Robot Cooperation and Swarm Intelligence

Nicole El Zoghby, Valeria Loscrì, Enrico Natalizio, Véronique Cherfaoui

Chapter 8

Robot Cooperation and Swarm Intelligence

Nicole El Zoghby[1], Valeria Loscrí[2], Enrico Natalizio[1] and Véronique Cherfaoui[1]

[1] *Heudiasyc Lab., UMR CNRS 7253, Compiègne, France,*
[2]*DEIS, Universitá della Calabria, Italy*

Abstract. *This chapter is devoted to illustrate and characterize the relationship between Swarm Intelligence and cooperation among robots. Individuals with very limited computational capabilities are able to carry out very complex tasks when they can work together. From a methodological point of view, Swarm Intelligence is a set of heuristic solutions inspired by animal swarm behaviors and capable to offer empirical solutions to many computationally hard problems pertaining to several disciplines. In this chapter, we will try to outline the main research directions in Swarm Intelligence implementation within a robot network through the cooperation among the robots. The latter topic will be presented along with its advantages, issues and challenges. The convergence of robot cooperation and Swarm Intelligence is leading towards a new discipline, called Swarm Robotics. In this chapter, we will introduce this new field of study, its most relevant works and its main research directions.*

8.1 Introduction

Swarm Intelligence is a powerful concept that pivots around the cooperation among the members of a community towards a common goal. From a methodological point of view, Swarm Intelligence is a set of heuristic solutions inspired by animal swarm behaviors and capable to offer empirical solutions to many computationally hard problems pertaining to several disciplines. In this chapter, we will focus on the relationship between Swarm Intelligence and cooperation among robots. We will start presenting Swarm Intelligence, its biological principles and the mechanisms that underlie collective behaviors, the most important Swarm Intelligence heuristics and their classical applications. After this introduction, we will survey the state of the art on cooperation among robots in order to present advantages, issues and challenges of this research field. Finally, we will try to bridge swarm intelligence and cooperation among robots towards the description of a very recent discipline: Swarm Robotics, of which we will show existing taxonomies and applications.

8.2 Swarm Intelligence

The word “swarm” evokes the image of a large number of small insects where each individual performs a simple task, but whose action produces a complex behavior as a whole [39]. The emergence of such a complex behavior extends beyond the swarms. Complex social structures are similar in bigger animals as well as other types of insects. Some examples are colonies of ants and termites, flocks of birds, schools of fish, colonies of bacteria, or even herds of terrestrial animals. Swarms are defined as collections of many simple individuals that interact with both other individuals and the surrounding environment [60]. The combination of their simple or microscopic behaviors causes considerably more complex and macroscopic actions, which enable the whole system to achieve remarkable results as a whole.

The term Swarm Intelligence was introduced for the first time by Gerardo Beni and Jing Wang in 1989 [9]. The Swarm Intelligence studies the collective behavior of systems composed of many individuals who interact locally with each other and with the surrounding environment, using forms of decentralized and self-organized control to achieve their objectives.

Therefore, the Swarm Intelligence provides a new framework for the design and implementation of systems consisting of many agents that are able to cooperate in order to solve highly complex problems. The potential benefits of such approach are several:

- robustness: the failure of individual elements does not degrade significantly the performance of the entire system;
- simplicity: the individual behavior is simple but still it allows to reduce the complexity of individuals;

- scalability: the control mechanisms used do not depend on the agents number within a swarm.

8.2.1 *Biological principles for swarm intelligence*

The Swarm Intelligence comes from biological insights related to the enormous capabilities that social insects possess to solve daily-life problems within their colonies. Insects belong to two families: the most ancient is the "Isoptera" (termites) entirely social, the second is the Hymenoptera, which includes ants, wasps, bees and also presents social structures. These insects, even if they belong to two distinct families, which are very far from each other in evolution, share three important characteristics:

(1) individuals of the same species cooperate in the care for the young ones;
(2) individuals share the reproductive division of labor, sterile individuals work for the benefit of fertile individuals;
(3) an overlap of at least two generations exists, so that the offspring can help parents to carry out the tasks necessary to the life of the colony.

On the basis of these three characteristics, the entomologists distinguish the true sociability, or *eusociality*, from behaviors that do not present all the three listed characteristics, and for this reason are defined pre-social. In what follows, we will analyze the main biological principles that govern the organization in the colony of insects, that is the mechanisms which give rise to complex collective behavior of social insects, the concept of stigmergy and the theory of self-organization in biological systems. After this, we will introduce the main metaheuristics of Swarm Intelligence.

Ants, wasps and termites are able to build sophisticated nests in cooperation, even if none of the individuals have an exact plan of how to proceed and no coordinator exists [11]. Another example is taken from the behavior of ants and bees during the search for food. The ants employ a strategy of indirect communication through the release of a chemical substance, called *pheromone*, in order to identify the shortest paths between nest and food sources (Fig. 8.1);

Bees are very efficient in finding the richest sources of food by using some explorers that communicate the information about newly-found sources of food by a *waggle dance* (so-called for the vibrations generated from the abdomen of the bees while flying) (Fig. 8.2).

An African species of termites, the *Macrotermes bellicosus*, builds mounds that can reach 30 m of diameter and 6 m of height [70] (Fig. 8.3).

Fig. 8.1 A group of ants following a pheromone trail.

Fig. 8.2 Bees' waggle dance.

These "skyscrapers" are the result of the biological work of millions of tiny individuals (1-2 mm long), which are completely blind. Even more fascinating than the size of these mounds is their internal structure.

The nests of the species *Apicotermes lamani* are probably one of the most complex structures ever built in the animal kingdom. The hive is a highly complex structure, high around 20 to 40 cm. On the outer surface there are a series of micro-structures which provide the air-conditioning and gas exchange with the external environment, while inside the hive, rooms are concatenated to each other by means of helical ramps. These spiral ramps are born from the twisting and welding of consecutive floors. There

Fig. 8.3 Mound built by Macrotermes bellicosus.

are different stairs on each floor and some of them cross the entire nest. Hence, even the most distant rooms are connected by these shortcuts. The complexity of these structures and the collective behavior do not reflect the relative simplicity of the individual behavior of a single insect. Of course, insects are complex entities, able to adapt their behavior according to many sensor inputs. However, the complexity of a single insect in terms of cognitive or communicative skills may be high according to an absolute perception, but it is not sufficient to control a large system and explain the complexity of all the behaviors that govern a colony [73]. In essence, a single insect is not able to find an efficient solution by itself to a problem of the colony, while the group to which it belongs manages to find, as a whole, a solution very easily. Behind this organization with no boss, there are several hidden mechanisms that enable groups of insects, whose members have to deal only with partial information about their surroundings, to face random situations and find solutions to complex problems.

8.2.1.1 *Mechanisms for collective behavior*

The study of the mechanisms that underlie the collective behavior of insects started more than a century ago. Initially, in order to justify the complexity of these behaviors, it was assumed that the individual insects possessed a minimum knowledge of the overall structure that needed to be produced and that, accordingly, they were able to make the appropriate decisions. In other words, it was thought that there was a causal relationship between the complexity of decisions, the patterns observed at the level of the colony, and the behavioral and cognitive complexity that was supposed to be required at the individual level to make these decisions and models. Therefore, it was assumed that the model which governed those companies was hierarchical and centralized. However, most of the research done in recent years, has revealed a completely different organization. Today, we know that the individual insects do not require representation, schema, or explicit knowledge of global structure they produce. A single insect is not able to evaluate the global situation to centralize the information about the state of the entire colony and later to control the tasks that need to be made by other workers. There is no supervisor in these colonies. A colony of social insects is quite similar to a decentralized system composed of autonomous units that are distributed in the environment, and could be described by simple probabilistic cause-effect behaviors [70].

The principles that underlie the interactions between insects are carried out through local information of a global model. Each insect follows a set of few rules. For example, the ants can perform, on average, approximately 20 different elementary behaviors. At the colony level, the organization emerges from the interactions that occur among individuals who show these simple behaviors. These interactions ensure the propagation of information within the colony and also organize the activity of each individual. With these sophisticated networks of interactions, social insects can solve a wide range of problems and respond to external challenges in a very flexible and robust way.

8.2.1.2 *Stigmergy*

The first scientific explanation of activities organization of social insects was given 40 years ago by the French biologist Pierre-Paul Grasse, who introduced the concept of *stigmergy* to explain some of his observations on the behavior of termites in the construction of termite mounds [70]. The self-organization of social insects requires interactions among themselves. This interaction may be direct or indirect. Direct interactions are obvious: sight contact and/or chemical; whereas indirect interactions are more subtle: two individuals interact indirectly when one of them modifies the environment and the other responds accordingly to the new environment [29]. This interaction is an example of stigmergy. This term, which comes from the greek words “stigma” that means sign and “ergon” that means work (led by stimuli), is a form of indirect communication in which each individual acts on the surrounding environment and other individuals that detect some changes in the environment react to the stimulus. Since the overhead of communications does not increase when the size of the group increases, the stigmergy allows great scalability. It should be noted that the stigmergy in itself does not explain how the communication takes place indirectly, but only provides a general mechanism that relates the behavior of the individual to the level of the colony. In his studies, Grasse showed that the coordination and regulation of activities of a colony did not depend on the workers, but they were mainly driven by the nest. In other words, the information coming from the local environment and the progress of the work can drive the individual activity. Each time that a worker performs an action, this action results in a modification of the local configuration. The new configuration will affect other subsequent actions and behaviors of the other workers in the colony. This process leads to an almost perfect coordination of collective labor and may give us the impression that the colony is following a definite plan.

