Task Allocation in Robotic Swarms: Explicit Communication Based Approaches

Aryo Jamshidpey[1(✉)] and Mohsen Afsharchi[2(✉)]

[1] Institute for Advanced Studies in Basic Sciences, Zanjan, Iran
aryo.jamshidpey@iasbs.ac.ir
[2] Department of Computer Science, University of Zanjan, Zanjan, Iran
afsharchim@znu.ac.ir

Abstract. In this paper we study multi robot cooperative task allocation issue in a situation where a swarm of robots is deployed in a confined unknown environment where the number of colored spots which represent tasks and the ratios of them are unknown. The robots should discover the spots cooperatively and spread proportional to the spots area. We proposed 4 self-organized distributed methods for coping with this scenario. In two different experiments the performance of the methods is analyzed.

1 Introduction

Swarm robotics is inspired by social insects and other nature colonies that show complex behavior although they have simple members. The action of assigning tasks to agents for performing is called task allocation. From a control architectural perspective Burger [3] distinguishes between Heteronomous, Autonomous and Hybrid methods in task allocation. In this paper, we introduce a practical scenario for the issue of task allocation in swarm robotics and 4 hybrid methods for solving it in unknown environments which, the number, locations and ratios of tasks are unknown to robots.

Market-based mechanism is one of the main approaches that tackle the task allocation problem. TraderBots is one of the works in this subject that is presented by Dias [6]. A comprehensive study of market-based multi-robot coordination can be found in [7]. Most solutions in self-organized task allocation is threshold-based that are inspired by models initially proposed to describe the behavior of insect societies [1]. In this case we can mention Krieger and Billeter work [9] which benefits from a simple threshold-based model for task allocation in a foraging scenario. Labella et al. [10] and Lui et al. [11] proposed two probabilistic task allocation approaches which use adaptive thresholds. Brutschy et al. in[2] presented a task allocation strategy in which robots specialize to perform tasks in the environment in a self-organized manner. Jones and Mataric [8] introduced an adaptive distributed autonomous task allocation method for identical robots. Dahl et al. [4] proposed a method that controls the group dynamics of task allocation. Dasgupta [5] presented a communication-based method for task

D. Barbosa and E. Milios (Eds.): Canadian AI 2015, LNAI 9091, pp. 59–67, 2015.
DOI: 10.1007/978-3-319-18356-5_6

allocation. Robots can only partially complete tasks and one after the other contribute to progressing them. Our proposed practical scenario like Dasgupta's one is about unknown environments.

2 Problem Definition

This scenario involves a colony of identical robots with limited energy levels that are rechargeable and an environment full of obstacles and colored spots which represent types of tasks. The individuals are unaware of the size of the population and the distribution of the other robots. At any moment, each robot is able to do only one of the *forage for green* or *forage for black* subtasks. Depending on the area of the spots, the number of robots to do cleaning and sampling actions in them varies. Obviously it is not necessary to fill the whole capacity of spots with robots. Even a robot is able to do cleaning and sampling actions on its own but it takes much time and is not desirable. In abstract the scenario can be defined in terms of finding more colored spots with minimal energy waste (One of the causes of this waste is unnecessary robot turns in the environment), maximum spreading of robots in different spots proportional to their area, avoiding robots from collision with obstacles and finally preventing robots from remaining idle.

3 Methods

In all proposed methods the robots are initially in random places. In each method the energy of each robot is divided into three parts. First part is for foraging. Second part is to do cleaning and sampling actions and the third part is dedicated to returning to the charging station. All methods are organized based on transferring messages and all messages have the same structure. Each message may have several rows that each of them presents a distinct spot's information. Information of each spot consists of 10 features. Message updating procedure is the same in all of proposed methods. Every robot has a message that contains information about the spots which is called private message. This message is empty at first and then updated via occurrence of two different events. First when a robot finds a non-observed spot for the first time, and second when it receives a message from one of its neighbors.

