World Automation Congress © 2010 TSI Press.

BUILDING A SWARM OF ROBOTIC BEES

ALEKSANDAR JEVTIĆ[(1)], **PEYMON GAZI**[(2)], **DIEGO ANDINA**[(1)], **MO JAMSHIDI**[(2)]

(1) Group for Automation in Signal and Communications, Technical University of Madrid, Spain
(2) Dept. of Electrical Engineering, College of Engineering, University of Texas at San Antonio, USA

ABSTRACT— Swarm Robotics refers to the application of Swarm Intelligence techniques where a desired collective behavior emerges from the local interactions of robots with one another and with their environment. In this paper, a modified Bees Algorithm is proposed for multi-target search and coverage by an autonomous swarm of robotic "bees". The objective is to find targets in an unknown area, send their estimated locations and fitness values to other robots in swarm which then provide the coverage of the found targets in a self-organized, decentralized way. The robots are equipped with ultrasonic sensors for obstacle avoidance, thermal sensors for target detection, and ZigBee modules for local communication. For the experiments, a small swarm of robots was built to test the performance of the modified Bees Algorithm. The experimental results show that the swarm is self-organized, decentralized and adaptive, and it can be successfully applied to the unknown area search and coverage.

Key Words: swarm robotics, swarm intelligence, bees algorithm, area search and coverage

1. INTRODUCTION

Robots can be utilized in many tasks that would be too risky or too demanding for humans. In applications where a fast response is crucial, multi-robot systems can play an important role thanks to their capability to cover the area. Possible applications for multi-robot systems are search and rescue, planet exploration, monitoring and surveillance, cleaning and maintenance, etc. In order to be successful in such applications, besides a high degree of autonomy and local intelligence, robots require a good level of cooperation and coordination in order to achieve objectives that are impossible for an individual unit. The set of robots should behave like a team and not merely as a set of entities.

Swarm-based systems typically consist of a population of simple agents interacting locally with one another and with their environment [1]. The benefits of cooperation can be significant in situations where global knowledge of the environment does not exist. Individuals within the group interact by exchanging locally available information such that the global objective is achieved more efficiently than it would be if performed by a single individual. Swarm Intelligence is a group of algorithms that are modeled on the social behavior present in animal colonies in nature, such as insects, fish or birds. Examples of emergent behavior in nature are numerous and they gave inspiration to some of the most popular algorithms. Ant Colony Optimization (ACO) models the foraging behavior of ant colonies [2]. Particle Swarm Optimization (PSO) was originally inspired by the social behavior of bird flocking or fish schooling [3]. Bees Algorithm (BA) is a model of the colony of bees in their search for the richer and closer food source [4]. Among others, these algorithms have been applied to optimization problems [5]-[6], and many that can be converted to optimization problems, such as the travelling salesman problem [7]-[8], quadratic assignment problem [9], data mining [10], data clustering [11], image processing [12]-[13], control [14]-[15], etc. Since the colonies in nature are autonomous multi-agent systems they are very suitable for modeling of multi-robot systems.

Swarm Robotics is a new approach to the coordination of multi-robot systems which consist of large numbers of usually, but not necessarily, simple physical robots. It is an interesting alternative to classical approaches to robotics because of some properties of problem-solving by social animals, which is flexible, robust, decentralized and self-organized. In this paper, we applied a modified Bees Algorithm (BA) on the swarm of robots in multi-target search and coverage scenario. The paper is organized as follows. In Section 2, we present the basic BA and the proposed modified BA. The implementation on a swarm of robots is described in Section 3. In Section 4 the experimental results are presented, followed by the discussion in Section 5. Finally, in Section 6 the conclusions are made.

2. PROPOSED METHOD

2.1 Problem statement

In the research proposal presented in this paper, we address the problem of multi–target search and coverage in an unknown area. It consists in distributing a swarm of robots on the map so that the targets with higher fitness values attract more robots. When a robot finds a target, it sends its estimated location and target's fitness value to the other robots in vicinity. If more than one target is found, the robots need to decide for which one to go. The decision is based on the probability which is calculated from the target's fitness value and distance. In such a manner, the system consisting of a large number of robots is capable of finding and covering targets on the map autonomously, in a decentralized way. The resulting distribution favors the most valuable targets but does not omit the targets with a lower fitness value.

