderino* [120] has six legs and has a hexapod toy-like design with an Arduino CPU, WLAN, and some reflected infrared sensors on a PCB.

I-Swarm (Intelligent Small-World Autonomous Robots for Tiny-Manipulation) is a swarming microrobot. Its sizes are $3 \times 3 \times 3$ mm, and it is solar-powered without a source. It travels by vibrating and communicates using infrared transceivers to establish a swarm of 1000 robots [121]. The prototype is on exhibit at the technology museum in Munich. The *Zooids* [122] human-computer interface is a novel type of HCI that handles interaction and presentation. It was built as a unique open-source robotics platform. Light patterns projected from an overhead projector regulate the swarming of *Zooids*. The APIS, or adaptable platform for an interactive swarm, comprises several components, i.e., the swarm's infrastructure and testing environment, software infrastructure, and simulation [123]. The focus is to experiment with human-swarm interaction. The platform uses an OLED display and a buzzer. With the help of the swarm, clean up the environment Wanda [124] is a robotics platform that might be useful. The authors have built the entire tool-chain from robot design and simulation to deployment. *Droplet* [125], a spherical robot that can organize itself into complex shapes with the help of vibration locomotion, is another ideal platform for education and study. The powered floor, which features alternating positive charge and ground stripes, has been used for both charging and communication between swarm robots. *Swarm-bots* [126,127] may automatically align themselves to various 3D shapes. Its design is open-source, and robots are made up of various insect-like shapes. They are built with low-cost, readily available components. They can adapt to any environment due to their self-assembling and self-organizing capabilities. The swarm can move heavy goods that would be too heavy for individual robots. *Swarmanoid* and its successor, are the first study of integrated design, development, and control of heterogeneous swarm robotics systems. It is open-source, and includes three types of autonomous robots. *Eye-bots* (UAVs that can stick to an interior ceiling), *Hand-bots* (UGVs that can climb), and *Footbots* (UGVs that can self-assemble) are among the varieties of UAVs that are developed [128]. Surprisingly, the *termes* robots [129] interact without the need for communication or GPS to build huge constructions using modular components. It is based on how termites construct their nests in nature, and they are block-carrying climbing robots that can also construct similar structures in unstructured situations. Other swarming platforms for research are *symbrion* and *replicator* [130]. They are two projects that are pretty much identical in terms of developing autonomous platforms for swarms. By physically connecting to each robot in the swarm, they may function individually or in a certain form and the goal was to devise a strategy for achieving robot organism evolvability. *PolyBots* [131] are self-configurable robots that can move in many ways. They have interchangeable object manipulation modules that may take on a variety of shapes depending on the situation, such as an earthworm for slithering over barriers or a spider for marching through hilly terrain. These robots are ideal for multitasking and usage in new areas. *M-TRAN I* [132], *M-TRAN II* [133], and *M-TRAN III* [134] are self-configurable robotics technologies. *ATRON* [135], *CONRO* [136], *sambot* [137], and *molecube* [138] are all open-source robotic systems and robots.

4.1.2. Aerial

Miniature and micro unmanned aerial vehicles (μUAVs) for swarming are affordable robots available for research and education [139] and Swetha et al. [140] both look into small-scale UAVs. Several off-the-shelf *Micro Air Vehicles* (μAVs) are available and famous in the gaming and commercial industries. Three rate gyroscopes and three accelerometers are used in UAVs developed for swarming robots in μAVs in [141], together with one ultrasonic sensor and four IR sensors. The Distributed Flight Array [142] is a popular platform used to construct *swarmanoid* [128] on it. Each UAV adds a single rotor to a big array. The module self-assembles into a multi-rotor system, in which all robots must exchange coordinates and local parameters for coordinated flying. *Crazyflies* [143], which are available commercially and open-source at Bitcraze, make use of a variety of sensors, including a high-precision pressure sensor, an accelerometer, a magnetometer, and a gyroscope. It can conduct experiments while minimizing the risk to humans because of its light weight of about 27 g. In *FINken-III* [144], is a powerful copter equipped with a better communication module (802.15.4) to communicate between ground station and other copters , and sensors like optical flow, infrared distance, and four sonar sensors.

4.1.3. Aquatic

The Collective Cognitive Robotics (CoCoRo) project has been developed with 41 heterogeneous Unmanned Underwater Vehicles (UUVs). Electric fields and sonar sensors are used to communicate, and the system applies to environmental monitoring, water pollution assessment in rivers and oceans, and global warming consequences. The *Monsun* [145] has two communication modes: a camera for identifying other swarm members and an underwater acoustic modem for transmitting data. *CORATAM* (Control of Aquatic Drones for Maritime Tasks) [146] has also been developed for swarms of USVs, with uses such as sea border patrols, marine life localization, and environmental monitoring. This open-source platform uses evolutionary computing to evaluate swarm methods [147].

4.1.4. Outer Space

NASA has developed swarmies to gather water, ice, and minerals on Mars. They have also established a swarmathon to aid academics in developing an ant-based swarm algorithm. In-situ Resource Utilization (ISRU) is the name given to this application. Twenty swarmies cover a distance of 42 km in around 8 h. Another NASA Innovation Advanced Concept (NIAC) program project aims to enrich knowledge on the Mars exploration swarm of Marsbees [148]. These have the size of the bumblebee for robotics flapping wing flyers. They can explore and discover themselves in an unfamiliar place. With NIAC financing, a flapping flyer with insect-like wings will be offered as a technical implementation.

4.2. Industrial Projects and Products

These include UAV, UGV, UUV, and USV swarm robots developed for industrial projects and products. The available robot with respective type has been shown in Figure 12.

4.2.1. Terrestrial

Agriculture is essential to a country's growth. Food demand is growing, but the output is still insufficient [149]. *SwarmBot 3.0* is being used to monitor fields autonomously using Unmanned Ground Vehicles (UGVs). Before beginning the specified task, this swarm collaborates via a centrally controlled timetable. The large area is automatically subdivided into smaller fields and then allocated to an individual robot in the swarm [150]. Their tasks include sowing, applying fertilizers to the assigned areas, harvesting, and irrigation which is the requirement of the agriculture sector. Another fascinating innovation from the Fendt firm is the *UGV Xaver*, which is used for seeding and is powered by a battery [151].

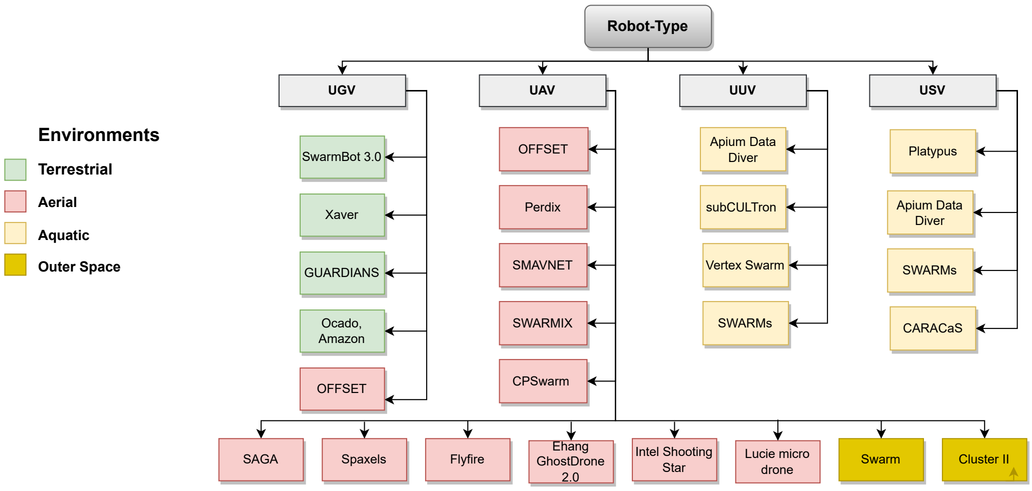

Figure 12. Classification of different industrial projects and products of swarm robotics.

