

# Counter-Ant Algorithm for Evolving Multirobot Collaboration

Abdelhak Chatty, Ilhem Kallel, Adel M.Alimi

► **To cite this version:**

Abdelhak Chatty, Ilhem Kallel, Adel M.Alimi. Counter-Ant Algorithm for Evolving Multirobot Collaboration. in IEEE International Conference on Soft Computing as Transdisciplinary Science and Technology, Oct 2008, France. pp.84-89. hal-00955864

**HAL Id: hal-00955864**

**<https://hal.archives-ouvertes.fr/hal-00955864>**

Submitted on 5 Mar 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Counter-Ant Algorithm for Evolving Multirobot Collaboration

Abdelhak Chatty

Research Group on Intelligent Machines  
Ecole Nationale d'Ingénieurs de Sfax  
University of Sfax, Tunisia  
+216 74 274088

abdelhak\_chatty@ieee.org

Ilhem Kallel

Research Group on Intelligent Machines  
Ecole Nationale d'Ingénieurs de Sfax  
University of Sfax, Tunisia  
+216 74 274088

ilhem.kallel@ieee.org

Adel M. Alimi

Research Group on Intelligent Machines  
Ecole Nationale d'Ingénieurs de Sfax  
University of Sfax, Tunisia  
+216 74 274088

adel.alimi@ieee.org

## ABSTRACT

The use of multirobot systems, is affecting our society in a fundamental way; from their use in hazardous environments, to their application in automated environmental cleanup. In an unknown environment, one of the most important problem related to multirobot systems, is to decide how to coordinate actions in order to achieve tasks in an optimal way. Ant algorithms are proved to be very useful in solving such distributed control problems. We introduce in this paper a modified version of the known ant algorithm, called Counter-Ant Algorithm (CAA). Indeed, the robots' collaborative behaviour is based on repulsion instead of attraction to pheromone, which is a chemical matter open to evaporation and representing the core of ants' cooperation. In order to test the performance of our CAA, we implement, simulate and test our algorithm in a generic multirobot environment. In practical terms, the subdivision of the cleaning space is achieved in emergent and evolving way. A series of simulations show the usefulness of our algorithm for adaptive and cooperative cleanup.

## Categories and Subject Descriptors

Problem Solving, Control Methods, and Search

## General Terms

Algorithms, Design, Reliability, Experimentation, Performance.

## Keywords

Counter-Ant algorithm, Multirobot collaboration  
Collaborative cleaning, pheromone, stagnation recovery

## 1. INTRODUCTION

Research on collective behaviour of social insects provides powerful methods for the design of optimization algorithms and distributed control. In addition to their ability, already surprising to solve a wide spectrum of "static" problems, these techniques offer a high degree of flexibility and robustness in dynamic environments they allow, in particular, addressing more effectively the problems of optimization. Swarm insects have the ability to solve problems with high flexibility (adaptability to environment changing) and robustness (task survival despite disappearance of some individuals). We can note different daily

problems solved by a colony: search for food, construction of the nest, labour division, tasks allocation between individuals, and so on. Most of these problems can be found in the engineering field, computer science and robotics, among others. Ethology researchers discovered that collective behaviour of some social insects is self-organized. The self-organizing characterizes the process in which structures emerge at the collective level, from multiple simple interactions between insects, without being explicitly coded at the individual level. Some interactions, such as an ant following the pheromone trace left by another, collectively help solve difficult problems, like finding the shortest path among several, leading to a food source. Computer scientists and engineers have been able to transform models of social insects' collective behaviour into methods for optimization and control. A new area of research has emerged which aims to convert Ethology knowledge to artificial problem solving: it is swarm intelligence, including "ants algorithm" inspired obviously from ants' behaviour. These algorithms demonstrated their efficiency to resolve optimisation problems by collaborative swarm behaviour [2, 4, 5, 16]. The intrusion of these algorithms in the robotic world improved the communication quality between robots except that optimisation, so the system convergence life, remains problematic [1,6,7,11,12,13,15,17,18]. Can we go by ants algorithms in order to achieve a more efficient multirobot collaboration, while minimising the system convergence time and realizing a best trade-off between optimization and goal to reach? Ant algorithms are proved to be very useful in solving such distributed control problems. We introduce in this paper a modified version of the known ant algorithm, called Counter-Ant Algorithm (CAA). Indeed, the robots' collaborative behaviour is based on repulsion instead of attraction to pheromone, which is a chemical matter open to evaporation and representing the core of ants' cooperation. Indeed, the presence of the pheromone does not inform any more about the presence of the food as it is used in the basic ant algorithm but rather on the location of robots cleaners. The robots' reaction consists henceforth in avoiding paths more covered by this chemical substance. In order to test the performance of our CAA, we had to look firstly for a suitable simulation platform that can represent both swarm behaviour and cooperative robot behaviour [3, 8, 10], and secondly an appropriate representation of evaporating pheromone. In practical terms, the subdivision of the cleaning space is achieved in emergent and evolving way. A series of experimentations show

