International Journal of Mechanical Engineering and Robotics Research

ISSN 2278 – 0149 www.ijmerr.com
Vol. 2, No. 2, April 2013

Research Paper

ROBOTS GROUPS PARTITION ACCORDING TO THE NONUNIFORM DISTRIBUTED RESOURCES IN THE ENVIRONMENT

Lida Talebsafa[1]* and Ali Ghafari[1]

*Corresponding Author: **Lida Talebsafa,** ✉ lida.talebsafa@gmail.com

In some tasks, such as moving objects in a swarm the robots are divided into several groups, each of which may vary. In situations where the robots are controlled by a central divide the group is easy. But swarm robots are generally without a central controller. In this paper, an algorithm is proposed to split the group of robots. Algorithm for swarm partition that inspired by honey bees colony in search of food, is designed according to sources in the environment. the group like bees, has participated in a decision. Each robot independently and without knowledge of the decision of other robots, based on observations of the distance to resources in the environment and their values select a resource. Swarm partition and scalability of the proposed algorithm have been studied for different number of robots and resources. For this purpose, the experiments were performed with different parameters, including number of robots, number of resources, and resources' values. Simulation results show that the robots are divided into groups according to the value of resources and the partition error decreased.

Keywords: Robotics team, Bee colony, Group decision-making, Group partition

INTRODUCTION

Biological inspiration is a common theme in robotics. Swarm robotics is a new approach that takes its inspiration from the social insects. The social insect communities of ants, bees, and termites provide a nearly inexhaustible supply of working algorithms and proven system designs that can be applied to robotic swarms. The hypothesis is that robots designed with sensors, actuators, and communications that are similar to those of their natural counterparts will also have similar constraints on how they perceive and interact with the world around them. If the problems we want our robots to solve are similar to those solved by insects, and they often are, then algorithms developed for insect survival can be used for inspirations, design guides, and

[1] Islamic Azad University, Tehran Jonob Branch, Mechatronics Department, Iran.

ultimately even for direct comparisons in performance (Sahin, 2005). In nature, bees are well known as social insects with well organized colonies. Their behaviors such as foraging (Lemmens *et al.*, 2008; and Aleksander *et al.*, 2010), mating and nest site location have been used by researchers to solve many difficult combinatorial optimization and functional optimization problems (Pham *et al.*, 2006a; Pham and Ghanbarzadeh, 2007).

In this paper, we propose the algorithm for divide the swarm of robots according to sources in the environment that inspired by the foraging behavior of colonies of bees. A colony of honey bees can fly on itself in multiple directions simultaneously to exploit a large number of food sources. In a colony, the foraging process starts by sending out scout bees to search for potential flower patches. The scout bees move from one patch to another randomly. When they find a food source, the scout bees return to the hive and perform a famous "waggle dance" (Pham *et al.*, 2007). This dance contains three pieces of information regarding a flower patch: its distance from the hive, the direction in which it will be found, and its quality rating (or fitness) (Tereshko and Loengarov, 2005; and Pham *et al.*, 2006b). This information helps the colony to send its bees to flower patches precisely, without using guides or maps. The information provides from the dance enables the colony to evaluate the relative merit of different patches according to both the quality of the food they provide and the amount of energy needed to harvest it. More follower bees are sent to more promising patches. This allows the colony to gather food in fast and efficiently.

With this idea the group like bees, has participated in a collective decision. Each robot independently and without knowledge of the decision of other robots, based on observations of the distance to resources in the environment and their values select a resource. Whit this choice more robots are attracted to the most valuable resources but other sources are not ignored. Each source can be a task that needs to be done by robots. For example in moving objects with different weights, To prevent leakage, searching for victims in Different environments in terms of risk or victims and such cases Robots need to be divided into groups unequally.

In this paper the scalability of the proposed algorithm has been studied. Scalability in its most general form is defined as how well a solution to some problem will work when the size of the problem increases. In the context of mobile multi robot systems, scalability refers to the overall system's performance if the number of robots increases in relation to the number of tasks at hand (Rana and Stout, 2000).