The *GUARDIANS* (Group of Unmanned Assistant Robots Deployed in Aggregative Navigation by Scent) [152] has been used for emergency and rescue missions. They are used in places where human presence is prohibited or where the environment severely impairs human senses. This project assists in searching and warns against toxic chemicals using mobile communication links. They can form and navigate using potential fields and achieve the assigned task without explicit communication between the robots.

Another autonomous *Ocado* [153] warehouse has been developed that has a swarm of homogeneous cuboids and is being utilized for grocery orders and dispatching. A total of 1100 collaborative swarms of robots are used for the order and dispatch, where the workers put the customer order together. Robots are controlled from a central location by a cloud server, and data are exchanged via cellular technology between the robot and the cloud. Amazon [154], which employs Kiva, is the most prominent player in the swarming of robots in warehouses. An A* algorithm (with visual tags on the ground) searches for humans who assemble the customer's order. WLAN is used for the communication of robots, and dispatching is organized centrally. A low battery of robots is handled by the charging stations automatically. *Alibaba* [155] retailers are using a similar system for the autonomous order of goods and dispatching.

4.2.2. Aerial

The *OFFSET* (OFFensive Swarm Enabled Tactics) [156] projects are mostly deployed in military applications, although they can also be applied in other situations. This project aims to improve intra-city observations using UAVs and UGVs. These swarms of robots are capable of detecting hazards from the surroundings. *Perdix* [157], a military application swarm supported by the business. It is capable of performing its tasks without human piloting and has the ability to communicate with other drones to work collaboratively and achieve a common goal. These drones operate in a swarm of 20 or more and coordinate their actions to accomplish the desired outcome. *Pentagon* consisting of 103 drones, is another military application swarm. This swarm is not controlled by a single leader and can adapt to UAVs. They can fly in formation and make decisions as a group, making them useful for covert operations and targeted assassinations. The autonomous swarm is developed to install and manage WLAN network [158] as part of the *Swarming Micro Air Vehicle Network* (SMAVNET) project [159] in the emergency and rescue application sector. The project aims to gather rescue teams when disaster places have been explored and located. *SWARMIX* [160] is a similar search and rescue initiative in which a swarm of heterogeneous agents, such as humans, dogs, and Unmanned Aerial Vehicles (UAVs), create a swarm and engage in a search and eventually rescue operation.

Using a swarm of autonomous aerial vehicles, the Swarm robotics for Agricultural Applications (SAGA) project [161] seeks to do weed-spread monitoring and mapping. A swarm's fitness is decided by trade-off exploration and weed detection time in smart farming. Weeds and plants are detected and identified using a visual approach.

Nowadays, swarms are also providing entertainment in terms of light shows. The UAVs are equipped with colorful LEDs and perform the formation of different patterns accomplished by music to create a beautiful scene. In *Spaxels*, *Flyfire*, *Ehang*, *Intel* [162], and *Lucie micro*, 1000 Unmanned Aerial Vehicles (UAVs) are controlled from a central location and follow pre-programmed patterns.

4.2.3. Aquatic

Swarms are commonly used in aquatic environments to monitor the environment. *Platypus* [163] offers autonomous swarm robotics boats as USVs. They are utilized to keep track of water quality, produce a dense map of defined bodies beneath the surface, and stratify salinity and oxygen levels. *Apium Data Diver* is a prototype vehicle with a maximum depth of 100 m. It is meant for swarm operations on the surface and underwater, with temperature, pressure, and GPS among the sensors on board. It finds its application areas in defense, oceanography, hydrographic survey, and aquaculture. This type of swarm can be found in UUVs and USVs. It can accept high-level commands from a human operator and build a wide range of patterns [164]. *Hydromea's Vertex Swarm* is available in UUVs and can assess water quality in various places up to 300 m deep. It generates 3D data with great spatial and temporal resolution that is faster and more precise than manual approaches. The major purpose of the SWARMs (Smart Networking Underwater Robots in Cooperation Meshes) project [165] is to develop surface and underwater vehicles that can operate in maritime and offshore operations. It is responsible for designing and developing software and hardware components for the next generation of maritime vehicles, as well as assisting in the improvement of autonomy, robustness, cooperation, dependability, and cost-effectiveness. It uses offshore installations, chemical pollution monitoring, and plume tracking. Research focus lies on reliable underwater communication [166] and leveraging topology control [167].

The military has employed the CARACaS software kit, which is used in aquatic environment. NASA developed CARACaS (Control Architecture for Robotic Agent Command and Sensing), which has now been upgraded by ONR (Office of Naval Research) for autonomous Navy operations in the United States where USVs communicate with one another [168]. It enables USVs to choose their courses, protect assets in the navel, and intercept enemy boats as a group. In a demonstration at the James River in Virginia in 2014, CARACaS was installed on rigid-hulled boats and proved to be magnificent and successful [169]. Based on the discoveries of the CoCoRo, Submarine Cultures (SubCULTron) conduct long-term robotic exploration of unusual environmental niches. It is used on UUV robots to assess factors such as learning and self-sustainability.

4.2.4. Outer Space

Swarm was launched in 2013 and is made up of three identical spacecraft, two of which are side-by-side at 450 km and the third at 530 km above the ground. The mission of each satellite was to research the earth's magnetic field, and each was nine meters long [169]. Cluster II is a tetrahedral arrangement of four identical cylindrical spacecraft that was launched in 2000. It was initially capable of sending three-dimensional solar wind data on the earth's magnetosphere to investigate the sun's influence on the environment [170].

5. Swarm Robotics: Past, Present and Future Perspective

Social insects, fish schools, and bird flocks are examples of naturally self-organizing systems that display emergent collective behavior based on simple local knowledge [171,172]. Swarm robotics emerged as a branch of swarm intelligence, or the computational modeling of collective, self-organizing activity, which has yielded many successful optimization methods [173,174] that are now used in fields ranging from telecommunications [175] to crowd simulation, and prediction [176]. In contrast, swarm behavior in robots necessitate the installation of swarm intelligence algorithms on current robotic systems. Because of the expected ubiquity of autonomous robots in real-world applications and the challenge of allowing them to interact with one another and with their human users while avoiding the drawbacks of centralized control, swarm robotics research is gaining traction. Swarm robotics research will be crucial in addressing complex coordination problems in future robotics applications. It includes cooperative (i.e., robots working together to complete a common task) and semi-cooperative (i.e., self-interested robots benefiting from a globally efficient organization of activities, such as autonomous vehicles) scenarios. In the future, it will become a new and powerful tool in precision medicine, allowing for personalized therapies such as minimally invasive surgery or direct polytherapy delivery to malignant cells inside the human body [177,178]. Large numbers of robots with limited computation and communication capabilities, on the other hand, will push swarm robotics to its limits, necessitating the development of new conceptual tools in addition to tiny hardware or robotics devices [179].

In lab settings, robot swarms are shown using a small number of tiny robots [128,180]. Although technology advancements are pushing the bounds to ever-smaller sizes [177,181] and greater numbers [6,7], but the road to real-world applications remains lengthy and arduous. For example, *group scale*, from a few dozen to millions of people constituting the swarm and *physical scale*, from micro/nanorobots to massive terrestrial, aerial, and aquatic robots. Swarms that display prompt intervention and adaptability in a quickly changing environment to robots that work on months-long missions are examples of *temporal scale* (e.g., on a distant planet) from small-scale deployments to large-scale deployments and geographical scale. Previous, current, and future robotics achievements in terms of software, hardware, or a combination of the two are explained in the Table 1. Figure 13 shows the evolution of swarm robotics, to the best of our knowledge, from algorithmic research to the real-time best-performing swarm of robots.