The area search and coverage are important tasks in various applications such as search and rescue (SAR), simultaneous localization and mapping (SLAM), area exploration, surveillance and monitoring, etc. Many include dangerous tasks (e.g. mine fields) or inaccessible territories (e.g. planet exploration) making them suitable for the applications of Swarm Robotics. A robotic swarm usually consists of a large number of robots and it is autonomous. The traditional, centralized approach faces the problem of information overload; also, the autonomy of the system is open to question if a robot needs to wait for the command from a higher instance, what we call a central control unit. In some cases, such as the planet exploration, it is impossible to wait for the command because of the time a signal needs to travel to reach its destination.

The solution Swarm Robotics offers has several advantages. Swarm Intelligence provides scalability and robustness. Adding more robots will improve the performance of the overall system and on the other hand, losing some robots will not cause the catastrophic failure. Swarms are decentralized, agents locally communicate with one another and with the environment substantially reducing the amount of the exchanged information. Many aspects of swarm behavior are self-organized which results in qualitatively different patterns that emerge at the global level of a system from interactions among its lower-level components. The swarm-based systems are also flexible, they can adapt to a dynamically changing environment.

2.2 Modified Bees Algorithm

The Bees Algorithm (BA) is a population-based search algorithm that mimics the food foraging behavior of swarms of honey bees [4]. In its basic version, the algorithm performs a neighborhood search combined with random search and can be used for optimization. In this paper, we propose to build an autonomous swarm of robotic "bees" capable of performing search in a two-dimensional space, i.e. the robot arena. The initial number of robots that make up the population is constant, and can only decrease due to robots' hardware failure. In order to apply the BA to swarm of robots some modifications need to be made.

Robots start the search from the preprogrammed initial locations. While moving through the search space they avoid obstacles and other robots. A robot that performs a random search can be referred to as an *available robot*. When it finds a target, or is directed to a target found by another robot, it becomes an *unavailable robot*. Based on their velocity, direction and time of movement, robots can estimate every new location they reach. When a target is found, the available robots that are in its vicinity receive the message that contains the estimated location and the fitness value of the target. If various targets are found, robots probabilistically determine the next target to move to. The probability that a kth robot is displaced to target i is given by:

$$p_i^k = \frac{\omega_i^{\alpha} \eta_i^{\beta}}{\sum_{j=1}^{N} \omega_j^{\alpha} \eta_j^{\beta}} \tag{1}$$

where ω_i and η_i are the fitness value and the visibility of the target at the location (x_i, y_i), respectively, N is the number of targets found in the kth robot's vicinity, and α and β are control parameters ($\alpha, \beta > 0; \alpha, \beta \in \Re$). The visibility of the target at (x_i, y_i) is inversely proportional to its Euclidean distance from the robot's location at (x_r, y_r):

$$\eta_i = \frac{1}{\sqrt{(x_i - x_r)^2 + (y_i - y_r)^2}} \tag{2}$$

It is important to notice that the goal of the original BA is to find a single value which represents the global optimum. The drawback of this algorithm is that the selection of the best sites and the recruitment of the bees are performed in a centralized manner. In function minimization, the lack of collective component in the original BA suggests that the algorithm would reach the same solution with only two bees (minimum needed for solution comparison) in the swarm, but would take longer time to compute. The modified BA we propose resolves this issue by relying on the local robot communication only. In the case scenario described in this paper, finding the global best is not the objective.