the usefulness of our algorithm for adaptive and cooperative cleanup.

The paper is organized as follows: section 2 overviews some collaborative swarm robots based on ant algorithms. Sections 3 and 4 present our proposal for a CAA in multirobot collaboration while providing solution to stagnation situation. Section 5 presents and discusses our Counter-Ants simulations and behaviours. Finally, last section concludes the paper.

## 2. COLLABORATIVE SWARM ROBOTS BASED ON ANT-ALGORITHM

Among swarm intelligence techniques, some are mature. In fact, the observation of ants' colony led to ants algorithms [5, 6]. These revolve around a key concept called "stigmergy" which is an indirect activities' coordination in an unknown environment: Ants are attracted to each other thanks to chemical material called "pheromone". It transpires that in an unknown environment, one of the most important problem related to multirobot systems, is to decide how to coordinate actions in order to achieve tasks in an optimal way [11]. Also, it is very important to know what needs to be accomplished and what is the number of robots required for the task. Also, among the multirobot systems cooperation, we note the absence of self-organizing cooperative behaviour. To solve this problem, researchers resort to stigmergy. For example, when a robot concludes that it can not conclude a mission, it marks the unfinished task with a quantity of pheromone. More difficult the task is, the greater the amount of pheromone increases. Thus, robots will be attracted to try to accomplish this task [17]. It is clear that box pushing is more natural if it is accomplished with ants' multirobot system: In fact, the ant tries to move the food alone. She spent moments to test the food resistance by varying the orientation of its body. Changing the direction of the applied strength can be enough to really move the cargo. Should the realignment would not be sufficient, the ant frees prey and find another position to seize the cargo. If multiple replacement attempts are not successful, the ant recruits other ants [11]. Many work such as cited by Kube and Zhang [12, 13] are based on ant algorithm to solve transportation problem. Other research consists to change the ant algorithm using a new technique for multirobot pheromone placement. This technique enables robots to place more pheromones in the tasks that are about to be fulfilled: A very large amount of pheromone is placed in the balls that are almost being moved or are nearly to their destinations, while a small amount of pheromone is placed in the balls, which require a great force to make them move. This helps to attract other robots to complete the task [17]. Except that the amount of pheromone is constant, whereas in [1, 18] is variable and depends on task difficulty and the robots strength. The problem of task allocation in the field of cooperative robotics was led by several studies especially in unknown environment. Each robot has to adapt to its environment without any training stage. Accordingly, swarm intelligence allows self-organization into an unfamiliar environment and adapting behaviours through simple individuals' interactions.

## 3. COUNTER-ANT ALGORITHM FOR MULTIROBOT COLLABORATION

We carried our interest in the cleaning field and we noticed that cleaning an unknown environment by a multirobot system, apparently simple, conceals some complexity in terms of inter-robots coordination and efficiency. In fact, we noticed a congestion in the cleaning zones which influences negatively the convergence time. To improve this function considering the dichotomy "optimization/objective", we propose another view of swarm intelligence inspired from ant colony optimization; it is based on a modified version of ant algorithm called "Counter-Ant-Algorithm" (CAA). To feign well our algorithm, we propose a nominal scenario describing the behaviour of cleaning ant-robots summarized by Figure 1: ants begin by moving randomly hunt rubbish in the ant-hill, which presents in our proposal an unknown environment. Ants move while marking out their paths by pheromone, in other words, they begin to circulate by secreting pheromone fearing that they are followed by other ants. The presence of this substance indicates the paths which are already gone through by other ants. In order to ensure a collaborative cleaning task, ants avoid these paths and circumvent blocked cleaning zones by changing motion direction according to the position and quantity of pheromone (see section 4.2).

