Cooperative Optimization of a Foraging Strategy for Swarm Robots

Y3864254
Department of Electronic Engineering,
University of York,
York, United Kingdom
Y3864254@york.ac.uk

***Abstract*— This article presents a cooperative foraging strategy based on local communication of swarm robotics. Using the proposed method, the robots perform task allocation through limited local communication, based on the value of food and the distance from nest. The results of Simulation demonstrate that this method transcends an existing foraging robot system.**

***Keywords*— swarm robots, foraging, task allocation**

I. Introduction

Swarms robotics is a study of the design process of large numbers of relatively simple physically embodied agents [1]Foraging robots are mobile robots capable of searching for and homing with objects to one or more designated collection points [2].Research on the foraging behaviour is a typical field of swarm robotics. In this task, the robot explores in an initially unknown environment, grabs food and sends them back to the 'nest' area.

The whole foraging behaviour can be divided into four processes, which are searching, grabbing, depositing, and homing. This paper mainly focus on the searching process, and optimize a random search foraging algorithm. The robots exchange information through local communication, ultimately perform task allocation and food collection.

This paper is organized as follows. Section 2 gives a brief review of the previous literature. Section 3 introduces the original algorithm and the optimized algorithm. These two strategies are simulated in Section 4. The simulating results are demonstrated and analysed in Section 5. And the paper is concluded in Section 6.

II. Literature Review

In 1989, Gerardo Beni first proposed the concept of group robots [3]. Many simple robots work together to form a robot system, resulting in collective effects [4]. Compared with highly intelligent robots, group robots have the characteristics of high scalability and strong robustness [4,5].

Foraging behaviour has now become a typical research area of swarm robotics [6]. The robot searches for the unknown environment and returns to the predetermined position after grabbing the food. This process is called foraging [7,8]. Foraging scenarios have been widely extended to various fields[9,10,11].

Task allocation can be used to divide different roles in the robot in an efficient manner to accomplish different tasks. High-quality task assignments enable global distribution and global tasks[12,13].In 2013, O.V Sanjay Sarma proposed a strategy for task-allocation foraging using simple communication messages[14].This solution allows the robots to communicate with each other through a short exchange of information, thus achieving cooperative foraging.

Inspired by this article, this paper designs an algorithm for task assignment through multi-layer information transfer and compares it with the original algorithm.

III. Implementation

The implementation is simulated in the ARGoS robot simulator. The walled arena is set as a 5m × 5m square venue with three random food placement sites, which are indicated by round black patches on the floor. These three sites do not overlap each other and are assigned food with different value when the sites are initialized. The nest is situated on one side of the arena and is represented by grey floor, and 10 robots were randomly placed in it. Four yellow lights are placed over the nest to help the robots return by performing phototaxis. The robot was not able to know the location and value of the food at the beginning.

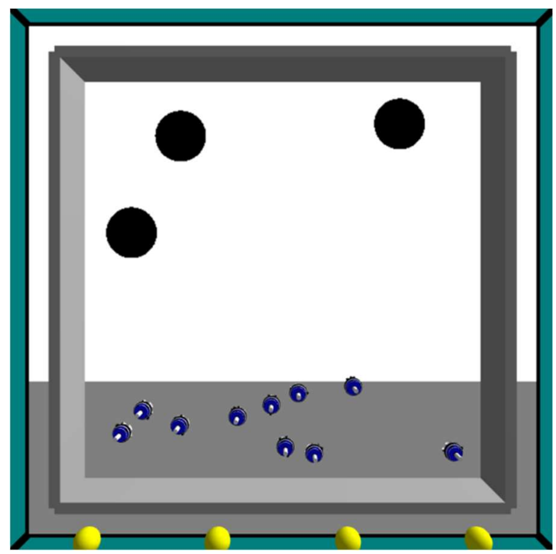

Fig. 1. ARGoS Arena.

A. Original Foraging Algorithm

The original foraging algorithm divides the state of the robots into two types, namely foraging and returning. This strategy allows the robots to perform obstacle avoidance during the foraging process and perform obstacle avoidance and phototaxis during the returning process. The foraging state becomes a returning state when the robot picks up the food. The Returning state becomes the foraging state When the robot puts down the food.

Pseudo code for individual robot is as follows.

TABLE I. ORIGINAL FORAGING ALGORITHM

```
robot.state = "foraging"
while(1)
 if robot.state == "foraging"
   if robot_has_food
     robot.state = "returning"
   else if robot_in_nest
     avoid obstacles
     perform anti-phototaxis
   else
     avoid obstacles
   end
 end
 if robot.state == "returning"
  if not robot_has_food
    robot.state = "foraging"
  else
    avoid obstacles
    perform phototaxis
  end
 end
end
```

Fig. 2. Pseudo code of original algorithm.

