A Survey on Efficiency of Multi-Swarm Systems through Collaboration and Communication Techniques

Shambhavi Singh[1], Sparsh K Kachhadiya[1], Chaudhari Mohit M[1], Ishita Jaiswal[1] and K.R. Anupama[2]

***Abstract*— Multi-swarm systems, characterized by the coordination of multiple individual swarms, are instrumental in addressing multi-modal problems where multiple local optima exist. This paper explores the factors influencing the efficiency of these systems, including the organization of sub-swarms, the dynamic processes of swarm formation and disbandment post-task completion, navigation strategies, the underlying communication network structure, and the communication ranges employed. We examine multi-swarm systems, particularly for collaborative robotic frameworks that tackle intricate challenges to optimize task-solving capabilities. Leveraging techniques such as grouping and dispersion, the systems demonstrate enhanced task completion within diverse environments. This study contributes valuable insights into designing and implementing multi-swarm systems for improved problem-solving capabilities.**

I. INTRODUCTION

Swarm systems, wherein multiple robots collaborate to accomplish tasks beyond the capacity of individual units, have garnered significant attention across diverse applications. In swarm robotics, agents are typically physically and behaviorally undifferentiated, aligning with ethological models of self-organizing natural systems. However, this design choice simplifies the intricate heterogeneity observed in natural counterparts, presenting an abstraction that aids modeling.

The extension of swarm robotics into multi-swarm systems enhances their capabilities by orchestrating multiple swarms to address various tasks within a shared environment. The allocation of tasks and collaboration among swarms hinge upon the complexity of designated tasks, leading to dynamic variations in swarm size for optimal performance [1]. Interaction between swarms within the same environment necessitates meticulous consideration during system design to ensure effective collaboration and resolve potential conflicts.

Swarm robots excel in executing a diverse range of tasks, encompassing odor source localization, deployment, task allocation, object assembly, self-assembly, coordinated motion, group size estimation, distributed rendezvous, collective decision-making, and human-swarm interaction. Collaborative manipulation, involving the joint effort of swarm robots to manipulate objects in the environment, is particularly valuable for tasks challenging for individual robots, such as object transportation and stick pulling. Evaluating the effectiveness of completed tasks is crucial, with metrics like completion time and formations providing valuable insights [2].

Fig. 1: Multiple swarms of birds in flight - demonstrating real-time dynamic interactions between swarms and collective coordinated movements of birds

Deployment, a critical aspect of swarm robotics, involves robots autonomously positioning themselves without central coordination. Direct communication or stigmergy, where robots respond to the actions of their predecessors, facilitates data sharing among robots. Task allocation strategies, including threshold-based methods and probabilistic approaches, contribute to successfully executing collaborative tasks such as foraging and collecting scattered items. The arrangement of swarms within multi-swarm systems is a pivotal determinant of their performance, with various configurations tailored to specific problem requirements. Particle Swarm Optimization (PSO) is one such method that mimics the collective behavior of birds searching for food, guiding robots to optimal solutions. In dynamic scenarios, the adaptability of PSO proves essential. Charged Particle Swarm Optimization (CPSO) introduces chaos in the orbits of repelling particles, maintaining diversity in swarms, while Quantum Swarm Optimization (QSO) employs a quantum model for enhanced exploration.

However, challenges arise in Multi-CPSOs due to the delicate balance between attractive forces in the nucleus and repulsive forces of orbiting robots, potentially resulting in swarms reaching equilibrium near suboptimal solutions. Proposed solutions include introducing competition between

*This work was done in partial fulfillment of a Course on Embedded Systems Design at BITS Pilani

[1]Shambhavi Singh, Sparsh Kanubhai Kachhadiya, Chaudhari Mohit Milind and Ishita Jaiswal graduated from BITS Pilani in 2023. {Shambhavi Singh f20180276@alumni.bits-pilani.ac.in}

[2]K. R. Anupama is a Professor at the Department of Electrical and Electronics Engineering at BITS Pilani K. K. Birla Goa Campus anupkr@goa.bits-pilani.ac.in

swarms, ensuring that the swarm with a higher score remains at a given optimum [2].