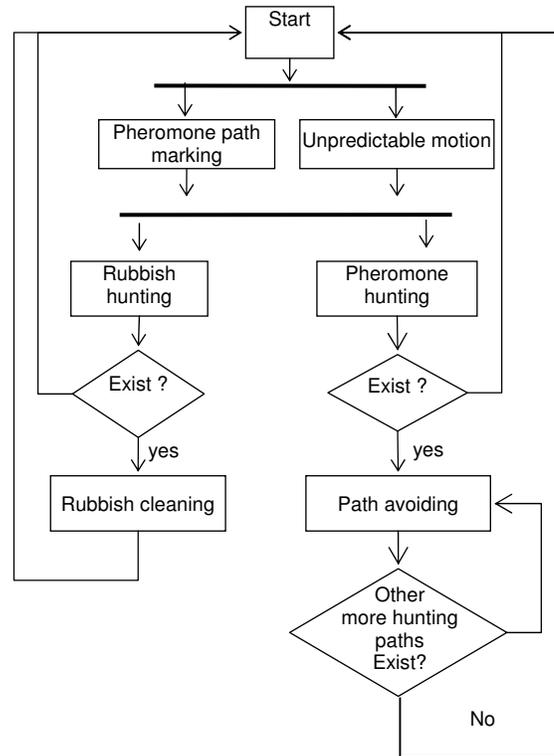


Figure 1. Collaborative cleaning task based on Counter-Ant Algorithm.

Given the evaporation property of the pheromone, the paths eventually disappear which enables ants expand their scanning areas. In other words, as old paths marked by the pheromone no longer exist, then the ants can reach, when they move randomly, new places to look for other rubbish, which were not detected by other ants. This leads us to say that the subdivision of the cleaning area is done in emerging and evolving behaviour.

Figure 2 expresses the pseudo-code of our CAA: the modules "task\_Achievement" encapsulates cleanup operation, "pheromone\_Update" formulates the evaporation phenomenon of the pheromone and "daemon\_Counter\_Reactions" expresses the reaction of the robots, that is repulsion instead of attraction to pheromone; it includes also the solution of stagnation recovery.

```

Procedure Counter-Ant
While (not_termination)
taskAchievement()
pheromoneUpdate()
daemonCounterReactions()
End While
End Procedure

```

Figure 2. Counter-Ant pseudo code.

### 3.1. Pheromone Update

Ants move and marking paths with a constant amount of pheromone ( $\Delta\tau_{x,y}$ ), after a fixed time, the paths of pheromone begin to disappear. The amount of pheromone in each zone of the cleaning environment is updated according the equation (1).

$$\tau_{x,y} = \rho\tau_{x,y} + \Delta\tau_{x,y} \quad (1)$$

Where,  $\tau_{x,y}$  is the amount of pheromone in the zone of coordinates  $x,y$ ;  $\rho$  is the rate of pheromone evaporation and  $\Delta\tau_{x,y}$  is the amount of pheromone deposited when an ant go through the zone  $x,y$ .

## 4. STAGNATION RECOVERY USING PHEROMONE

A problematic case appears: the stagnation of robots' motion due to dead ends. This means that robots lost totally the possibility of moving in their unknown environment. We propose to include in our CAA a stagnation recovery which allows the robots to overtake dead ends and to resolve stagnation problems. The unpredictable motion of ants' robots is accompanied by a secretion of pheromone so that the other ants do not borrow the already marked paths. After a while, the ant-hill (the robot environment) will be blocked by pheromone, this can create what we call "stagnation situations" (SS). In our system, the SS are similar to dead ends [1, 11, 17]: they refer to situations where ants are surrounded by the pheromone; this causes the blocking of their actions because they have to avoid the perceived pheromone.