A good example of stigmergic behavior is the search for food of the ants. The ants communicate with each other through the use of pheromones, chemicals that attract other ants. When an ant finds a source of food, it quickly returns to the nest and releases a pheromone trail. This trail will then lead the others from the nest to the food source. While returning to the nest, the ants release their pheromones along the path, thus reinforcing the trail. The formation of the trail therefore derives from a positive feedback: the greater the number of ants that follow the path, the more the path will become attractive and appealing. Of course, the trail will disappear after a while, if the reinforcement is too weak, and this may happen when the food source is exhausted. The interesting thing is that this system of maintenance of the trail is not only a mechanism used to quickly gather a large number of purveyors around the source of food, but it also allows the colony to take efficient decisions such as the choice of the shortest path that leads to the source of food. From this description, other properties characterizing the stigmergy have emerged. In fact, stigmergy affects the overall behavior of the population by two key elements of self-organization that have already been implicitly introduced: the positive feedback and its dual, the negative feedback. The positive feedback is the phenomenon by which the marks on the environment, deployed by individuals, encourage other members to release additional marks in the same place, making the population converge toward the reinforcement of the solution. The negative feedback is the opposite phenomenon, i.e., areas marked weakly tend to be overlooked by the individuals of the population, leading to the impoverishment of the solution. Another property is that the emerged stigmergy is a form of communication limited in time, i.e., the changes on the environment vanish over some time. For example, the pheromone released by ants evaporates with time, therefore, it is necessary to operate the reinforcement of the path to keep it alive.

8.2.1.3 *Principles of self-organization*

Self-organization is a set of dynamic mechanisms by means of which the structure of a system appears at the global level as a set of interactions of its components at the local level. It has four basic components: the positive feedback that derives from the execution of simple behavioral rules that support the creation of structures; the negative feedback that counteracts the positive feedback and leads to the stabilization of the collective behavior; the amplification of fluctuations through the positive feedback; direct and multiple interactions or stigmergic interactions among individuals to produce deterministic results and the appearance of large-scale durable structures. In addition to the components described so far, the self-organization is also characterized by some key properties:

(1) Self-organizing systems are dynamic. As mentioned previously, the production of structures, as well as their persistence, requires constant interactions among the permanent members of the colony and their surroundings.
(2) Self-organizing systems exhibit emerging properties. They show more complex properties of the single contribution of each individual. These properties arise from a combination of non-linear interactions between the members of the colony.
(3) Together with emergent properties, nonlinear interactions lead self-organized systems to bifurcations. A bifurcation is the appearance of new stable solutions when there is a change of the parameters of the system. This corresponds to a qualitative change in the collective behavior.
(4) Finally, the self-organizing systems can be multi-stable. Multi-stability means that, for a given set of parameters, the system is able to achieve different stable states that depend on the initial conditions and random fluctuations.

8.2.1.4 *Collective behaviors*

The processes of self-organization described above may produce a wide variety of collective behaviors that are intended for the resolution of a given problem. In their studies, Camazine et al. [15] have proposed to categorize social behaviors of a colony of insects, according to four types of tasks: individual, group, team and shared tasks. Following this categorization, each global task in the colony (e.g., nest building) can be hierarchically split into sub-tasks belonging to any of the mentioned types. This method can be seen as the decomposition of a problem into simpler tasks, which are essential for the resolution of the problem. Another way to characterize social insects collective behavior consists in defining specific functions to describe insects' tasks. It is possible to identify four main categories of functions: coordination, cooperation, evaluation and collaboration. These categories are not mutually exclusive and they contribute together to the fulfillment of the various collective tasks of the colony. Below, we provide a first definition of each of these functions and subsequently explain their respective roles in some examples of collective behavior of social insects.

Coordination. Coordination is the appropriate organization in space and in time of all the tasks necessary to solve a specific problem. This leads to specific spatio-temporal distributions of individuals, of their activities and/or outcomes of their activities, in order to achieve a certain goal. For example, the coordination occurs in the organization of the movement of swarms of bees and locusts. In this case, the interactions between individuals generate synchronized movements (temporal organization) and oriented (spatial organization) of individuals towards a specific goal.

Cooperation. The cooperation is a phenomenon that occurs when a task can not be performed by a single individual but requires a set of them. Therefore, individuals must combine their efforts in order to successfully solve a problem that goes far beyond their individual capabilities. For example, cooperation is required from ants to remove a long wooden stick that obstructs the entrance of their nest. In this situation, the ants combine their efforts to pull the stick away from the hole. Some ants raise the stick while others put their head inside the entrance in order to avoid that the stick can fall back inside. In the end, the combined efforts lead the group to remove the stick. The function of cooperation represents the mechanisms that go beyond the limitations of individuals.

Evaluation. The term evaluation refers to the mechanisms that occur when a colony is faced with several opportunities. These mechanisms are the result of a collective choice of at least one of opportunity. For example, when the ants Lasius niger find different food sources, or different routes that lead to one food source, they generally choose only one among the various possibilities. The evaluation is usually guided by competition between the chemical trails that underlie each possibility. In most cases, the ants will end up choosing the richest food source reach via the shortest path.

Collaboration. With the term collaboration we mean the various activities which are carried out simultaneously by a group of specialized individuals. This specialization stems from both a behavioral and morphological differentiation of individuals. The most striking expression of this division of labor is the existence of castes. For example, the workers in a colony of ants that shred the leaves may belong to four different castes and their size is closely related to the tasks they perform. Only the workers who have a head size greater than 1.6 mm are able to shred the leaves, which are used to grow fungi that are the primary source of food for these colonies. Conversely, only the worker ants that have a smaller head size of about 0.5 mm are able to take care of fungi cultivation.

Most of the organization of the collective behavior of social insects can be seen as the combination of the four functions of coordination, cooperation, evaluation and collaboration. Each of these functions emerge at the collective level by the continuous interactions among insects. Together, the four functions of organization produce solutions to the problems of the colony and could give the impression that the colony act as a whole, planning the work to achieve the colony's goals.

8.2.2 *Main meta-heuristics of swarm intelligence*

The main disadvantage present in the algorithms that rely on constructive methods or iterative improvements is that they generate only a limited number of solutions [26]. In order to try to solve these problems, it is possible to use metaheuristics. The term metaheuristic comes from two Greek words: heuristic derives from the verb heuriskein, i.e., search, while the suffix meta means beyond to a higher level [11]. A metaheuristic is a heuristic method, i.e., a general algorithm or a set of algorithmic concepts applicable to a diverse number of optimization problems that, with slight modifications, can be adapted to describe a specific problem. Below we will explain the most popular Swarm intelligence metaheuristics.

8.2.2.1 *Ant colony optimization*

Ant colony optimization (ACO) is a metaheuristic within which a colony of artificial ants cooperate in order to obtain good solutions to difficult discrete optimization problems. Cooperation is the key component of ACO algorithms. In fact these algorithms allocate computational resources to a number of relatively simple agents (artificial ants) that communicate indirectly through the stigmergy [26]. ACO algorithms can be used to solve combinatorial optimization problems, both static and dynamic. The static problems are those in which the characteristics of the problem are known a priori, when the problem is defined, and do not change during the resolution of the problem. A classic example is the TSP (traveling salesman problem), where cities and the distances among cities are part of the definition of problem and do not change during the execution time of the algorithm. The dynamic problems are defined as functions of a number of quantities, the values of which are set by the dynamics of the problem. The instance of the problem therefore changes at the execution time and, therefore, the optimization algorithm must be able to adapt to a changing environment. An example of a dynamic problem can be identified in the routing problems in networks, in which the data traffic and network topology may change very frequently.

Description.

An artificial ant is a constructive and stochastic process that incrementally builds a solution by adding appropriate components to the partial solution. Therefore, the ACO metaheuristic can be applied to some combinatorial optimization problems, for this reason, it can be defined as a constructive heuristic. If we consider an optimization problem (S, f, Ω), where S is a set of possible solutions, f is the objective function that assigns a cost to each candidate solution s belonging to the set S, and $\Omega(t)$ is a set of conditions to satisfy. The parameter t indicates that the objective function and the conditions imposed are both functions of time. The goal is to find a globally optimal solution s^*, that is a possible solution at minimal cost. From this, the ants develop artificial solutions of performing random paths on a fully connected graph $G_c = (C, L)$, called construction graph with L connections. In many applications, the ants construct feasible solutions, even if, in some cases, it is necessary to allow them to construct improbable solutions. For example, a component $c_i \in C$ and a connection $l_i \in L$ may be associated with a trace of pheromone τ (τ_i if the trace is associated only to a component, τ_{ij} if it is associated also to a connection), and a heuristic value μ (μ_i and μ_{ij}, respectively). The pheromone trace encodes a long-term memory of the entire process, and it is updated by the ants themselves. Instead, the heuristic value, often called heuristic information, is an input given *a priori* information and based on the current problem instance, or otherwise it is a run-time input provided by the ants through different sources. In many cases, μ represents a cost, or at least an estimate of the cost of adding components or connections to the solution under certain conditions. The heuristic values are used into ants in order to make decisions on how to navigate the probabilistic graph. It is important to note that the ants act simultaneously and independently of each other. Even if it is very complex for each single ant to find a solution to the proposed problem, good quality solutions can still be obtained as a result of a collective cooperation among the ants. All this is obtained through an indirect communication represented by the traces of pheromone. In more specific terms it can be said that an ACO algorithm can be represented through a combination of three different procedures: *Construct Ants Solutions*, *Update Pheromones*, and *Daemon Actions*.