B. Optimized foraging algorithm

In this algorithm, the robot is allowed to perform local communication within one meter. The robot is divided into four states, namely exploring, foraging, returning, and marking. Different from the above algorithm, this algorithm has an initial exploration phase. In this phase, each robot's state is exploring, independently exploring the environment without communicating. The first robot that discovered the new food site changes its state to marking and stopped at that site as a beacon.

At the end of the exploration phase, each robot's state changes to one of the other three states and communicates with the neighbours.

Beacons marking with different food sites exchange information through robots in between to determine the beacon with better food resources. Other robots are classed according to the number of communication intermediaries (number of robots) they need from the beacon. The resulting information exchange network is a tree topology. In addition to the beacon robots, other robots move toward the minimum class neighbour robot during the foraging phase.

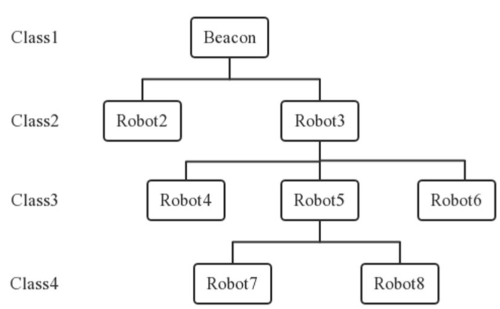

Fig. 3. Network is a tree topology.

C. Threshold preset

In the foraging process, the robot judges its own class by acquiring the class information of the neighboring robot, that is, the minimum_class + 1. During the foraging process, the network is at risk of being self-circulating due to the detachment of the beacon.

Based on this, the threshold is set to determine whether or not the beacon is included in the robot network. When the robot finds that the neighbouring robot's class is greater than this value, it determines that its own network has left the beacon, and chooses to re-randomly explore until it joins the network with the beacon. The threshold should be set according to the environment, the number of robots, and the distance of data transmission. In this simulation the threshold is set to 4.

D. Judgment of food resources

The value of a food site*(V_f)* should be defined by its unit food value*(V_u)* in distance to nest*(D_n)*:

$$V_f = \frac{V_u}{D_n} \tag{1}$$

However, for a beacon robot, it is difficult to obtain the distance from the nest, especially when the food site is far from the nest. Therefore, the measured value of the brightness of the nest*(V_l)* light is used to indirectly express the distance from the nest, which decreases as the distance increases. The food point values in this paper are defined as follows:

$$V_f = V_u V_l \tag{2}$$

In this strategy, the robot uses the range_and_bearing system in ARGoS to communicate within one meter. The system allows a robot, upon receiving information from another robot, also detects the position of the sender in its local frame. It is worth noting that the robot that sends data can only broadcast it in a preset range, which means that it is not possible to pass personalized information to the specified individual robot. And the system can exchange information only when there are no other obstacles between the two robots.

Pseudo code for foraging and marking states is as follows.

TABLE II. OPTIMIZED FORAGING ALGORITHM

```
if robot.state == "foraging"
  receive and send the information of classes and food sites
  if robot_has_food
    robot.state = "returning"
  end
  its_own_class =  the _minimum_neighbor_class + 1
  if the _minimum_neighbor_class > 4
    move random
  else
    follow the neighbor with the minimum class
  end
end

if robot.state == "marking"
  receive and send the information of classes and food sites
  if the value of site < known_maximum_value
    robot.state == "returning"
  end
end
```

Fig. 4. Pseudo code of Optimized algorithm.

IV. RESULTS

The algorithms are simulated in ARGoS.

The original strategy uses the colour of the light to mark the state of the foraging, where foraging is marked in red and returning is marked in green.

For an optimized algorithm, the colour of the light is used to mark the robot's class. The beacon is marked in red, the class2 robot is orange, the class3 robot is yellow, and the remaining robots are marked in green.

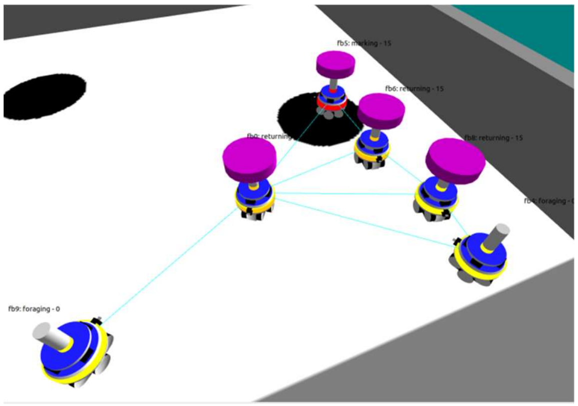

Fig. 5. The led lights of robotics.

