Distributed Bees Algorithm for Task Allocation in Swarm of Robots

Aleksandar Jevtić, *Member, IEEE,* Álvaro Gutiérrez, *Member, IEEE,* Diego Andina, *Senior Member, IEEE,* and Mo Jamshidi, *Fellow, IEEE*

Abstract—In this paper, we propose the distributed bees algorithm (DBA) for task allocation in a swarm of robots. In the proposed scenario, task allocation consists in assigning the robots to the found targets in a 2-D arena. The expected distribution is obtained from the targets' qualities that are represented as scalar values. Decision-making mechanism is distributed and robots autonomously choose their assignments taking into account targets' qualities and distances. We tested the scalability of the proposed DBA algorithm in terms of number of robots and number of targets. For that, the experiments were performed in the simulator for various sets of parameters, including number of robots, number of targets, and targets' utilities. Control parameters inherent to DBA were tuned to test how they affect the final robot distribution. The simulation results show that by increasing the robot swarm size, the distribution error decreased.

Index Terms—Multirobot systems, scalability, swarm intelligence, task allocation.

I. INTRODUCTION

IN applications that are too risky or too demanding for humans, or where a fast response is crucial, multirobot systems can play an important role thanks to their capability to cover the area. Possible applications are planetary exploration, urban search and rescue, monitoring, surveillance, cleaning, maintenance, among others. In order to successfully perform the tasks, robots require a high degree of autonomy and a good level of cooperation. The set of robots should behave like a team and not merely as a set of entities.

In scenarios that require area coverage, dozens, hundreds, or even thousands of robots can be used. Such a large group of robots, if organized in a centralized manner, could experience information overflow that can lead to the overall system failure. For this reason, the communication between the robots can be realized through local interactions, either directly with

one another or indirectly via environment. The large group of robots acting in such a manner is referred to as swarm [1].

As a result of the growing interest in coordination of multi-robot systems, multirobot task allocation (MRTA) has become an important research topic. The goal is to assign the tasks to the robots in a way that, through cooperation, the global objective is achieved more efficiently. In the scenario proposed in this paper, tasks are represented by targets defined by their qualities and their location in the robot arena. We developed a simulator that implements a multiforaging scenario, and the experimental setup addresses the issue on how to, in a distributed way, assign the robots to the found targets with a expected distribution determined from the targets qualities.

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 multirobot systems, scalability refers to the overall system's performance if the number of robots increases in relation to the number of tasks at hand [2]. The resulting effect on the system's performance can be determined in terms of metrics associated with a particular platform or an operating environment, which in our work refers to dispatching a robot to a remote site marked as a target.

For a swarm of robots engaged in a multiforaging scenario we propose using the distributed bees algorithm (DBA) [3] inspired by the foraging behavior of colonies of bees in nature. When they find a food source, the scout bees return to the hive and perform a famous "waggle dance" in order to recruit other bees. The information about the richness and location of the source is passed using direct communication. Some models of the cooperative behavior of bee colonies with centralized communication or no communication have already been proposed [4]–[6]. In order to avoid the centralized concept of the beehive dance floor, our robots were designed to use broadcast communication to inform other robots in the range about the estimated location and the quality of the found target.

The objective of the proposed algorithm is to assign the robots in a swarm to the found targets in such a way that the final distribution is proportional to the targets' qualities. The targets with associated qualities represent a distributed "food" that requires a usually nonuniform distribution of robots in the area. The algorithm has been previously validated through experiments with real robots [3]. This paper presents the analysis of the proposed algorithm's scalability in a simulated

environment. The simulations were performed for various sets of parameters, including number of robots, number of targets and targets' quality values. Control parameters inherent to the DBA were tuned to test how they affect the final robot distribution.

The rest of this paper is organized as follows. Section II provides a summary of the related work. In Section III, we give a problem statement and describe a general mechanism for distributed task allocation in multiforaging domain. In this section, we present a mathematical model of the proposed DBA algorithm. In Section IV, we define the simulation environment and propose the experimental evaluation for the multirobot system's scalability. We present the experimental results and their statistical analysis, and give a discussion of the approach and the results. Finally, in Section V we make the conclusions.

II. RELATED WORK

Multirobot systems offer the possibility of enhanced task performance, increased task reliability and decreased cost over more traditional single-robot systems. However, multirobot systems must be designed having these issues in mind. Research field of multirobot systems is not new and various architectures that differ in size and complexity have been proposed. Dudek *et al.* [7] provided a taxonomy that categorizes the existing multirobot systems along various axes, including size (number of robots), team organization (e.g., centralized versus distributed), communication topology (e.g., broadcast versus unicast), and team composition (e.g., homogeneous versus heterogeneous).

Rather than characterize architectures, Gerkey and Matarić [8] categorized instead the underlying coordination problems with a focus on MRTA. They distinguish: single-task (ST) and multitask (MT) robots, single-robot (SR) and multirobot (MR) tasks, and instantaneous (IA) and time-extended (TA) assignment. The authors showed that many MRTA problems can be viewed as instances of well-studied optimization problems in order to analyze the existing approaches, but also to use the same theory in the synthesis of new approaches. In order to estimate a robot's performance, they defined utility that depends on two factors, namely expected quality of task execution and expected resource cost. Given a robot R and a task T one can define Q_{RT} and C_{RT} as the quality and cost, respectively, expected to result from the execution of T by R. The resulting nonnegative utility measure is as follows:

$$U_{RT} = \begin{cases} Q_{RT} - C_{RT}, & \text{if } R \text{ is capable of executing} \\ & T \text{ and } Q_{RT} > C_{RT} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

This, however, is not a strict definition of utility which is a flexible measure of performance and can entail arbitrary computation. The only constraint on utility estimators is that they must each produce a single scalar value that can be compared for the purpose of assigning robots for tasks. The problem that we address in this paper is categorized as a "single-task robots, multirobot tasks, instantaneous assignment (ST-MR-IA)," which Gerkey and Matarić proposed to be

solved as a set partitioning problem. However, this requires the combined utilities of all the robots to be known in advance, which is not the case. The method we propose is described in detail in Section III.