3.1 Static Communication Based Method

At the beginning, one of the two subtasks of exploring for green or black spot is assigned to each robot which is called worker statically (With a probability of 0.5). Each robot moves in the environment by random walk and avoids collision with other robots, obstacles and walls by the help of its distance computing sensors. Each robot updates its private message periodically and broadcasts it to every other robot in the coverage area of its radio frequency transmitter. By this process, messages will propagate among the robots. If a robot could not

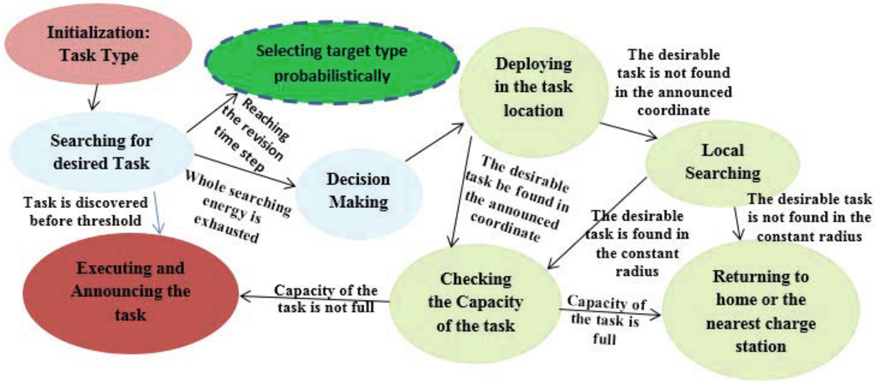

Fig. 1. Method1 and Method 2 state diagram

find a match before running out of its searching energy, it would go to one of the spots indicated in its private message. By having this policy, our goals for preventing idle robots existence, maximum spot coverage and also preventing spot starvation will be guaranteed in a desirable level. The decision step is as follows: The robot should sort its private message based on three fields; color, hop count and its current distance from the spots. At first this sort is done based on color field, so that spots with the same color as the robot current state will be placed at the top. Then these lists are sorted based on the hop count field. Each sorted list then will be categorized according to sets of 5 hops. Robots prefer to go to spots with low hop count number to prevent starvation. In the third step the sorted list will be further sorted based on Euclidean distance. So in each hop category, spots are sorted based on their distance from the robot. The state diagram of the robot's controller while using this method is shown in Figure 1.

3.2 Dynamic Communication Based Method

This method (figure 1) is similar to the first method in a way that it supports dynamic task allocation in such a way that the robot will change its target color probabilistically after a time step. For example consider that a robot has 4 spots in its private list after 100 iterations containing 1 black and 3 green. The robot will set its target color to black by probability 1/4 and also will set it to green by probability 3/4.

3.3 Decentralized Chapar Method

In this method (figure 2 (a)) we have some radio turrets called Chapar stations which are used for radio communications and also we assume two groups of robots; workers and Chapars. Chapars transfer messages between workers and Chapar stations. High speed robots with simple structure are considered

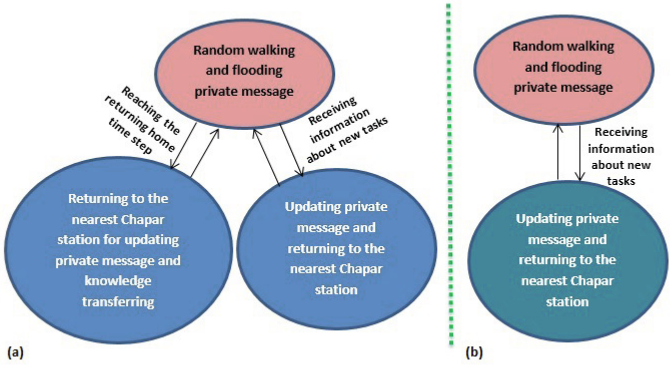

Fig. 2. (a): Method 3 State Diagram, (b): Method 4 State dDiagram

as Chapars. Chapars broadcast their updated private messages which are modified by some workers to the area of their radio frequency coverage. Once a Chapar realizes a new row in its private message it quickly goes to the nearest Chapar station and sends its private message to it and also updates the message based on the content of the messages sent by the Chapar stations. In addition each Chapar goes to the nearest Chapar station periodically to update its private message. Chapar stations are also in the coverage of each other and so they replicate messages to keep the whole system up to date.

3.4 Centralized Chapar Method

This method (Figure 2 (b)) is similar to the decentralized one in which there is only one Chapar station that covers the entire environment. In this method, Chapars have a single task which is to transfer messages from robots to the Chapar station and they do it once they realize a new row in their private messages.In spite of the centralized method, each Chapars only goes to the Chapar station when it encounters a new spot in its private message.

