Survey of Methods and Algorithms of Robot Swarm Aggregation

N E Shlyakhov, **I V Vatamaniuk**, **A L Ronzhin**

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, 39, 14 line, St. Petersburg, Russia.

E-mail: ronzhin@iias.spb.su

Abstract. The paper considers the problem of swarm aggregation of autonomous robots with the use of three methods based on the analogy of the behavior of biological objects. The algorithms substantiating the requirements for hardware realization of sensor, computer and network resources and propulsion devices are presented. Techniques for efficiency estimation of swarm aggregation via space-time characteristics are described. The developed model of the robot swarm reconfiguration into a predetermined three-dimensional shape is presented.

1. Introduction

The concept of swarm robotics was formed on the basis of scientific paradigms, such as multi-agent technology and swarm robotics, where the principles of the decentralized functioning of autonomous groups of robots based on pairwise interactions were first formulated. The characteristic feature of swarm robotics is the use of heterogeneous unified mobile robotic systems that solve different problems and have different on-board resources and propulsion devices. In the field of multi-agent swarm robotics, multi-agent technologies are used to simulate the interaction of large groups of simple homogeneous robots. The limited resources of individual robots significantly affect the configuration and capabilities of the system. However, the distributed swarm intelligence, which is based on data retrieved during mass pairwise interactions of robots, ensures that a swarm exists and solves the required tasks.

Let us consider some tasks which require different numbers of robots to solve them. Systems composed of a large number of autonomous agents (robots) may be used to perform collective tasks when the tasks can not be performed by a single robot, or they can be performed much more efficiently by a group of robots. In [1], the following task categories for robots are defined:

1) Tasks that are exclusively performed by one agent;
2) Tasks that can be performed more efficiently by a large number of agents;
3) Tasks traditionally performed by a large number of agents;
4) Tasks requiring a large number of agents.

Figure 1 shows examples of problems that can be solved by using single robots and robotic swarm interaction. Swarm robotics focuses on the last three categories. Most studies have shown [2,3] that the use of a large number of robots for executing a task allows working with robots characterized by simplified functionality. If a single robot solves the problem, it usually has a more complex structure.

In recent years, the principles of swarm intelligence have been widely studied and implemented to solve various tasks by a group of autonomous robots using the so-called distributed approach (without a centralized group coordination) [2-4].