Construct Ants Solutions manages a colony of ants concurrently and asynchronously in the search for adjacent states of a given problem, moving through neighboring nodes of the problem graph G_c. They move by applying a stochastic and local policy decision, which is based on the use of pheromone trace and heuristic information. In this way, the ants are able to search for the solution to the optimization problem in an incremental manner. As soon as an ant has found a solution, or while the solution is to be added, the ant evaluates the (partial) solution that will be used during the following procedure *Update Pheromones* to determine how much pheromone must be deposited.

Update Pheromones is the process by which the traces of pheromone are updated. The importance of the path can both increase, by depositing pheromone on the components and the connections use, or decrease due to the evaporation of the pheromone itself. From the practical point of view, the deposit of new pheromone increases the likelihood that these components/connections are used again by ants future. Otherwise, the evaporation of the pheromone realizes a form of forgetfulness, in order to avoid a too rapid convergence of the algorithm towards a sub-optimal region. Therefore, it promotes the exploration of a new area within the search space.

Finally there is *Daemon Actions*, a procedure used to implement centralized actions that cannot be performed by the individual ants. Examples of such procedure are the activation of a local optimization procedure, or the collection of global information that can be used to decide whether it is useful or not to deposit additional pheromone, in order to influence the research process considering a non-local perspective. As a practical example, the daemon can observe the path found by each ant in the colony and select one or more ants, which can thereafter afford to deposit of additional pheromone on components or connections they have been using.

Applications.
ACO metaheuristic finds application in most of the cases located problems in combinatorial optimization. The most classical problem of this type is the the Traveling Salesman Problem (TSP) [59]. It is a problem of a salesman that, starting from her home city, is looking for the possible shortest paths within a given set of customer cities, visiting each city at least once before returning home. The TSP can be represented by a weighted and complete graph. TSP is a problem based on finding the shortest Hamiltonian path length in the graph, where the Hamiltonian path is a closed path that passes through each node of the graph exactly once. The trace of pheromone τ_{ij} in the TSP refers to the desire to visit node j directly after passing through the node i. The heuristic information μ_{ij} is typically inversely proportional to the distance between two cities i and j, $\mu_{ij} = 1/d_{ij}$. Each ant is initially placed on a random city and at each step it adds in its path a city that was not yet visited, The construction of the solution ends when all cities have been visited. Therefore, TSP is a NP-hard combinatorial optimization problem that attracted a lot of research. The TSP has a central role in ACO problems, in fact it has been used for Ant System, the first ACO algorithm.

8.2.2.2 *Particle Swarm Optimization*

The Particle Swarm Optimization (PSO) algorithm is defined as a population-based algorithm characterized by a set of candidate solutions, where each solution is “a particle” moving in a search space [37]. The PSO was introduced for the first time in 1995 by Kennedy and Eberhart, whose objective was to introduce a new methodology in the computational intelligence [46]. This technique uses simple analogies with social interactions, rather than purely individual cognitive skills [67]. The roots of this metaheuristic bring back to ideas and techniques investigated for computer graphics and social psychology. In the field of computer graphics, the first work that proposed a PSO algorithm can be traced in the work of Reeves (1983), who proposed particle systems to model dynamic objects that could not be easily represented by polygons and surfaces (fire, smoke, water and clouds). The social psychology, in particular the theory of dynamic social impact, was another source of inspiration for the development of the first PSO algorithm. The principle that governs the movement of a particle in a search space of a problem can also be compared with a model of human social behavior, in which individuals adapt their behaviors to satisfy those of their peers. In PSO, simple entities, called particles, are located in an area of research of a specific problem or function, and evaluate a fitness function in their current location. This fitness function is based on a more important and generic function that characterizes the behavior of the entire swarm, called the objective function. Each particle can determine the displacement within the search space by combining some aspects of the its best position history with those of one or more members of the swarm through some random perturbations. Next iteration takes place when there is an update of all the particles. Each individual particle in the swarm is characterized by three-dimensional vectors D, where D represents the size of the search space and it is associated with the following values:

- current position, x_i,
- best past position, p_i,
- particle speed, v_i.

The current position x_i can be seen as a point in the space. At each iteration of the algorithm, the current position is considered as a solution to the problem. If the location is the best obtained up to that point, its coordinates are loaded into the vector p_i. The value of the best result can be loaded in a variable named p_{best_i} in order to perform a comparison with the results that will be obtained from subsequent iterations. The goal is to preserve the best position achieved and load it in In the PSO, the single particle itself has no power to resolve the problem: the progress occurs only when a particle interacts with the other. In the process of particle swarm optimization, the velocity of each particle is updated iteratively in such a way that the particles oscillate stochastically around the value of p_i. PSO attracted a lot of attention and several different versions have been presented over the years, interested readers can refer to [30] for a survey.

Population dynamics: Algorithm

(1) Initialize an array of particles with random positions and velocities on a D-dimensional area of research.
(2) Loop
(3) For each particle, evaluate the fitness function.
(4) Compare the fitness function of the particle with its value p_{best_i}, if the current value is better than the value in variable p_{best_i}, assign the current value to p_{best_i} and put p_i equal to the current position x_i.
(5) Individuate the particle in the population that has obtained the best

fitness function and assign its position to a variable p_g.

(6) Change the speed and the position of the particle in accordance with the following equations:

$$\vec{v_i} \leftarrow \vec{v_i} + \vec{U}(0, \phi_1) \otimes (\vec{p_i} - \vec{x_i}) + \vec{U}(0, \phi_2) \otimes (\vec{p_g} - \vec{x_i}) \tag{8.1}$$

$$\vec{x_i} \leftarrow \vec{x_i} + \vec{v_i} \tag{8.2}$$

where $\vec{U}(0, \phi_i)$ is a random matrix of numbers uniformly distributed between $[0,\phi_i]$, which is randomly generated at each iteration and for each particle.

(7) If the criterion is met (usually a threshold value of the fitness function or a maximum number of iterations), exit the loop.

Population dynamics: Parameters definition

A small, but not insignificant, advantage of PSO is given by the relatively small number of parameters to be set. A fundamental parameter is the population size, this parameter is set in an empirical way on the basis of the size and perceived difficulty of the problem. The parameters ϕ_1 and ϕ_2 determine the relative magnitudes of random forces in the direction of the best particle $\vec{\phi_i}$ and of the best neighborhood $\vec{\phi_g}$, and are often called acceleration coefficients. The behavior of the PSO can change with the values of ϕ_1 and ϕ_2. Interestingly, the components $\vec{U}(0, \phi_1) \otimes (\vec{p_i} - \vec{x_i})$ and $\vec{U}(0, \phi_2) \otimes (\vec{p_g} - \vec{x_i})$ can be interpreted as attractive forces. When we change ϕ_1 and ϕ_2, we can get a PSO more reactive but possibly unstable, in which the velocity of the particles increases without any control.

Applications

The first practical application of the PSO was in the field of neural networks, which was presented with the same algorithm. Many other application areas have been explored since then, including telecommunications, control, data mining, design, combinatorial optimization, signal processing, and many others. Although the PSO has been used mainly to troubleshoot problems with a single goal and without constraints, PSO algorithms have been developed to solve problems with constraints, multi-objective optimization problems, problems with dynamic changes of the landscape, and multiple solutions. For a survey on PSO applications, interested readers can refer to [3].

8.2.2.3 *Stochastic diffusion search*

The Stochastic Diffusion Search (SDS), which was introduced by Bishop in 1989, is a research technique that uses a diffusion process to find the best fit of a given model within an area of research [10]. It is a well-characterized, robust, and global metaheuristic of the family of Swarm Intelligence, able to efficiently solve problems of research and optimization through composite structures. The SDS uses a form of direct communication between the agents, in a manner more akin to the *tandem calling* mechanism used by a particular ant species called Leptothorax acervorum. SDS is an algorithm for matching of a model based on a population of agents. Each agent takes care of the information related to the area of research in order to identify the best solution for a given model objective. The research space and the target model require to be split into micro features through a predefined set or alphabetical order [58].

Description of the algorithm SDS In general, the SDS can be easily applied to optimization problems, in which the objective function is decomposable into elements that can be assessed independently. To locate the optimal solutions for a given function objective, SDS employs a set of n agents, each of which stores a hypothesis x_i, in the range of optimal solutions. An iteration of the SDS algorithm involves testing and spread until one of the agents of the swarm does not converge to an excellent hypothesis. The agents in the SDS cooperate in a synchronous manner and appear to be subject to the steps explained below:

Algorithm 8.1 Standard SDS algorithm

```
Initialization;
repeat
  Test;
  Diffusion;
until (Termination criterium)
```

The first step sets the initial hypothesis of each agent. Generally, its value is selected randomly and uniformly within the search space. However, any information about the probable solutions available a priori, can be useful in the setting of assumptions. Then, each agent randomly selects a function f_i, with $i \in 1, \ldots, n$, and performs the evaluation of its own hypothesis $s_h \in S$. Based on this evaluation, the agents are divided into two groups: active and inactive. For the active agents we have that $f_i(s_h) = 0$, while for the inactive ones we have that $f_i(s_h) = 1$. It should be noted that, since f_i is a probabilistic function, it is possible that different evaluations of $f_i(s_h)$ give different results [43]. In the diffusion step each inactive agent randomly selects another agent to communicate with. If the latter is active, it can duplicate the hypothesis of the passive agent, hence the dissemination of information. If the selected agent is inactive, there can be no transfer of information between the two agents, and so the chosen agent will adopt a new random value. In contrast, the active agents in the standard SDS can not initiate any communication with other agents. While the iterations proceed, groups of agents assuming the same hypothesis constitute, for convergence, the largest group of agents that defines the optimal solution. The convergence is defined by two termination criteria:

- *Strong Halting Criterion:* after having determined that a group of agents larger than a threshold, verify that the size of the group is (stochastically) stable over a certain number of iterations.
- *Weak Halting Criterion:* it simply checks the stability and the minimum size of the total number of active agents (the total activity is strongly dependent on the current best solution found).