4 Simulations

We have used e-puck robots in simulations. Since the purpose of this article is to involve a wide range of robots and the use of simpler hardware, we considered them without any camera. All experiments have been implemented in a 3m x 3m square environment enclosed with walls by using of Webots as robotic simulation software. In our general experiments, performance of the four proposed methods is evaluated individually before and after energy consumption and in both of them 10 robots which are called workers with IDs from 1 to 10 are used. Initially foraging mode of the robots with IDs from 1 to 5 is adjusted to green and the robots with IDs from 6 to 10 is adjusted to black. For simplicity, the details of cleaning and sampling operations are ignored, the colored spots are considered as 30cm x 30cm squares and finally the length of each robot's communication radius is considered larger than the diameter of each colored square. In both experiments, for each 300 square centimeters of each spot one worker is sufficient for covering it desirably.

4.1 First Experiment: Performance Evaluation of Proposed Methods Before the Threshold Energy

Since the third and fourth methods use the worker robot which its controller is that of the first or second method, it is not necessary to compare the performance of all methods before foraging energy consumption (before the threshold). So we have compared only the performance of the first and second methods before the threshold. For this purpose four environments covered by green and black spots with obstacles are considered. The first environment has 3 green spots and 3 black ones, the second has 4 green spots and 2 black ones, the third includes 5 green spots and 1 black ones and finally the fourth contains only 6 green spots. In each of the four areas, both the first and second methods are tested 10 times separately. The average number of successful robots before the energy threshold for the fist and second methods is shown in Figure 3 (b).

It can be seen that, except for the first environment in which the average number of successful robots before the threshold are equal for both methods, in other environments, the average number of successful robots in the second method is higher than the first one. This disparity grows by moving from the first environment to the fourth and its reason is the changing attitude mechanism which is used by second method's robots during their foraging operation. As a result this leads to increasing in the number of robots which their attitude changes when the number of green spots rise and the number of black ones falls respectively. Subsequently this process will result in forming approximate stability in the number of successful robots before the threshold. As it is shown in Figure 3 (b), we can conclude that in unknown environments where the number of colored spots and the ratios of them are unknown, the second method is more successful than the first one in terms of the robots' attempts in finding spots by themselves before energy threshold.

In the next step the average number of green spots that have been found by workers before the threshold is shown in Figure 3 (a). In this figure the obtained curve from the second method is steeper than the first method's one. The reason that the second method's curve is more steeper, is increasing of the number of workers which search for green spots during the search time. As mentioned before this is because of changing the robot's attitudes during foraging operations. Figure 3 (a) shows that in unknown environments where the number of colored spots and their ratios are unknown, the second method is more successful than the first one in spot finding before the energy threshold.

Figure 4 shows the average number of robots deployed in the green spots before the end of the foraging energy in both the first and second methods. It can be observed that the steepness of the second method's curve is ascending linear but the steepness of the first method's curve is sub linear. In the first method

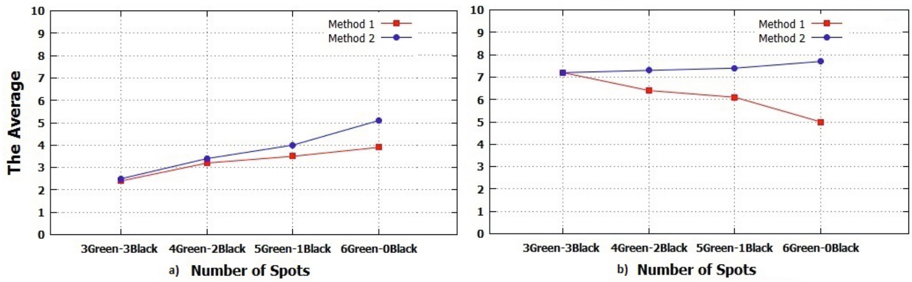

Fig. 3. (a): Average number of discovered spots. (b): Average number of successful robots before the threshold.

the maximum number of robots in green spots is 5 because only 5 robots have green initial foraging modes and there is no any changing attitude mechanism before threshold. But in the second method there is changing attitude and so the maximum number of robots in green spots might be 10. We can conclude that in unknown environments where the number of colored spots and the ratios of them are unknown, adaptability has a significant positive impact on the performance of robots. To sum up, from the above results in unknown environments with the features which are mentioned above, the second method is more efficient than the first one before the energy threshold.