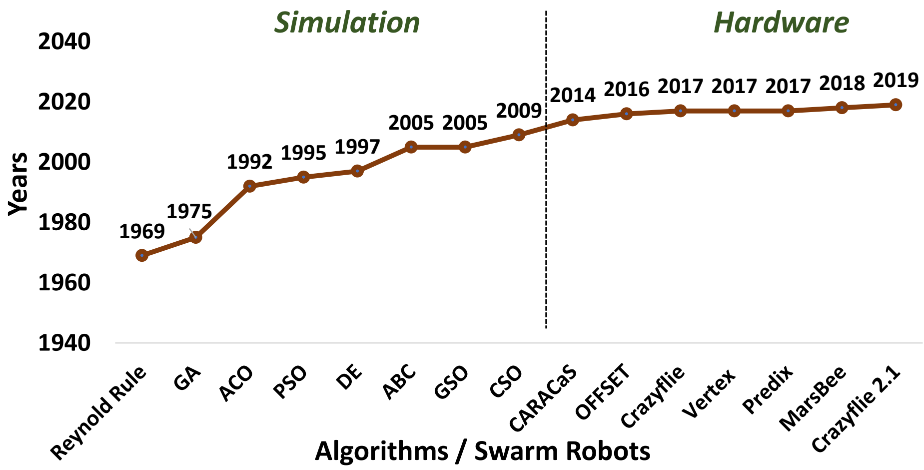

Figure 13. Evolution of swarm algorithms and swarm robotics.

6. Conclusions

Swarm robotics aims to develop simple, autonomous or self-governing robots that can cooperate to solve real-world problems collectively. Intelligent swarm algorithms are needed to enable the robots to interact autonomously and coordinate together without centralized control. The research on the swarm robotics domain started in the late 1900s, and the development work started in the early 2000s, gradually evolving the previous research and simulation work towards the actual real-world projection of swarm robotics. But there is a gap between theoretical and industrial research in swarm robotics. Theoretical research mainly focused on simulating swarm behaviours, while industrial research focuses on designing hardware that can execute swarm behaviour. Therefore, it is crucial to deploy swarm algorithms on hardware that can accommodate swarm behaviour functionality.

This article provides a comprehensive overview to new researchers of the swarm robotics field. It classifies the definition of swarm robots and identifies the difference between a multi-agent system and an actual swarm of agents. A detailed review of the swarm's most emerging swarm behaviors, and swarm intelligence algorithms is captured, keeping in view the limitation and the transformation towards the industrial application and development of the swarm robotic platform. In addition to the industrial application, this paper reviewed several research hardware platforms specifically designed to demonstrate or replicate any swarm behaviour. Finally, this paper concludes by reviving the era of swarm robotics from the past, present, and future projections with expected timelines of evolving the system and having real-world application, agnostic of swarm robotics platforms.

This article provides valuable insights for researchers in swarm robotics by highlighting various areas of research gaps, including algorithmic and hardware implementation. It emphasizes the importance of addressing these gaps to enable effective collaboration among robots. Researchers can bridge the gap between theoretical and industrial research in swarm robotics, leading to advancements in the field.

Author Contributions: M.M.S.: contributes to the main part of the research focused on the hardware including the swarm of robots, their limitations, future projections, and the overall organization of the paper structure. Z.S.: researched on swarm intelligence algorithms, swarm behaviors, and paper formatting. A.A.: review, rewriting, formatting and editing of the paper. H.M.: Supported in the review of industrial application, abstract, and conclusion. M.H.Y.: researched on the era of swarm robotics, allied applications and future projections. N.K.B. & F.H.: co-supervision of research work, paper reviewing, editing and structural guidance. All authors have read and agreed to the published version of the manuscript.

Funding: We acknowledge the Higher Education Commission (HEC) of Pakistan, for funding this project through a grant titled "Establishment of National Centre of Robotics and Automation (NCRA)" under Grant DF-1009-31.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: The authors acknowledge the funding and support from National Centre of Robotics and Automation (NCRA) Pakistan for this research work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Schranz, M.; Umlauft, M.; Sende, M.; Elmenreich, W. Swarm robotic behaviors and current applications. *Front. Robot. AI* **2020**, *7*, 36. [CrossRef] [PubMed]
2. Arnold, R.; Carey, K.; Abruzzo, B.; Korpela, C. What is a robot swarm: A definition for swarming robotics. In Proceedings of the 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 10–12 October 2019; pp. 74–81.
3. Dorigo, M.; Theraulaz, G.; Trianni, V. Reflections on the future of swarm robotics. *Sci. Robot.* **2020**, *5*, eabe4385. [CrossRef] [PubMed]
4. Floreano, D.; Mattiussi, C. *Bio-Inspired Artificial Intelligence*; MIT Press: Cambridge, MA, USA, 2008; Volume 5, pp. 335–396.