Since the tests occur with high frequency in the points of the solution space that show a good objective value, on average the agents spend more time on these optimal solutions, and at the same time, attract other agents. However, limited resources (a finite size of the population) ensure that only the best solution discovered up to that moment is able to maintain a stable group of agents. This different resources allocation allows the largest group of agents to determine the optimal solution, without requiring the individual agents to evaluate the objective function in an explicit way. Three recruitment strategies have been introduced for SDS: the passive recruitment (the standard mechanism), the active recruitment and the double recruitment. The passive recruitment has been briefly introduced above. The active recruitment is modeled on the behavior of the species of insects that swarm. They actively try to recruit other members to direct them towards a preferred direction, which can be a source of food or the selection of the site to build the nest. A practical example is the waggle dance of the bees in the hive. The waggle dance is performed to indicate to other bees the location of a promising source of food. During the deployment phase, in the active recruitment, the active agents seek the passive agents to communicate their hypotheses. Each active agent randomly contacts another agent B, if B is passive, it will be recruited from A (the hypothesis of A is communicated to B). Unlike the passive recruitment, in which the larger group, in theory, may increase disproportionately at each iteration, the active recruitment allows, at most, to double, in the size of the group, at each iteration (if any active agent chooses a passive agent). Finally, in the double recruiting, the mechanism of active and passive recruitment act simultaneously. Thus, both active and passive agents choose the agents and the hypotheses are transferred from the active agents to the passive ones. This mix of recruitment mechanisms in a system is considered to be the most truthful, from a biological point of view, and is therefore of particular interest. This may lead to a conflict in the allocation of hypotheses, so it is necessary to define some priorities among an active agent that assigns a hypothesis to a passive agent (active priority) and a passive agent that copies hypothesis from an active agent (passive priority). The SDS has a greedy assignment process, that is, once a good solution has been found, a large portion of the swarm is allocated for the operations, making these agents not available for further explorations. A mechanism that frees some of these resources without significantly affecting the stability properties of the groups of agents could increase the efficiency of SDS for many types of problems, in particular in dynamic optimization.

Applications

The SDS has been applied to many different problems of research and optimization such as the site selection for wireless networks, the identification of the sequence in bio-informatics, self-localization of mobile robots, object recognition, motion tracking of eyes and lips, and text search.

8.3 Robot Cooperation

Robot cooperation is a challenging domain that researchers have been investigating since the 1980's. It is the ability of solving a task by a group of robots. Robots cooperate as a team in order to achieve a common goal. Multi-robot cooperation increases efficiency of robots and allows the achievement of complex tasks, which cannot be accomplished by a single robot. Multi-robot cooperation comes into applications and extends research on single robot for many reasons:

- Tasks are basically too complex for a single robot to achieve because single robot is spatially limited;
- Using multiple simple robots may be cheaper and simpler than handling one complex robot;
- Multi-robot systems are more flexible and fault-tolerant than single robots acting alone.

Robots act through a cooperative behavior. They are aware of their teammates, they share goals and their actions are useful for the whole group. Cao et al. [82] have defined robots cooperative behavior over a multi-robot system as it follows: *Given some task specified by a designer, a multiple-robot system displays cooperative behavior if, due to some underlying mechanism (i.e., the "mechanism of cooperation"), there is an increase of the overall system utility.*
In other works, scientists classified multi-robot cooperative systems into two categories: active and passive. In active system, robots communicate between them in order to exchange information. They can organize their tasks and make decisions. In passive systems, there is no communications link between robots which makes the system easy to design and robust. Robots do not share information and do not make decisions together.

Robots community has been interested in this domain in the last decades, where many researchers wrote a number of surveys on multi-robot systems. Cao et al. [82] summarized the research into a taxonomy of cooperative systems. They surveyed five main research directions: group architecture, resource conflicts, origins of cooperation, learning problems, and geometric problems. L.E. Parker organized current research works in multi-robot systems by principal topic areas [64], and focused on the interaction of multiple mobile robots in chapter "Multiple mobile robots systems" of the handbook of robotics [75]. Verret gave a brief history of robotics and detailed some inspirations and influences in multi-robot systems [79]. Cai et al. explored few research fields in the multi-robot systems [14]. In this section, we will explore several robot cooperation aspects such as distributed fusion, cooperative localization and architectures and we will discuss communication effects on a multi-robot system. We will focus on active cooperative systems and consider the basic functionalities of such systems. Another aspect that should be developed is related to artificial intelligence such as control, planning and task allocation. Issues related to distributed artificial intelligence will be explored in the second part of this chapter.

8.3.1 *Communication*

Accomplishing a cooperative task needs some form of communication. Researchers distinguished between implicit and explicit communications. Implicit communication allows robots to communicate through their environment. In this case, communication is based upon the environment change

or behavior of other robots. Robots are equipped with sensors to observe the changes. Whereas, in explicit communication, robots exchange messages to transfer various information like positions, current status, future actions, etc. They devise also effective cooperative schemes.
Several works explored communication effect on performance of multi-robot systems in different cooperative tasks. They concluded that communication between robots can multiply their capabilities and can improve their efficiency. It can provide benefit for many tasks. Exchanging a small amount of information can lead to better performances.
Using wireless communication among multi-robot systems has become an important area of research [65] and a requirement for different scenarios ([20], [52]). It has contributed to the cooperative systems using explicit communication. Robots use this technology for exploration, distributed sensing or tracking, environmental monitoring and surveillance [40]. They should be able to send and receive information at any time. Even though there is no clear conclusion on which type of communication is better for robot cooperation: implicit communication can fulfill some tasks, while explicit communication can improve flexibility of multi-robot systems. Recent work took advantage of implicit and explicit communications in order to improve cooperation and competition between robots [81].

8.3.2 *Research fields*

8.3.2.1 *Distributed sensor data fusion*

Robots may be equipped with different sensors such as vision sensors (camera), sensitive sensors, distance measurement (radar and laser scanner) or position sensors (odometry, GPS, etc.). These sensors help robots discover their environment and are considered as important input for the perception task. This process consists in fusing data collected by the sensors, exploiting redundant information and reducing uncertainty. Different fusion algorithms have been developed and used in literature for single robots. Main fusion methods include weighted average method, Bayesian inference, Dempster Shafer theory, Kalman filter, fuzzy logic and neural networks. Zhao et al. presented a survey on robot multi-sensor fusion technology and explored different applications of multi-sensor fusion [85].

Multi-sensor fusion improves robots sensing and decision making while accomplishing different tasks. That is why these methods are exploited for multi-robot systems. Thanks to communication capabilities, robots can exchange information that can be used to enhance their perception performances. They exchange data with their teammates through two different approaches. In the first approach, the robot sends its own perception, in this case the fusion algorithm considers data sent by robots as it comes from an off-board sensor. In the second approach, robot sends data resulting from the fusion of all the received information. The same information can be received many times; this is what is called “data incest” [61]. Due to the latter, robots suffer from cycles of data dissemination where the same information is provided by independent sources and can be combined many times. Distributed multi-sensor fusion process should manage data incest issue and take account of the latency of distributed data and the different references systems. The main fusion method that had been appropriated to robot cooperation and that can manage all these latter are the Bayesian inference, Dempster Shafer theory and Kalman filter.

Bayesian inference combines multi-sensor information according to rules of probability theory based on observational evidences. It depends on the prior knowledge. In general, robots share their fused data. To avoid the data incest the method of covariance intersection is used ([41], [17]). For example, Santos et al. [72] developed a multi-robot cooperative object localization based on a decentralized Bayesian approach. Their method is composed of a local filter and a team filter. The local filter receives a reduced dimension representation of its teammates sample belief about the object location. The team filter receives Gaussian Mixture Model (GMM) representations of the object in the world frame, from the sensor teammates, and fuses them all performing Covariance Intersection among GMM components.

Dempster Shafer theory, which generalizes the Bayesian inference, deals with incomplete and uncertain data. It represents the knowledge by mass functions, updates the beliefs and combines the evidences. Wang Shuo et al. were interested in map-building task for multi-robot system [80]. A robot detects its environment using its own sensors and can exchange its sensing information with other robots to build a global map. Information is fused by using the Dempster operator adapted to the combination of independent sources. Authors explored the cooperative strategies in order to avoid invalid sensing information. Nowadays, different studies were interested in using Dempster–Shafer theory for distributed data fusion in vehicle networks [28]. This method is based on the cautious rule of combination [22] that allows combination of dependent sources due to its idempotent properties [86]. It can be suited for data combination in multi-robot systems.

Kalman filter uses statistical characteristics of the system model to recursively determine estimates for data fusion. It deals with dynamic models and fuses the low level redundant data of independent sources. Fused data is local to each robot and is not a result of combination. The communication regarding the robots' positions increases certainty and reduces imprecision about the robots' own poses [4]. Kalman filter was applied to improve the ball position estimation for a robotic soccer team [76] and its extensions were explored for cooperative behavior of mobile robots [68]. It is also applied for robot localization, we will detail this in the next section.