What follows is a survey of various multirobot system architectures that have been proposed for solving different problems. We tend to use the above mentioned taxonomies to categorize them.

One of the common approaches for solving the ST-SR and ST-MR problems is a market-based approach which uses auctioning mechanism for task allocation. Matarić *et al.* [9] proposed four different strategies for dynamical task allocation in two different emergency-handling scenarios. The robots bid for tasks and decisions are made by auctioning. The authors concluded that there is no overall best strategy and that the success of a strategy is task-related. Michael *et al.* [10] proposed a market-based approach for robots formation control. They associated multiple tasks with predefined spatial locations that define a formation.

A thorough overview of market-based approaches for MRTA was given by Dias *et al.* [11]. A common drawback of these approaches is the underlying auctioning mechanism which requires all the bids from the robots to be gathered at one auctioning point. The main advantage of the method we propose is that, although it imposes certain communication cost for sending the information of the found targets, the robots make decisions autonomously and in a distributed manner. This is not the case with market-based approaches that feature a partial distribution, where robots are divided into subteams that take decisions in a centralized manner. For this reason, scalability in market-based approaches is often limited by the computation and communication needs that arise from increasing auction frequency, bid complexity, and planning demands.

Environment exploration and mapping are common applications for multirobot systems. Franchi *et al.* [12] proposed a sensor-based random graph method for cooperative robot exploration. They addressed the issue of system's performance with respect to exploration time and traveled distance. The authors showed that by adding more robots the system could scale-up, but its performance was highly dependent on the initial team deployment, giving better results when the robots started grouped in a cluster than if scattered in the environment. Another approach proposed by Burgard *et al.* [13] treats the unknown environment exploration as a ST-SR problem, where individual robots select a new target location based on its distance and utility. Although the experimental results show the advantages of collaboration, the proposed centralized approach cannot be applied if not all robots can communicate with each other.

Decentralized coordination of robots has various advantages over more traditional centralized approaches. It can be applied to reduce the communication burden on multirobot system [14], especially for large teams of robots. In some applications communication can be difficult to implement or no communication exists at all. Joordens and Jamshidi [15] proposed a decentralized coordination for a swarm of underwater robots which is based on consensus control. Another decentralized

strategy for dynamical allocation of tasks that requires no communication among robots was proposed by Berman *et al.* [16]. But often, as in case of multirobot area coverage [17], the decentralized coordination and distributed decision-making is applied having one goal in mind, that the global objective is achieved more efficiently.

A. Bio-Inspired Coordination of Multirobot Systems

Robot swarms are multirobot systems that typically consist of a large population of simple robots interacting locally with one another and with their environment [18]. These systems draw inspiration from animal swarms in nature but their design is not constrained by biological plausibility. Their main feature is decentralized coordination which results in a desired behavior that emerges from the rules of local interactions.

The self-organizing properties of animal swarms such as insects have been studied for better understanding of the underlying concept of decentralized decision-making in nature [19], but it also gave a new approach in applications to multiagent system engineering and robotics. Bio-inspired approaches have been proposed for multirobot division of labor in applications such as exploration and path formation [20], multisite deployment [21], or cooperative transport and prey retrieval [22], [23].

The bottom-up design topology inherent to bio-inspired multirobot systems provides them with one or more of the following features, such as being autonomous, scalable, robust and adaptive to changes in their environment. On the other hand, the collective behavior has emergent properties that give them the ability to produce unpredictable patterns. One way of dealing with the unpredictability issue is statistical analysis through experiments, as proposed in this paper.

B. Scalability

Task allocation scenarios include a set of tasks that may have different priorities and require one or more robots to be assigned to their execution. A very important property of multirobot systems is the ability to scale-up with respect to the number of robots or the number of tasks at hand. However, scalability of multirobot systems and multiagent systems in general has been analyzed from various perspectives including the total number of agents involved, the size of the communication data, the number of rules the agents operate with, or the agents' diversity [2].

In order to evaluate the scalability of a given multirobot system we need to identify a performance metrics. Various MRTA methods exist but, to the best of our knowledge, a comprehensive analysis tool for the scalability of such methods has not been given. Some mathematical models that have been proposed could serve as guidelines in multirobot system design, but different scenarios to which these systems are applied usually do not permit us to maintain within the proposed framework.

Lerman *et al.* [24] proposed a mathematical model for MRTA in dynamical environments. The authors assumed that robots were able to observe tasks in order to discriminate their types, but also to discriminate the tasks that other robots were

assigned to. Robots had limited sensing capabilities and could not directly communicate. The lack of communication made the system more robust to failures, but also more susceptible to noise from the sensors, and requires more time for exploration of available tasks.

Top-down design methodologies apply the classical control theory for performance estimation of distributed agent-based systems. While it is possible to establish bounds on the system behavior and provide performance guarantees, they heavily rely on the available bandwidth for robot communication and they are more sensitive to noise. The need for resources becomes even a bigger issue as the number of robots increase. There is therefore a very natural tendency to apply bottom-up methodologies which produce autonomous, scalable and adaptable systems requiring minimal communication [25].

Broadcast communication provides quick propagation of tasks' information within the multirobot system but extensive use of communication channel can affect the system's scalability. Previously described market-based approaches suffer from a large requirement in terms of communication bandwidth as they use broadcast messages to auction for the tasks. Farinelli *et al.* [26] proposed a mechanism based on token passing for cooperative object retrieval, which scales up for reliable sending of broadcast messages. The authors made a comparison of their method with market-based approaches and the ones based on iterative broadcast communication. Their results show that the ability of the system to adjust to the available communication bandwidth provides guarantees for better performance.