3. IMPLEMENTATION

3.1 Robot hardware

The robots were assembled to have the same selection of hardware components (Fig. 1). We used the Lynxmotion Terminator Sumo Robot Kit with four Spur Gear Head Motors as a base to build the robots, although any platform that could support the listed hardware components would be suitable. The DC motors were powered with 12VDC, with 200 RPM, torque of 63.89oz.in (4.6 Kg-cm), 30:1 reduction and 6mm shaft diameter, and they were paired as left-hand and right-hand, in order to be able to perform rotation on-the-spot. Devantech MD22 Motor Driver was used to control the motors' rotation speed and direction. It averaged the PWM signal received from the Arduino microcontroller board to provide a proportional value of the 12VDC from the battery. The four switches on the motor driver's board are used to define the working mode. In the experiments, we used the analog mode which provided satisfactory speed control.

Arduino Duemilanove microcontroller board with ATMEGA328 microcontroller was powered with the 6VDC battery. Zigbee module used for robot communication was connected using the Arduino Xbee shield. Sensors were programmed for I2C communication protocol with the microcontroller. One ultrasonic sensor was mounted on each robot for obstacle detection. Thermal sensor was used to detect the targets. The odometry error inherent to all mobile robots affected their precise localization. It cannot be eliminated, but it can be reduced by using for example the averaging method proposed in [16].

3.2 Coordinator and communication

A computer with a Pentium IV processor at 3 GHz with 2 GB of RAM was used to program the robots and to connect a ZigBee communication module that created a mesh communication network (the coordinator). The ZigBee modules mounted on robots were able to detect a reserved communication channel and connect with the coordinator. This allowed the communication between the robots and the robots with the computer coordinator. We used the broadcast mode, since it allowed each module to communicate with any other module in the network.

4. EXPERIMENTAL RESULTS

The main objective of the experimental setup was to test the performance of the proposed algorithm and not the sensing and pattern recognition capabilities of the robots. Although the robots performed well in detecting the source of heat and sending the estimated location and measured temperature, which was tested in initial experiments, in order to test the performance of the modified BA algorithm a simplified scenario was arranged. A small swarm of three real and two simulated robots was used in search of two targets.

The experiments were performed in 12x12 feet (3.65x3.65m) arena, with randomly distributed obstacles. Robots were placed at the preprogrammed initial locations. When the command was sent from the coordinator, the robots started the random search. After a certain period of time t, the information of two targets was sequentially sent from the coordinator. The information included the targets' estimated locations and their temperature (fitness) values. The robots calculated the probabilities to move to each target as in (1). The real robots were not able to recognize the message sender; therefore, the coordinator was able to simulate two "real" robots that found two different targets.

Two types of experiments were performed. In the first one, the random search time t was changed to test its influence on the odometry error. Single robot was used in order to avoid the collision with other robots, and only one robot was simulated from the coordinator. With each of three robots 30 experiments were conducted in order to obtain the average odometry error value. The results of the first experimental setup are shown in Fig. 2. The search was considered successful if the robot was able to get as close as 30,48 cm (1 foot) from the target. From Fig. 2, we can see that, as expected, while increasing the initial random search time of the robot, the odometry error increased as well. This happened due to the imperfect calibration of the DC motors, non-constant battery voltage, friction of the ground, etc. We can also notice that for the $t \leq 90$ s the experiment success rate was 100%.

The results from the first experimental setup were used to define the second experimental setup. The random search initial time value was set to $t = 30$ s, which guaranteed that the odometry error would be below the success threshold. The scenario involved all three robots in search for two targets whose information was sent form the coordinator. The choice of having two targets in the scenario was based on the number of real robots that were at our disposal. By having three robots and two targets we could test how the robots distributed in the arena based on the targets' fitness values. Four possible events could occur: 1) all robots go for the first target (T1); 2) all robots go for the second target (T2); 3) two robots go for the target T1 and one for the target T2; and 4) one robot goes for the target T1 and two robots go for the target T2.

Since the scenario arena was relatively small, the visibility of the targets was set to be constant, $\eta_1 = \eta_2 = 1$, and the decision on which target to go to was made only on the target's fitness value. The values of the control parameters were also set to $\alpha = \beta = 1$. In order to test the self-organized behavior of the swarm of robots, the fitness values of the two targets sent from the coordinator to the robots were changed. The experimental results are shown in Table I. We can notice that when the fitness values were equal, $\omega_1 = \omega_2 = 50$, in most cases two robots would go for one target and one would go for the other. This was expected since the probabilities of choosing one target or another were equal. By increasing the difference between the fitness values, the distribution would change in favor of the target with the higher fitness value since the probability that a robot chooses that target would also increase.