Distributed multi-sensor fusion is inevitable in different applications of multi-robot systems. Robots can receive different kinds of information from their teammates when cooperating to achieve a task. This information should be combined with the best fusion algorithm that can avoid data incest, latency and exploit the redundancy and complementarity of information.

8.3.2.2 *Cooperative localization*

Localization is an essential problem in multi-robot systems. Robots should be able to estimate their positions in order to navigate autonomously in an environment, which may be known or unknown for robots. Usually, most robots tasks require information about their positions and orientations. Distributed tasks require information about the whole group of robots or at least those detected in the scene. To supply multi-robot systems with a solution for these needs, cooperative localization has been introduced. It consists in locating each robot in a group within the same environment. Nowadays, we find localization methods based on proprioceptive sensors, inertial unit and/or GPS. Other methods take the assumption of known environment and locate robots with exteroceptive sensors.

For a strong cooperation, each robot should know positions of other members of the team. In general, they use their sensors in order to detect other robots and to recognize their environment. They communicate to exchange their pose, their maps, and the state of team. Therefore, communication capabilities allow cooperative localization. Some recent studies investigated the cooperative localization approach especially to improve localization accuracy. In the following, we will explore different examples.

Different localization methods dedicated for single robots were extended to multi-robot systems. In [32] collaborative robot localization of indoor robots is developed based on Markov Localization. Robots can localize themselves in the same environment, maps are supposed to be known. Whenever one robot detects another, probabilistic methods are used to synchronize each robot's belief. The Kalman filter and its extensions were deployed for cooperative localization in different studies. In [69], each robot shares the information regarding its own motion with the rest of the team. The Kalman filter is used to process the available positioning information from all the team members. It estimates a pose for each robot. The authors showed that the Kalman filter estimator can be distributed in a number of smaller communicating filters, one filter for each robot. Each filter processes sensor data collected by its robot and communication with other filter when two robots detect each other and measure their relative pose. Martinelli et al. [55] extended what have been developed in [69] and introduced an EKF approach by considering the most general relative observation between two robots. Karam et al. described a cooperative approach for collective localization of a heterogeneous group of vehicles where each vehicle updates its group state with its own sensor data [45]. The vehicles exchange their information about the positions of the rest of the group, then fuse it in order to obtain the global state of the system. In [50], Lee et al. presented a cooperative localization method for a multi-robot system. They incorporated different sensors such as GPS, odometer and gyro sensor to localize absolute and relative position. They utilized correlation between GPS errors and differential position data between the robots to refine their position data. In [66], the authors treated the problem of absolute localization of a team of robots for unknown initial robot positions. They proposed a particle clustering method which reduces the complexity of the overall localization algorithm.

Other methods were based on exteroceptive sensors and used communication to exchange information. Franchi et al. in [33] took the assumption that each robot is equipped with a sensor that measure the relative pose of nearby robots without their identity. They proposed a two stage localization system where data is processed by an associator, and EKF is used to isolate and treat the best estimates. The localization approach presented in [77] is based on stereo vision system helping each robot to recognize the others and its environment. The authors used serial and parallel fusion. The first approach identifies the position uncertainty of an observed robot while the second method reduces errors of the position.

Different studies explored robot formation for localization. Hidaka et al. proposed a method for optimizing the geometry of robots formation [38]. They evaluated the trace of the steady-state covariance of the robots position estimates. This method was applied on heterogeneous robots teams. The authors studied the effects of optimal formation on robot's localization.

As presented above, researchers have been interested in cooperative localization. Some of them used method initially designed for single robots. Others developed new approaches for multi-robot system. They profit from the advantages of collecting and integrating sensors information from different robots. They proved that cooperative localization can give a system better localization performance and improve robustness of localization for each robot in the group.

8.3.2.3 *Control architectures*

Control architecture is needed to handle robot control system complexity. This is because robots need to interact in a certain environment. The control architecture influences the system robustness. It should allow robots to act in real time and to control sensors and actuators. L.E. Parker described in the chapter "Multiple Mobile Robot Systems" in [75] four types of architectures: centralized, hierarchical, decentralized and hybrid.

- In **Centralized architecture** one agent controls the remaining agents in the system. The centralized controller provides a strategy for cooperation and decision making. Each robot in the team takes the commands.
- **Hierarchical architecture** is based on an approach where one robot supervises the action of a group of robots. Each robot of this group supervises another group of other robots and so on. Each robot receives a part of the task to execute.
- **Decentralized architecture** does not employ a central agent that controls all remaining agents. Robots take actions based on information detected in their environment. This architecture is very robust to failure, flexible and scalable, since control is not centralized. Each robot is responsible of its own actions.
- **Hybrid architecture** profits from the advantages of the above control strategies. In this strategy, agents are decentralized while a centralized planner supervises the team.

Different control architectures have been developed over the years. The interested readers can refer to [84] for a survey on the topic.

8.3.3 *Applications*

Cooperative robotics may be applied to different domains such as underwater and space exploration, building surveillance, large objects transportation and hazardous environment (Landmine detection, de-mining, etc.). They may also be implemented for air and underwater pollution monitoring, forest fire detection, service robotics in both public and private domains, transportation systems, search and rescue operations after large-scale disasters and the entertainment field:

- **Exploration** is an important application because it concerns areas where humans cannot access easily. Belbachir et al. treated the problem of underwater exploration where autonomous underwater vehicles exchange their information and cooperate to optimize their motions [8]. Leitner studied space applications [51], while Bautin et al. presented a frontier-based exploration method for multi-robot system [6]. In [48] air and ground robots were deployed.
- **Object transportation and manipulation** is a task where cooperation between a group of robots is essential. Robots surround the object and transport it to the desired destination by pushing it (Figure 8.4). Different works in literature treated this application and proposed different control approaches and motion planning ([83], [56]).

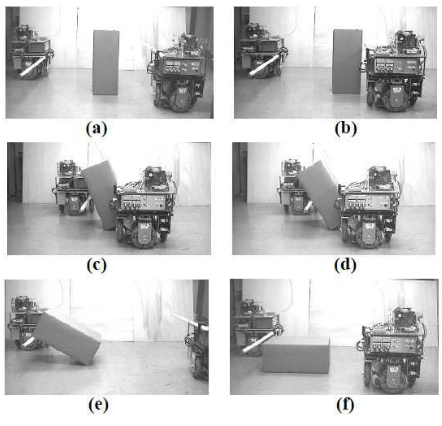

Fig. 8.4 Experimental Results of tumbling a rectangle with two robots [83].

- **Robots soccer** is considered as a dynamical and adversarial application. A ball moves and robots should act to catch and shoot it. Robots of one team should cooperate to compete with the opponent team. This application requires a lot of knowledge in different research areas such as robotics, intelligent control, communication, computer and sensor technology, image processing, mechatronics, and artificial intelligence. It is an important application for robot cooperation and competition. Figure 8.5 shows the RoboCup Soccer platform league in Robocup 2010. J.-H. Kim et al. [47] explored a soccer robot system and presented two control schemes: vision-based and robot-based. They discussed both control structure and action selection mechanisms. D.-H. Lee [49] proposed a task and role selection strategy where each robot in a team selects its task and role.

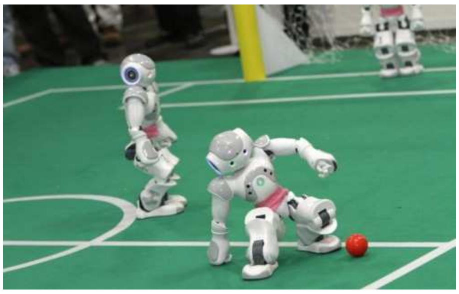

Fig. 8.5 RoboCup soccer league in 2010 (www.robocup.org)(Copyright ©1998–2013).

A plethora of applications was developed for multi-robot cooperation. Nowadays, researchers explore applications in different domains such as intelligent vehicles and swarm robotics, about which we will talk in the next section.

8.3.4 *Challenges*

Distributed information engenders a main challenge in robot cooperation. Robots exchange information through communication, thus increasing cooperation vulnerability towards errors. Communication between robots requires bandwidth and solution for data dissemination. Researchers should choose a robust network architecture capable of handling connection failures and ensuring message passing. Furthermore, by sharing information, the system is confronted to issues due to the different representation of the environment that each robot can have. This is due to several reasons such as errors in localization of each robot or different sensor equipment available on different robots. Data incest remains a challenging issue for multi-robot system. Scientists should tackle this problem by developing appropriate fusion algorithms. Several research challenges still remain such as robot computational capabilities, control architecture design, organization of high number of robots...

8.4 Swarm Robotics

It is difficult to define properly a swarm robotics, due to wide range of applications. Maybe, the most appropriate definition is: "Swarm robotics is the study of how large the number of relatively simple physically embodied agents can be designed such that a desired collective behavior emerges from the local interaction among agents and between the agents and the environment" [71]. In this definition, the main characteristics of a swarm robotics are summarized: simplicity of robots, fully distributed system, scalability, robustness. Swarm Robotics are required to be characterized with specific key advantages such as:

- Parallelism: typically a big, complex task is divided in many sub-task and each unit accomplishes a given task quicker than a single robot;
- Robustness: the system is required with a high degree of fault tolerance. In practice, if some robot fails the execution of its task, the system will evolve in a novel and dynamic configuration that will reestablish the correct functioning of the system;
- Scalability: the increment of the number of devices does not degrade the performance of the whole system;
- Heterogeneousness: each unit can be characterized with specific properties that will be effectively exploited to accomplish suitable tasks;
- Flexibility: a system has to be reconfigurable in order to accomplish different tasks and execute different applications;
- Complex Tasks: generally, a single unit is not able to accomplish a complex task, whereas a swarm is able to, because of the joint capabilities of the single devices;
- Cheap Alternative: devices are simple, easy to build and cheaper than a single powerful robot.