5. Maxim, P.M.; Spears, W.M.; Spears, D.F. Robotic chain formations. *IFAC Proc. Vol.* **2009**, *42*, 19–24. [CrossRef]
6. Rubenstein, M.; Cornejo, A.; Nagpal, R. Programmable self-assembly in a thousand-robot swarm . *Science* **2014**, *345*, 795–799. [CrossRef] [PubMed]
7. Slavkov, I.; Carrillo-Zapata, D.; Carranza, N.; Diego, X.; Jansson, F.; Kaorp, J.; Hauert, S.; Sharpe, J. Morphogenesis in robot swarms . *Sci. Robot.* **2018**, *3*, eaau9178. [CrossRef]
8. Carrillo, D.; Sharpe, J.; Winfield, A.F.; Giuggioli, L.; Hauert, S. Toward controllable morphogenesis in large robot swarms. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3386–3393. [CrossRef]
9. Bayındır, L. A review of swarm robotics tasks. *Neurocomputing* **2016**, *172*, 292–321. [CrossRef]
10. Sion, A.; Reina, A.; Birattari, M.; Tuci, E. Controlling robot swarm aggregation through a minority of informed robots. In Proceedings of the Swarm Intelligence: 13th International Conference, ANTS 2022, Málaga, Spain, 2–4 November 2022; Springer: Cham, Switzerland, 2022; pp. 91–103.
11. Ducatelle, F.; Di Caro, G.A.; Förster, A.; Bonani, M.; Dorigo, M.; Magnenat, S.; Mondada, F.; O'Grady, R.; Pinciroli, C.; Rétornaz, P.; et al. Cooperative navigation in robotic swarms. *Swarm Intell.* **2013**, *8*, 1–33. [CrossRef]
12. Varga, M.; Bogdan, S.; Dragojević, M.; Miklić, D. Collective search and decision-making for target localization. *Math. Comput. Model. Dyn. Syst.* **2012**, *18*, 51–65. [CrossRef]
13. Firat, Z.; Ferrante, E.; Zakir, R.; Prasetyo, J.; Tuci, E. Group-size regulation in self-organized aggregation in robot swarms. In *Swarm Intelligence, Proceedings of the 12th International Conference, ANTS 2020, Barcelona, Spain, 26–28 October 2020*; Springer: Cham, Switzerland, 2020; pp. 315–323.
14. Christensen, A.L.; O'Grady, R.; Birattari, M.; Dorigo, M. Exogenous fault detection in a collective robotic task. In Proceedings of the Advances in Artificial Life: 9th European Conference, ECAL 2007, Lisbon, Portugal, 10–14 September 2007; Springer: Berlin/Heidelberg, Germany, 2007.
15. Alitappeh, R.J.; Jeddisaravi, K. Multi-robot exploration in task allocation problem. *Appl. Intell.* **2022**, *52*, 2189–211. [CrossRef]
16. Moussa, M.; Beltrame, G. On the robustness of consensus-based behaviors for robot swarms. *Swarm Intell.* **2020**, *14*, 205–231. [CrossRef]
17. Tang, W.; Zhang, C.; Zhong, Y.; Zhu, P.; Hu, Y.; Jiao, Z.; Wei, X.; Lu, G.; Wang, J.; Liang, Y.; et al. Customizing a self-healing soft pump for robot. *Nat. Commun.* **2021**, *12*, 2247. [CrossRef] [PubMed]
18. Bonabeau, E.; Dorigo, M.; Theraulaz, G.; Theraulaz, G. *Swarm Intelligence: From Natural to Artificial Systems* ; Oxford University Press: New York, NY, USA, 1999.
19. Goldberg, D.E. *Genetic Learning in Optimization, Search and Machine Learning*; Addisson Wesley: Boston, CA, USA, 1994.
20. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
21. Grefenstette, J.J. Navy Center for Applied Research in Artificial Intelligence Naval Research Laboratory, Washington, DC 20375-5000. *Mach. Learn. Proc. 1989* **2014**, *28*, 340.
22. Bhattacharjya, D.R. *Introduction to Genetic Algorithms*; IIT Guwahati: Guwahati, India, 1998.
23. Devooght, R. Multi-Objective Genetic Algorithm. 2010, 1–39. Available online: epb-physique.ulb.ac.be/IMG/pdf/devooght_2011.pdf (accessed on 1 January 2014).
24. Uzel, O.; Koc, E. Basic of Genetic Programming, Graduation Project I. 2012, 1–25. Available online: http://mcs.cankaya.edu.tr/proje/2012/guz/omer_erdem/Rapor.pdf (accessed on 3 January 2014).
25. de Oliveira, G.M.; Silva, R.G.; do Amaral, L.R.; Martins, L.G. An evolutionary-cooperative model based on cellular automata and genetic algorithms for the navigation of robots under formation control. In Proceedings of the 2018 7th Brazilian Conference on Intelligent Systems (BRACIS), Sao Paulo, Brazil, 22–25 October 2018; pp. 426–431.
26. Shao, J. Robot Path Planning Method Based on Genetic Algorithm. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2021; Volume 1881, p. 022046.
27. Godinho, F.M.; Barco, C.F.; Tavares Neto, R.F. Using Genetic Algorithms to solve scheduling problems on flexible manufacturing systems (FMS): A literature survey, classification and analysis. *Flex. Serv. Manuf. J.* **2014**, *26*, 408–431. [CrossRef]
28. Cheng, C.; Yang, Z.; Xing, L.; Tan, Y. An improved genetic algorithm with local search for order acceptance and scheduling problems. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence in Production and Logistics Systems (CIPLS), Singapore, 16–19 April 2013; pp. 115–122.
29. Sachdeva, J.; Kumar, V.; Gupta, I.; Khelwal, N.; Ahuja, C.K. Multiclass brain tumor classification using GA-SVM. In Proceedings of the 2011 Developments in E-systems Engineering, Dubai, United Arab Emirates, 6–8 December 2011; pp. 182–187.
30. Khuntia, A.K.; Choudhury, B.B.; Biswal, B.B.; Dash, K.K. A heuristics based multi-robot task allocation. In Proceedings of the 2011 IEEE Recent Advances in Intelligent Computational Systems, Trivandrum, India, 22–24 September 2011; pp. 407–410.
31. Yang, Q.; Yu, M.; Liu, S.; Chai, Z.M. Path planning of robotic fish based on genetic algorithm and modified dynamic programming. In Proceedings of the 2011 International Conference on Advanced Mechatronic Systems, Zhengzhou, China, 11–13 August 2011; pp. 419–424.
32. Kang, C.C.; Chuang, Y.J.; Tung, K.C.; Chao, C.C.; Tang, C.Y.; Peng, S.C.; Wong, D.S. A genetic algorithm-based boolean delay model of intracellular signal transduction in inflammation. *BMC Bioinform.* **2011**, *12*, 17. [CrossRef]
33. Foschini, L.; Tortonesi, M. Adaptive and business-driven service placement in federated Cloud computing environments. In Proceedings of the 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 27–31 May 2013; pp. 1245–1251.

34. Fu, H.H.; Li, Z.J.; Li, G.W.; Jin, X.T.; Zhu, P.H. Modeling and controlling of engineering ship based on genetic algorithm. In Proceedings of the 2012 International Conference on Modelling, Identification and Control, Wuhan, China, 24–26 June 2012; pp. 394–398.
35. Mahmudy, W.F.; Marian, R.M.; Luong, L.H. Optimization of part type selection and loading problem with alternative production plans in flexible manufacturing system using hybrid genetic algorithms-part 1: Modelling and representation. In Proceedings of the 2013 5th International Conference on Knowledge and Smart Technology (KST), Chonburi, Thailand, 31 January 2013–1 February 2013; pp. 75–80.
36. Jing, X.; Liu, Y.; Cao, W. A hybrid genetic algorithm for route optimization in multimodal transport . In Proceedings of the 2012 Fifth International Symposium on Computational Intelligence and Design, Hangzhou, China, 28–29 October 2012; pp. 261–264.
37. Murata, T.; Ishibuchi, H. MOGA: Multi-objective genetic algorithms. In Proceedings of the IEEE International Conference on Evolutionary Computation, Piscataway, NJ, USA, 29 November 1995; pp. 289–294.
38. Dorigo, M. Optimization, Learning and Natural Algorithms. Ph.D. Thesis, Politecnico di Milano, Milan, Italy, 1992.
39. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [CrossRef]
40. Abreu, N.; Ajmal, M.; Kokkinogenis, Z.; Bozorg, B. *Ant Colony Optimization*; University of Porto: Porto, Portugal, 2011.
41. Selvi, V.; Umarani, R. Comparative analysis of ant colony and particle swarm optimization techniques. *Int. J. Comput. Appl.* **2010**, *5*, 1–6. [CrossRef]
42. Valdez, F.; Chaparro, I. Ant colony optimization for solving the TSP symetric with parallel processing. In Proceedings of the 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), Edmonton, AB, Canada, 24–28 June 2013; pp. 1192–1196.
43. Tosun, U.; Dokeroglu, T.; Cosar, A. A robust island parallel genetic algorithm for the quadratic assignment problem. *Int. J. Prod. Res.* **2013**, *51*, 4117–4133. [CrossRef]
44. Yagmahan, B.; Yenisey, M.M. A multi-objective ant colony system algorithm for flow shop scheduling problem. *Expert Syst. Appl.* **2010**, *37*, 1361–1368. [CrossRef]
45. Abdelaziz, A.Y.; Elkhodary, S.M.; Osama, R.A. Distribution networks reconfiguration for loss reduction using the Hyper Cube Ant Colony Optimization. In Proceedings of the 2011 International Conference on Computer Engineering & Systems, Cairo, Egypt, 29 November–1 December 2011; pp. 79–84.
46. Kumar, S.B.; Myilsamy, G. Multi-target tracking in mobility sensor networks using Ant Colony Optimization. In Proceedings of the 2013 IEEE International Conference On Emerging Trends in Computing, Communication and Nanotechnology (ICECCN), Tirunelveli, India, 25–26 March 2013; pp. 350–354.
47. Agrawal, P.; Kaur, S.; Kaur, H.; Dhiman, A. Analysis and synthesis of an ant colony optimization technique for image edge detection . In Proceedings of the 2012 International Conference on Computing Sciences, Phagwara, India, 14–15 September 2012; pp. 127–131.
48. He, Y.; Zeng, Q.; Liu, J.; Xu, G.; Deng, X. Path planning for indoor UAV based on Ant Colony Optimization. In Proceedings of the 2013 25th Chinese Control and Decision Conference (CCDC), Guiyang, China, 25–27 May 2013; pp. 2919–2923.
49. Abdallah, H.; Emara, H.M.; Dorrah, H.T.; Bahgat, A. Using ant colony optimization algorithm for solving project management problems. *Expert Syst. Appl.* **2009**, *36*, 10004–10015. [CrossRef]
50. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
51. Poli, R.; Kennedy, J.; Blackwell, T. Particle swarm optimization: An overview. *Swarm Intell.* **2007**, *1*, 33–57. [CrossRef]
52. Del Valle, Y.; Venayagamoorthy, G.K.; Mohagheghi, S.; Hernandez, J.C.; Harley, R.G. Particle swarm optimization: Basic concepts, variants and applications in power systems. *IEEE Trans. Evol. Comput.* **2008**, *12*, 171–195. [CrossRef]
53. Shi, Y.; Eberhart, R. A modified particle swarm optimizer. In Proceedings of the 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360), Anchorage, AK, USA, 4–9 May 1998; pp. 69–73.
54. Yan, X.; Wu, Q.; Liu, H.; Huang, W. An improved particle swarm optimization algorithm and its application. *Int. J. Comput. Sci. Issues* **2013**, *10*, 316.
55. Arumugam, M.S.; Rao, M.V.; Tan, A.W. A novel and effective particle swarm optimization like algorithm with extrapolation technique . *Appl. Soft Comput.* **2009**, *9*, 308–320. [CrossRef]
56. Kiranyaz, S.; Ince, T.; Yildirim, A.; Gabbouj, M. Fractional particle swarm optimization in multidimensional search space. *IEEE Trans. Syst. Man Cybern. Part B* **2009**, *40*, 298–319. [CrossRef] [PubMed]
57. Banks, A.; Vincent, J.; Anyakoha, C. A review of particle swarm optimization. Part I: Background and development. *Nat. Comput.* **2007**, *6*, 467–484. [CrossRef]
58. Banks, A.; Vincent, J.; Anyakoha, C. A review of particle swarm optimization. Part II: Hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications. *Nat. Comput.* **2008**, *7*, 109–124. [CrossRef]
59. Gong, D.; Lu, L.; Li, M. Robot path planning in uncertain environments based on particle swarm optimization. In Proceedings of the 2009 IEEE Congress on Evolutionary Computation, Trondheim, Norway, 18–21 May 2009; pp. 2127–2134.
60. Gong, M.; Cai, Q.; Chen, X.; Ma, L. Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *IEEE Trans. Evol. Comput.* **2013**, *18*, 82–97. [CrossRef]