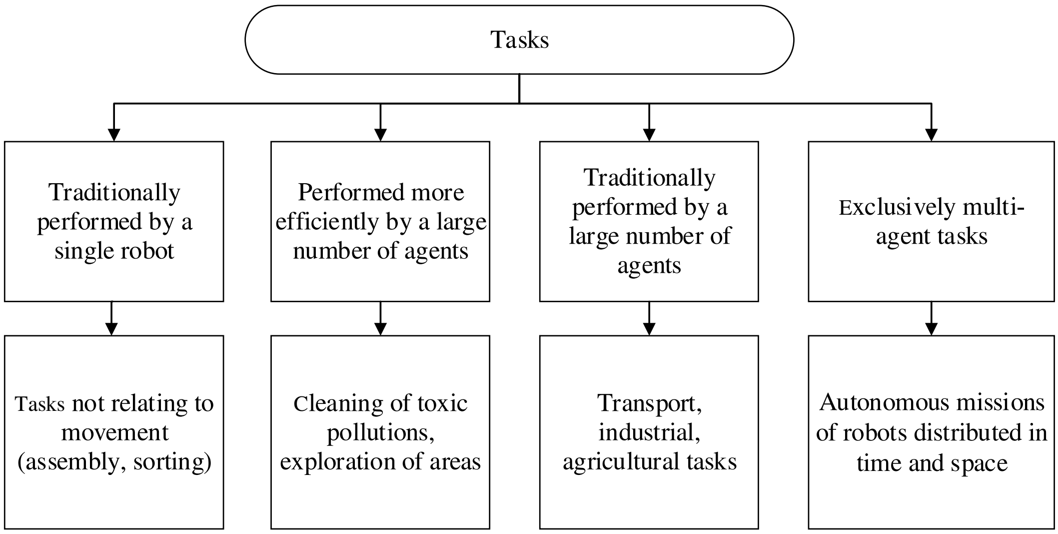

Figure 1. Classification of tasks performed by robots

2. Robot aggregation methods

Let us consider the problem of decentralized cooperation of robots when aggregated. Self-aggregation, i.e. the grouping of a certain number of autonomous objects in one place is a very frequent behavioral pattern in the natural world [5, 6]. An example of aggregation of autonomous robots is shown in Figure 2.

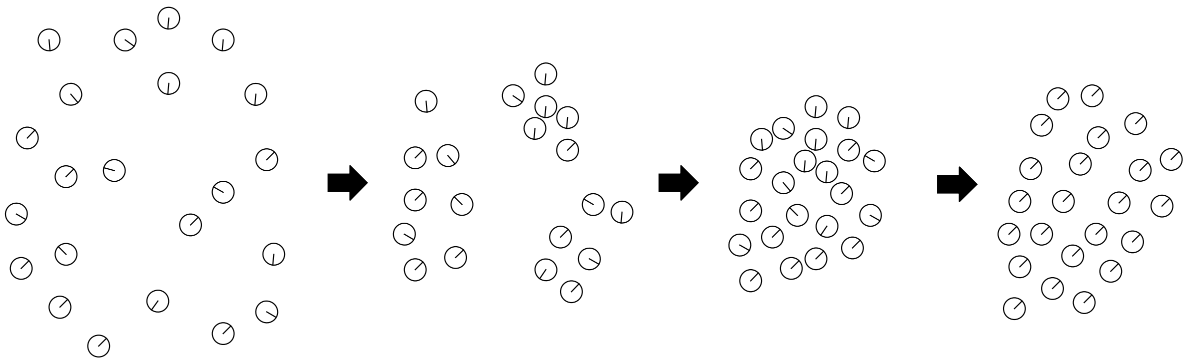

Figure 2. An example of aggregation of autonomous agents

As a rule, the aggregation problem is studied either as an independent problem, or as part of more specialized tasks implying grouping a number of agents. Hereinafter, self-organized groups of robots will be called aggregates in accordance with the common terminology [7, 8]. The analysis of approaches to the aggregation of a robot swarm has shown that the most effective approaches include the method of virtual forces, probabilistic methods and evolutionary methods. Let us consider them in more detail.

2.1. The method of virtual forces

The behavior of autonomous robots is often modeled with the use of the method of virtual forces. It is based on calculating the forces that determine the motion of robots relative to each other taking into account the location of the surrounding objects. Many organized groups of animals (for example, swarms of insects, flocks of birds, schools of fish) can be modeled with the help of attractive forces (due to which neighboring animals tend to stay close to each other) and repulsive forces (which prevent the occurrence of collisions between animals) [7] . Each autonomous robot moves according to the force acting on it from the neighboring robots and depending on the distance between them. Typically, repulsive forces act at short distances, while attractive forces act at distances greater than some predetermined value.

The method of virtual forces is successfully used for the formal description of the aggregation of a robot swarm [8-10]. However, its realization in systems with real robots creates a series of strict requirements for sensors of each robot, which are difficult and expensive to implement. The most simple robots with autonomous sensors are characterized by a low visibility range, which greatly reduces their ability to distinguish between the other robots in the environment. Errors may occur when determining the relative position of agents in the environment, especially when using infrared sensors. Moreover, mechanical constraints create the so-called saturation effect in robot's actuators, which limits the amplitude of input signals to regulate robot's movement [9]. Despite these limitations of sensors, the method of virtual forces is widely used to control the movements of robots in the simplest systems [6].