These considerations underscore the importance of individual swarm characteristics in achieving solutions, as exemplified by the challenges faced by multi-CPSOs in dynamic environments. This paper delves into the intricate interplay of factors influencing the efficiency and effectiveness of multi-swarm systems, shedding light on their application in solving complex tasks.

II. COMMUNICATION IN HETEROGENEOUS SWARMS

Interoperability is crucial in multi-agent systems, especially considering their autonomous operation and interaction with other agents or non-agent entities for coordinated actions. This characteristic underscores the need for effective communication among individual systems. The swarm is expected to autonomously execute tasks while simultaneously communicating with other systems to optimize actions for the overall benefit of the entire swarm. Integration implies that each system can communicate and interact (control) with the System of Systems (SoS) regardless of hardware and software characteristics. Ensuring compatibility is essential for seamless communication [5].

A. Spatially Targeted Communication

In cooperation between two homogeneous swarms, communication is typically specific to the system and its applications. To our knowledge, the most explored heterogeneous swarm system involves cooperation between two different swarms: foot-bots and eye-bots. Foot-bots are small autonomous robots specialized in moving on both even and uneven terrains, capable of self-assembling and transporting objects or other robots. Eye-bots are autonomous flying robots that can attach to an indoor ceiling, analyzing the environment from a privileged position to collectively gather information inaccessible to foot-bots [3].

Communication between eye-bots and foot-bots is achieved through spatially targeted control signals and control policy statuses of the particular bots. The status of a foot-bot is communicated through LEDs fixed on their bodies to the eye-bot. Positioned on the ceiling, eye-bots provide directional instructions to the foot-bots on the ground, guiding them towards the source or the target location [4].

The introduction of a third swarm system, hand-bots, to this existing setup has also been explored. Hand-bots are autonomous robots capable of climbing vertical surfaces and manipulating small objects. These systems operate without centralized control, relying on continued local and non-local interactions to produce collective self-organized behavior. The dynamic self-reconfigurability of these robots allows them to form ad hoc coalitions or integrated structures locally on a need basis, enhancing their capacity to perform more complex tasks [6].

B. Strategies Implemented on the Network Layers

To achieve some of the main characteristics of a system of systems approach, communications play a vital role. Task management is proposed by dividing the swarm network. A control plane interface connects An evolving base station to the nodes. Two-stage control data plane splitting (TSCDP) is introduced to enhance network usage by optimizing unwanted access. This approach improves task allocation and resource management in heterogeneous swarms and increases the data retention rate during transmission [5].

Considering robotic swarm systems in heterogeneous environments (land, water, and air) as a larger System of Systems [6], three channels of communication in complex heterogeneous swarms are discussed:

1) Over-the-Air and Underwater Communication: Communication can be implemented in various ways, including radio frequency modulation, acoustic propagation, and fiber-optic communication. Among these, radio modems, specifically Zigbee modules, are chosen for over-the-air communication due to their low-power wireless communication technology and an international protocol for wireless networking. Zigbee modules reduce data size, allowing for lower-cost network construction with a simplified protocol and limited functionality. RF signals are deemed suitable for underwater communication, and a wideband communication solution has been chosen to address potential degradation in salty water. The communication process involves nodes sending information to the Ground node to establish communication, including the robot's position and a unique identification number for each node [7], [8].

2) Underwater to Surface Communication: A surface vessel acts as a gateway between swarms in both underwater and on-land environments. Directional antennas, optic fibers, or wire-line mechanisms establish communication between the surface vessel and the underwater swarm. Intermediate relay swarm robots can facilitate communication between the surface vehicle's antenna and the underwater swarm. The chosen hardware includes XBee Pro modules for land-based vehicle communication and OFDM for underwater vehicle communication [9].