The rest of this paper is organized as follows. (1) Brief we give a brief description of agent and search area properties. (2) Described the detailed proposed algorithm. (3) Various various simulation experiment results were provided and analyzed in detail. Finally summarizes the main conclusions of this work.

PROBLEM DEFINITION

The aim of the project is that the members shall be divided in accordance with the resources. To achieve this goal the Algorithm based on bee colony in search of more and closer food sources, is proposed.

The proposed scenario is presented under the following assumptions, these assumptions are taken for simplicity to analyze the performance of the system:

- All robots are aware of the value of resources.
- Collective decision-making robot runs only once.
- After the decision, the robot should move toward the source and they are not allowed to move to another source.

Suppose our swarm system is composed of *N* autonomous robots and *M* sources. $Q \in \{q1, ..., qM\}$ is the set of normalized qualities of resources. the number of robots on the resource $i \in \{1, ..., M\}$ is *ni*, the fraction of that is *fi* = *ni*/*N* and the vector of population fraction is *f* = [*f*1, ..., *fM*]*T*. Because the robot does not know the other members decision and select the most valuable resource Alone is not effective for distributed swarm of robots decision making were influenced by the behavior of bees.

ALGORITHM DESCRIPTION

Robots must fit the data resources, select the appropriate resource. The usefulness of this resource for the robot is calculated. Usefulness depends on the value of the resource and Cost, which can be distance or energy required to achieve resource. It must produces a single scalar value that can be compared for the purpose of selecting the resources by robots. Given a robot *R* and a resource *T* one can define *QRT* and *CRT* as the quality and cost, respectively.

The utility can be expressed as follows:

$$U_{RT} = Q_T - C_{RT} \quad ...(1)$$

Details of the proposed algorithm is as follows:

Costs: The cost of a *i* for robot *k* is calculated as (4) that d_i^k is normalized Euclidean distance between the robot and the resource in a 2-D arena. Where (*xi*, *yi*) and (*xk*, *yk*) represent resource's and robot's coordinates in the arena, respectively.

$$D_i^k = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad ...(2)$$

$$d_i^k = \frac{D_i^k}{\sum_{j=1}^{M} D_j^k} \quad ...(3)$$

$$C_{ki} = \log\left(10 \times d_i^k\right) \quad ...(4)$$

Qualities: The quality is a scalar value that represents priority of the resource. Normalized qualities are calculated in (5).

$$q_i = \frac{Q_i}{\sum_{j=1}^{M} Q_j} \quad ...(5)$$

Computing Utilities: The utility of a robot as proposed in (6) depends on both, cost and quality of the chosen resource.

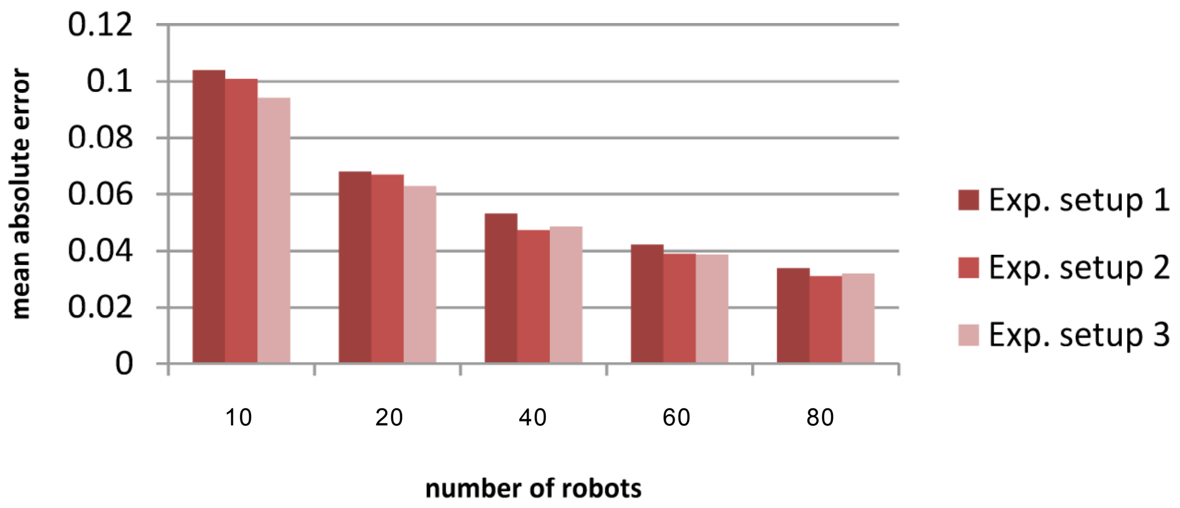
$$U_{ki} = q_i - \alpha \times \log\left(10 \times d_i^k\right) \quad ...(6)$$