61. Sivakumar, P.; Grace, S.S.; Azeezur, R.A. Investigations on the impacts of uncertain wind power dispersion on power system stability and enhancement through PSO technique. In Proceedings of the 2013 International Conference on Energy Efficient Technologies for Sustainability, Nagercoil, India, 10–12 April 2013; pp. 1370–1375.
62. Li, F.; Li, D.; Wang, C.; Wang, Z. Network signal processing and intrusion detection by a hybrid model of LSSVM and PSO. In Proceedings of the 2013 15th IEEE International Conference on Communication Technology, Guilin, China, 17–19 November 2013; pp. 11–14.
63. Jun, Z.; Kanyu, Z. A Particle Swarm Optimization Approach for Optimal Design of PID Controller for Temperature Control in HVAC. In Proceedings of the 2011 Third International Conference on Measuring Technology and Mechatronics Automation, Shanghai, China, 6–7 January 2011; pp. 230–233.
64. Atyabi, A.; Luerssen, M.; Fitzgibbon, S.P.; Powers, D.M.W. PSO-Based Dimension Reduction of EEG Recordings: Implications for Subject Transfer in BCI. *Neurocomputing* **2013**, *119*, 319–331. [CrossRef]
65. Mohan, S.; Mahesh, T.R. Particle Swarm Optimization based Contrast Limited enhancement for mammogram images. In Proceedings of the 2013 7th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India, 4–5 January 2013; pp. 384–388.
66. Gorai, A.; Ghosh, A. Hue-preserving colour image enhancement using particle swarm optimization. In Proceedings of the 2011 IEEE Recent Advances in Intelligent Computational Systems, Trivandrum, India, 22–24 September 2011; pp. 563–568.
67. Na, L.; Yuanxiang, L. Image Restoration Using Improved Particle Swarm Optimization . In Proceedings of the 2011 International Conference on Network Computing and Information Security, Guilin, China, 14–15 May 2011; pp. 394–397.
68. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
69. Price, K.; Storn, R.; Lampinen, J. *Differential Evolution: A Practical Approach to Global Optimization*; Springer: Berlin/Heidelberg, Germany, 2005; Volume 104, p. 542.
70. Wu, Y.; Lee, W.; Chien, C. Modified the Performance of Differential Evolution Algorithm with Dual Evolution Strategy. *Int. Conf. Mach. Learn. Comput.* **2011**, *3*, 57–63.
71. Muñoz, J.; López, B.; Quevedo, F.; Barber, R.; Garrido, S.; Moreno, L. Geometrically constrained path planning for robotic grasping with Differential Evolution and Fast Marching Square. *Robotica* **2023**, *41*, 414–432. [CrossRef]
72. Mahmoud, Z.S.; Powers, D.; Sammut, K.; Yazdani, A.M. Differential evolution for efficient AUV path planning in time variant uncertain underwater environment. *arXiv* **2016**, arXiv:1604.02523.
73. Dragoi, E.; Curteanu, S.; Vlad, D. Differential evolution applications in electromagnetics. In Proceedings of the 2012 International Conference and Exposition on Electrical and Power Engineering, Iasi, Romania, 25–27 October 2012; pp. 636–640.
74. Myeong-Chun, L.; Sung-Bae, C. Interactive differential evolution for image enhancement application in smart phone. In Proceedings of the IEEE Congress on Evolutionary Computation, Brisbane, QLD, Australia, 10–15 June 2012; pp. 1–6.
75. Yilmaz, A.R.; Yavuz, O.; Erkmen, B. Training multilayer perceptron using differential evolution algorithm for signature recognition application. In Proceedings of the 2013 21st Signal Processing and Communications Applications Conference (SIU), Haspolat, Turkey, 24–26 April 2013; pp. 1–4.
76. Chiou, J.; Chang, C.; Wang, C. Hybrid differential evolution for static economic dispatch. In Proceedings of the 2014 International Symposium on Computer, Consumer and Control, Taichung, Taiwan, 10–12 June 2014; pp. 950–953.
77. Karaboga, D. *An Idea Based on Honeybee Swarm for Numerica Optimization*; Technical Report TR06; Erciyes University: Talas/Kayseri, Türkiye, 2005.
78. Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numeric function optimization: Artificial Bee Colony (ABC) algorithm. *J. Glob. Optim.* **2007**, *39*, 459–471. [CrossRef]
79. Karaboga, D. Artificial bee colony algorithm. *Scholarpedia* **2010**, *5*, 6915. [CrossRef]
80. Singh, A. An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem. *Appl. Soft Comput.* **2009**, *9*, 625–631. [CrossRef]
81. Abu-Mouti, F.S.; El-Hawary, M.E. Overview of Artificial Bee Colony (ABC) algorithm and its applications. In Proceedings of the 2012 IEEE International Systems Conference SysCon, Vancouver, BC, Canada, 19–22 March 2012; pp. 1–6.
82. Gerhardt, E.; Gomes, H.M. Artificial Bee Colony (ABC) algorithm for engineering optimization problems. In Proceedings of the International Conference on Engineering Optimization, Rio de Janeiro, Brazil, 1–5 July 2012; pp. 1–11.
83. Tan, L.; Shi, J.; Gao, J.; Wang, H.; Zhang, H.; Zhang, Y. Multi-UAV path planning based on IB-ABC with restricted planned arrival sequence. *Robotica* **2023**, *41*, 1244–1257. [CrossRef]
84. Sharma, T.K.; Pant, M. Golden search based artificial bee colony algorithm and its application to solve engineering design problems. In Proceedings of the 2012 Second International Conference on Advanced Computing & Communication Technologies, Rohtak, India, 7–8 January 2012; pp. 156–160.
85. Lee, C.H.; Park, J.Y.; Park, J.Y.; Han, S.Y. Application of artificial bee colony algorithm for structural topology optimization. In Proceedings of the 2012 8th International Conference on Natural Computation, Chongqing, China, 29–31 May 2012; pp. 1023–1025.
86. Lee, T.E.; Cheng, J.H.; Jiang, L.L. A new artificial bee colony based clustering method and its application to the business failure prediction. In Proceedings of the 2012 International Symposium on Computer, Consumer and Control, Taichung, Taiwan, 4–6 June 2012; pp. 72–75.