3) Air to Ground Communication: RF technology is the most suitable form of communication, considering various parameters. ZigBee-based radio modems provide the PHY and MAC layers for a communication protocol, offering the freedom to use a custom protocol for the swarm of robots. Serial Line Internet Protocol (SLIP) is used to benefit from low overhead requirements and create a customized protocol tailored to the needs of the swarm. A cyclic redundancy check (CRC) is introduced to reduce errors in SLIP, and when bandwidth and processing resources permit, extensible markup language may be employed in messages to reduce errors. The ground-based vehicles relay messages to underwater vehicles, forming a System of Systems Communication for heterogeneous independently operable systems [10].

C. Hybrid Particle Swarm Optimization

Understanding Heterogeneous Particle Swarm Optimization (PSO) algorithms requires an introduction to Homogeneous PSO algorithms.

*1) **Homogeneous PSO algorithms**:* In commonly studied Homogeneous PSO algorithms, certain variants are briefly reviewed here:

1) *Position and Velocity Updates:*

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{1}$$

$$\begin{aligned} v_{ij}(t) = wv_i j(t) + c_1 r_{1j}(t)(y_{ij}(t) - x_a(t)) + \\ c_2 r_{2j}(t)(\hat{y}_j(t) - x_{ij}(t)) \end{aligned} \tag{2}$$

Here $x_{ij}(t)$, $y_{ij}(t)$ and $y_i(t)$ refer to particle i's position, personal best position, and global best position in dimension j at time step t respectively. The constants c_1 and c_2 is the acceleration coefficients, and $r_{1j}(t), r_{2j}(t) \approx U(0,1)$. In the above, w is the inertia weight. Equation 2 results in particles with a balance of exploration and exploitation depending on the values of the parameters, w, c_1, and c_2. Suppose parameters are changed such that c_1 is initially larger than c_2 and is decreased linearly over time, while c_2 is linearly increased. In that case, the model will focus on exploration during the initial search steps and move towards more exploitation as the number of iterations increases. [12].

2) *Cognitive-Only Model:*

$$v_{ij}(t+1) = wv_i j(t) + c_1 r_{1j}(t)(y_{ij}(t) - xij(t)) \tag{3}$$

The cognitive-only velocity update removes the social component, resulting in a model that encourages exploration as each node becomes a hill-climber[13].

3) *Social-Only Model:*

$$v_{ij} = wv_{ij}(t) + c_2 r_{2j}(t)(\hat{y}_i(t) - x_{ij}(t)) \tag{4}$$

The social-only velocity update removes the cognitive component, resulting in a model where each node is a stochastic hill-climber [13]. The position update equation is the same as Equation 1. In this model, each node is a stochastic hill-climber node.

Communication between robotic nodes depends on the roles assigned to them. New variants of PSO have been designed to work well in dynamic environments, involving splitting the population into interacting swarms. These swarms communicate locally through an exclusion parameter and globally through a new anti-convergence operator. Additionally, each swarm maintains diversity either through charged or quantum particles.

*2) **Heterogeneous PSO algorithms**:* Examples of existing approaches of Heterogeneous PSO algorithms where nodes implement different behaviors include:

1) *Static HPSO (sHPSO):* Behaviors are randomly assigned during initialization and remain constant throughout the search process.
2) *Dynamic HPSO (dHPSO):* Particle behaviors can change randomly during the search process. When a particle fails to improve its personal best position over a window of recent iterations, it randomly selects new behaviors from the behavior pool.
3) *Division of labor PSO:* Nodes can switch to a local search near the end of the search process. [14]
4) *Charged PSO:* Some nodes have a charge, while others do not. Charged nodes add repelling force to the velocity update rule.[15]
5) *Life-cycle PSO:* Nodes follow a life-cycle, transitioning from a PSO particle to a genetic algorithm individual to a stochastic hill-climber. Individuals may follow different behaviors at any time.[16]
6) *Predator-prey PSO:* The swarm contains predator and prey nodes. Predator nodes are attracted only to the global best position, exploiting, while prey nodes repel from the position of predator nodes.[17]
7) *Niche PSO:* Developed to locate multiple solutions, a main swarm of nodes is used, where nodes implement a cognitive-only velocity update. Sub-swarms are formed around optima, with nodes following the guaranteed convergence PSO [18].
8) *Guaranteed convergence PSO:* The global best particle follows a different and comparatively higher exploitative search behavior than all the other nodes. [19]

III. GROUPING AND DEPLOYMENT OF HETEROGENEOUS SYSTEMS

In a self-deployment scenario, robots must deploy themselves in an environment without central coordination. This task has many potential practical applications, from mapping unknown environments to autonomous surveillance systems. The swarms are deployed and grouped initially following one of the below methods. The best individual in each inactive swarm will be a good starting point for searching for new (local) optima. Therefore, the best individual of each inactive swarm serves as the seed of a new active swarm. After an active swarm has become inactive or obsolete, the respective members will either remain inactive or join another active swarm for optimal use of resources.

A. Direct Communication

Direct robot-to-robot communication is the most used mechanism for cooperation in self-deployment tasks. Communication can occur using explicit messages or implicitly by sensing other robots' nearby presence and relative position. Information acquired from nearby robots can be used to implement simple mechanisms of robot avoidance or, more often, to regulate the position and velocity of a robot according to a desired behavior. In the latter case, robot movement can be determined following principles of artificial physics to preserve connectivity between swarm members or obtaining formations described by a specific geometric relationship between neighboring robots.

B. Stigmergy

Stigmergic communication gives robots moving in an area of the environment an indication of previous actions done by other robots in the same area. In various scenarios (e.g., in the foraging task), it is used with a positive feedback mechanism, i.e., an action done by a robot increases the probability that other robots repeat the same action. Conversely, for the deployment task, a negative feedback mechanism can be put in place, preventing different robots from repeating the same action and specifically preventing the same areas of the environment from being explored multiple times or, in tasks where the environment must be covered repeatedly, maximizing the time between two successive visits to the same place. Stigmergic communication has been implemented in past works using simulated pheromone traces: the presence and intensity of a pheromone at a given location is used to indicate that the location has been visited before. In some studies, pheromone is assumed to evaporate over time, analogously to what happens in nature with chemical traces, and this property is used to optimize repeated coverage of the same area or to dynamically assign non-overlapping patrolling areas to different robots.