### 4.1. Stagnation Recovery Using Evaporative Pheromone

The preservation of the pheromone evaporation property has allowed the ants to enlarge their sweeping areas on one hand; on the other hand, reduced the stagnation situations since the paths end up disappearing, which allows the ants to continue cleaning up the dynamic environment. Evaporation of the pheromone is not instantaneous; it requires some time to be completed. This helps to inform, in an interval of time, the maximum number of ants already passed. However, the lifetime of the pheromone is itself a factor of stagnation situations: ants can be found surrounded by tracks of pheromones that have not yet evaporated. The first idea for resolving these problematic situations is to take advantage from evaporation: ants stuck and have to wait till evaporation of one condemning issue, but this causes a lack of cleaning workforce! If all ants are imprisoned, our algorithm will enter a waiting phase until their release. This issue will negatively affect the convergence time of our system because it will be affected by pending active ants that are in stagnation situations.

### 4.2. Stagnation Recovery Using Positions' Pheromone

The solution is then to give ant the ability to become localized according to pheromone position: if its location is parallel to the pheromone, it has to change direction in order to find another way, elsewhere it may be in the presence of other ants. On the other hand, if its location is not parallel, it can move to overcome this stagnation. In this case, the ant doesn't avoid the pheromone since both don't have the same direction and therefore we no longer refer to congestion areas. This idea is encapsulated in daemonCounterReactions() function. Figure 3 presents a flowchart algorithm for stagnation situation.

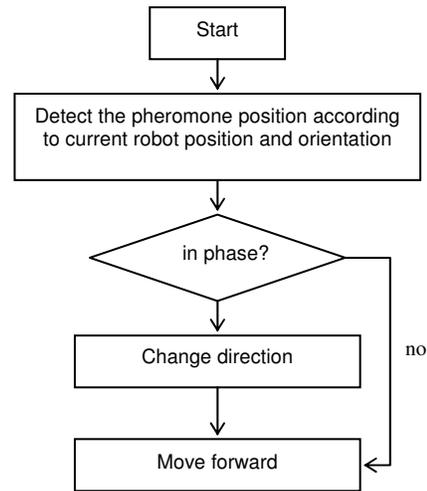


Figure 3. Flowchart of stagnation situations solution.

## 5. SIMULATION AND DISCUSSION OF THE COUNTER-ANTS' BEHAVIOUR

Collaboration between robots is done in a completely unknown environment. It is made up mainly by garbage and cleaning robots which are implemented and generated by the simulation platform "Madkit"<sup>1</sup>. These cleaning robots are provided by a skeletal sensor enabling them to detect the pheromone trace. The robots are completely blind and their motion is completely random. For the cleaning operation, it is question to remove meted. This enables us to focus on our Counter-Ants Algorithm to emphasize our multirobot collaboration method. Moreover, we can consider also that our robots are provided by tanks able to contain a collection of garbage. Figure 4a presents our simulation environment at the starting time  $t=0$  containing five cleaning robots and 80 rubbish. The time of pheromone evaporation is firstly fixed to 45s. Figures 4b is a screen caption after a running time  $t=58,63s$ ; according to pheromone traces, wan can see that the robots move away or cross other marked paths, since the robots' behaviour is based on repulsion instead of attraction to pheromone. The cleaning robot motion is accompanied by the activation of pheromone detectors in order to locate the recent robots' trajectories in the environment. The figure 4c shows that after  $t=133,55s$  the robots are in the course of cleaning meted rubbish. As the paths end up disappearing, it is what explains the existing number of pheromone traces in the environment. The figure 4d shows the end of the cleaning operation after  $t=331,58s$ , and after an additional 45s, all pheromone traces will disappear.