87. Karaboga, D.; Gorkemli, B.; Ozturk, C.; Karaboga, N. A comprehensive survey: Artificial bee colony (ABC) algorithm and applications. *Artif. Intell. Rev.* **2014**, *42*, 21–57. [CrossRef]
88. Krishnan, K.N.; Ghose, D. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intell.* **2009**, *3*, 87–124. [CrossRef]
89. Krishnan, K.N.; Ghose, D. Glowworm swarm optimisation: A new method for optimising multi-modal functions. *Int. J. Comput. Intell. Stud.* **2009**, *1*, 93–119.
90. Krihnanand, K.N.; Amruth, P.; Guruprasad, M.H. Glowworm-inspired Robot Swarm for Simultaneous Taxis towards Multiple Radiation Sources. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, (ICRA), Orlando, FL, USA, 15–19 May 2006; pp. 958–963.
91. Zainal, N.; Zain, A.M.; Radzi, N.H.; Udin, A. Glowworm swarm optimization (GSO) algorithm for optimization problems: A state-of-the-art review. *Appl. Mech. Mater.* **2013**, *421*, 507–511. [CrossRef]
92. Zhang, Y.; Ma, X.; Miao, Y. Localization of multiple odor sources using modified glowworm swarm optimization with collective robots. In Proceedings of the 30th Chinese Control Conference, Yantai, China, 22–24 July 2011; pp. 1899–1904.
93. Goel, U.; Varshney, S.; Jain, A.; Maheshwari, S.; Shukla, A. Three dimensional path planning for UAVs in dynamic environment using glow-worm swarm optimization. *Procedia Comput. Sci.* **2018**, *133*, 230–239. [CrossRef]
94. Khan, A.; Aftab, F.; Zhang, Z. Self-organization based clustering scheme for FANETs using Glowworm Swarm Optimization. *Phys. Commun.* **2019**, *36*, 100769. [CrossRef]
95. He, D.X.; Zhu, H.Z.; Liu, G.-Q. Glowworm swarm optimization algorithm for solving multi-constrained QoS multicast routing problem. In Proceedings of the 2011 Seventh International Conference on Computational Intelligence and Security, Sanya, China, 3–4 December 2011; pp. 66–70.
96. Krishnan, K.N.; Ghose, D. Detection of multiple source locations using a glowworm metaphor with applications to collective robotics. In Proceedings of the 2005 IEEE Swarm Intelligence Symposium, Pasadena, CA, USA, 8–10 June 2005; pp. 84–91.
97. Senthilnath, J.; Omkar, S.N.; Mani, V.; Tejovanth, N.; Diwakar, P.G.; Shenoy, A. Multi-spectral satellite image classification using glowworm swarm optimization. In Proceedings of the 2011 IEEE International Geoscience and Remote Sensing Symposium, Vancouver, BC, Canada, 24–29 July 2011; pp. 47–50.
98. McGill, K.; Taylor, S. Comparing swarm algorithms for large scale multi-source localization. In Proceedings of the 2009 IEEE International Conference on Technologies for Practical Robot Applications, Woburn, MA, USA, 9–10 November 2009; pp. 48–54.
99. Menon, P.P.; Ghose, D. Simultaneous source localization and boundary mapping for contaminants. In Proceedings of the 2012 American Control Conference (ACC), Montreal, QC, Canada, 27–29 June 2012; pp. 4174–4179.
100. Yang, X.S.; Deb, S. Cuckoo search via Lévy flights. In Proceedings of the 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), Coimbatore, India, 9–11 December 2009; pp. 210–214.
101. Yang, X.S.; Deb, S. Engineering optimisation by cuckoo search. *Int. J. Math. Model. Numer. Optim.* **2010**, *1*, 330–343. [CrossRef]
102. Yang, X.S.; Deb, S. Multi-objective cuckoo search for design optimization. *Comput. Oper. Res.* **2013**, *40*, 1616–1624. [CrossRef]
103. Xie, C.; Zheng, H. Application of improved Cuckoo search algorithm to path planning unmanned aerial vehicle. In Proceedings of the Intelligent Computing Theories and Application: 12th International Conference, ICIC 2016, Lanzhou, China, 2–5 August 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 722–729.
104. Chaowanawatee, K.; Heednacram, A. Implementation of cuckoo search in RBF neural network for flood forecasting. In Proceedings of the 2012 4th International Conference on Computational Intelligence, Communication Systems and Networks, Phuket, Thailand, 24–26 July 2012; pp. 22–26.
105. Kumar, A.; Chakarverty, S. Design optimization for reliable embedded system using Cuckoo Search. In Proceedings of the 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, India, 8–10 April 2011; Volume 1, pp. 264–268.
106. Khodier, M. Optimisation of antenna arrays using the cuckoo search algorithm. *IET Microwaves Antennas Propag.* **2013**, *7*, 458–464. [CrossRef]
107. Vo, D.N.; Schegner, P.; Ongsakul, W. Cuckoo search algorithm for non-convex economic dispatch. *Gener. Transm. Distrib.* **2013**, *7*, 645–654. [CrossRef]
108. Yang, X.S.; Deb, S.; Karamanoglu, M.; Xingshi, N. Cuckoo search for business optimization applications. In Proceedings of the 2012 National Conference on Computing and Communication Systems, Durgapur, India, 21–22 November 2012; pp. 1–5.
109. Jati, G.K.; Manurung, H.M.; Suyanto, S. Discrete cuckoo search for traveling salesman problem. In Proceedings of the 2012 7th International Conference on Computing and Convergence Technology (ICCCT), Seoul, Republic of Korea, 3–5 December 2012; pp. 993–997.
110. Sharkey, A.J. Swarm robotics and minimalism. *Connect. Sci.* **2007**, *19*, 245–260. [CrossRef]
111. Programmable Robot Swarms. Available online: https://wyss.harvard.edu/technology/programmable-robot-swarms/ (accessed on 6 November 2019).
112. Caprari, G.; Balmer, P.; Piguet, R.; Siegwart, R. The Autonomous Micro Robot "Alice": A platform for scientific and commercial applications. In Proceedings of the MHA'98, 1998 International Symposium on Micromechatronics and Human Science Creation of New Industry (Cat. No. 98TH8388), Nagoya, Japan, 25–28 November 1998; pp. 231–235.
113. Arvin, F.; Samsudin, K.; Ramli, A.R. Development of a miniature robot for swarm robotic application. *Int. J. Comput. Electr. Eng.* **2009**, *1*, 436–442. [CrossRef]