2.2. Probabilistic methods

In the probabilistic approach, the behavior of each robot has a random component and is adjusted in the process of robot's interaction with the environment. This type of behavior is often found in the natural world in social insects such as bees or cockroaches. On the basis of observations of social insects behavior, probabilistic algorithms were created to control movements of robots. These algorithms are based on a finite state machine (FSM) with two main states - "go" and "wait" [7] - which correspond to behavioral algorithms. In some cases, the state "go" is divided into two states: when a robot tries to get close to other robots or, conversely, to move away from the neighbors [8]. The decision to change the state can be made completely at random, or based on local signals (e.g. the presence of robots around), or more complex algorithms and signaling mechanisms. A designer of swarm usually chooses the parameters of the finite state machine, such as probability of switching between states; however, recently there have been alternative methods based on automatic techniques for determining parameters [11].

A common characteristic of all probabilistic aggregation algorithms is the presence of unstable aggregates, which robots are constantly entering and leaving. Aggregation dynamics occurs due to changes in the random behavior of the robot while detecting adjacent robots. While disaggregated robots usually move in space randomly, dynamics of aggregated robots is deterministic. However, a random component of the behavior of aggregated robots is often necessary for the formation of small amounts of larger aggregates to avoid situations in which the presence of small aggregates does not allow robots to be attached to larger aggregates. In the studies, where FSM-based algorithms are not used, usually there is no clear distinction between the aggregated and non-aggregated robots. However, the swarm dynamics can be determined by a special metric, such as the average distance between robots; and the randomness of robot's movement can be changed on a continuous scale [12].

2.3. Evolutionary methods

In the case of the evolutionary control method, aggregation dynamics is achieved by using robot controllers, the parameters of which are selected in the process of artificial evolution. Examples of the controllers, using this method, are neural networks. Depending on the algorithm being used, the sensor inputs may include devices able to receive information about the environment, and the outputs of actuators may include devices allowing robots to communicate with each other. Examples of algorithms, used for the method of artificial evolution, are the genetic algorithm or tournament selection [13-14]. Artificial evolution applies the standard paradigm of the natural evolution of population in the wild. This paradigm is based on the concept of adaptation, which determines the ability of a selected population of individuals to adapt to the task.

In contrast to the evolutionary method based on the concept of adaptation, the novelty search method implies a privileged position of those robots, whose behavior differs from the behavioral patterns in previous generations. This method helps to prevent possible negative consequences of the approach based on the concept of adaptation, in which the local maximum of the fitness function in the parameter space may exclude the possibility of studying all the other parts of this space and thus limit the evolutionary process. In [15], the novelty search algorithm is applied to solve the aggregation problem, and the behavioral characteristic of a selected population is based on the metrics, such as the average distance from each robot to the center of masses of all the robots or the total number of aggregates. These parameters were measured several times in the simulation process, and their values (average values of different simulations performed) were added to the vector of the behavioral characteristic used to determine the similarity of different behavioral patterns. The experimental results of the simulation with the use of the novelty search algorithm and the adaptation algorithm have shown that the latter algorithm is better at finding the optimal parameters to perform the task of aggregation for many "generations". However, the novelty search algorithm has proved to be better at an early stage of evolution, and has only slightly reduced its performance in subsequent populations, which generally has given results close to the evolutionary method based on adaptation.

Novelty search methods are based on the determination of similarities between robot’s behavioral patterns, which helps detect the novelty of the behavior of each individual robot. In [16], two methods for similarity determination are presented, which do not depend on a certain level of complexity of a swarm problem and, therefore, can be used without a clearly defined task. Both methods are based on the assessment of the state of a neuro-controller mounted on each robot. This state is defined as a vector of controller’s inputs and outputs for a certain period of time. The first method, called a combined state count, characterizes the behavior model discrediting the possible states of the controller and calculates the occurrence of each state (over time) during the experiment conducted. The second method is called a sampled average state and is based on the calculation of the vector which characterizes the average state of a robot swarm (i.e. the state obtained by averaging the states of all robots) measured over the specified time intervals. Applied to solve the aggregation problem, both methods have shown the results comparable with those when using the method for determining domain-specific similarity of robots.