The simulation was repeated twenty times in each of the two strategies, each time refreshing a new random map and run 5000 steps. During the simulation, the number of steps and the value of the food each time when the food is returned to the nest is recorded.

V. Analysis

A. Qualitative analysis

The total value of the food obtained in 5000 steps is shown in the table below.

TABLE III. TOTAL FOOD VALUE

Original Algorithm	550	495	610	425	595	530	360	590	500	455
	400	725	475	485	230	470	530	385	525	400
Optimized Algorithm	1240	970	1220	755	860	940	920	940	620	670
	840	885	690	495	790	1460	1080	1070	1075	1090

Fig. 6. Total food value.

The distribution of values is shown Fig.7.

As can be seen from the figure, the total amount of food obtained has a better distribution in the improved algorithm and has been significantly improved.

The original foraging algorithm is less affected by the map due to its random exploration characteristics. The improved algorithm is sensitive to the distance of food due to information exchange that relies on a limited distance. The further the distance between the food station and the nest, the greater the possibility that the robot network will be out of the beacon .

B. Quantitative analysis

Unit step value *(V_{us})* is used as the primary evaluate performance in this foraging behaviour.

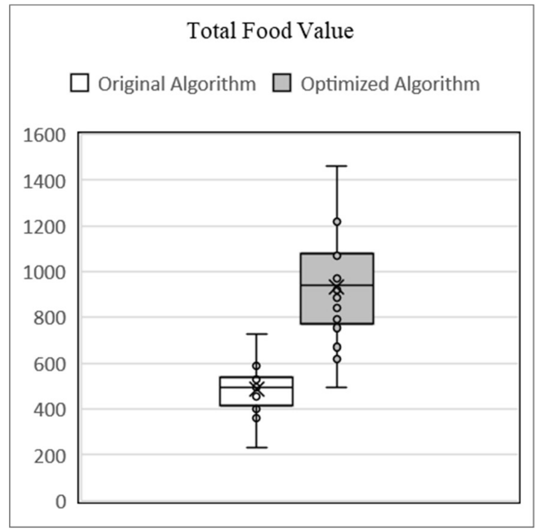

Fig. 7. Distribution of values.

$$V_{us} = \frac{Food\ Value}{Step} \tag{3}$$

We compare the median of average unit step value, which are 0.098 and 0.188. The performance of the improved system has been significantly improved.

In the middle steps of obtaining food twice, it can be seen that the value is equal. So we define the unit step value as:

$$V_{us} = \frac{single\ value}{Next\ get\ food\ step - last\ get\ food\ step} \tag{4}$$

In the 20 simulation experiments, the probability distribution of the unit step value is shown in Fig.8.

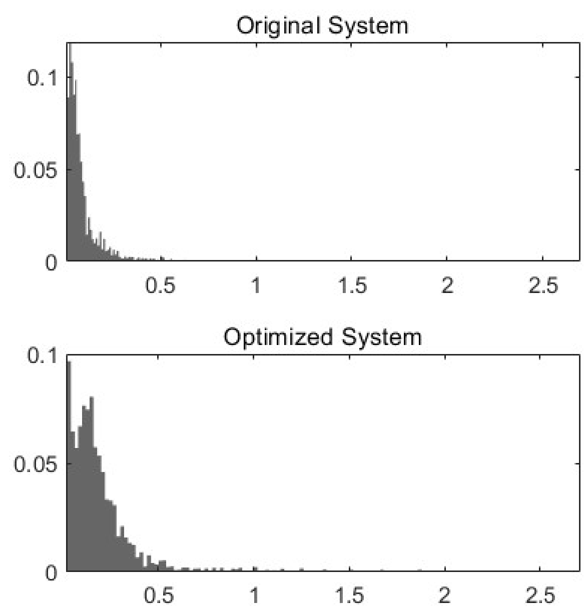

Fig. 8. Probability distribution.

It can be seen that in the optimized algorithm, the unit step has a higher probability of obtaining greater value.

The growth curve of food value reflects the characteristics of the system. We take the step as the abscissa and the average values of the 20 times as the ordinate to draw the Fig. 9.