114. Arvin, F.; Murray, J.; Zhang, C.; Yue, S. Colias: An autonomous micro robot for swarm robotic applications. *Int. J. Adv. Robot. Syst.* **2014**, *11*, 113. [CrossRef]
115. Soares, J.M.; Navarro, I.; Martinoli, A. The Khepera IV mobile robot: Performance evaluation, sensory data and software toolbox. In Proceedings of the Robot 2015: Second Iberian Robotics Conference: Advances in Robotics, Lisbon, Portugal, 19–21 November 2015; Volume 1 , pp. 767–781.
116. Pickem, D.; Lee, M.; Egerstedt, M. The GRITSBot in its natural habitat-a multi-robot testbed. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 4062–4067.
117. Jones, S.; Studley, M.; Hauert, S.; Winfield, A.F. A two teraflop swarm. *Front. Robot. AI* **2018**, *5*, 11. [CrossRef]
118. Riedo, F.; Chevalier, M.; Magnenat, S.; Mondada, F. Thymio II, a robot that grows wiser with children. In Proceedings of the 2013 IEEE Workshop on Advanced Robotics and Its Social Impacts, Tokyo, Japan, 7–9 November 2013; pp. 187–193.
119. Wilson, S.; Gameros, R.; Sheely, M.; Lin, M.; Dover, K.; Gevorkyan, R.; Haberl, M.; Bertozzi, A.; Berman, S. Pheeno, a versatile swarm robotic research and education platform. *IEEE Robot. Autom. Lett.* **2016**, *1*, 884–891. [CrossRef]
120. Jdeed, M.; Zhevzhyk, S.; Steinkellner, F.; Elmenreich, W. Spiderino-a low-cost robot for swarm research and educational purposes. In Proceedings of the 2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES), Hamburg, Germany, 12–13 June 2017; pp. 35–39.
121. Seyfried, J.; Szymanski, M.; Bender, N.; Estana, R.; Thiel, M.; Wörn, H. The I-SWARM project: Intelligent small world autonomous robots for micro-manipulation. In Proceedings of the Swarm Robotics: SAB 2004 International Workshop, Santa Monica, CA, USA, 17 July 2004.
122. Le Goc, M.; Kim, L.H.; Parsaei, A.; Fekete, J.D.; Dragicevic, P.; Follmer, S. Zooids: Building blocks for swarm user interfaces. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology, Tokyo, Japan, 16–19 October 2016; pp. 97–109.
123. Dhanaraj, N.; Hewitt, N.; Edmonds-Estes, C.; Jarman, R.; Seo, J.; Gunner, H. Adaptable platform for interactive swarm robotics (apis): A human-swarm interaction research testbed. In Proceedings of the 19th International Conference on Advanced Robotics (Belo Horizonte), Belo Horizonte, Brazil, 2–6 December 2019; pp. 720–726. [CrossRef]
124. Kettler, A.; Szymanski, M.; Wörn, H. The wanda robot and its development system for swarm algorithms. In *Advances in Autonomous Mini Robots, Proceedings of the 6-th AMiRE Symposium, Bielefeld, Germany, 23–25 May 2011*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 133–146.
125. Klingner, J.; Kanakia, A.; Farrow, N.; Reishus, D.; Correll, N. A stick-slip omnidirectional powertrain for low-cost swarm robotics: Mechanism, calibration, and control. In Proceedings of the 2014 IEEE International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 846–851. [CrossRef]
126. Mondada, F.; Pettinaro, G.C.; Kwee, I.W.; Guignard, A.; Gambardella, L.M.; Floreano, D.; Nolfi, S.; Deneubourg, J.L.; Dorigo, M. *SWARM-BOT: A Swarm of Autonomous Mobile Robots with Self-Assembling Capabilities*; Technical Report; ETH-Zürich: Zürich, Switzerland, 2002.
127. Groß, R.; Bonani, M.; Mondada, F.; Dorigo, M. Autonomous self-assembly in swarm-bots. *IEEE Trans. Robot.* **2006**, *22*, 1115–1130.
128. Dorigo, M.; Floreano, D.; Gambardella, L.M.; Mondada, F.; Nolfi, S.; Baaboura, T.; Birattari, M.; Bonani, M.; Brambilla, M.; Brutschy, A.; et al. Swarmanoid: A novel concept for the study of heterogeneous robotic swarms . *IEEE Robot. Autom. Mag.* **2013**, *20*, 60–71. [CrossRef]
129. Petersen, K.H.; Nagpal, R.; Werfel, J.K. Termes: An autonomous robotic system for three-dimensional collective construction. In *Robotics: Science and Systems VII*; MIT Press: Cambridge, CA, USA, 2011 .
130. Kernbach, S.; Meister, E.; Schlachter, F.; Jebens, K.; Szymanski, M.; Liedke, J.; Laneri, D.; Winkler, L.; Schmickl, T.; Thenius, R.; et al. Symbiotic robot organisms: REPLICATOR and SYMBRION projects. In Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems, Gaithersburg, MD, USA, 19–21 August 2008; pp. 62–69.
131. Duff, D.; Yim, M.; Roufas, K. Evolution of polybot: A modular reconfigurable robot. In Proceedings of the Harmonic Drive Intelligent Symposium, Nagano, Japan, 19 November 2001.
132. Murata, S.; Yoshida, E.; Kamimura, A.; Kurokawa, H.; Tomita, K.; Kokaji, S. M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME Trans. Mechatronics* **2002**, *7*, 431–441. [CrossRef]
133. Kurokawa, H.; Kamimura, A.; Yoshida, E.; Tomita, K.; Kokaji, S.; Murata, S. M-TRAN II: Metamorphosis from a four-legged walker to a caterpillar. In Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No. 03CH37453), Las Vegas, NV, USA, 27–31 October 2003; Volume 3, pp. 2454–2459.
134. Kurokawa, H.; Tomita, K.; Kamimura, A.; Kokaji, S.; Hasuo, T.; Murata, S. Distributed self-reconfiguration of M-TRAN III modular robotic system. *Int. J. Robot. Res.* **2008**, *27*, 373–386. [CrossRef]
135. Brandt, D.; Christensen, D.J.; Lund, H.H. ATRON robots: Versatility from self-reconfigurable modules. In Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, Harbin, China, 5–8 August 2007; pp. 26–32.
136. Castano, A.; Behar, A.; Will, P.M. The Conro modules for reconfigurable robots. *IEEE/ASME Trans. Mechatronics* **2002**, *7*, 403–409. [CrossRef]
137. Wei, H.; Cai, Y.; Li, H.; Li, D.; Wang, T. Sambot: A self-assembly modular robot for swarm robot. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 66–71.
138. Zykov, V.; Mytilinaios, E.; Desnoyer, M.; Lipson, H. Evolved and designed self-reproducing modular robotics. *IEEE Trans. Robot.* **2007**, *23*, 308–319. [CrossRef]