3. Examples of algorithms of robot aggregation

In free aggregation algorithms, robots have to come together, but there are no preferences regarding the meeting point. Accordingly, the robots can come together with the same probability in any part of the area where they move.

A widely studied probabilistic aggregation algorithm is largely based on observations of the behavior of cockroaches. In a simplified model of cockroach aggregation, these insects move randomly in space and stop at a location depending on the detected number of neighbors. Probability of a stop is a function that depends on the number of robots detected within a certain radius of the robot. The larger the number of robots, the greater the likelihood of stopping. Conversely, a stopped robot can continue to move accidentally in space at any time, possibly even leaving the aggregation group. The less the number of adjacent robots, the greater the probability of transition to the moving state. Accordingly, switching from the waiting mode occurs in two cases: if the robot no longer sees the other robots around it, or if the robot has neighbors, it can switch to the state of free movement with a priori given probability. In such a simple behavioral pattern, free movement of robots in a certain region and their collisions with each other lead to the natural formation of aggregations, as demonstrated by various simulation experiments [17].

In those cases, when a finite state machine with three states [18] (namely, free movement, approach and wait) is used to control robots, the movement state lasts for a fixed period of time after which the robot analyzes the environment. If it finds other robots, then a transition to the approach state occurs, and the robot starts to move to the nearest found robot and then switches to the waiting mode. If in the process of analyzing, the robot does not detect other robots, it just switches to the waiting mode. From the waiting mode the robot again can switch to the state of free movement with a predetermined probability. When using this algorithm, the whole dynamics of aggregation is determined by the probability with which the robot can detect other robots in the process of movement, considering that it is easier to detect larger aggregation clusters than individual robots.

There is another algorithm based on finite state machine [19], where the aggregation behavior is achieved by using four states: search, wait, leave a group and change direction. The robot is looking for other robots and aggregates with them; then there is a transition to the waiting state in such a way that the robot tries to keep a fixed distance from each of its neighbors. It allows creating aggregation groups of almost round shape. As in the standard version of the probabilistic algorithm described earlier, in order to avoid situations where little aggregations inhibit the formation of large aggregates, robots can leave their group at any time with a predetermined probability.

In [20], an approach is shown in which the aggregation is achieved in a simulated robotic system where each robot is equipped with an omnidirectional loudspeaker and a set of microphones. Robots use sound waves to determine their relative position. Basic states of finite state machine, implemented in this case, are approaching, waiting and repulsion. In the state of approach the robot moves in the direction of the loudest sound source, while in the state of repulsion robots move in opposite directions. When the robot detects another robot in the state of approach at close range, it switches to the waiting state in which it holds its current position. The robot switches to the state of repulsion with a predetermined probability, and then returns to the state of approach with another predetermined probability.

In [21], Triani system may be of interest. The robots are provided with a source of light radiation, which can be used to alert the other robots. With the help of sensory perception (robots can detect the presence of other robots nearby and measure the intensity of the light emitted by another robot), each robot generates its own idea of the surrounding environment. At each time point, the robot randomly selects a behavioral pattern from pre-designed templates (including movement to and from other robots as well as switching on and off the light emitter). The probabilities of selection of different behavior models are determined on the basis of robot's perception of the environment. Such genetic algorithm can be adapted to different collective tasks, if we define how exactly the robot will interpret the information received by the sensor, and determine the probabilities of activation of the basic behavior patterns on the basis of various environments. This approach allows us to perform the task of aggregation.

In [22], the probability that the robot will leave its group is determined by robot's orientation to other robots in the group. The robot facing the center of the aggregate is less likely to leave it than the robot oriented to other directions. Following the examples of various natural phenomena, such as molecular composition, the stability of aggregates is described by the energy of relations between the robots, which is a function of the relative position of robots. For simplicity, the authors have studied an aggregate consisting of two robots, where the energy of units equals the energy of relations between two robots.