III. DISTRIBUTED TASK ALLOCATION

A. Problem Definition

Based on the described taxonomy, our multirobot system can be categorized as homogeneous and distributed, using broadcast communication. We address a problem of single-task robots, multirobot tasks and instantaneous assignment (ST-MR-IA). The task allocation scenario we study considers the environment that contains a number of tasks that could be of same or different importance and robots that are equally capable of performing each task but can only be assigned to one at any given time. More specifically, the tasks are targets with their associated qualities. The quality of a target is an application-specific scalar value that may represent target's priority or complexity, where a higher value requires more robots to be allocated. For example, it could represent the richness of the mineral or water source on a planet that we want to harness, the amount of garbage to be collected in a public space, or the number of injured people in a need for assistance in urban search and rescue scenario. In this paper, we do not consider how these values are obtained.

The proposed scenario is presented under the following assumptions.
  1) All the targets are made available to all the robots. This is done by setting the broadcast communication range of the robots to cover the entire arena.
  2) Robots take decisions once all the targets in the arena are found, unless they were the ones that found a target

in which case they are automatically allocated to that target. The total number of targets is preset in robots' internal memory and it depends on the experimental setup.

3) Reallocation to another target is not allowed.

These assumptions are taken for simplicity; otherwise, it would be difficult to analyze the performance of the system due to the unpredictability of the robots' distribution prior to target allocation.

Consider a population of N robots to be allocated among M targets. Let $Q \in \{q_1, \ldots, q_M\}$ denote the set of normalized qualities of all available targets. We denote the number of robots on the target $i \in \{1, \ldots, M\}$ by n_i, a nonnegative integer. The population fraction allocated to target i is $f_i = n_i/N$, which represents the target's relative frequency, and the vector of population fraction is $\mathbf{f} = [f_1, \ldots, f_M]^T$. The expected distribution is the set of desired population fractions for each target, $\mathbf{f^d} = [f_1^d, \ldots, f_M^d]^T$, where $f_i^d = q_i$. The usage of fractions rather than integers is practical for scaling, but it also introduces a distribution error as the fractions can take only certain values that are defined by the swarm size.

A relevant concept from set theory could be used to observe this as a set partitioning problem. A family X is a partition of a set E if and only if the elements of X are mutually disjoint and their union is E

$$\bigcap_{x \in X} = \emptyset$$
$$\bigcup_{x \in X} = E. \tag{2}$$

However, for the proposed scenario the system optimization based on the maximum utility cannot be applied because the combined utilities of the robots are unknown as robots have no knowledge of the decisions taken by other robots. Therefore, we propose the DBA.

B. Distributed Bees Algorithm

When a robot receives information about the targets it calculates the utilities with respect to those targets. The utility depends on the target's quality value and the related cost, i.e. the robot's distance from the target. This is the basic concept behind the DBA that was introduced in [3], but here it is described in details.

1) *Costs:* The cost of a target i for robot k is calculated as the Euclidean distance between the robot and the target in a 2-D arena

$$d_i^k = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \tag{3}$$

where (x_i, y_i) and (x_k, y_k) represent target's and robot's coordinates in the arena, respectively.

However, to calculate the utility we use the target's visibility defined as the reciprocal value of the distance

$$\eta_i^k = \frac{1}{d_i^k}. \tag{4}$$

2) *Qualities:* The quality is a scalar value that represents priority, or the complexity of the target. Normalized qualities are calculated as fractions of the sum of qualities of all available targets

$$q_i = \frac{Q_i}{\sum_{j=1}^M Q_j} \tag{5}$$

where Q_i is a quality of the target i. In real-world scenarios, the quality of a region of interest is an estimated value that is as a result of sensor-readings or a previously acquired knowledge.

3) *Computing Utilities:* The utility of a robot as proposed in (1) depends on both, cost and quality of the chosen target. We define the utility as a probability that the robot k is allocated to the target i, and it is calculated as follows:

$$p_i^k = \frac{q_i^\alpha \eta_i^\beta}{\sum_{j=1}^M q_j^\alpha \eta_j^\beta} \tag{6}$$

where α and β are control parameters that allow us to bias the decision-making mechanism toward the quality of the solution or its cost, respectively ($\alpha, \beta > 0$; $\alpha, \beta \in \Re$). From (6) it is easy to show that

$$\sum_{i=1}^M p_i^k = 1. \tag{7}$$

4) *Decision-Making:* The underlying decision-making mechanism of the DBA algorithm adopts the roulette rule, also known as the wheel-selection rule. That is, every target has an associated probability with which it is chosen from a set of available targets. Once all the probabilities are calculated as in (6), the robot will choose a target by "spinning the wheel."

It should be noticed that the resulting robots' distribution depends on their initial distribution in the arena, i.e. their distances from each target prior to target allocation. Therefore, robots' utilities will differ with respect to the same target if their distances from that target are not equal. Since a combined robots utility cannot be computed due to a distributed nature of the proposed algorithm, the quality of the targets is used as the only measure for the expected robots' distribution. Although the overall cost efficiency of the swarm is not analyzed in this paper, target's visibility as used in (6) makes closer targets more attractive to robots.

IV. EXPERIMENTAL EVALUATION

In the following, we describe the simulation environment and experimental setup, and we report the simulation results in order to analyze the scalability of the proposed system.