Decision-Making: Each robot calculates the utility values for all sources and Selects the most useful of them. The usefulness would be for a different source if not identical distances from the source. The usefulness of a source would be different if the robot's distances to the source are not identical.

RESULTS AND DISCUSSION

In this paper we choose three different experimental setups to compare and study performance and scalability of the proposed

Figure 1: The Robots' Distribution Mean Absolute Error (MAE)



algorithm. The experiments were performed in the same arena where the number of robots, number of resources and resources' quality values were changed as shown in Table 1. For each experimental setup and each swarm size described in this table, 50 experiments were repeated. In order to test the scalability of the proposed algorithm with respect to the size of the swarm, the experiments were performed with 10, 20, 40, 60, and 100 robots. The number of resources was also changed, from two in the experimental setup 1 to four in the experimental setup 2, in order to test the performance of the algorithm with respect to the number of resources. In the experimental setup 3, we used four resources with different quality values to show the adaptability of the swarm to non uniform resources.

We define the mean absolute error (*MAE*) of the robots' distribution As the algorithm performance metrics, which is given by:

$$MAE = \frac{1}{M}\sum_{i=1}^{M}\left| f_i - f_i^d \right| \quad ...(7)$$

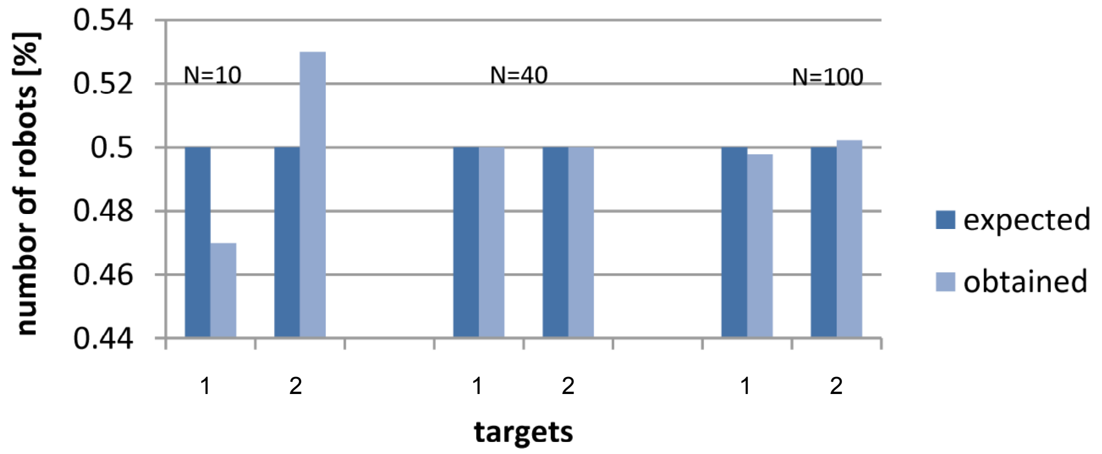
where $f_i^d = q_i$

The mean absolute error is the average value of the absolute distribution error (per resource) that is the result of discrepancy between the expected and the resulting robots' distribution.