5. DISCUSSION

The experimental results show that the swarm of robots is self-organized, and adaptive. It is self-organized because the localization of new targets would change the distribution of the robots in the arena accordingly. This also demonstrates that the swarm is adaptive as the robots were able to react to that change in the environment, e.g. when a new target is found. Although the decentralized concept of the swarm was not directly demonstrated through the presented experiments, the robots were not aware of the message sender's identity. Even though the message containing the information about the new targets was sent from the coordinator, it was actually a simulation of the real robots sending that same message. The robots were equipped with the ZigBee communication module, hence capable of sending and receiving the messages, which was previously tested in the initial experiments.

TABLE I
ROBOTS' DISTRIBUTION VS. TARGETS' FITNESS VALUES

Fitness values	T1:T2	Occurrence	Occurrence [%]
ω1 = ω2 = 50	2:1	11	36.67
ω1 = ω2 = 50	3:0	4	13.33
ω1 = ω2 = 50	1:2	13	43.33
ω1 = ω2 = 50	0:3	2	6.67
ω1 =70; ω2 = 30	2:1	18	60.00
ω1 =70; ω2 = 30	3:0	9	30.00
ω1 =70; ω2 = 30	1:2	2	6.67
ω1 =70; ω2 = 30	0:3	1	3.33
ω1 =90; ω2 = 10	2:1	8	26.67
ω1 =90; ω2 = 10	3:0	21	70.00
ω1 =90; ω2 = 10	1:2	1	3.33
ω1 =90; ω2 = 10	0:3	0	0.00

First column represents the fitness values of the two targets.
Second column represents the number of robots, out of three, that went for the targets T1 and T2, respectively.
Third column represents the number of experiments that the distribution from the second column occurred. Total number of experiments for the constant values ω1 and ω2 was 30.
The last column represents the occurrence from the third column compared to total number of experiments, in percentage.

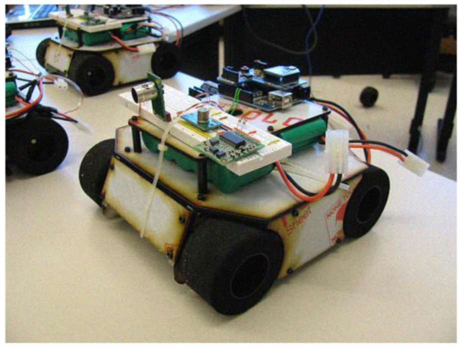

Fig. 1 Sumo robot used in the experiments

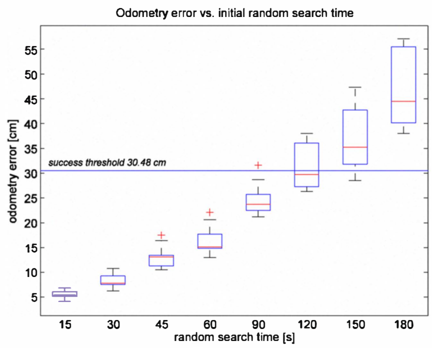

Fig. 2 Results of the first experimental setup: Average odometry error vs. initial random search time

The distribution of the robots depended on the fitness values of the found targets. The decision for which target the robot would go was based on the probability calculated in (1). The targets with higher fitness values attracted more robots in the swarm, which was the goal of the multi-target search scenario. The odometry error inherent to mobile robots was used as an advantage in order to gather the robots in the vicinity of the found targets and not at their exact locations. Still, there is a necessity to maintain the odometry error within the acceptable limits, and this might be a part of the future work.