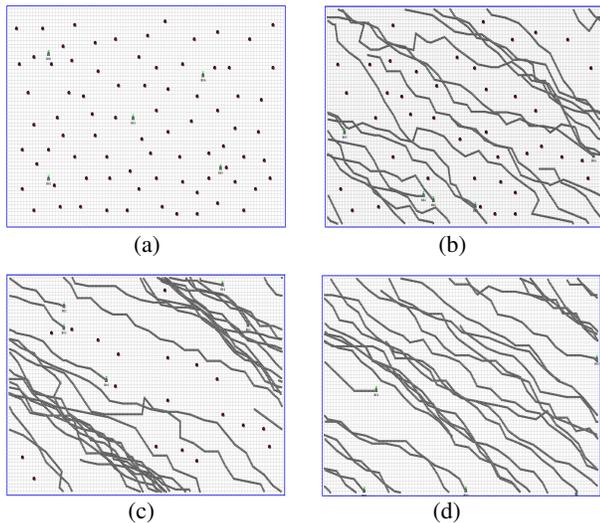


Figure 4. Some simulation' screen captions

### 5.1. Influence Variation of the Time of Evaporation of Pheromone and Number of Robots and Number of Rubbish

Figure 5 shows (from 1 to 2 robots) that in spite of the fixed time of evaporation to 8s, the convergence time is improved, from 1816,09s with only one robot to 1426,17s with two robots. This emphasizes the effectiveness of swarm robotics in the realization

of cleaning operation. This figure also shows (from 2 to 5 robots) that the long life of pheromone paths guarantees the reduction of obstruction zones, since their presence allow robots to recognize previously explored zones. This is ensured thanks to the reaction of robots with respect to the pheromone which consists of repulsion and not attraction. The increase in the number of robots with the persistence of the traces does nothing but improve time of convergence since the robots any more will not seek rubbish in open zones but rather in closed spaces. This carries out us to say that the subdivision of the space of cleaning is done in evolutionary and emergent way.

Figure 5 shows also the existing relations between the convergence time, the number of rubbish and the number of robots. The advantage of swarm robotics is obvious: (from 1 to 2 robots) it makes decrease considerably the convergence time following the fixing of the number of rubbish and to the increase of the number of robots. Increase in the number of rubbish with an increase in the number of robots (from 2 to 5 robots) made decrease the convergence time. This is explained by the adequate place of rubbish compared to the positions of the robots. Indeed, the increase in rubbish and consequently its progressive proximity of the robot facilitate its detection considerably.

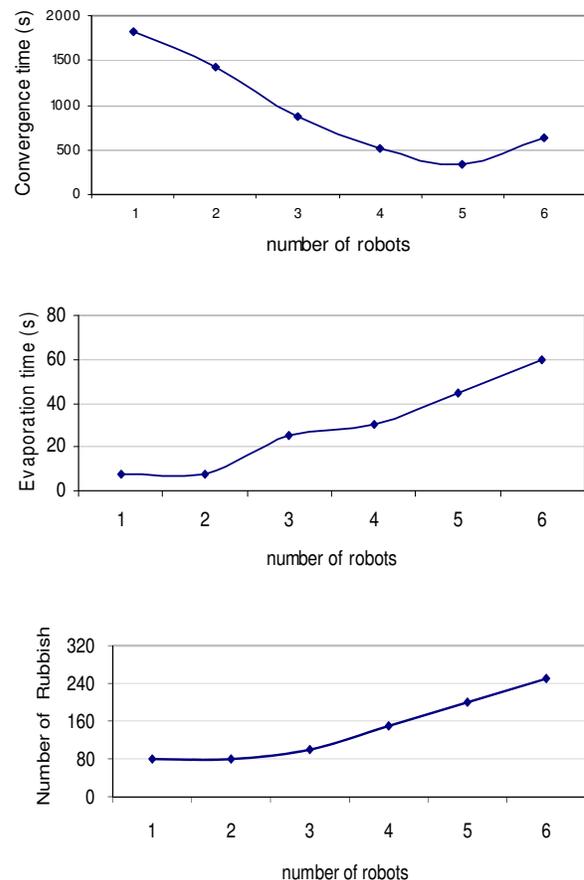


Figure 5. Evaluation of the convergence time following the variation of the number of robots and the evaporation time and the number of rubbish.

<sup>1</sup> <http://www.madkit.org>

These results enabled us to validate our Counter-ants algorithm while focusing on our method of inter-robots collaboration. We can also prove our solution of going beyond stagnation: in spite of the long life of the pheromone evaporation, the robots are able to progress while accelerating the convergence time. However, (from 5 to 6 robots) the persistence of the pheromone trace for a long life (45s to 60s) and the increase of the number of rubbish, increase the convergence time: first the environment will be encumbered by pheromone traces, thing that limits the robot motion, second the overload of the environment by rubbish causes deceleration of the time of convergence.