A. Simulator

Our simulation platform is a fast, specialized multirobot simulator for the e-puck robots described in [27]. It is a simple and effective simulator implementing 2-D kinematics. A screenshot of the simulator is shown in Fig. 1. In our simulations, the e-puck is modeled as a cylindrical body of 3.5 cm in radius that holds eight infrared (IR) proximity sensors distributed around the body, three ground sensors on the lower-front part of the body and a range and bearing

TABLE I
PARAMETERS DESCRIBING THREE ARENAS USED IN EXPERIMENTS

	Arena 1					Arena 2					Arena 3				
Area dimensions [m^2]	1.5 × 2.125					1.5 × 2.125					1.5 × 2.125				
Number of robots	10	20	40	60	100	10	20	40	60	100	10	20	40	60	100
Experiment duration [time steps]	400	400	400	300	200	400	400	400	300	200	400	400	400	300	200
Time step duration [s]	0.1					0.1					0.1				
Initial area radius [m]	0.4	0.4	0.4	0.4	0.5	0.4	0.4	0.4	0.4	0.5	0.4	0.4	0.4	0.4	0.5
Number of targets	2					4					4				
Target radius [m]	0.09					0.09					0.09				
Target 1 location (x, y) [m]	(−0.45, 0.75)					(−0.45, 0.75)					(−0.45, 0.75)				
Target 2 location (x, y) [m]	(0.45, −0.75)					(0.45, −0.75)					(0.45, −0.75)				
Target 3 location (x, y) [m]	N/A					(−0.45, −0.75)					(−0.45, −0.75)				
Target 4 location (x, y) [m]	N/A					(0.45, 0.75)					(0.45, 0.75)				
Target 1 quality (q_1)	0.5					0.25					0.1				
Target 2 quality (q_2)	0.5					0.25					0.2				
Target 3 quality (q_3)	N/A					0.25					0.3				
Target 4 quality (q_4)	N/A					0.25					0.4				

*Targets have a form of a circle. Without loss of generality, their radius and location were intuitively chosen.
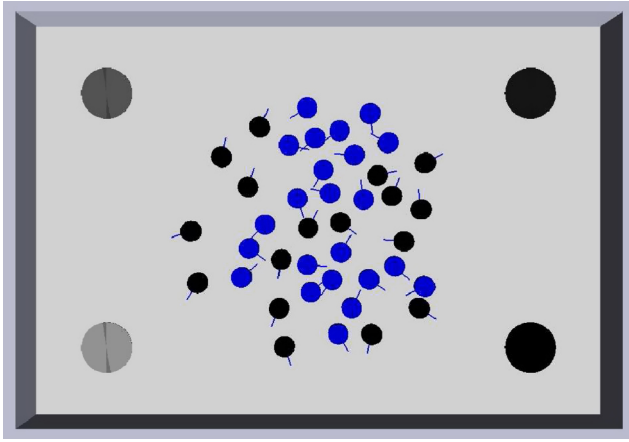

Fig. 1. Simulator screenshot. Experimental setup included 40 robots engaged in search for four targets of different qualities represented by different grey-level intensity. Robots are programmed for obstacle avoidance, when robot detects an obstacle its color changes from black to blue to mark his new state. Once the robot has taken a new direction, its color goes back to black.

communication sensor. IR proximity sensors have a range of 5 cm, while the communication range of the E-puck Range & Bearing module was set to cover the whole arena. For the three types of sensors, real robot measurements were sampled and the data was mapped into the simulator. Furthermore, uniformly distributed noise was added to the samples in order to effectively simulate different sensors; ±20% noise is added to the IR sensors and ±30% to the ground sensors. In the range and bearing sensor, noise is added to the range (±2.5 cm) and bearing (±20°) values. A differential drive system made up of two wheels is fixed to the body of the simulated robot. At each time step (100 ms), the robot senses the environment and actuates. The robot speed has been limited to 6 cm/s when moving straight and 3 cm/s when turning.

B. Experimental Setup

Three different experimental setups have been chosen to compare and study performance and scalability of the proposed DBA algorithm. The setups were carried out in the

same arena where the number of robots, number of targets and targets' quality values were changed as shown in Table I. Additional experimental setup was created in order to analyze the effect of the control parameters α and β on the resulting distribution. Each experiment was repeated 50 times in order to perform a statistical analysis of the results.

C. Simulation Results and Discussion

In order to test the scalability of the proposed DBA with respect to the size of the swarm, the experiments were performed with 10, 20, 40, 60, and 100 robots for the experimental setup 1, and 20, 40, 60, and 100 robots for the experimental setup 2 and the experimental setup 3. The number of targets 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 targets. In the experimental setup 3, we used four targets with different quality values to show the adaptability of the swarm to a nonuniform distribution of the "food" in the environment. This is also the most realistic scenario. Finally, the experimental setup 4 was created to test how by changing the ratio of the control parameters α and β we can affect the resulting robots' distribution.

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

$$MAE = \frac{1}{M} \sum_{i=1}^{M} \left| f_i - f_i^d \right| \qquad (8)$$

where $f_i^d = q_i$.

As the name suggests, the mean absolute error is the average value of the absolute distribution error (per target) that is the result of discrepancy between the expected and the resulting robots' distribution. For each experimental setup and each swarm size described in Table I, 50 experiments were performed. The average and the maximum values of MAE obtained from the experiments are presented in Table II and graphically shown in Fig. 2. We can notice that the average MAE and maximum MAE values decrease as the size of the
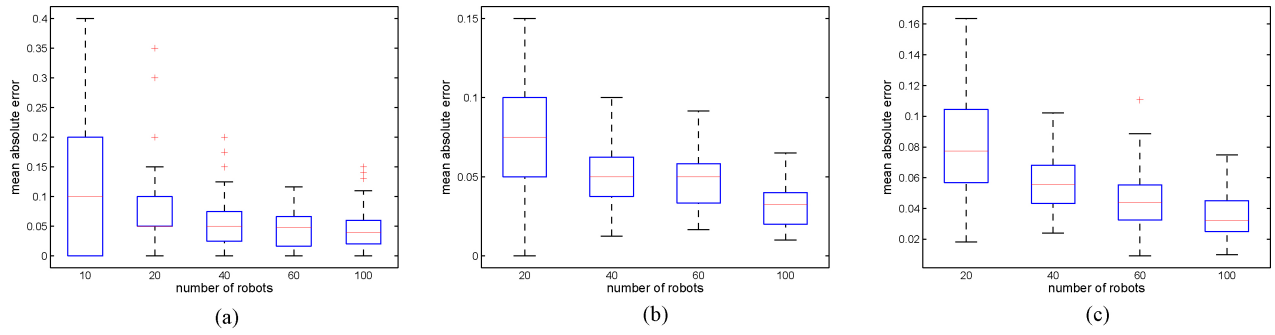
(a) (b) (c)