In [23], minimalist aggregation algorithms are used, in which the signal received at the sensor input is limited to one binary variable that determines whether there is another robot in sight. The study [24] discusses the algorithm according to which the robot is commanded to move back along a curved path if there is no robot in sight, or otherwise to turn on the spot. This simple mechanism allows achieving emergent aggregation in case robot's sensors can work at a quite long distance. However, due to the lack of the behavioral pattern similar to the free movement, aggregation can not be guaranteed in case robots are initially at a greater distance from each other than the distance of sensor sensitivity.

In [25], a neural network, the parameters of which are determined by the genetic algorithm, controls robots, equipped with microphones, sensors, wheels and a speaker. Two types of collective behavior were studied: static and dynamic aggregation. Both types of aggregates are shown in Figure 3.

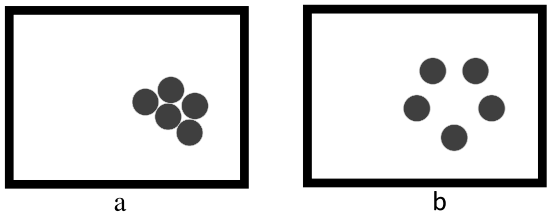

Figure 3. A formed aggregate for different types of behavior of agents: (a) static aggregation; (b) dynamic aggregation

The first type determines the formation of the static and compact aggregates, which, however, can not be scaled, as a large number of robots in one region tend to form multiple disjoint clusters. In the case of dynamic aggregation, formed units are less compact, but they continue to move through the area, and when there is a large number of robots in the area, different units tend to merge and form a single unit. Accordingly, dynamic aggregation has a greater scalability.

4. Methods for estimating robot aggregation efficiency

Metrics is usually based either on the determination of individual groups of robots, forming aggregates, or on the spatial distribution of robots in a certain area. The former case requires a formal definition of aggregates. Most often an aggregate is a group of robots such that for any pair of robots in the group, there is a chain connecting them and consisting of robots located at the maximum allowable distance from each other. The maximum allowable distance is usually chosen on the basis of robot's parameters: working distance of communication modules and sensor sensitivity. In studies, where the robots are controlled by a finite state machine, the concept of aggregate can be defined as a group of robots, whose controllers are in the "waiting" state.

Therefore, depending on the method for robot control, an aggregation efficiency index can be calculated as the ratio of the number of robots constituting the largest aggregate to the total number of robots [18], or as the average size of aggregates in the area [20]. A more thorough analysis of aggregation dynamics can be conducted by observing the distribution of the robots located in aggregates of different sizes. In tasks, where the objective is the aggregation of robots in a certain area, the standard metric is the percentage of robots that are in the target area or at a certain distance from the place of aggregation [20, 26]. Figure 4a shows robots, some of which formed an aggregate inside the circle, and this part accounts for 45%.

The second type of metric involves determining the location of all robots in the area and their relative position. In [27], the sum of distances between each pair of robots is used to assess the aggregation (Figure 4b). In [23], the value of the average distance from robots to the center of mass of the swarm (Figure 4c) is used. In the study [24], the so-called "second moment of robots" was used, which is calculated as the sum of the squares of the distances from each robot to the center of mass. In [28], the so-called " bounding box ratio" was used, defined as the ratio of the surface of the smallest rectangle containing all the robots over the total surface of the arena (Figure 4d).