## 5.2. Counter-Ants vs. Random Behaviour

For better point out the effectiveness of counter-ant behaviour and stagnation recovery, we propose to compare it with a random behaviour. Both simulation environments contain 80 pieces of rubbish and 5 robots. 45s is the time of pheromone evaporation for counter-ant behaviour. The figure 6a watches the initial state of the random system. The figures 6b, 6c shows with  $t=135,72s$  the random operation of cleaning. Starting from the movements carried out by the robots. It is important to mention that the traces carried which concretize the random behavior, do not present the pheromone but rather the movement of the robots. We can notice with  $t=523,25s$  the presence of a cumulating of tracings informing us about their purely random displacement. This contributes to increase the time of convergence. The figure 6d shows a screen caption after  $t=843,56s$  with a cleaned environment. Whereas, the conversion time with our CAA is  $t=331,58s$  according to the figure 4d.

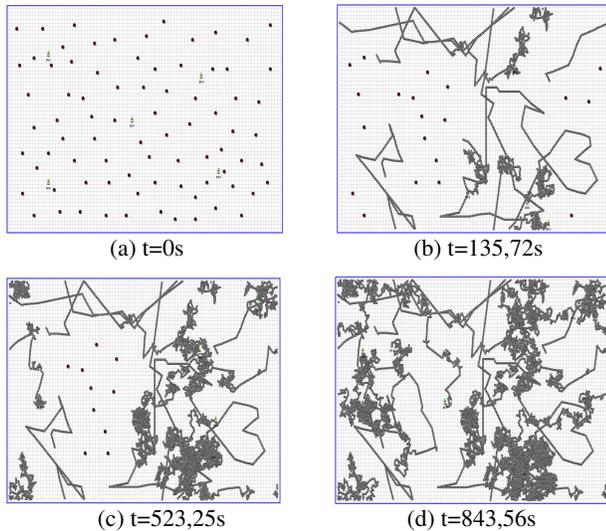


Figure 6. Simulation of hazardous behaviours.

## 5.3. Counter-Ants vs. Alliance Collaborative Cleaning

The *Alliance* architecture suggested by Parker [14] based on faults tolerance is a supervised architecture which consists on collaborating robots to clean two sources of garbage. The communication between robots is done through the diffusion of messages about sources localization via a supervision system. Alliance is certainly based on the faults tolerance however, the

supervision of the robots present in itself a limit of the solution. It is important to mention that the quality of the exploration of the environment was not approached. Alliance presents a specific solution for garbage cleaning gathered and not scattered. However, our CAA can be well applied to scattered garbage cleaning in unknown environment. It is an unsupervised solution with “*stigmergy*” as the base of communication and collaboration. In fact, it possible to solve the cleaning problem while taking into account the quality of the environment exploration and the dilemma “convergence time/goal to reach”.

Table 1. Counter-Ants vs. Alliance

critierion	Counter-Ants	Alliance
<b>Supervision Structure</b>	No	Yes
<b>Method of Communication</b>	Stigmergy : pheromone	sensors+ diffusion of Messages
<b>Faults Tolerance</b>	Yes	Yes
<b>Based Environment</b>	pheromone	Behavior
<b>Exploration</b>	assured	-
<b>Optimization</b>	Yes	No
<b>Application</b>	Cleaning	Cleaning
<b>Convergence Time</b>	Medium	Long

## 6. CONCLUDING REMARKS

This work consists to propose and apply a modified version of the known ant algorithm, called Counter-Ant Algorithm (CAA). It consists on a new collaborative behaviour in relation to the pheromone in a multirobot system. This technique enables robots to place the pheromone while moving randomly. This will no longer attract other robots, but rather to avoid motion paths (already explored and cleaned areas) by other robots. As a result, this method allows improving system efficiency and minimizes the creation of cleaning congested areas. As the old paths marked by the pheromone will disappear, new places become open to explore and clean, which were not detected by other ant-robots. This leads us to say that the subdivision of the cleaning area is done in a evolving, emerging and dynamic manner [9]. As perspectives, we intend to hybridize our algorithm with a soft computing technique, such as fuzzy rule base system, in order to surround more fluently the stagnation situations

## 7. ACKNOWLEDGEMENTS

Authors thank the financial support of the Tunisian General Direction of Scientific Research and Technological Renovation (DGRSRT), under the ARUB program 01/UR/11 02.