Fig. 2. Box-plot comparison shows the robots' distribution mean absolute error (MAE) with respect to the swarm size. (a) Experimental setup 1. (b) Experimental setup 2. (c) Experimental setup 3. Each box-plot comprises observations ranging from the first to the third quartile. The median is indicated by a horizontal bar, dividing the box into the upper and lower parts. The whiskers extend to the farthest data points that are within 1.5 times the interquartile range. Outliers are shown with a plus symbol. The values were obtained from 50 experiments performed for each swarm size within each experimental setup.
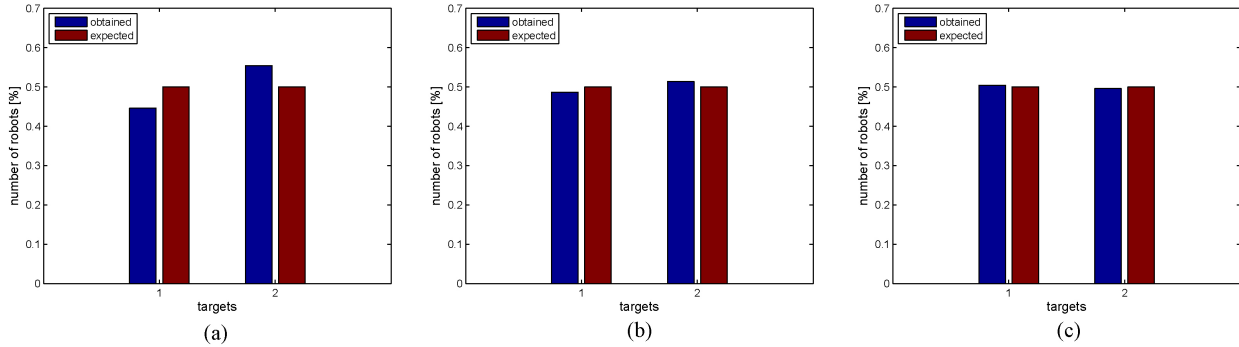

(a) (b) (c)

Fig. 3. Bar-plot comparison of the expected (red) versus the obtained (blue) robots' distribution on two targets of same quality values, $q_1 = q_2 = 0.5$. Fifty experiments were performed for each of the following swarm sizes. (a) 10 robots. (b) 40 robots. (c) 100 robots.

TABLE II
MEAN ABSOLUTE ERROR (MAE) OF THE ROBOTS' DISTRIBUTION

	Num. of Robots	Average MAE	Maximum MAE
Exp. setup 1	10	0.1140	0.4000
	20	0.0820	0.3500
	40	0.0555	0.2000
	60	0.0482	0.1167
	100	0.0461	0.1100
Exp. setup 2	10	0.0941	0.1750
	20	0.0720	0.1500
	40	0.0500	0.1000
	60	0.0475	0.0917
	100	0.0313	0.0650
Exp. setup 3	10	0.0979	0.2500
	20	0.0790	0.1500
	40	0.0526	0.0875
	60	0.0478	0.0790
	100	0.0343	0.0750

* Parameters for each experimental setup are described in Table I.
** The average MAE and the maximum MAE values were obtained from 50 experiments performed for the each swarm size within the each experimental setup.

robot swarm increases regardless of the number of targets or their quality values. This was expected because of the probabilistic target allocation mechanism applied in (6).

The effectiveness of the algorithm in terms of increased number of targets is shown in Figs. 3 and 4. The results show that the average and the maximum MAE values decreased for larger swarms in case of four targets of the same quality.

It should be noticed that the allocation of ten robots to four targets produces an error that is the result of the cardinality of the robot swarm. It is not physically possible to partition the swarm in order to obtain the expected target allocation (2.5 robots per target).

Another inherent source of error results from the assumption that the robots that had found a target are not allowed to reallocate to another target, therefore they are not involved in the decision-making process. Also, it is assumed that the robots wait for a predetermined number of targets to be found before they make a decision, which can result in the same target being found by more than one robot. This fraction of the robot swarm also produces an error in the final distribution because they cannot reallocate to another target. The algorithm's performance is analyzed having these issues mind.

