Triangle Formation Based Multiple Targets Search Using a Swarm of Robots

Jie Li[1,2] and Ying Tan[1,2](✉)

[1] Key Laboratory of Machine Perception and Intelligence, Ministry of Education, Peking University, Beijing, China
ytan@pku.edu.cn

[2] Department of Machine Intelligence, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China

Abstract. As a distributed system, swarm robotics is well suited for multiple targets search tasks. In this paper, a new approach based on triangle formation and random search is proposed for high efficiency, demonstrating excellent abilities of exploration and exploitation in experiments. In addition, a new random walk strategy of linear ballistic motion, integrated with triangle estimation, is put forward as a comparison algorithm, the performance of which can serve as a benchmark.

Keywords: Swarm robotics · Multiple targets search · Triangle formation · Random search · Exploration and exploitation

1 Introduction

Swarm robotics, inspired from the self-organization phenomena in nature, is a relatively new field, on which people have done lots of various research work [1]. With large number of individuals, swarm robotic system is appropriate for tasks involving area coverage [2], such as searching for multiple targets. When the targets can generate fitness values in certain range and can be collected, it comes to the issue we concern [3]. The multi-target search strategy has a broad prospect of application, such as hunting a submarine [11], searching for victims and wreckage after air crash or shipwreck, monitoring the leak water quality [2], exploring and destroying battlefield targets, and so on.

Behavior-based design methods are commonly used in the task of searching for targets, such as methods based on artificial potential functions [4] or methods adapted from some heuristic algorithms [5,6]. GES [7] and IGES [8] we proposed before borrowed some ideas from the FWA [9], a heuristic algorithm inspired by the firework explosion. Mathematical physics methods are also used to analyze the foraging and migratory behaviors of animals, which is often referred to as "random search" [11,12] or "stochastic optimal foraging theory" [13].

Another thing needed to be introduced here is the formation control for multiple robots or vehicles, and what we used is the behavior-based control [14],

Y. Tan et al. (Eds.): ICSI 2016, Part II, LNCS 9713, pp. 544–552, 2016.
DOI: 10.1007/978-3-319-41009-8_59

where each robot determines its proper position based on a reference point which can be a leader, a neighbor or the unit-center of the whole group.

The rest of the paper is organized as follows. In Sect. 2 , the multiple targets search problem and an idealized model are stated. In Sect. 3, the Triangle Formation Search strategy is described. In Sect. 4, experimental results and discussions are presented. Finally, the work is concluded in Sect. 5.

2 Problem Statement

In the multiple targets search problem, a swarm of robots are delivered into a vast unknown space, where multiple targets are distributed randomly. Robots are expected to search and collect the targets as soon as possible using some collaborative mechanism. In the simplest case, only three kinds of objects are considered: environment space, robots and targets. Obstacles, decoys [10] and inference sources can also be introduced into the problem [3]. Since we focus on the search efficiency in this paper, only the simplest case are studied.

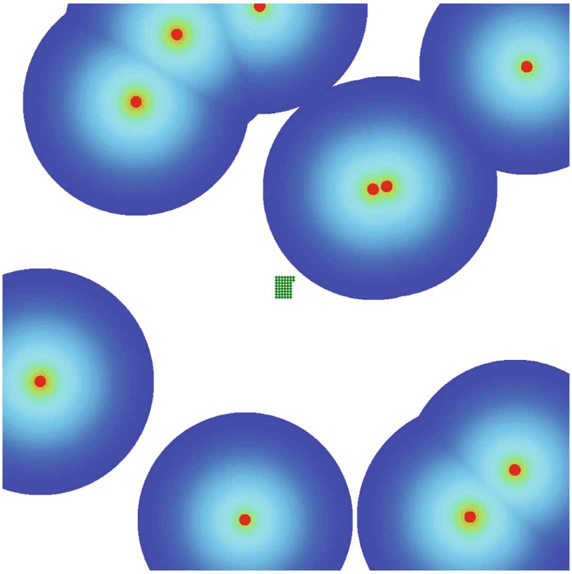

Fig. 1. A screenshot of the problem at the beginning of a simulation. Red rounds stand for the targets. The background color illustrates the fitness value of that position. The robot phalanx is in the center of the figure. (Color figure online)