## 8. REFERENCES

- [1] Borzello E. and L. D. Merkle, "Multi-Robot Cooperation Using the Ant Algorithm with Variable Pheromone Placement", Proceedings of the IEEE, 2005
- [2] Casillas J., O. Cerdón, I. Fernández de Viana, F. Herrera, Learning cooperative linguistic fuzzy rules using the best-worst ant system algorithm, International Journal of Intelligent Systems 20:4, 2005.
- [3] Cao Y.U., A.S. Fukunaga, A.B. Khang, Cooperative mobile robotics: antecedents and directions, Autonomous Robots 4 (1) (1997) 7–27.
- [4] Dorigo M. and G. DiCaro, The Ant Colony Optimization Meta-Heuristic, New Ideas in Optimization 1999. (New York: McGraw-Hill)
- [5] Dorigo M. and T. Stützle, *Ant colony optimization*. Cambridge, MA, USA: The MIT Press, 2004.
- [6] Dudek G., M. Jenkin, E. Milius and D. Wilkes, A taxonomy for swarm robots, Intelligent Robots and Systems, IROS '93. Proc. of the 1993 IEEE/RSJ, , Vol. 1, July 1993, pp. 441-447.
- [7] Holland O. and C. Melhuish, Stigmergy, self-organization, and sorting in collective robotics, Artificial Life, vol. 5, no. 2, 1999, pp. 173–202.
- [8] Johnson, J. and M. Sugisaka, Complexity science for the design of swarm robot control systems, Industrial Electronics Society, 2000. IECON 2000. 26th Annual Conference of the IEEE, Vol.1, Oct. 2000, pp. 695-700
- [9] Kallel I., S. Mezghani and A. M. Alimi, Towards a Fuzzy Evaluation of the Adaptivity Degree of an Evolving Agent, Proc. of the 3rd International Workshop IEEE GEFS, March 2008, pp. 29-34.
- [10] Kasabov N. and D. Filev, "Evolving intelligent systems: methods, learning, & applications," in Proc. Int. Symposium on Evolving Fuzzy Systems EFS, 2006, pp. 8–18.
- [11] Kube C. R. and E. Bonabeau, "Cooperative transport by ants and robots," Robotics and Autonomous Systems, vol. 30, 2000, pp. 85–101.
- [12] Kube, C.R. and H. Zhang, Collective robotics: From social insects to robots, Adaptive Behavior 2, 189-218, 1994
- [13] Kube, C.R and H. Zhang, Stagnation recovery behaviors for collective robotics, in: Proceedings of the 1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems, IEEE Computer Society Press, Los Alamitos, CA, 1995, pp. 1883–1890.
- [14] Parker, L., "ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation," IEEE Transactions on Robotics and Automation, vol. 14, no. 2, 1998, pp.220-240.
- [15] Purnamadajaja, A.H. and R.A Russell, Pheromone communication: implementation of necrophoric bee behaviour in a robot swarm, Robotics, Automation and Mechatronics, 2004 IEEE, Dec. 2004, Vol. 2, pp.638-643.
- [16] Szu H., P. Chanyagorn, W. Hwang, M. Paulin and T. Yamakawa, Collective and distributive swarm intelligence: evolutionary biological survey, International Congress Series, Volume 1269, August 2004, Pages 46-49.
- [17] Yingying D., H. Yan, J. Jingping, "Multi-Robot Cooperation Method Based On The Ant Algorithm", Proc. of the IEEE, 2003.
- [18] Zhang D., G. Xie, Ji Yu, and L. Wang, "Adaptive task assignment for multiple mobile robots via swarm intelligence approach," Robotics and Autonomous Systems, vol. 55, no. 7, 2007, pp. 572–588.