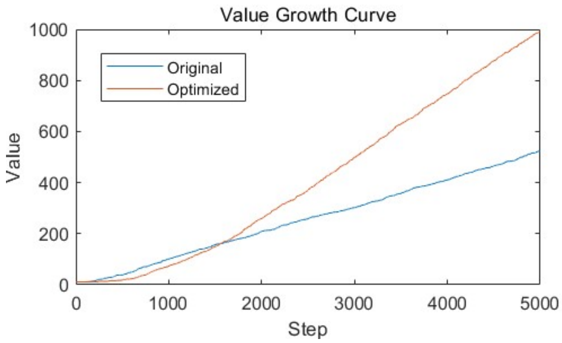

Fig. 9. Value growth curve.

As we can see in the figure, the improved system has a greater value growth rate in the later stage. It is worth noting that in the early stage of the foraging process, the improved system performed less well than the random system. The reason is mainly in the early stage of the exploring period (first 1500steps), the robot only performs a small amount of communication, and the beacon robot stops foraging. This reduces the total amount of robots at work. After the exploring period, the robot gradually established an information network, and the speed of foraging is significantly improved. In general, although task allocation can reduce work robots, the value they receive is significant.

C. *Reality gap*

There are many other factors that need to be considered when applying simulation to reality.

For the original foraging algorithm, the robot needs to beequipped with a distance sensor for obstacle avoidance, which can be implemented with an ultrasonic distance sensor. A light sensor for sensing the nest light is also necessary. In addition, a sensor for identifying the ground grayscale should be provided for judging whether it is in a nest or whether it is in a food site. For real robots, the behaviour is not automatic, so the device that grabs and lowers also needs to be equipped. The value of food can be preset to be expressed in different qualities, so the robot should have a quality measuring device for judging the value of the food.

For improved foraging behaviour, communication between robots is a must. Signal communication and distance measurement at various angles can be achieved by equipping the robot with IR sensors at multiple angles.

REFERENCES

[1] Sahin, Erol. (2005). Swarm Robotics: From Sources of Inspiration to Domains of Application. Lect. Notes Comput. Sci.. 3342. 10-20. 10.1007/978-3-540-30552-1_2.

[2] Winfield A.F. (2009) Foraging Robots. In: Meyers R. (eds) Encyclopedia of Complexity and Systems Science. Springer, New York, NY

[3] G. Beni and J. Wang. Swarm intelligence. In Proc. of the Seventh Annual Meeting of the Robotics Society of Japan, pages 425–428, Tokyo, Japan, 1989.

[4] Y. Uny Cao, Alex S. Fukunaga and Andrew B. Kahng, “Cooperative Mobile Robotics: Antecedents and Directions”, Autonomous Robots, vol. 4, 1997, pp. 1-23.

[5] Bernardine Dias M. and Anthony (Tony) Stenz, “A Market Approach to Multirobot Coordination”, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, August 2001.

[6] M. Kreiger, J. Billeter, "The call of duty: Self-organized task allocation in a population of up to twelve mobile robots", Robotics & Autonomous Systems, vol. 30, no. 1, pp. 65-84, January 2000.

[7] H. Hamann, H. Rn, "An analytical and spatial model of foraging in a swarm of robots", Swarm Robotics Second International Workshop SAB 2006, pp. 43-55, September 30-October 1.

[8] W. Liu, A. Winfield, S. Jin, J. Chen, L. Dou, "Strategies for energy optimisation in a swarm of foraging robots", International Conference on Swarm Robotics Springer-Verlag, pp. 14-26, 2006

[9] Z. Zakaria, T. K. A. Rahman and E. E. Hassan, "Economic load dispatch via an improved Bacterial Foraging Optimization," 2014 IEEE 8th International Power Engineering and Optimization Conference (PEOCO2014), Langkawi, 2014, pp. 380-385.

[10] Y. Liu, K. Hu, Y. Zhu and H. Chen, "Color image segmentation using multilevel thresholding-cooperative bacterial foraging algorithm," 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, 2015, pp. 181-185.

[11] Y. Lin, Y. Leu and W. Lu, "Fuzzy bacterial foraging system and its applications in control of servo motors," 2012 International conference on Fuzzy Theory and Its Applications (iFUZZY2012), Taichung, 2012, pp. 203-207.

[12] Baghaei K R , Agah A . Task Allocation and Communication Methodologies for Multi-Robot Systems[J]. Intelligent Automation & Soft Computing, 2003, 9(4):217-226.

[13] Brian P. Gerkey and Maja J Mataric, “A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems”, The International Journal of Robotics Research, vol. 23, 2004, pp. 939- 954.

[14] O. V. Sanjay Sarma, T. Vishwanath Lohit and S. Pugazhenthi, "Development and testing of a foraging strategy for low cost Swarm Robots," 2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE), Jabalpur, 2013, pp. 1-6.