Typically, Swarm Robotics operate based on some sense of biological inspiration [74]. In this sense, the application of Swarm Intelligence to collective robotics can be identified as "Swarm Robotics". The sense of the interaction between bio-inspiration, Swarm Intelligence and Self-organized and Distributed System can be explained through the Fig. 8.6.

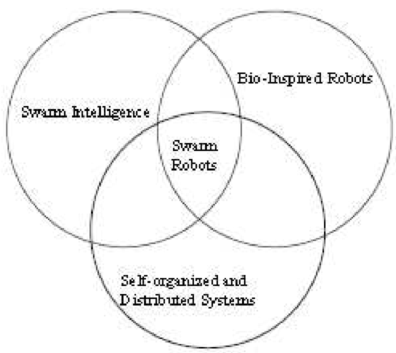

Fig. 8.6 Swarm Robotics as the intersection between bio-inspired systems, robots Swarm Intelligence and Self-Organized and Distributed Systems [63].

From an historical point of view, the first experiments on systems of robots that could be identified as Swarm Robotics, were realized in late 1940s. Grey Walter and his team showed a system of simple robots interacting in a seemingly social manner and by exhibiting "complex behavior" [24], but Swarm Robotics becomes an active field of research only in the 1990's. G. Beni [9] introduced the concept of Swarm Robotics by discussing cellular robotics systems. In the 1990's Deneubourg et al. introduced the concept of stigmergy in robots that behave like ants [21], [7]. Since then, numerous researchers have developed collective and self-organized systems [18] and have introduced robots' behaviors inspired by insects' social organization [53], [25], [16].

8.4.1 *Classification of swarm robotics*

Different types of classification have been proposed for Swarm Robotics. In [1] authors propose a taxonomy and classify existing studies. Specifically, they split existing studies into the most important research directions. The five fields they identify are: modeling, behavior design, communication, analytical studies and problems. The taxonomy is summarized in Fig. 8.7.

Concerning *modeling*, authors found that modeling is a very suitable method for Swarm Robotics. In fact, there are some risks related to the

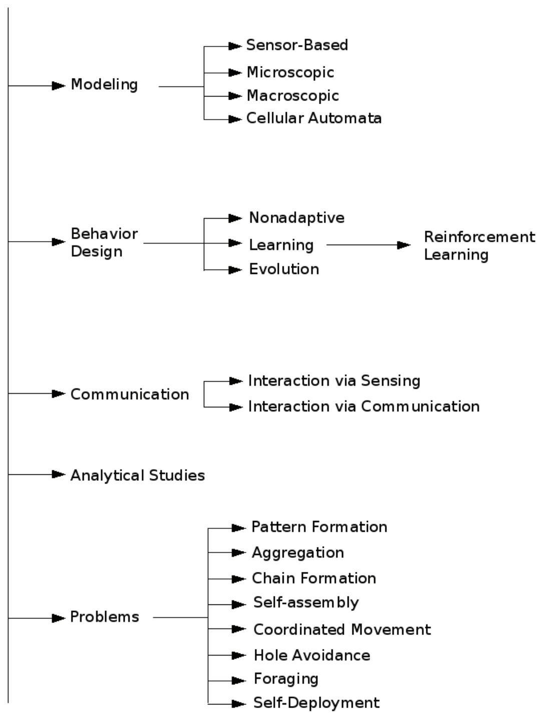

Fig. 8.7 Classification of Swarm Robotics literature [1].

robots that require a human to follow the experiments. Typically, to validate results, a high number of experiments is required and simulation and modeling of the experiments seem to be an effective way to make the system work. Another important aspect related to modeling in Swarm Robotics is scalability. Generally, demonstration of scalability of some control algorithm requires hundreds of robots. Costs related to the use of such a number of robots could be prohibitive and modeling could become the only viable solution.

In a biological system, individuals may fine-tune their behaviors in their lifetime. In practice, they learn how to survive and to stay better when external conditions change. In Swarm Robotics, researchers considered the *behavioral adaptation* to control large number of robots to accomplish a task collectively.

Communication is sub-divided into three types. The first kind is via sensing and represents the simplest type of communication based on the capacity of a robot to distinguish between other robots and the objects in the environment. When robots use interaction via the environment, they consider it as a communication medium (i.e., pheromones used from ants). Interaction via communication involves explicit communication through direct messages.

Analytical studies include studies that contribute to the theoretical understanding of swarm systems. In this category, methods for solution of different problems can be included. Furthermore, mathematical tools that allow a deeper comprehension of the details of Swarm Robotics systems can be considered as part of analytical studies.

The last point of the taxonomy formulated in [1] is the *problem* axis, where authors individuate general problems that have been investigated in Swarm Robotics. A researcher could find useful the individuation of this point when he tries to solve a specific problem and can try to make its problem matchable with a more general problem already faced in literature.

In [36] authors classified Swarm Robotics literature in different domains. They suggest a classification based on the characteristics of the swarm as a whole rather than the architectural characteristics of individual robots. Authors individuate domains like communication range, communication topology, swarm size, communication bandwidth, etc.

In [82] authors present a survey of cooperative robotics in a hierarchical way, as we have already mentioned in the previous section. They individuate five main axes: group architecture, resource conflicts, origins of cooperation, learning and geometric problems. Group architecture represents the necessary infrastructure where the cooperative behavior must rely. Resource conflicts is strictly related to the communication of the robots, the management of the shared environment, etc. Origin of cooperation means how cooperative behavior is achieved and actuated. Learning axis is strictly related to adaptability and flexibility that represent essential traits in a task-solving Swarm Robotics. These first four axes are mostly related to the cooperative aspects of Swarm Robotics. The fifth axis individuated by the authors is defined as geometric problems and covers research issues tied to the embedding of robot tasks in a two- or three-dimensional world.

In [44] authors survey existing works on modeling collective behavior of robot swarms with macroscopic models. Specifically, they consider very simple robots that can be represented as stochastic Markov processes. A macroscopic model describes the collective behavior of the robotic swarm. The choice of macroscopic vs microscopic models, by taking into account the behavior of some *average* quantity that represents the system, is related to the inherent simplicity and the analytical tractability of such macroscopic descriptions.

A more recent and interesting classification of Swarm Robotics is given in [13]. Authors propose a classification based on two taxonomies as shown in Fig. 8.8.

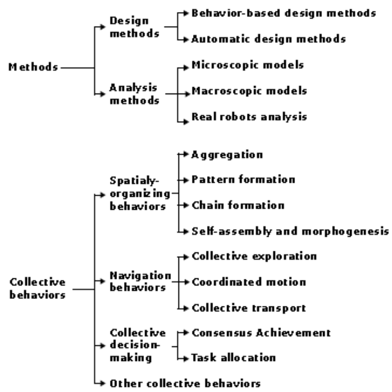

Fig. 8.8 Taxonomies for Swarm Robotics [13].

Authors classify literature regarding Swarm Robotics according to the methods used to design or analyze the swarms and its main collective behaviors.

8.4.2 *Applications of swarm robotics*

Concerning potential domains where Swarm Robotics can be applied, various scenarios can be individuated:

- *Foraging* - In this scenario, a robot is able to collect the objects and deliver them to some predefined location. This scenario requires many fundamental skills from a Swarm Robotic system, such as collective exploration, efficient task allocation, etc. It seems that the first contribution in terms of implementation of foraging using a group of real robots is given in [62]. In [57], Mataric considers a formulation of reinforcement learning in a concurrent multi-robot learning domain. In order to validate the fach, the author proposes an experiment involving four mobile robots learning a foraging task. Sugawara et al. investigate the collective and cooperative behavior of interacting agents [42].

The task assigned to the swarm consists into pick up and collect pucks distributed in a field. This task is reformulated as a problem of ants foraging even if the movement and interaction of agents (robots) are more simplified than ants. In [31], Ducatelle et al. study self-organization of heterogeneous swarms robotic to solve a specific task. Specifically, they consider two swarms that need to mutually adapt to each other and the swarm, as a whole, has to solve the task. Their work is related to the problem on self-organized foraging, where robots have to optimize a path to follow back and forth between a source and a target [34].