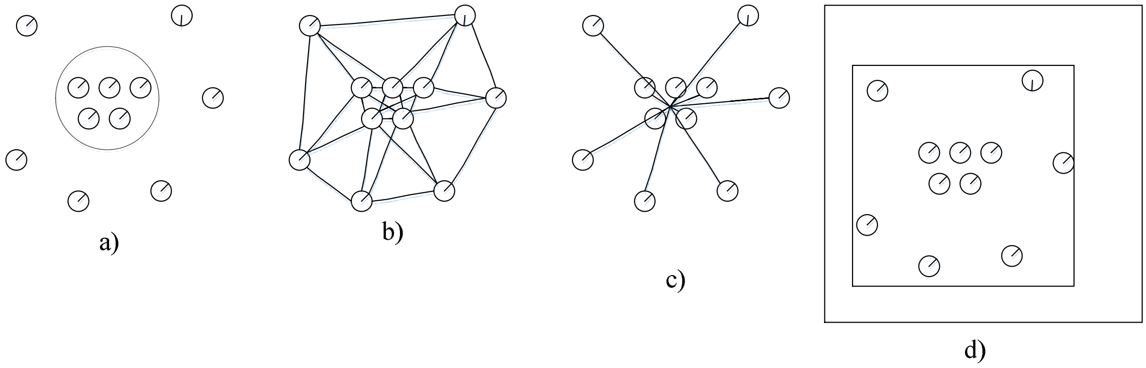

Figure 4. Graphic illustrations of different kinds of robot aggregation metrics: a) assessment of aggregates of robots within a given circle; b) the sum of the distances between all pairs of robots; c) the average distance from robots to the center of mass of the swarm; d) "bounding box ratio"

Another important feature of the robot swarm aggregation is time estimation. In [29], the rate at which a swarm performs aggregation is used as time estimation. Usually, such metric is calculated as the time passed from the stage where the robots are arranged in a random way and until the moment when robots form an aggregate with set parameters.

5. Software and an algorithmic model of the targeted robot swarm reconfiguration

As a part of preliminary research into swarm robotics, we have developed principles and a conceptual model of reconfiguration of the spatial position of a group of robots. This takes into account restrictions on the size of homogeneous robots and occupied area of their initial position; spatial characteristics; density of robots' location; as well as ways of defining the target coordinates of robots' position in a new spatial configuration [30, 31]. The developed mathematical model of reconfiguration of the robot swarm is aimed at the control and navigation of autonomous homogeneous mobile robots involved in the formation of a given convex surface. The computational complexity of the proposed model is quadratic, as the number of targeted points is limited to the total number of robots (active ones, involved in the reconfiguration, and passive ones). The complexity does not depend on the time and the length of robots' movements (Figure 5). During analytical modeling of reconfiguration of the swarm, with 10 to 10.000 robots participating in the formation of surfaces of convex figures, we have estimated the number of collisions (Figure 6), occurring at rectilinear motion of robots to their targeted points, assuming that robots start simultaneously.

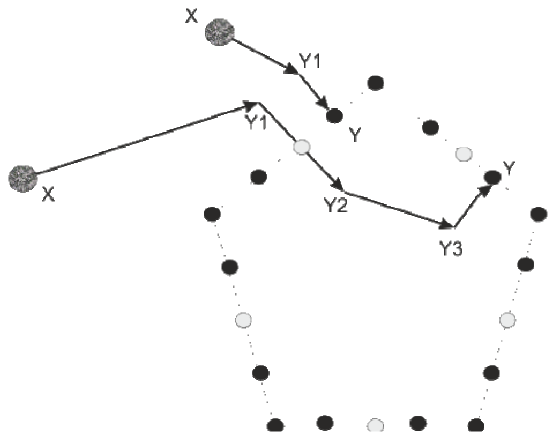

Figure 5. The example of robot's trajectories: crossing and non-crossing a formed surface

As a result of the simulation, it was revealed that:

(a) the number of pairs of robots with paths, leading to collisions, does not exceed 2% of the total number of pairs of active robots;

(b) the total number of calculations in resolving conflicts increases less than the logarithm of the total number of pairs of the robots and, therefore, makes a negligible contribution to the overall complexity of calculations.