4.2 Second Experiment: Performance Evaluation of Proposed Methods After the Threshold Energy

We use an environment consisting of 3 green spots and 3 black ones for evaluating the performance of the methods after foraging energy consumption. In this area, all proposed methods are separately tested 10 times with random initial distribution of robots. In the third and fourth methods, in addition to 10 workers, another 3 Chapar robots that are faster than the workers are used too. It should be mentioned about Chapar station that the third method is equipped with 3 Chapar stations which their communication radius cover each other sequentially and the fourth method is equipped with one of them, which is omniscient. Table 1 shows the results of the second experiment. The absorption percentage is the percentage of successfulness of finding spots by robots after the threshold by applying the decision making mechanism.

As expected, the fourth method has the highest absorption percentage which means 100%. This is due to the use of global message transferring system. The third method is in second place with 87.09% and after it the first and the second methods with approximate absorption 76% are both in third place.

5 Discussion

Most studies in swarm robotic define their own test scenario, which is then used to build a concrete swarm robotic system capable of solving the problem. This leads

Table 1. Results for the second experiment

Method Number	Meth. 1	Meth. 2	Met. 3	Meth. 4
The Average Number of Successful Robots Before the Threshold.	7.2	7.3	6.9	7.3
The Average Number of Unsuccessful Robots Before the Threshold.	2.8	2.7	3.1	2.7
Absorption Percentage.	75	77.78	87.09	100
Absorption Percentage of Robots Appropriate to Their Initial Foraging State.	46.42	37.03	70.96	83
The Average Number of Absorbed Robots.	2.1	2.1	2.7	2.7
The Average Number of Absorbed Robots Appropriate to Their Initial Foraging State.	1.3	1	2.2	2.2

to a huge amount of differently designed global missions and as a result to many different solutions which are hard to compare[3]. Thus in most of the proposed methods in this area, researchers have only introduced their own methods and refrained from comparing with other methods. Our scenario also possesses different features, goals and finally distinct global foraging mission compared to previous scenarios in task allocation field. Accordingly applying current approaches in our scenario and consequently comparing them with each other is so difficult and in most of the times is impossible, except for changing the scenario which in turn leads to changing both the problem and the solutions. As pointed out before, Dasgupta‘s method is practical for unknown environments with this difference that its global mission is different from what have been proposed here. However our proposed methods have distinct advantageous over Dasgupta's.

The proposed methods have the least waste time for robots as apposed to that of Dasgupta in which, robots might be idle in environment for a long time. These methods have been designed for the energy constrained scenario compared to Dasgupta's. Consequently the priority of discovering new spots is higher than accomplishing a task and then continuing unlimited foraging. Further more in contrast with Dasgupta's method(which robots may be in idle mode for a long time) our methods are close to optimal in the sense of idle mode. On the other hand, our approaches are very practical for scenarios which detecting a task should be done in a limited time. However in Dasgupta's approach, since after detecting a task other neighbor robots quit foraging and wait for accomplishing the task sequentially, the probability of identifying other tasks in a limited time will decrease. Moreover the proposed methods in this paper are practical for blind robots while Dasgupta's method is practical for robots with cameras which ignoring use of them obviously reduces cost in swarm robotics. Also there is no more need to apply sophisticated techniques of machine vision in order to determine starvation (which may have some errors).

5.1 Scalability and Robustness in Term of Single Point of Failure

If by adding or removing robots the performance of a method drops off, we will call it unscalable. Accordingly, both the first and second methods are scalable

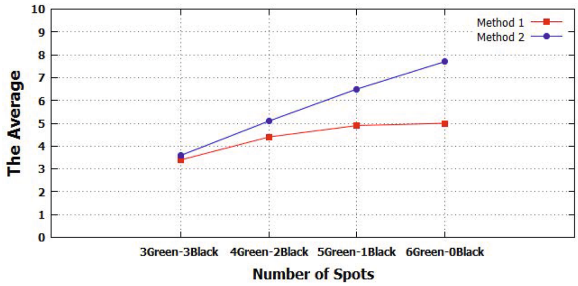

Fig. 4. Average number of robots deployed in green spots

because both are distributed, autonomous and based on the local communication.So Adding or removing robots will maintain their effectiveness. The decentralized Chapar method also has scalability property. This is due to using of the limited range Chapar station, the worker robots and the Chapar robots that are distributed and behave locally. But the centralized Chapar method, due to using of omniscient Chapar station with respect to its expected efficiency does not have scalability property.