C. Dispersion Algorithm

In the dispersion task, swarm members must position themselves away from one another to maximize the area covered globally by the swarm and/or minimize the time needed to cover the area. For example, a robot dispersion technique can be applied in scenarios where robots must find particular locations in the environment or objects located in unknown places (as in the case of foraging robots). Thus, the dispersion task can be used as a sub-task of more complex activities. In some cases, an additional constraint is given by the requirement that the connectivity of the swarm must be preserved, i.e., each robot must be able to sense or communicate with at least another robot so that there are no isolated groups. It is intuitively understood that programming robots so that they avoid each other while moving in the environment increases the capability of the swarm to cover a large area, compared to a simple random walk technique. Kuyucu et al. [20] used a genetic algorithm to evolve a set of parameters (e.g., pheromone production rate) that influence the swarm performance in the deployment task. According to simulation results, parameter values obtained with the evolutionary method lead to better performance than manually tuned values. Stigmergic communication allows a group of robots to coordinate to dynamically partition an area into contiguous territories, with each territory patrolled by one robot; an adaptive variant allows swarm members to dynamically learn the optimal size of their respective territories based on the arena size and the total number of robots. Some studies utilized artificial physics methods based on the concept of virtual potential fields and virtual forces to regulate the mutual distance between robots where the objective is to disperse the swarm. Howard et al. [21] proposed a control law for the velocity of robots based on a potential field determined by the presence of other robots and obstacles. Nearby entities and moves repel each robot near other robots or obstacles according to the virtual force determined by this repulsion. This mechanism leads the swarm to optimize the occupation of the arena according to the total number of robots. Podury and Sukhatme [23] used potential fields to maximize the area covered by a swarm of robots with defined sensing and communication ranges, with the constraint that each robot must stay within the communication range of a minimum number of other robots. In [24], maximizing the area covered by a swarm of connected robots is tackled with an automatic design method using probabilistic finite state machines, where the parameters of robot controllers are selected with an optimization algorithm. In many deployment tasks, and especially in those using potential field approaches, it is assumed that a robot can measure the distance and relative orientation of nearby robots with reasonable precision. When using real robots, infrared technology usually offers this capability: line-of-sight communication with highly directive signal radiation patterns with known attenuation characteristics. In [25], the swarm deployment task is performed by robots using radio frequency communication, characterized by a much less predictable mapping between signal strength and distance; relative orientation cannot generally be inferred from the received signal. Despite these difficulties, the algorithm proposed successfully disperses a robot swarm in the environment.

D. Pattern Formation Algorithm

Pattern formation is a variant of the deployment task where robots occupy relative positions such that when viewed globally, their ensemble can be described by a specific pattern. For example, such formations can be used in surveillance tasks where each robot is assigned a specific area to be monitored, and the swarm must prevent situations with uncovered spots. The capability of a robot to measure the relative distance and orientation of its neighbors allows a high degree of flexibility in determining the desired positions of neighbors, from which multi-robot formations can emerge. Thus, by using local rules, if each robot in a swarm positions itself to obtain a desired distance and orientation with respect to neighboring robots (i.e., forming a geometric shape with its neighbors), at a global level, the swarm can converge to a state where it is deployed optimally in the environment. In [22], an extensive analysis is performed on the dynamics of the formation of different patterns with robots controlled by virtual forces. The authors described how two and three-dimensional hexagonal lattices of self-controlled particles can be obtained using attraction and repulsion forces. In addition, particles are subjected to a vicious friction force proportional to the particle speed and whose purpose is to avoid continuous oscillations around an equilibrium state. In [26], each robot chooses two other robots among its neighbors and then positions itself to form a triangular shape with those neighbors. The distance between the robot and its neighbors is chosen based on a measured local characteristic of the environment. If all robots operate with the same algorithm, and if the environment characteristic that determines the distance has the same value in the entire covered area, this technique leads the formed triangles to be equilateral, and thus, a regular mesh pattern is observed at the swarm level. Another analogous technique is used in a three-dimensional space, where each robot selects three

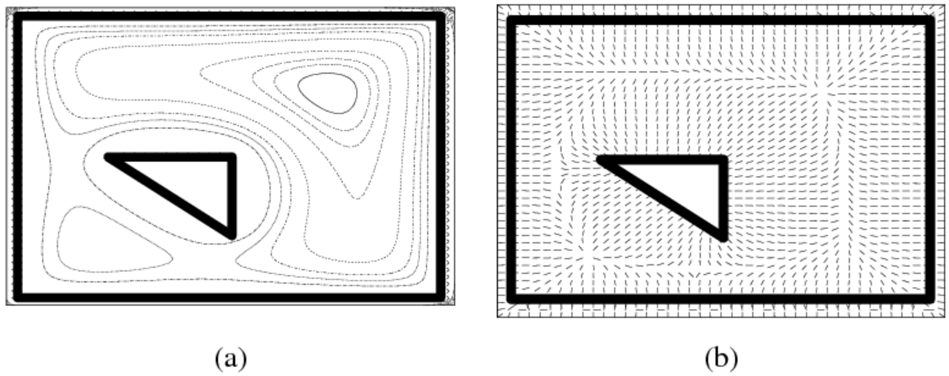

Fig. 2: (a) Potential field generated by a simple environment; the contours show the lines of equal potential. (b) This potential generates force fields; the arrows indicate the force's direction (but not magnitude).