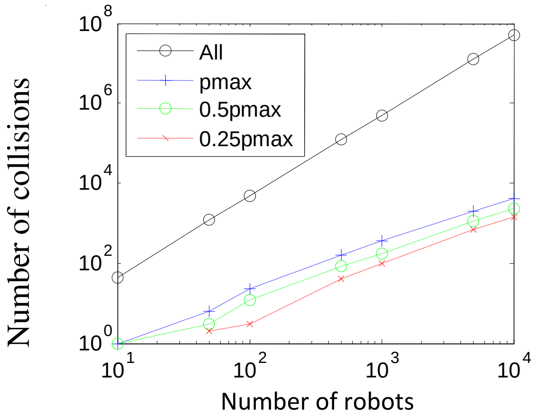

Figure 6. Evaluation of the number of collisions when the number of robots increases and at different density pmax of robots' location on the surface

6. Conclusion

This article discusses the problems of aggregation of a robot swarm that are mainly related to the simplest computational and embedded propulsion devices as well as limited resources of a swarm of homogeneous robots. An overview of the main methods for solving the problem of aggregation (the method of virtual forces, probabilistic and evolutionary methods) has shown that the choice of the method, first of all, depends on the computational and network resources of robots. We have given examples of different aggregation algorithms as well as limitations on design solutions due to technical requirements for the implementation of these algorithms.

To evaluate the effectiveness of robot aggregation, spatial and temporal assessments are mainly used. The choice of the metric depends on the parameters of aggregates, which should be achieved, as well as on the aggregation method that is technically possible to implement to control a given robot swarm. As a result of the analysis, it can be concluded that systems using neural networks to control a robot swarm are the most promising in terms of further improvement of aggregation algorithms. However, their implementation requires large computational onboard resources. The developed mathematical and software support for control and navigation of a swarm of autonomous homogeneous mobile robots has been tested in forming a given convex surface. Further research will be aimed at solving the problems of interaction of a swarm of robots in constructing more complex shapes taking into account a larger number of physical parameters.

7. Acknowledgments

The study is supported by RFBR (grant no 16-29-04101-ofi_m) and the fundamental program of the Presidium of the Russian Academy of Sciences I.31P "Actual problems of robotics".

References

[1] Dudek G, Jenkin M, Milios E, Wilkes D. 1993 A taxonomy for swarm robots. *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems '93, IROS'93.* vol. **1**. pp. 441–447

[2] Gorodetsky V I, Samoylov V V, Trotsky D V 2015 Artificial intelligence, basic ontology of the collective behavior of autonomous agents and its extension. *Izvestiya RAS. Theory and control of systems* **5** 102–121

[3] Kalyaev A I, Kalyaev I A 2015 Decentralized control method for a group of robots in tasks flow. *Robotics and technical cybernetics* **1(6)** 26–35

[4] Pavlovsky V E, Pavlovsky V V 2013 Scalable robot control system robocon-1. *Information-measuring and operating systems* **11(4)** 80–92