6. CONCLUSIONS

The proposed robot swarm behavior is a decentralized model of the foraging behavior of bee colonies in nature. For that, a modified Bees Algorithm (BA) was applied to multi-target search and coverage. The experimental results proved that the robot swarm is autonomous and self-organized. The odometry error of the mobile robotic units was used as an advantage to group the robots around the areas of interest, i.e. where the targets were found. Future work may include the research on how to reduce the odometry error for the scenarios that take long time to execute. The scalability of the swarm with a larger number of robots may also be studied through the scenarios where the robot's communication range does not cover the whole search area and where the local communication may get its full effect.

REFERENCES

1. A. Jevtić and D. Andina, "Swarm intelligence and its applications in swarm robotics," *Proc. of the 6th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS '07)*, Tenerife, Canary Islands, Spain, 14 -16 December, 2007, pp. 41-46.
2. M. Dorigo, M. Birattari and T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, Nov. 2006, 1(4):28-39.
3. J. Kennedy and R.C. Eberhart, "Particle swarm optimisation," *Proceedings of the IEEE international conference on Neural Networks Vol. IV*, IEEE service center, Piscataway, NJ, 1995, pp. 1942-1948.
4. D.T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm – a novel tool for complex optimisation problems," *Proc. of the IPROMS 2006 Conference*, 2006, pp. 454-461.
5. D. T. Pham, A. J. Soroka, A. Ghanbarzadeh, E. Koc, S. Otri, and M. Packianather, "Optimising neural networks for identification of wood defects using the bees algorithm," *Proc. of the IEEE International Conference on Industrial Informatics*, Singapore, 2006, pp. 1346-1351.
6. D. T. Pham and A. Ghanbarzadeh, "Multi-objective optimisation using the bees algorithm," in *3rd International Virtual Conference on Intelligent Production Machines and Systems (IPROMS 2007)*: Whittles, Dunbeath, Scotland, 2007.
7. M. Dorigo and L.M. Gambardella, "Ant colony system: a cooperating learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computing*, 1997, 1:1-24.
8. A. Jevtić, A. Jaimes, J. Gomez, D. Andina and M. Jamshidi, "Unmanned aerial vehicle route optimization using ant system algorithm," *Proc. of the 5th IEEE International Conference on System of Systems Engineering (SoSE 2010)*, Loughborough, UK, 22-24 June, 2010.
9. M. Dorigo and T. Stutzle, "ACO algorithms for the quadratic assignment problem," *New ideas in optimization*, McGraw-Hill NewYork, 1999, pp. 33- 50.
10. A.A. Freitas, R.S. Parpinelli and H.S. Lopes, "Data mining with an ant colony optimization algorithm," *IEEE Transactions on Evolutionary Computing*, special issue on Ant Colony Algorithms, 2002, 6(4):321-332.
11. J. Handl and B. Meyer, "Ant-based and swarm-based clustering," *Swarm Intelligence*, 2007, 1(1): 95-113.
12. A. Jevtić, J. Quintanilla-Dominguez, M.G. Cortina-Januchs and D. Andina, "Edge detection using ant colony search algorithm and multiscale contrast enhancement," *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics (SMC2009)*, San Antonio, Texas, USA, 11-14 October, 2009, pp. 2193-2198.
13. A. Jevtić, I. Melgar and D. Andina, "Ant based edge linking algorithm", *Proc. of the 35th Annual Conference of the IEEE Industrial Electronics Society (IECON 2009)*, Porto, Portugal, 3-5 November, 2009, pp. 3353-3358.
14. D.T. Pham, M. Castellani, and A.A. Fahmy, "Learning the inverse kinematics of a robot manipulator using the bees algorithm," in *17th IFAC World Congress COEX*, South Korea, 2008, pp. 493-498.
15. A.A. Fahmy, M. Kalyoncu, and M. Castellani, "Automatic design of control systems for robot manipulators using the bees algorithm," *International Journal of Computer Aided Engineering and Technology*, Special issue for the Bees Algorithm Applications, 2010.
16. A. Gutiérrez, A. Campo, F.C. Santos, F. Monasterio-Huelin, and M. Dorigo, "Social odometry: imitation based odometry in collective robotics," *International Journal of Advanced Robotic Systems*, June 2009, 6(2):129-136.