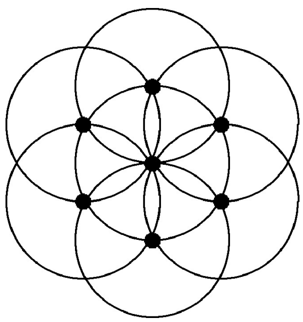

Fig. 3: Six circles can be drawn on the perimeter of a central circle, forming a hexagon at the intersection of the circles.

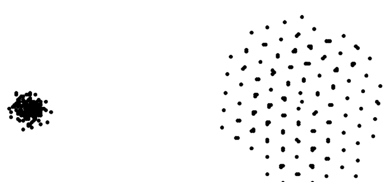

Fig. 4: Initially, the particles are assumed to be in a tight cluster t = 0 (left). Then particles repel, and after 1,000 time steps form a good hexagonal lattice (right).

neighbors and tries to form a tetrahedron.

III. IV. TASK ALLOCATION

Task allocation mainly studies how to use the swarm of robots to complete a series of tasks. It is challenging for swarm robots to allocate tasks efficiently through self-organization in an unknown dynamic environment, and such conditions require adaptive task allocation for optimal working performance. Various algorithms have been formulated based on intelligent systems in nature, such as ant colonies, fish swarms, bird flocks, bacterial foraging, etc.

A. Ant Colony Optimization (ACO)

The success of many insect colonies in their methods of labor division suggests that they are an excellent inspiration for solving complex coordination problems in multi-agent systems. Ants can determine the most efficient path between their nest and food sources and alert other ants. Based on ants' behavior, the ACO algorithm consists of two phases - the development of the shortest path between food and the nest and a pheromone update. These two phases are repeated over and over until the shortest path is found. Various adapted ACO algorithms are analyzed below. [27]

1) Ant Task Allocation (ATA) Algorithm: This algorithm is proposed based on a honeybee's task selection model. Each agent selects its current task randomly according to the probability defined by the following: When an agent finishes its current task, the threshold is updated according to the agent's working performance. This algorithm differs from ACO, as ants in ACO share a common pheromone field because they need to build a common optimal solution to the problem. In contrast, each individual ant in ATA keeps a private record of its response thresholds for different task categories to achieve specialization. [28]

B. Bacterial Foraging Optimization (BFO)

The social foraging behavior of Escherichia coli inspires this algorithm. Bacteria search for nutrients to maximize energy obtained per unit of time, and individual bacteria communicate with others by sending out signals. Chemotaxis is the process in which a bacterium moves by taking small steps while searching for nutrients, and the key idea of BFO is mimicking the chemotactic movement of virtual bacteria in the problem search space. BFO mimics the four principal mechanisms observed in a real bacterial system: chemotaxis, swarming, reproduction, and elimination and dispersal. [29]

C. Optimal Mass Transport (OMT)

This theory is based on economic principles of supply and demand. The goal is to find the mapping between two distributions so that the mapping is the least expensive for a given metric. This theory is used to carry out the dynamic allocation of swarm robots by taking robots as suppliers and tasks as the demand. The mapping of robots to tasks is obtained by solving the optimal mass transport equations. [30]

IV. CONCLUSION

Existing swarm robotics systems are limited by displaying simple proof-of-concept behaviors under laboratory conditions. Several papers reviewed point out the drawback of the almost universal insistence on homogeneous system components. To be applicable to more real-world systems, heterogeneous systems should be developed more. The field of swarm robotics currently lacks methods and tools to leverage the heterogeneity of naturally occurring systems and emulate them through swarms. This paper reviewed existing research on the important aspects of developing a multi-swarm system and the challenges faced in such systems. Several optimization algorithms can be applied to multiple swarms for better performance and robustness to failures and external disturbances. The use of multi-swarm systems to solve multiple problems in an environment is a resource-consuming process. The deployment and grouping techniques after a task is finished for the redeployment of the swarm into the same environment will increase the efficiency of the multi-swarm system as we can make optimal use of the resources available. The use of the different arrangements of individual swarms depending on the characteristics of the environment and the problem will also contribute to the accuracy of the solution.