- *Aggregation* - Aggregation is one of the fundamental behaviors of swarm in nature. In Swarm Robotics, self-organized aggregation is required to form a robot cluster and is a very common goal but the approaches are very diverse. In [27], authors consider an evolving neural network with 12 neurons to reach robot aggregation. Additionally, aggregation is a requisite for Swarm Robotics behaviors such as self-assembly and pattern formation [78]. In [12], authors show how the spatial separation of two conflictive spots affects the cooperation behavior. In [5], authors investigate aggregation behavior as a case, and systematically studies the performance and the scalability of aggregation behaviors of perceptron controllers evolved for a simulated Swarm Robotic system with different parameter settings. Baldassarre et al. successfully evolved controllers for a swarm of robots to aggregate and move towards a light source in a clustered formation [35].
- *Clustering and Sorting* - Clustering and sorting are mostly influenced by the nest building behavior of termites and wasps. Distributed clustering, and more recently sorting, by a swarm of robots have served as benchmarks for swarm intelligence based robotics [29]. In [2] a new method for distributed object sorting by a swarm of robots is introduced. In this work it is shown how an unloaded agent seeks an isolated object to pick up, and an agent already carrying an object seeks an existing cluster of the same type to deposit its load. Authors employ only on-board sensing. In [23] authors use the concept of spatial awareness to accomplish the cluster task and support task allocation that are spatially differentiated.
- *Exploration* - Exploration of an unknown environment is a fundamental issue in mobile robotics. One of the main advantages in the usage of multiple robots instead of a single one lies on the speed of convergence of the exploration process, accuracy of the solution, and fault tolerance. Most significant research topics in multi-robot-exploration

are task sharing and navigation. A useful contribution in terms of multi-robot exploration is given in [54]. The exploration method proposed by these authors minimizes the overall exploration time, making it possible to efficiently localize fire sources. In GUARDIANS (Group of Unmanned Assistant Robots Deployed In Aggregative Navigation by Scent)FP6, EU funded project, there is a group of robots whose task is the exploration of the unknown operative environment. In [19] authors propose two techniques based on Particle Swarm Optimization and Darwinian Particle Optimization to perform in an effective way exploration task, by explicitly taking obstacle avoidance into account.

8.5 Conclusion

In the first part of this chapter, we tried to express the essence of what Swarm Intelligence is, namely: "A single ant or bee is not smart, but their colonies are. The study of Swarm Intelligence is providing insights that can help humans manage complex systems, from truck routing to military robots" [60]. After considering the main applications of Swarm Intelligence itself, we presented the most important results about cooperation of robots, trying to analyze in a critical fashion the issues and challenges related to this field and presenting the main advantages. The last contribution of this chapter is represented by the synergic "fusion" of Swarm Intelligence and cooperation among robots, that we referred as Swarm Robotics. Specifically, we presented some interesting taxonomies and typical applications of this new field of research.

Bibliography

[1] A., B. and E., S. (2007). A review of studies in swarm robotics, *Turkish journal of electrical engineering and computer science* **15**, pp. 115–147.

[2] A., V. (2012). Accelerated patch sorting by a robotic swarm. in *Ninth Conference on Computer and Robot Vision (CRV)*, pp. 314–321.

[3] AlRashidi, M. and El-Hawary, M. (2009). Applications of computational intelligence techniques for solving the revived optimal power flow problem, *Electric Power Systems Research* **79**, 4, pp. 694 – 702, doi:10.1016/j.epsr.2008.10.004, URL `http://www.sciencedirect.com/science/article/pii/S0378779608002757`.

[4] Andersson, L. and Nygards, J. (2003). On sensor fusion between a pair of heterogeneous robots, in *Proceedings of the 6th International Conference on Information Fusion, FUSION 2003* (Cairns, Australia), pp. 1287–1294.

[5] Baheci, I. and Sahin, E. (2005), Evolving aggregation behaviours for swarm robotic systems: a systematic case study. In *Proceedings of IEEE swarm intelligence symposium.*, pp. 333–340.

[6] Bautin, A., Simonin, O. and Charpillet, F. (2011). Towards a communication free coordination for multi-robot exploration, in *6th National Conference on Control Architectures of Robots, CAR 2011* (Grenoble, France).

[7] Beckers, R., Holland, O. and Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics, in *Artificial Life*.

[8] Belbachir, A., Lacroix, S., Ingrand, F., Perrier, M. and Opderbecke, J. (2010). Cooperative-adaptive algorithms for targets localization in underwater environment, in *IEEE-OES Autonomous Underwater Vehicles* (Monterey, CA).

[9] Beni, G. and Wang, J. (1989). Swarm intelligence in cellular robotic systems, in *Proceedings of NATO Advanced Workshop on Robots and Biological Systems* (Italy).

[10] Bishop, J. (1989). Stochastic searching networks, in *1st IEE Conf. Artificial Neural Networks* (Reading University, UK).

[11] Blum, C. and Merkle, D. (2008). *Swarm Intelligence: Introduction and Applications* (Natural Computing Series. Springer).

[12] Bodi, M., Thenius, R., Schmickl, T. and Crailsheim, K. (2011). How two cooperating robot swarms are affected by two conflictive aggregation spots, in *Proceedings of the 10th European conference on Advances in artificial life: Darwin meets von Neumann - Volume Part II*, ECAL'09 (Springer-Verlag, Berlin, Heidelberg), ISBN 978-3-642-21313-7, pp. 367–374, URL `http://dl.acm.org/citation.cfm?id=2017762.2017813`.

[13] Brambilla, M., Pinciroli, C., Birattari, M. and Dorigo, M. (2012). Property-driven design for swarm robotics, in *IAAMAS*, pp. 139–146.

[14] Cai, Y. and Yang, S. X. (2012). A survey on multi-robot systems, in *World Automation Congress (WAC), 2012* (Puerto Vallarta, Mexico).

[15] Camazine, S., Deneubourg, J.-L., Franks, N., Sneyd, J., Theraulaz, G. and Bonabeau, E. (2001). *Self-organization in biological systems* (Princeton University Press).

[16] Campo, A., Gutiérrez, A., Nouyan, S., Pinciroli, C., Longchamp, V., Garnier, S. and Dorigo, M. (2010). Artificial pheromone for path selection by a foraging swarm of robots, *Biological Cybernetics* **103**, 5, pp. 339–352, doi: http://dx.doi.org/10.1007/s00422-010-0402-x.

[17] Chen, L., Arambel, P. O. and Mehra, R. K. (2002). Estimation under unknown correlation: covariance intersection revisited, *IEEE Transactions on Automatic Control* **47**, pp. 1879–1882.

[18] Correll, N. and Martinoli, A. (2006). System identification of self-organizing robotics swarms, in *DARS*, pp. 31–40.

[19] Couceiro, M. (2011). A novel multi-robot exploration approach based on particle swarm optimization algorithms. in *In Proceedings of IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 327–332.

[20] Cui, Y. A. and Dai, Y. . J. (2011). Robots active linking in unknown environment, *Advanced Materials Research* **186**, pp. 84–88.

[21] Deneubourg, J. L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C. and Chrétien, L. (1991). The dynamics of collective sorting: Robots-like ants and ants-like robots. from animal to animats, in *Artificial Life*, pp. 356–363.

[22] Denoeux, T. (2008). Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence, *Artificial Intelligence* **172**, pp. 234–264, doi:10.1016/j.artint.2007.05.008.

[23] Di Caro, G. A., Ducatelle, F. and Gambardella, L. (2012). A fully distributed communication-based approach for spatial clustering in robotic swarms, in *Proceedings of the 2nd Autonomous Robots and Multirobot Systems Workshop (ARMS), affiliated with the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)* (Valencia, Spain, June 5), pp. 153–171.

[24] Dorf, R. (1990). *Concise international encyclopaedia of robotics: Applications and automation* (Wiley-Interscience).

[25] Dorigo, M. (ed.) (1996). Special Issue on Learning Autonomous Robots, *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, 26:361–506 (IEEE Press), doi:http://ieeexplore.ieee.org/xpl/tocresult.jsp?isYear=1996\&isnumber=10822\&Submit32=Go+To+Issues.

[26] Dorigo, M. and Stützle, T. (2004). *Ant Colony Optimization* (MIT Press, Cambridge, USA).

[27] Dorigo, M., Trianni, V., Şahin, E., Groß, R., Labella, T. H., Baldassarre, G., Nolfi, S., Deneubourg, J.-L., Mondada, F., Floreano, D. and Gambardella, L. M. (2004). Evolving self-organizing behaviors for a swarm-bot, *Autonomous Robots* **17(2-3)**, pp. 223–245.

[28] Ducourthial, B., Cherfaoui, V. and Denoeux, T. (2012). Self-stabilizing distributed data fusion, in *14th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2012)*, Vol. 7596 (Toronto), pp. 148–162.

[29] E. Bonabeau, M. D. and Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems* (Oxford University Press, New York, USA).

[30] Eslami, M., Shareef, H., Khajehzadeh, M. and Mohamed, A. (2012). An effective particle swarm optimization for global optimization, *Computational Intelligence and Intelligent Systems* , pp. 267–274doi:10.1007/978-3-642-34289-9_30, URL `http://dx.doi.org/10.1007/978-3-642-34289-9_30`.

[31] F.Ducatelle, Di Caro, G. A., Förster, A. and Gambardella, L. (2010). Mobile stigmergic markers for navigation in a heterogeneous robotic swarm, *Swarm Intelligence, Proceedings of the 7th International ANTS Conference* .

[32] Fox, D., Burgard, W., Kruppa, H. and Thrun, S. (2000). A probabilistic approach to collaborative multi-robot localization, *Autonomous Robots* **8**, pp. 325–344.

[33] Franchi, A., Oriolo, G. and Stegagno, P. (2009). Mutual localization in a multi-robot system with anonymous relative position measures, in *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (St. Louis, MO), pp. 3974–3980.

[34] Fujisawa, R., Dobata, S., Kubota, D., Imamura, H. and Matsuno, F. (2008). Dependency by concentration of pheromone trail for multiple robots, *Proc. of the 6th Int. Conf. on Ant Colony Optimization and Swarm Intelligence (ANTS)* .

[35] G., B., S., N. and D., P. (2003). Evolving mobile robots able to display collective behavior, in *Artificial Life*, Vol. 9(3), pp. 255–267.

[36] G., D., M., J. and Wilkes, D. A. (1993). Taxonomy of swarm robots, *IEEE Int. Conf. on Intelligent robots and systems* , pp. 441–447.