In order to test the ability of the robot swarm to adapt to a nonuniform distribution of "food" in the environment, the experiments were performed for four different targets (experimental setup 3). The robots' distribution changed according to a new set of targets' quality values, as shown in Fig. 5. In the same figure we can also notice that the resulting robots' distribution, with respect to the expected distribution, is slightly in favor of the less valuable targets. This is another consequence of the robots that had found a target not being able to reallocate, and it is especially evident for smaller robot swarms. For example, let us consider a swarm of ten robots in search of four different targets, as shown in Fig. 5(a). If
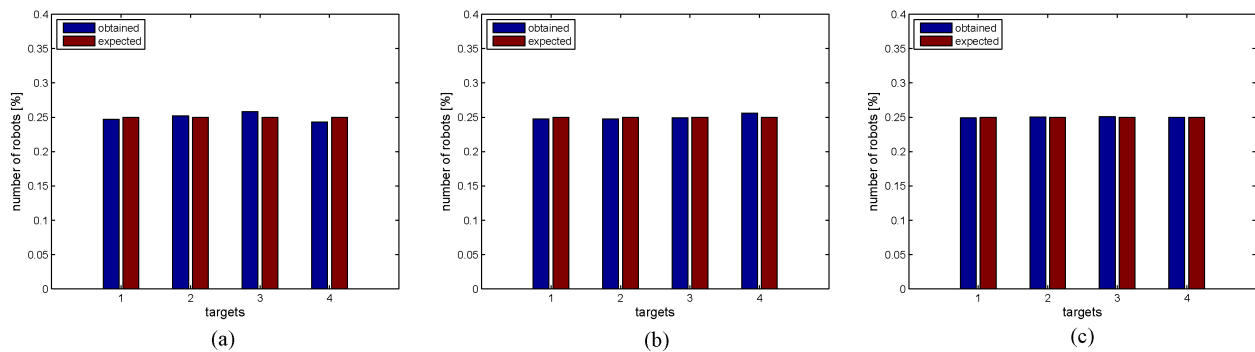
Fig. 4.   Bar-plot comparison of the expected (red) versus the obtained (blue) robots' distribution on four targets of same quality values, $q_1 = q_2 = q_3 = q_4 = 0.25$. Fifty experiments were performed for each of the following swarm sizes. (a) 20 robots. (b) 60 robots. (c) 100 robots.
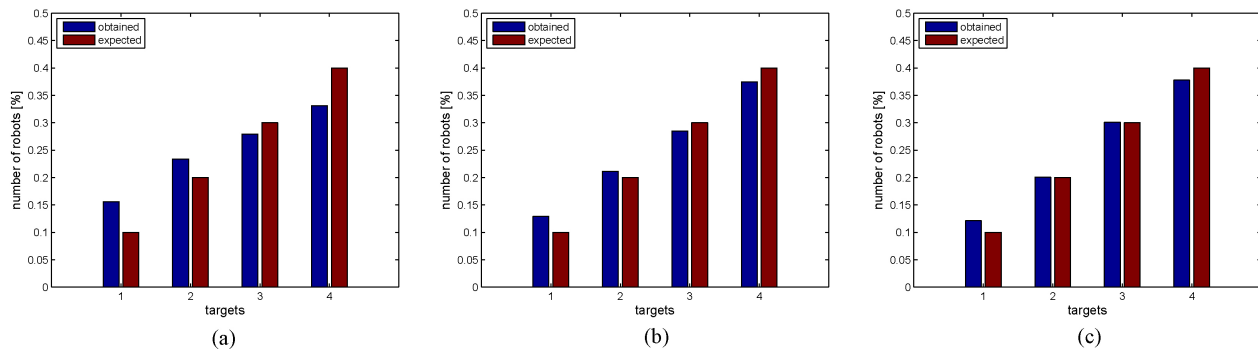

Fig. 5.   Bar-plot comparison of the expected (red) versus the obtained (blue) robots' distribution on four targets of different quality values, $q_1 = 0.1$, $q_2 = 0.2$, $q_3 = 0.3$, and $q_4 = 0.4$. Fifty experiments were performed for each of the following swarm sizes. (a) 20 robots. (b) 60 robots. (c) 100 robots.
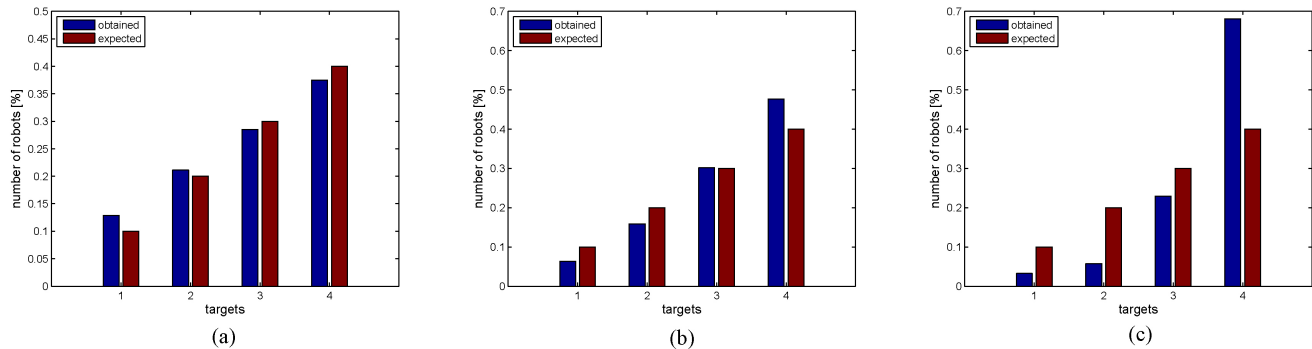

Fig. 6.   Effects of the control parameters, α and β, on the final robots' distribution. Target allocation was performed with 60 robots as described in the experimental setup 3 consisting of four targets with different quality values: $q_1 = 0.1$, $q_2 = 0.2$, $q_3 = 0.3$, and $q_4 = 0.4$. The results of the robots' distribution per target is shown for the following values of α/β ratio. (a) $\alpha/\beta = 1$. (b) $\alpha/\beta = 2$. (c) $\alpha/\beta = 5$. The values were obtained from 50 experiments for each scenario.

in the random target search process two robots find the target with the associated quality value of 0.1, then the final relative frequency for this target cannot be less than 0.2 (2 out of 10 robots) which is already above the expected value of 0.1. Although for the larger swarms the effect of the initial robot distribution becomes less relevant, it is always present.

The control parameters, α and β, were introduced in (6) to compensate for the biased distribution, but also to allow us to give more relevance to either the quality of the targets or the cost of reaching them. Therefore, in the experimental setup 4 we increased the α/β ratio to give more relevance to the quality value of the targets on the expense of their distances from the robots. The resulting robots' distributions per target for different values of the α/β ratio are presented

TABLE III

EFFECTS OF CONTROL PARAMETERS ON ROBOTS' DISTRIBUTION

α/β Ratio	Average MAE	Maximum MAE
1	0.0478	0.1083
2	0.0525	0.1000
5	0.1415	0.2083

∗ The values were obtained from 50 experiments performed on the swarm of 60 robots in search for four targets with different quality values (experimental setup 3).

in Fig. 6. Results show that, by tuning the control parameters, the final robot distribution can change in favor of the more valuable targets but with an increase in the average MAE (see Table III). It is reasonable to expect that by decreasing

the α/β ratio the cost efficiency of the robot swarm would improve in terms of the distance traveled, however, the *MAE* is also expected to increase. Further analysis of the effect of the control parameters will be a part of the future work.