REFERENCES

[1] Bayındır, Levent. “A Review of Swarm Robotics Tasks.” Neurocomputing, vol. 172, 2016, pp. 292–321., https://doi.org/10.1016/j.neucom.2015.05.116.

[2] Blackwell, Tim, and Jürgen Branke. “Multi-Swarm Optimization in Dynamic Environments.” Lecture Notes in Computer Science, 2004, pp. 489–500., https://doi.org/10.1007/978-3-540-24653-$4_5$0.

[3] Mathews et al. ”Establishing Spatially Targeted Communication in a Heterogeneous Robot Swarm,” Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010), May 10–14, 2010, Toronto, Canada, pp 939-946.

[4] Frederick Ducatelle, Gianni A. Di Caro, and Luca M. Gambardella. 2010. ”Cooperative self - organization in a heterogeneous swarm robotic system.” In Proceedings of the 12th annual conference on Genetic and evolutionary computation (GECCO ’10). Association for Computing Machinery, New York, NY, USA, 87–94.

[5] Ferat Sahin, Prasanna Sridhar, Ben Horan, Vikraman Raghavan and Mo Jamshidi, ”Systems of Systems Approach for Threat Detection and Integration of Heterogeneous Independently Operable Systems,” in Proc. IEEE Int. Conf. Syst, Man, Cybern, 2007.

[6] K. Nagothu, T. Shaneyfelt, P. Benavidez, M. A. Jordens, S. kota and M. Jamshidi, ”Systems of Systems Communication for heterogeneous independent operable systems,” 2009 IEEE International Conference on System of Systems Engineering (SoSE), 2009, pp. 1-6.

[7] Kranthimanoj Nagothu, Matthew Joordens, and Mo Jamshidi, ”Communications for underwater Robotics Research platforms,” in proc IEEE Systems Conference, Montreal, Canada, 2008, p. 6.

[8] Sinem Coleri Ergen, ”Zigbee/IEEE 802.15.4 Summary”, Sep. 2004.

[9] Zhou, S., Wang, Z. (2014). ”OFDM for underwater acoustic communications.” John Wiley Sons.

[10] Benavidez, Patrick Nagothu, Kranthimanoj Ray, Anjan Shaneyfelt, Ted Kota, Satyanarayana Behera, Laxmidhar Jamshidi, Mo. (2008). ”Multi-domain robotic swarm communication system.” 1 - 6. 10.1109/SYSOSE.2008.4724189.

[11] Eberhart, R., Kennedy, J.: ”A New Optimizer using Particle Swarm Theory.” In: Proceedings of the Sixth International Symposium on Micromachine and Human Science, pp. 39–43 (1995)

[12] Kennedy, J., Eberhart, R.: ”Particle Swarm Optimization.” In: Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 1942–1948 (1995)

[13] Kennedy, J.: ”The Particle Swarm: Social Adaptation of Knowledge.” In: Proceedings of the IEEE International Conference on Evolutionary Computation, pp. 303–308 (1997)

[14] Vesterstrøm, J., Riget, J., Krink, T.: ”Division of Labor in Particle Swarm Optimization.” In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1570–1575. IEEE Press, Los Alamitos (2002)

[15] Blackwell, T., Bentley, P.: ”Dynamic Search with Charged Swarms.” In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 19–26 (2002)