[37] H. Zhu, K. E., C. Pu and Gu, J. (2009). Euclidean particle swarm optimization, in *In Intelligent Networks and Intelligent Systems (ICINIS)*.

[38] Hidaka, Y. S., Mourikis, A. I. and Roumeliotis, S. I. (2005). Optimal formations for cooperative localization of mobile robots, in *Proc. of the IEEE International Conference on Robotics and Automation* (Barcelona, Spain), pp. 4137–4142.

[39] J. Kennedy, R. C. E. and Shi, Y. (2004). *Swarm Intelligence* (Morgan Kaufmann Publishers).

[40] Jia, Y. and Xi, N. (2011). Coordinated formation control for multi-robot systems with communication constraints, in *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM2011)* (Budapest, Hungary).

[41] Julier, S. and Uhlmann, J. (2001). *Handbook of Multidimensional Data Fusion, Chap. 12*, chap. General decentralized data fusion with covariance intersection (CI) (Hall D.and Llians J. eds, CRC Press).

[42] K., S. and M., S. (1997). Cooperative acceleration of task performance: Foraging behavior of interacting multi-robots system. *Physica* , pp. 343–354.

[43] K. De Meyer, S. N. and Bishop, J. (2006). *Stochastic Diffusion Search: partial function evaluation in swarm intelligence dynamic optimisation*, Vol. 31 (Abraham A. and Grosan C. and Ramos, V., eds, Springer Verlag, Germany).

[44] K. Lerman, A. G., A. Martinoli (2005). *A review of probabilistic macroscopic models for swarm robotic systems*, number 3342 in LNCS (Springer-Verlag, Berlin Heidelberg).

[45] Karam, N., Chausse, F., Aufrre, R. and Chapuis, R. (2006). Localization of a group of communicating vehicles by state exchange, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (Beijing, China), pp. 2816–2821.

[46] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization, in *In Proceedings of the Fourth IEEE International Conference on Neural Networks* (Perth, Australia), pp. 1942–1948.

[47] Kim, J.-H., Shim, H.-S., Jung, M.-J., Kim, H.-S. and Prahlad., V. (1997). Cooperative multi-agent robotic systems: From the robot-soccer perspective, in *Proc. of MiroSot97 Workshop*, pp. 3–14.

[48] Lacroix, S. and Besnerais, G. L. (2011). Issues in cooperative air/ground robotic systems, *Robotics Research* , pp. 421–432.

[49] Lee, D.-H., Na, K.-I. and Kim, J.-H. (2010). Task and role selection strategy for multi-robot cooperation in robot soccer, *Trends in Intelligent Robotics Communications in Computer and Information Science* **103**, pp. 170–177.

[50] Lee, J., Jo, K. and Jang, C. (2007). Cooperative localization for multirobot incorporative proprioceptive/exteroceptive position sensors, in *6th Int. Conf Field and Service Robotics* (Chamonix, France).

[51] Leitner, J. (2009). A survey of mutli-robot cooperation in space, *Advanced Technologies for Enhanced Quality of Life* , pp. 144–151.

[52] Lopez, J., Perez, D. and Zalama, E. (2011). A framework for building mobile single and multi-robot applications, *Robotics and Autonomous Systems* **59**, 3-4, pp. 151–162.

[53] M., K., J.-B., B. and L., K. (2000). Ant like task allocation and recruitment in cooperative behaviour, in *Nature*, ISBN 406(6799).

[54] Marjovi, A., Nunes, J., Marques, L. and Almeida, A. (2009). Multi-robot exploration and fire searching, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).*

[55] Martinelli, A., Pont, F. and Siegwart, R. (2005). Multi-robot localization using relative observations, in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* (Barcelona, Spain), pp. 2808–2813.

[56] Mas, I. and Kitts, C. (2012). Object manipulation using cooperative mobile multi-robot systems, in *Proceedings of the World Congress on Engineering and Computer Science, WCECS 2012* (San Francisco, USA).

[57] Mataric, M. (1997). Reinforcement learning in the multi-robot domain. *Autonomos robots* , pp. 73–83.

[58] Meyer, K. D. (2004). Foundations of stochastic diffusion search, *Culture* .

[59] Miettinen, K. (ed.) (1999). *Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, GE* (John Wiley & Sons, Inc., New York, NY, USA), ISBN 0471999024.

[60] Miller, P. (2007). The genius of swarm, *National Geographic Magazine* .

[61] Mitchell, H. (2007). *Multisensor Data Fusion: An introduction* (Springer).

[62] M.J. Mataric, M. M. (1993). Synthesizing complex behaviours by composing simple primitives. in *European conference on Artificial life*, Vol. 2, pp. 698–707.

[63] P. Muniganti, A. O. P. (2010). A survey on mathematical models of swarm robotics. in *Physical Agents Network, XI Workshop of Physical Agents.*

[64] Parker, L. E. (2003). Current research in multirobot systems, *Journal of Artificial Life and Robotics* **7**, pp. 1–5.

[65] Pimentel, B. S. and Campos, M. F. M. (2003). Cooperative communication in ad hoc networked mobile robots, in *Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems* (Las Vegas, Nevada), pp. 2876–2881.

[66] Prorok, A., Bahr, A. and Martinoli, A. (2012). Low-cost collaborative localization for large-scale multi-robot systems, in *IEEE International Conference on Robotics and Automation (ICRA)* (RiverCentre, Saint Paul, Minnesota, USA).

[67] R. Poli, J. K. and Blackwell, T. (2007). Particle swarm optimization, *Swarm Intelligence* **1**, 1, pp. 33–57.

[68] Rigatos, G. (2008). Distributed motion planning and sensor fusion for cooperative behavior of mobile robots, in *Proceedings of the EURON/IARP International Workshop on Robotics for Risky Interventions and Surveillance of the Environment* (Benicassim, Spain).

[69] Roumeliotis, S. and Bekey, G. A. (2002). Distributed multirobot localization, *IEEE Transactions on Robotics and Automation* **18**, 5, pp. 781–795.

[70] S. Garnier, J. G. and Theraulaz, G. (2007). The biological principles of swarm intelligence, *Swarm Intelligence* **1**.

[71] Sahin, E. (2005). Swarm robotics: From source of inspiration to domains of applications in swarm robotics, in *Springer Verlag*, pp. 10–20.

[72] Santos, J. and Lima, P. (2010). Multi-robot cooperative object localization-decentralized bayesian approach, in *Robot Soccer World Cup XIII, Lecture Notes in Artificial Intelligence*, Vol. 5949 (Graz, Austria), pp. 332–343.

[73] Seeley, T. (2002). When is self-organization used in biological systems? *Biological Bulletin* , pp. 314–318.

[74] Sharkey, A. (2006). The application of swarm intelligence to collective robots, in *Advances in Applied Artificial Intelligence* (John Fulcher, Idea Group Publishing), pp. 157–185.

[75] Siciliano, B. and Khatib, O. (eds.) (2008). *Springer Handbook of Robotics* (Springer).

[76] Silva, J., Lau, N., Rodrigues, J., Azevedo, J. L. and Neves, A. J. R. (2009). Sensor and information fusion applied to a robotic soccer team, in Springer (ed.), *Robocup 2009: robot soccer worls cup XIII. Lecture notes in artificial intelligence*.

[77] Song, K.-T., Tsai, C.-Y. and Huang, C.-H. C. (2008). Multi-robot cooperative sensing and localization, in *IEEE International Conference on Automation and Logistics, ICAL 2008.*, pp. 431–436.

[78] Soysal O., S. E., Bahceci E. (2007). Aggregation in swarm robotic systems: Evolution and probabilistic control. *Turkish Journal Electrical Engineering* **15(2)**, pp. 199–225.

[79] Verret, S. (2005). Current state of the art in multirobot systems, Technical report, Defence R&D Canada, Suffield.

[80] Wang, S. and Tan, M. (2000). Multi-robot cooperation and data fusion in map-building, in *Proceedings of the* 3^{rd} *World Congress on Intelligent Control and Automation* (Hefei, P.R. China), pp. 1261–1265.

[81] Wang, T., Dang, Q. and Pan, P. (2011). A predict-fuzzy logic communication approach for multi robotic cooperation and competiton, *Journal Of Communications* **6**, 3, pp. 225–231.

[82] Y., C., A., F. and A., K. (1997). Cooperative mobile robotics: Antecedents and directions, *Autonomous robots* **4(1)**, pp. 7–27.

[83] Yamashita, A., Fukuchi, M., Ota, J., Arai, T. and Asma, H. (2000). Motion planning for cooperative transportation of a large object by multiple mobile robots in a 3d environment, in *Proccedings of the 2000 IEEE International*

Conference on Robotics and Automation (San Francisco, CA), pp. 3144–3151.

[84] Yan, Y. and Zhenmin, T. (2009). Control artchitecture for autonomous multi-robot system: Survey and analysis, in *Second International Conference on Intelligent Computation Technology and Automation, 2009. ICICTA'09*, Vol. 4 (USA), pp. 376–379.

[85] Zhao, X., Luo, Q. and Han, B. (2008). Survey on robot multi-sensor information fusion technology, in *Proceedings of the 7th World Congress on Intelligent Control and Automayion* (Chongqing, China), pp. 5014–5018.

[86] Zoghby, N. E., Cherfaoui, V., Ducourthial, B. and Denoeux, T. (2012). Distributed data fusion for detecting sybil attack in vanets, in *Belief Functions: Theory and Applications*, Vol. 164, pp. 351–358.