[5] Ishikawa T, Locsei J T, Pedley T J 2008 Development of coherent structures in concentrated suspensions of swimming model micro-organisms. *Journal of Fluid Mechanics*. **615** 401–431
[6] Fetecau R C 2011.Collective behavior of biological aggregations in two dimensions: anon local kinetic model. *Math. Models Methods Appl. Sci*. **21(7)** 1539–1569
[7] Kernbach S , Thenius R, Kernbach O, Schmickl T 2009 Re-embodiment of honeybee aggregation behavior in an artificial micro-robotics system . *Adapt. Behav*. **17(3)** 237–259
[8] Bayındır L 2012 A probabilistic geometric model of self-organized aggregation in swarm robotic systems: Ph.D. thesis. Middle East Technical University.
[9] Vanualailai J, Sharma B 2010 A Lagrangian-based swarming behavior in the absence of obstacles. *Workshop on Mathematical Control Theory, Kobe University*. pp. 8–10
[10] Hackett-Jones E J, Landman K A, Fellner K 2012 Aggregation patterns from nonlocal interactions: discrete stochastic and continuum modeling. *Phys. Rev*. **85(4)** 041912
[11] Francesca G, Brambilla M, Brutschy A, Trianni V, Birattari M 2014 Auto Mo De: a novel approach to the automatic design of control software for robot swarms. *Swarm Intell*. **8(2)** 89–112
[12] Burger M, Haškovec J, Wolfram M T 2013 Individual based and mean-field modeling of direct aggregation. *Physica D: Nonlinear Phenom*. **260** 145–158
[13] Blickle T, Thiele L 1995 A Comparison of Selection Schemes used in Genetic Algorithm, 2 Edition . TIK-Report. 67 p.
[14] Batishchev D I, Isayev S A 1997 Optimization of multi functions using genetic algorithms. I*nteruniversity collection of scientific papers "High technologies in engineering, medicine and education"*. (Voronezh: VGTU) pp. 4–17
[15] Gomes J, Urbano P, Christensen A L 2013 Evolution of swarm robotics systems with novelty search. *Swarm Intell*. **7(2-3)** 115–144
[16] Gomes J, Christensen A L 2013 Generic behavior similarity measures for evolutionary swarm robotics. *Proceeding of the Fifteenth Annual Conference on Geneticand Evolutionary Computation, ACM, NewYork*. pp. 199–206
[17] Correll N, Martinoli 2007 A Modeling Self-Organized Aggregation in a Swarm of Miniature Robots. *IEEE 2007 International Conference on Robotics and Automation Workshop on Collective Behaviors inspired by Biological and Biochemical Systems*. pp. 1
[18] Schmickl T, Möslinger C, Crailsheim K 2007 Collective perception in a robot swarm. *Swarm Robotics*, (Springer, Berlin) pp.144–157
[19] Bayındır L 2012 A probabilistic geometric model of self-organized aggregation in swarm robotic systems: Ph.D. thesis. Middle East Technical University.
[20] Soysal O, Şahin E 2005 Probabilistic aggregation strategies in swarm robotic systems. *Proceedings 2005 IEEE Swarm Intelligence Symposium*, 2005. SIS 2005, IEEE Press, Piscataway. pp. 325–332
[21] Trianni V, Labella T H, Groß R, Şahin E, Dorigo M, Deneubourg J L 2002 Modeling Pattern Formation in a Swarm of Self-assembling Robots. *Technical Report, IRIDIA*, Université Librede Bruxelles
[22] Mermoud G, Brugger J, Martinoli 2009 A Towards multi-level modeling of self-assembling intelligent micro-systems. *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*. **1** 89–96
[23] Gauci M, Chen J, Dodd T J, Groß R 2014 Evolving aggregation behaviors in multi-robot systems with binary sensors. *Distributed Autonomous Robotic Systems*. pp. 355–367
[24] Gauci M, Chen J, Li W, Dodd T J, Groß R 2014 Self-organized aggregation without computation. *Int. J. Robot. Res*
[25] Trianni V, Groß R, Labella T H, Şahin E, Dorigo M 2003 Evolving aggregation behaviors in a swarm of robots. *Advances in Artificial Life*, Springer, Berlin. pp. 865–874

[26] Garnier S, Jost C, Jeanson R, Gautrais J, Asadpour M, Caprari G, Theraulaz G 2005 Aggregation behavior as a source of collective decision in a group of cockroach-like-robots. *Advances in Artificial Life*, Springer, Berlin. pp.169–178
[27] Soysal O, Şahin E 2005 Probabilistic aggregation strategies in swarm robotic systems. *Proceedings 2005 IEEE Swarm Intelligence Symposium*. SIS 2005, IEEE Press, Piscataway. pp. 325–332
[28] Fatès N 2010 Solving the decentralized gathering problem with a reaction – diffusion-chemotaxis scheme. *Swarm Intell* **4(2)** 91–115
[29] Arvin F, Turgut A E, Bellotto N, Yue S 2014 Comparison of different cue-based swarm aggregation strategies. *Advances in Swarm Intelligence* pp.1–8
[30] Vatamaniuk I V, Panina G U, Ronzhin A L 2015 Reconfiguration of the spatial position of swarm robots. *Managing large systems* **58** 285–305
[31] Vatamaniuk I V, Panina G U, Ronzhin A L 2015 Simulation of the path of the robot systems with the reconfiguration of the spatial position of the swarm. *Robotics and Technical Cybernetics*. **3(8)** 52–57