[16] Krink, T., vberg, M.L.: ”The Life Cycle Model: Combining Particle Swarm Optimization, Genetic Algorithms and Hill Climbers.” In: Guervos, J.J.M., Adamidis, P.A.,Beyer, H.-G., Fernandez-Villacanas, J.-L., Schwefel, H.-P. (eds) PPSN 2002. LNCS, vol. 2439, pp. 621–630. Springer, Heidelberg (2002)

[17] Silva, A., Neves, A., Costa, E.: ”An Empirical Comparison of Particle Swarm and Predator Prey Optimisation.” In: O’Neill, M., Sutcliffe, R.F.E., Ryan, C., Eaton, M., Griffith, N.J.L. (eds) AICS 2002. LNCS (LNAI), vol. 2464, pp. 103–110. Springer, Heidelberg (2002)

[18] Brits, R., Engelbrecht, A., van den Bergh, F.: ”A Niching Particle Swarm Optimizer.” In: Proceedings of the Fourth Asia-Pacific Conference on Simulated Evolution and Learning, pp. 692–696 (2002)

[19] Van den Bergh, F., Engelbrecht, A.: ”A New Locally Convergent Particle Swarm Optimizer.” In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, pp. 96–101 (2002)

[20] Kuyucu, Tüze, et al. ”Evolutionary Optimization of Pheromone-Based Stigmergic Communication.” Applications of Evolutionary Computation, 2012, pp. 63–72., https://doi.org/10.1007/978-3-642-29178-4_7.

[21] Howard, Andrew, et al. ”Mobile Sensor Network Deployment Using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem.” Distributed Autonomous Robotic Systems 5, 2002, pp. 299–308., https://doi.org/10.1007/978-4-431-65941-$9_3$0.

[22] Spears, William M., et al. ”Distributed, Physics-Based Control of Swarms of Vehicles.” Autonomous Robots, vol. 17, no. 2/3, 2004, pp. 137–162., https://doi.org/10.1023/b:auro.0000033970.96785.f2.

[23] Poduri, S., and G.S. Sukhatme. ”Constrained Coverage for Mobile Sensor Networks.” IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA ’04. 2004, https://doi.org/10.1109/robot.2004.1307146.

[24] Francesca, Gianpiero, et al. ”An Experiment in Automatic Design of Robot Swarms.” Lecture Notes in Computer Science, 2014, pp. 25–37., https://doi.org/10.1007/978-3-319-09952-1_3.

[25] Ugur, Emre, et al. ”Dispersion of a Swarm of Robots Based on Realistic Wireless Intensity Signals.” 2007 22nd International Symposium on Computer and Information Sciences, 2007, https://doi.org/10.1109/iscis.2007.4456899.

[26] ”Self-Configurable Mobile Robot Swarms: Adaptive Triangular Mesh Generation.” Networking Humans, Robots and Environments, 2013, pp. 59–75., https://doi.org/10.2174/9781608057313113010008.

[27] A. Nayyar and R. Singh, ”Ant Colony Optimization — Computational swarm intelligence technique,” 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), 2016, pp. 1493-1499.

[28] J. Du, Z. Ling, Q. Peng, S. Zhen and L. Yang, ”Task allocation in multi-agent systems with swarm intelligence of social insects,” 2010 Sixth International Conference on Natural Computation, 2010, pp. 4322-4326, doi: 10.1109/ICNC.2010.5583611.

[29] Das S., Biswas A., Dasgupta S., Abraham A. (2009) Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications. In: Abraham A., Hassanien AE., Siarry P., Engelbrecht A. Foundations of Computational Intelligence Volume 3. Studies in Computational Intelligence, vol 203. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-01085-9_2.

[30] Wang, Qiuzhen, and Xinjun Mao. 2020. ”Dynamic Task Allocation Method of Swarm Robots Based on Optimal Mass Transport Theory.” Symmetry 12, no. 10: 1682. https://doi.org/10.3390/sym12101682.