V. Conclusion

Various applications for large multirobot systems require efficient task allocation in terms of individual and combined robots' utilities. The quality of the solution is analyzed using a defined performance metrics, which in our case was a mean absolute error of the resulting robots' distribution with respect to the qualities of the available targets in the robot arena. In case of large, autonomous, multirobot systems, the scalability and the ability to adapt to different environments are the features of utmost importance. Our experiments through simulation showed that the proposed DBA provides the robot swarm with scalability in terms of the number of robots and number of targets, but also with adaptability to a nonuniform distribution of the targets' qualities.

The importance of the control parameters, α and β, is that they provide a mechanism to adjust the robot swarm behavior depending on the task at hand and the available resources. In this paper, we changed the values of α and β in order to bias the resulting robots' distribution toward the more favorable targets. Future work will include the analysis of the effect of these parameters in terms of the task-allocation cost with respect to the distance traveled by the robots.

References

[1] G. Beni and J. Wang, "Swarm intelligence in cellular robotic systems," in *Proc. NATO Adv. Workshop Robot. Biol. Syst.*, 1989.

[2] O. F. Rana and K. Stout, "What is scalability in multiagent systems?" in *Proc. 4th Int. Conf. Autonomous Agents*, 2000, pp. 56–63.

[3] A. Jevtić, P. Gazi, D. Andina, and M. Jamshidi, "Building a swarm of robotic bees," in *Proc. World Autom. Congr.*, Sep. 2010, pp. 1–6.

[4] D. T. Pham and M. Castellani, "The bees algorithm: Modeling foraging behavior to solve continuous optimization problems," *Proc. ImechE, Part C: J. Mech. Eng. Sci.*, vol. 223, no. 12, pp. 2919–2938, Dec. 2009.

[5] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, 2009.

[6] P. Bailis, R. Nagpal, and J. Werfel, "Positional communication and private information in honeybee foraging models," in *Proc. Swarm Intell.*, LNCS 6234. 2010, pp. 263–274.

[7] G. Dudek, M. Jenkin, and E. Milios, "A taxonomy of multirobot systems," in *Robot Teams: From Diversity to Polymorphism*, T. Balch and L. Parker, Eds. Natick, MA: A. K. Peters, 2002, pp. 3–22.

[8] B. P. Gerkey and M. J. Matarić, "A formal analysis and taxonomy of task allocation in multirobot systems," *Int. J. Robot. Res.*, vol. 23, no. 9, pp. 939–954, Sep. 2004.

[9] M. J. Matarić, G. S. Sukhatme, and E. H. Østergaard, "Multirobot task allocation in uncertain environments," *Autonomous Robots*, vol. 14, nos. 2–3, pp. 255–263, Mar. 2003.

[10] N. Michael, M. M. Zavlanos, V. Kumar, and G. J. Pappas, "Distributed multirobot task assignment and formation control," in *Proc. IEEE ICRA*, May 2008, pp. 128–133.

[11] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, "Market-based multirobot coordination: A survey and analysis," *Proc. IEEE*, vol. 94, no. 7, pp. 1257–1270, Jul. 2006.

[12] A. Franchi, L. Freda, G. Oriolo, and M. Vendittelli, "The sensor-based random graph method for cooperative robot exploration," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 2, pp. 163–175, Apr. 2009.

[13] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multirobot exploration," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 376–386, Jun. 2005.

[14] A. K. Ray, P. Benavidez, L. Behera, and M. Jamshidi, "Decentralized motion coordination for a formation of rovers," *IEEE Syst. J.*, vol. 3, no. 3, pp. 369–381, Sep. 2009.

[15] M. A. Joordens and M. Jamshidi, "Consensus control for a system of underwater swarm robots," *IEEE Syst. J.*, vol. 4, no. 1, pp. 65–73, Mar. 2010.

[16] S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar, "Optimized stochastic policies for task allocation in swarms of robots," *IEEE Trans. Robot.*, vol. 25, no. 4, pp. 927–937, Aug. 2009.

[17] M. Schwager, J. McLurkin, J.-J. Slotine, and D. Rus, "From theory to practice: Distributed coverage control experiments with groups of robots," in *Experimental Robotics* (Springer Tracts in Advanced Robotics, vol. 54), O. Khatib, V. Kumar, and G. Pappas, Eds. Berlin/Heidelberg, Germany: Springer, 2009, pp. 127–136.

[18] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford Univ. Press, 1999.

[19] S. Camazine, N. R. Franks, J. Sneyd, E. Bonabeau, J.-L. Deneubourg, and G. Theraulaz, *Self-Organization in Biological Systems*. Princeton, NJ: Princeton Univ. Press, 2001.

[20] R. Groß, S. Nouyan, M. Bonani, F. Mondada, and M. Dorigo, "Division of labor in self-organized groups," in *Proc. 10th Int. Conf. Simul. Adaptive Behav. From Animals to Animats*, 2008, pp. 426–436.

[21] S. Berman, A. Halasz, V. Kumar, and S. Pratt, "Bio-inspired group behaviors for the deployment of a swarm of robots to multiple destinations," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 2318–2323.

[22] T. H. Labella, M. Dorigo, and J.-L. Deneubourg, "Division of labor in a group of robots inspired by ants' foraging behavior," *ACM Trans. Autonomous Adaptive Syst.*, vol. 1, no. 1, pp. 4–25, Sep. 2006.

[23] A. Campo and M. Dorigo, "Efficient multiforaging in swarm robotics," in *Proc. 9th ECAL*, 2007, pp. 696–705.