Table 1: Parameters Describing Three Arenas

	Arena 1	Arena 2	Arena 3
Area Dimensions [m^2]	3×3	3×3	3×3
Number of Robots	10, 20, 40, 60, 100	10, 20, 40, 60, 100	10, 20, 40, 60, 100
Number of Resources	2	4	4
Resource 1 Quality (q_1)	0.5	0.25	0.1
Resource 2 Quality (q_2)	0.5	0.25	0.2
Resource 3 Quality (q_3)	N/A	0.25	0.3
Resource 4 Quality (q_4)	N/A	0.25	0.4

Figure 2: Bar-Plot Comparison of the Expected vs. the Obtained Robots' Distribution on Two Resources of Same Quality Values, $q_1 = q_2 = 0.5$



The average and the maximum values of MAE obtained from the experiments are presented in Table 2.

We can notice that the average *MAE* and maximum MAE values decrease as the size of the robot swarm increases regardless of the number of resources or their quality values. In Figure 1 *MAE* values changes with the increasing number of robots in each experiment and Resources Is shown. Error is reduced by increasing the number of robots in the group. Also, the error rate has been reduced by increasing the number of sources.

Table 2: Mean Absolute Error (MAE) of the Swarm Robots

	Number of Robots	Average MAE	Maximum MAE
Exp. Setup 1	10	0.104	0.4
	20	0.068	0.2
	40	0.053	0.175
	60	0.0423	0.15
	100	0.0338	0.12
Exp. Setup 2	10	0.1010	0.25
	20	0.067	0.15
	40	0.0473	0.112
	60	0.039	0.075
	100	0.031	0.06
Exp. Setup 3	10	0.094	0.2
	20	0.068	0.125
	40	0.0485	0.1
	60	0.0387	0.091
	100	0.032	0.065

Figure 3: Bar-Plot Comparison of the Expected vs. the Obtained Robots' Distribution on Four Resources of Same Quality Values, $q_1 = q_2 = q_3 = q_4 = 0.25$

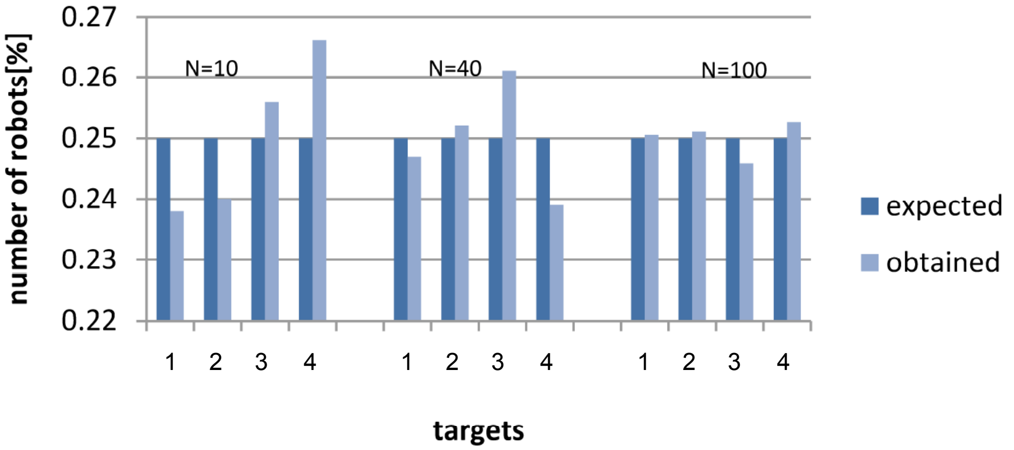
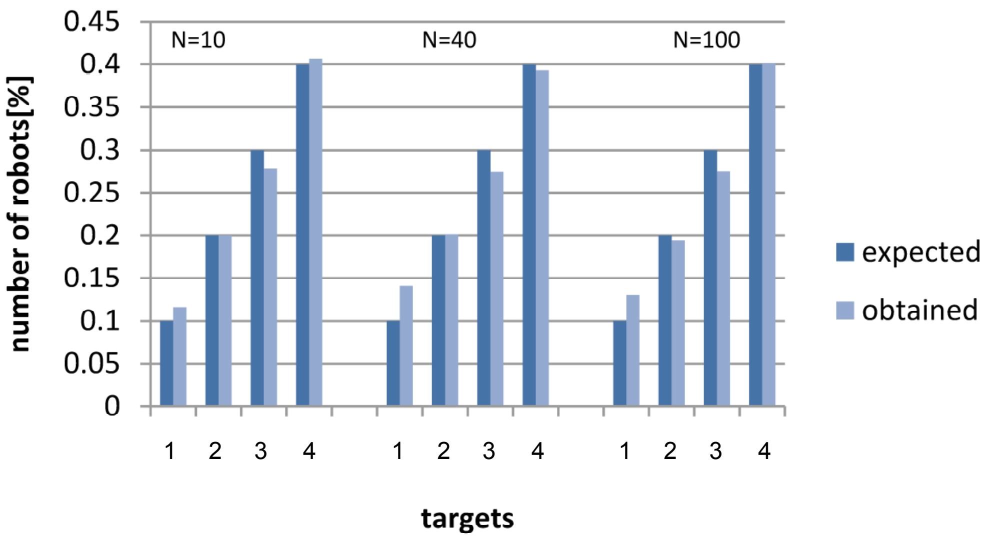