With respect to robustness, the first and second methods are completely robust against single point of failure because they behave in a distributed and autonomous manner. If a robot fails outside a spot, others will consider it as an obstacle. If it fails inside a spot and before announcing the center of it, there is no problem because it looks like the situation in which the spot is not discovered yet. But if it fails after the center announcement, it causes only decreasing in speed of the operations in the spot, because they can be carried out by other active robots on the spot yet.

The third method is less robust. The strength of the method for worker robots is such as the first and second methods. In the case of Chapar robots failures, there will be not any problem because in the worst case the speed of communication will be reduced and other robots largely compensate this loss. But about Chapar station as we have mentioned previously in describing the method to benefit the high-speed informing relative to the cost, it is necessary that their communication radius cover each other sequentially. Thus, if one of Chapar stations fails, the connection between two parts of the environment will be lost. It can be said that this method is robust against single point of failure. But this robustness is less than the first two methods. Centralized Chapar method has the lowest robustness than the previous three methods because this method is only have a wide range Chapar station.

6 Conclusions

In this paper a new practical scenario in swarm robotics is presented which is about task allocation in unknown environments. Here we consider that there is limited source of energy for each Robot. It is pointed out that energy management in the form of a 3 level structure is essential and four self-organized threshold-based methods are proposed for solving the scenario. Moreover, in two general experiments, the performance of them is analyzed.

References

1. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: From natural to artificial systems. Oxford University Press (1999)
2. Brutschy, A., Tran, N.L., Baiboun, N., Frison, M., Pini, G., Roli, A., Birattari, M.: Costs and benefits of behavioral specialization. Robotics and Autonomous Systems **60**(11), 1408–1420 (2012)
3. Burger, M.: Mechanisms for Task Allocation in Swarm Robotics (Doctoral dissertation, Diploma thesis, Ludwig-Maximilians-Universitt Mnchen) (2012)
4. Dahl, T.S., Matari, M., Sukhatme, G.S.: Multi-robot task allocation through vacancy chain scheduling. Robotics and Autonomous Systems **57**(6), 674–687 (2009)
5. Dasgupta, P.: Multi-Robot Task Allocation for Performing Cooperative Foraging Tasks in an Initially Unknown Environment. In: Jain, L.C., Aidman, E.V., Abeynayake, C. (eds.) Innovations in Defence Support Systems -2. SCI, vol. 338, pp. 5–20. Springer, Heidelberg (2011)
6. Dias, M.B.: Traderbots: A new paradigm for robust and efficient multirobot coordination in dynamic environments (Doctoral dissertation, Carnegie Mellon University) (2004)
7. Dias, M.B., Zlot, R., Kalra, N., Stentz, A.: Market-based multirobot coordination: A survey and analysis. Proceedings of the IEEE **94**(7), 1257–1270 (2006)
8. Jones, C., Mataric, M.J.: Adaptive division of labor in large-scale minimalist multi-robot systems. In: Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2003, vol. 2, pp. 1969–1974. IEEE. Integration. Technical report, Global Grid Forum (2002)
9. Krieger, M.J., Billeter, J.B.: The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. Robotics and Autonomous Systems **30**(1), 65–84 (2000)
10. Labella, T.H.: Division of labour in groups of robots (Doctoral dissertation, Ph. D. thesis, Universite Libre de Bruxelles) (2007)
11. Liu, W., Winfield, A.F.T., Sa, J., Chen, J., Dou, L.: Strategies for Energy Optimisation in a Swarm of Foraging Robots. In: Şahin, E., Spears, W.M., Winfield, A.F.T. (eds.) SAB 2006 Ws 2007. LNCS, vol. 4433, pp. 14–26. Springer, Heidelberg (2007)