[24] K. Lerman, C. Jones, A. Galstyan, and M. J. Matarić, "Analysis of dynamic task allocation in multirobot systems," *Int. J. Robot. Res.*, vol. 25, no. 3, pp. 225–241, Mar. 2006.

[25] V. Crespi, A. Galstyan, and K. Lerman, "Top-down versus bottom-up methodologies in multiagent system design," *Autonomous Robots*, vol. 24, no. 3, pp. 303–313, Apr. 2008.

[26] A. Farinelli, L. Iocchi, D. Nardi, and V. A. Ziparo, "Assignment of dynamically perceived tasks by token passing in multirobot systems," *Proc. IEEE*, vol. 94, no. 7, pp. 1271–1288, Jul. 2006.

[27] A. Gutiérrez, A. Campo, F. Monasterio-Huelin, L. Magdalena, and M. Dorigo, "Collective decision-making based on social odometry," *Neural Comput. Applicat.*, vol. 19, no. 6, pp. 807–823, Sep. 2010.

Aleksandar Jevtić (S'10–M'11) received the M.Sc. degree in electrical engineering from the University of Belgrade, Belgrade, Serbia, in 2005, and the Ph.D. degree in computer science from the Technical University of Madrid, Madrid, Spain, in 2011.

He is currently with the Department of Signals, Systems and Radiocommunications, Technical University of Madrid. On many occasions he has been invited to universities in the U.S., U.K., and Serbia, where he worked as a Visiting Researcher or Lecturer. His current research interests include swarm robotics and swarm intelligence applications to distributed multiagent systems, image processing, and cluster analysis.

Dr. Jevtić was the Chair or Co-Chair of several special sessions at the international conferences, including IEEE conferences. His work on distributed coordination of multirobot systems was awarded the 2010 World Automation Congress First Place Best Paper Award and the 2010 IEEE Systems Conference First Place Best Paper Award.

Álvaro Gutiérrez (S'99–M'08) received the M.Sc. degree in electrical engineering in 2003 and the Ph.D. degree in computer science in 2009, both from the Universidad Politécnica de Madrid (UPM), Madrid, Spain.

He was a Visiting Researcher with the Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle, Université Libre de Bruxelles, Brussels, Belgium, in 2008. He is currently an Assistant Professor of automatic and control systems with ETSI Telecomunicación, UPM. He has authored and co-authored three book chapters and more than 20 refereed journal papers and conference proceeding papers. His current research interests include swarm robotics, control systems, sensors networks, and demand-side management applications especially focused on smart-grids and renewable energies.

Diego Andina (SM'10) was born in Madrid, Spain, where he received simultaneously the two Masters degrees in computer science and in electronics and communications in 1990 and the Ph.D. degree in 1995 from the Technical University of Madrid (UPM), Madrid, Spain.

He currently heads the Group for Automation in Signals and Communications, Department of Signals, Systems and Radiocommunications, UPM. He is the author or co-author of about 250 national and international publications, having being the Director of more than 50 research and development projects.

Dr. Andina is an Associate Editorial Member of several international journals and transactions, and has participated in the organization of more than 60 international technology events.

Mo Jamshidi (S'66–M'71–SM'74–F'89) received the B.S. degree in electrical engineering from Oregon State University, Corvallis, in 1967, the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois at Urbana-Champaign, Urbana, in June 1969 and February 1971, respectively, and the Honorary Doctorate degrees from the University of Waterloo, Waterloo, ON, Canada, in 2004 and the Technical University of Crete, Crete, Greece, in 2004.

Currently, he is the Lutcher Brown Endowed Chaired Professor and the Leader of the Sustainable Energy Research Group, Department of Electrical and Computer Engineering, University of Texas, San Antonio. He was the Founding Director of the Center for Autonomous Control Engineering (ACE, http://ace.utsa.edu), University of New Mexico, Albuquerque, in 1995, and moved the center to the University of Texas, San Antonio, in 2006. He is an Honorary Professor with three Chinese universities and Deakin University, Melbourne, Australia. He is the Lead on the Sustainable Energy Research Group, University of Texas, San Antonio. He was a Senior Research Advisor with the U.S. Air Force Research Laboratory, Kirtland Air Force Base, NM, from 2002 to 2005 and from 1984 to 1990. He was a Consultant with the U.S. Department of Energy Office of Industrial Technologies and the Department of Energy Laboratories, Oak Ridge National Laboratory, Oak Ridge, TN, Sandia National Laboratories, Albuquerque, NM, and Los Alamos National Laboratory, Los Alamos, NM. He was an advisor for the NASA headquarters from 1998 to 2004 and on the NASA JPL's Pathfinder Project Mission and Surface Systems Track Review Board. He has worked in various academic and industrial positions at various national and international locations, including with IBM and GM Corporations. In 1999, he was a NATO Distinguished Professor in Portugal conducting lectures on intelligent systems and control. He has over 600 technical publications, including 62 books (12 textbooks), research volumes, and edited volumes. His most recent edited books are on system of systems engineering. Six of his books have been translated into at least one foreign language.

Dr. Jamshidi is the Founding Editor or Co-Founding Editor or Editor-in-Chief of five journals, including the IEEE CONTROL SYSTEMS MAGAZINE and the IEEE SYSTEMS JOURNAL. He is a fellow or a member of eight societies and academies. He was the recipient of the IEEE Centennial Medal, the IEEE Control Systems Society Distinguished Member Award, the IEEE CSS Millennium Award, and the NASA National Service Award in 2005. He was on the Board of Governors of the IEEE Society on Systems, Man and Cybernetics and is currently on the Board of the IEEE Systems Council. In October 2005, he received the IEEE Norbert Weiner Research Achievement Award. He was a U.K. Royal Academy of Engineering Distinguished Fellow with Cardiff University, Wales, U.K., from 2009 to 2010. He is a fellow of ASME, AAAS, TWAS, NYAS, A. Fellow of AIAA, and a Foreign Fellow of HAE.