Figure 4: Bar-Plot Comparison of the Expected vs. the Obtained Robots' Distribution on Four Resources of Different Quality Values, $q_1 = 0.1$, $q_2 = 0.2$, $q_3 = 0.3$ and $q_4 = 0.4$



CONCLUSION

There are different applications for robots Robots can be divided into smaller groups and do part of the job, finally, to complete the tasks to be performed. In this paper, we propose the algorithm that inspired by the foraging behavior of colonies of bees and the robots are able to make decisions independently of each other and finally a proportionate division of the swarm have. The algorithm performance is evaluated using the mean absolute error. The error is obtained from the mean value of the difference between the expected number of robots in the groups and the results of simulations. On a broader level, scalability and compatibility with different environments is also important. Experiments in simulation show the scalability of the swarm for different number of robots and distributed resources.

REFERENCES

1. Aleksander Jevtic, Peymon Gazi, Diego Andina and Mo Jamshidi (2010), "Building A Swarm of Robotic Bees", *World Automation Congress*.
2. Lemmens N, Jong S, Tuyls K and Nowe A (2008), "Bee Behaviour in Multi-Agent Systems: A Bee Foraging Algorithm", in Tuyls K, Nowe A, Guessoum Z and Kudenko D (Eds.), *Adaptive Agents and Multi-Agent Systems III: Adaptation and Multi-Agent Learning*, *Lecture Notes in Artificial Intelligence*, Vol. 4865, pp. 145-156.
3. Pham D T, Soroka A J, Ghanbarzadeh A, Koc E, Otri S and Packianather M (2006a), "Optimising Neural Networks for Identification of Wood Defects Using the Bees Algorithm", Proc. of the IEEE International Conference on Industrial Informatics, Singapore.
4. Pham D T, Ghanbarzadeh A, KoC E, Otri S, Rahim S and Zaidi M (2006b), "The Bees Algorithm—A Novel Tool for Complex Optimisation Problems", Proc. of the IPROMS 2006 Conference.
5. Pham D T and Ghanbarzadeh A (2007), "Multi-Objective Optimisation Using the Bees Algorithm", in 3rd International Virtual Conference on Intelligent Production Machines and Systems (IPROMS 2007), Whittles, Dunbeath, Scotland.
6. Pham D T, Koc E, Lee J Y and Phrueksanat J (2007), "Using the Bees Algorithm to Schedule Jobs".
7. Rana O F and Stout K (2000), "What is Scalability in Multiagent Systems?", in Proc. 4th Int. Conf. Autonomous Agents, pp. 56-63.
8. Sahin E (2005), "Swarm Robotics: From Sources of Inspiration to Domains of Application", in E Sahin and W M Spears (Eds.), *Swarm Robotics, Lecture Notes in Computer Science*, Springer, pp. 10-20.
9. Tereshko V and Loengarov A (2005), "Collective Decision Making in Honey-Bee Foraging Dynamics", *Comput Inf. Syst.*, Vol. 9, No. 3, pp. 1-7.