139. Cheng, J.; Cheng, W.; Nagpal, R. Robust and self-repairing formation control for swarms of mobile agents. In Proceedings of the Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, Pittsburgh, PA, USA, 9–13 July 2005; Volume 5.
140. Swetha, S.; Anandan, R.; Kalaivani, K. An investigation of micro aerial vehicles (μAV). *Int. J. Eng. Technol.* **2018**, *7*, 174–177. [CrossRef]
141. Roberts, J.F.; Stirling, T.S.; Zufferey, J.; Floreano, D. Quadrotor using minimal sensing for autonomous indoor flight. In Proceedings of the European Micro Air Vehicle Conference and Flight Competition (Toulouse), Toulouse, France, 20 September 2007.
142. Oung, R.; D'Andrea, R. The distributed flight array. *Mechatronics* **2011**, *21*, 908–917. [CrossRef]
143. Preiss, J.A.; Honig, W.; Sukhatme, G.S.; Ayanian, N. Crazyswarm: A large nano-quadcopter swarm. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3299–3304.
144. Heckert, A. Entwicklung eines Dynamischen Modells und Parameterschätzung für den FINken 3 Quadkopter. Bachelor's Thesis, Fakultät für Elektrotechnik und Informationstechnik, Otto-von-Guericke-Universitat Magdeburg, Magdeburg, Germany, 2016 .
145. Osterloh, C.; Pionteck, T.; Maehle, E. MONSUN II: A small and inexpensive AUV for underwater swarms. In Proceedings of the 7th German Conference on Robotics, Munich, Germany, 21–22 May 2012; pp. 1–6.
146. Christensen, A.L.; Duarte, M.; Postolache, O.; Sargento, S.; Oliveira, M.J.; Santana, P.; Nunes, L.; Velez, F.; Oliveira, S.M.; Sebastião, P.; et al. Design of communication and control for swarms of aquatic surface drones. *Des. Commun. Control. Swarms Aquat. Surf. Drones* **2015**, 548–555. [CrossRef]
147. Duarte, M.; Costa, V.; Gomes, J.; Rodrigues, T.; Silva, F.; Oliveira, S.M.; Christensen, A.L. Evolution of collective behaviors for a real swarm of aquatic surface robots. *PLoS ONE* **2016**, *11*, e0151834. [CrossRef] [PubMed]
148. Kang, C.K.; Fahimi, F.; Griffin, R.; Landrum, D.B.; Mesmer, B.; Zhang, G.; Lee, T.; Aono, H.; Pohly, J.; McCain, J.; et al. Marsbee-Swarm of Flapping Wing Flyers for Enhanced Mars Exploration. 2019. Available online: https://ntrs.nasa.gov/citations/20190002496 (accessed on 1 March 2023) .
149. Tilman, D.; Balzer, C.; Hill, J.; Befort, B.L. Global food demand and the sustainable intensification of agriculture. *Proc. Natl. Acad. Sci. USA* **2011**, *108*, 20260–20264. [CrossRef] [PubMed]
150. Ball, D.; Ross, P.; English, A.; Patten, T.; Upcroft, B.; Fitch, R.; Sukkarieh, S.; Wyeth, G.; Corke, P. Robotics for sustainable broad-acre agriculture. In Proceedings of the Field and Service Robotics: Results of the 9th International Conference, Brisbane, Australia, 9–11 December 2013; Springer: Berlin/Heidelberg, Germany, 2015; pp. 439–453.
151. Blender, T.; Buchner, T.; Fernandez, B.; Pichlmaier, B.; Schlegel, C. Managing a mobile agricultural robot swarm for a seeding task. In Proceedings of the IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016; pp. 6879–6886.
152. Saez-Pons, J.; Alboul, L.; Penders, J.; Nomdedeu, L. Multi-robot team formation control in the GUARDIANS project. *Ind. Robot. Int. J.* **2010**, *37*, 372–383. [CrossRef]
153. Telegraph, T. The Science Behind the Swarm of Robots Picking Your Grocery Order. Available online: https://www.telegraph.co.uk/technology/2018/06/04/science-behind-swarm-robots-picking-grocery-order/ (accessed on 2 August 2018).
154. Brown A. Rise of the Machines? Amazon's Army of More Than 100,000 Warehouse Robots Still Can't Replace Humans Because They Lack 'Common Sense'. Available online: http://www.dailymail.co.uk/sciencetech/article5808319/Amazon-100-000-warehouse-robots-company-insists-replacehumans.html (accessed on 2 August 2019).
155. Pickering, J. Take a Look Inside Alibaba's Smart Warehouse Where Robots do 70% of the Work. Available online: https://www.businessinsider.com/inside-alibaba-smart-warehouse-robots-70-per-cent-work-technologylogistics-2017-9?IR=T (accessed on 2 August 2019).
156. Chung, T.H. OFFensive Swarm-Enabled Tactics (OFFSET). Available online: https://www.darpa.mil/attachments/OFFSET_ProposersDay.pdf (accessed on 2 April 2019).
157. Mizokami, K. The Pentagon's Autonomous Swarming Drones Are the Most Unsettling Thing You'll See Today. Available online: https://www.popularmechanics.com/military/aviation/a24675/pentagon-autonomousswarming-drones/ (accessed on 2 April 2018).
158. Varga, M.; Basiri, M.; Heitz, G.; Floreano, D. Distributed formation control of fixed wing micro aerial vehicles for area coverage. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 669–674.
159. Hauert, S.; Zufferey, J.C.; Floreano, D. Evolved swarming without positioning information: An application in aerial communication relay. *Auton. Robot.* **2009**, *26*, 21–32. [CrossRef]
160. Flushing, E.F.; Gambardella, L.M.; Di Caro, G.A. A mathematical programming approach to collaborative missions with heterogeneous teams. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 396–403.
161. Albani, D.; Manoni, T.; Arik, A.; Nardi, D.; Trianni, V. Field coverage for weed mapping: Toward experiments with a UAV swarm. In Proceedings of the Bio-inspired Information and Communication Technologies: 11th EAI International Conference, BICT 2019, Pittsburgh, PA, USA, 13–14 March 2019; Springer: Cham, Switzerland, 2019.
162. Barrett B. Intel Lights Up the Night with 500 'Shooting Star' Drones. Available online: https://www.wired.com/story/more-inclusive-fourth-of-july-drone-show/ (accessed on 2 August 2019).

163. Jeradi, A.; Raeissi, M.M.; Farinelli, A.; Brooks, N.; Scerri, P. Focused Exploration for Cooperative Robotic Watercraft. In Proceedings of the 2015 International Workshop on Artificial Intelligence and Robotics, Las Vegas, NV, USA, 9–10 May 2015; pp. 83–93.
164. Maccready, T. Multiscale vorticity from a swarm of drifters. In Proceedings of the 2015 IEEE/OES Eleveth Current, Waves and Turbulence Measurement (CWTM), St. Petersburg, FL, USA, 2–6 March 2015; pp. 1–6.
165. Real-Arce, D.A.; Morales, T.; Barrera, C.; Hernández, J.; Llinás, O. Smart and networking underwater robots in cooperation meshes: The swarms ECSEL: H2020 project. *Instrum. Viewp.* **2016**, *19*.
166. Rodríguez-Molina, J.; Bilbao, S.; Martínez, B.; Frasheri, M.; Cürüklü, B. An optimized, data distribution service-based solution for reliable data exchange among autonomous underwater vehicles. *Sensors* **2017**, *17*, 1802. [CrossRef]
167. Li, N.; Cürüklü, B.; Bastos, J.; Sucasas, V.; Fernandez, J.A.; Rodriguez, J. A probabilistic and highly efficient topology control algorithm for underwater cooperating AUV networks. *Sensors* **2017**, *17*, 1022. [CrossRef]
168. Smalley, D. Autonomous Swarmboats: New Missions, Safe Harbors. *Off. Nav. Res.* **2016**, *14*.
169. Hsu, J. US navy's drone boat swarm practices harbor defense. *IEEE Spectr.* **2016**, *19*.
170. Escoubet, C.P.; Fehringer, M.; Goldstein, M. Introduction the cluster mission. In *Proceeding Annual Geophysics*; Copernicus GmbH: Göttingen, Germany, 2001; Volume 19, pp. 1197–1200.
171. Sneyd, J.; Theraula, G.; Bonabeau, E.; Deneubourg, J.L.; Franks, N.R. *Self-Organization in Biological Systems*; Princeton University Press: Princeton, NJ, USA, 2003.
172. Sumpter, D.J. Collective animal behavior. In *Collective Animal Behavior*; Princeton University Press: Princeton, NJ, USA, 2010.
173. Dorigo, M. Ant colony optimization. *Scholarpedia* **2007**, *2*, 1461. [CrossRef]
174. Dorigo, M.; de Oca, M.A.; Engelbrecht, A. Particle swarm optimization. *Scholarpedia* **2008**, *3*, 1486. [CrossRef]
175. Di Caro, G.; Dorigo, M. AntNet: Distributed stigmergetic control for communications networks. *J. Artif. Intell. Res.* **1998**, *9*, 317–365. [CrossRef]
176. Moussaïd, M.; Helbing, D.; Theraulaz, G. How simple rules determine pedestrian behavior and crowd disasters. *Proc. Natl. Acad. Sci. USA* **2011**, *108*, 6884–6888. [CrossRef]
177. Jeon, S.; Kim, S.; Ha, S.; Lee, S.; Kim, E.; Kim, S.Y.; Park, S.H.; Jeon, J.H.; Kim, S.W.; Moon, C.; et al. Magnetically actuated microrobots as a platform for stem cell transplantation . *Sci. Robot.* **2019**, *4*, eaav4317. [CrossRef]
178. Dong, X.; Sitti, M. Controlling two-dimensional collective formation and cooperative behavior of magnetic microrobot swarms. *Int. J. Robot. Res.* **2020**, *39*, 617–638. [CrossRef]
179. Sitti, M. *Mobile Microrobotics*; MIT Press: Cambridge, MA, USA, 2017.
180. Nouyan, S.; Groß, R.; Bonani, M.; Mondada, F.; Dorigo, M. Teamwork in self-organized robot colonies. *IEEE Trans. Evol. Comput.* **2009**, *13*, 695–711. [CrossRef]
181. Xie, H.; Sun, M.; Fan, X.; Lin, Z.; Chen, W.; Wang, L.; Dong, L.; He, Q. Reconfigurable magnetic microrobot swarm: Multimode transformation, locomotion, and manipulation. *Sci. Robot.* **2019**, *4*, eaav8006. [CrossRef]