2.1 An Idealized Model

An idealized model of the problem is shown in Fig. 1. The environment is a 1000*1000 square while the robot is a unit square. Robots can memorize information (positions and fitness values) of 10 iterations. Each target is abstracted to be a round with radius of r_t (10 units), and robots in the round can locate the target directly. Positions and fitness values of targets are generated randomly, and fitness ranges from $F_{max} - 2$ to F_{max} (40 units). Influence scopes of targets are presented as a series of annuli, and each annulus is $0.5r_t$ width which is also the maximum speed limit of robots to ensure the variance of fitness values in two adjacent iterations is small. From the inside out, the fitness value decreases by 1 unit till 0, and greater ones are chosen as the fitness values in overlapped areas. Discrete fitness values are adopted because the hardware design in swarm robotics should be as simple as possible which may lead to low quality sensors and fault sensing results [15]. 10 iterations are required for one robot to collect a target while 10 robots can do that in one iteration. The sphere of local interaction between robots is a round of radius $2r_t$. One problem that has not been considered here is avoiding collisions of robots resulting from route intersection.

3 Triangle Formation Search Strategy

3.1 Characteristics of the Problem

- Compared with each individual robot, the entire search space is vast, so the swarm is supposed to have nice ability of exploration. The influence scopes of all targets cover a large proportion of the entire search space, so the swarm should bear excellent ability of exploitation.
- In order to prevent excessive concentration of resources and give full play to the group exploration ability, the entire swarm should disperse as much as possible in the initial stage.
- The integration of local information is essential to improve the group exploitation ability, so each robot should ensure certain degree of connection with neighboring robots, i.e. form local groups (or niches) with other robots.

In our triangle formation search (TFS) strategy, the swarm is divided into three-robot teams which are arranged in a triangle, including one leader and two other members. According to the conclusions above, the TFS strategy may be a promising approach, for the three-robot teams can balance the exploration and exploitation of the swarm.

3.2 Five Stages of the TFS Strategy

- Initial grouping: Divide the whole swarm into three-robot teams, and robots insufficient to form a team will search alone.
- Initial diffusion: Firstly, the leaders will count the number of neighboring robots and select a sparse direction.
- Search in areas without fitness: The leader will search randomly, and the step lengths are submitted to some type of probability distribution, such as a Lévy or exponential distribution.
- Search in areas with fitness: The leader will estimate the gradient direction according to the information obtained by the team and update its position.

– Target collecting: Robots having found targets will broadcast the information within the team and the other two will move towards the target.

In stages of diffusion and search, members in teams will follow the leader and maintain the formation. Since the strategy involves formation control which increases the complexity of the system, for convenience, we restrict the exchange of information within the team, and the formation cannot be restored once broken.

3.3 Key Issues to be Tackled in the TFS Strategy

Unified Grouping. Taking into account the initial formation of the robot phalanx and the simplicity of implementation, we assign a global ID for each robot in an “S” shape order. Then, the robot whose ID is a multiple of 3 serves as a leader, closely followed by two other members of the team.

Diffusion Control. In order to make full used of the exploration ability of the swarm, an initial diffusion stage is introduced, in which he leaders will monitor the number of neighbors and terminate diffusion if the number has fallen below a certain threshold. We carried out experiments to determine the threshold from 3 to 10, and the optimal value is about 3.

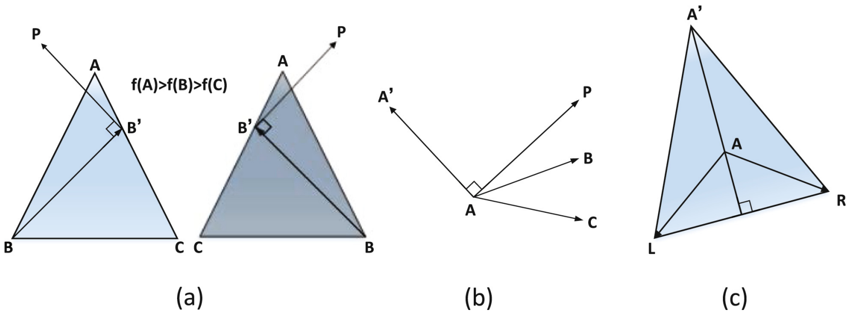

Fig. 2. (a) Calculate the gradient direction when all robots have different fitness values. (b) Robots determine their own roles in role switching process. (b) Robots maintain the triangle formation with the aid of their internal compasses.

Random Search. As is mentioned above, the leaders will search randomly in areas without fitness. And there are already some conclusions drawn from the one-dimensional and bidimensional cases of random search [13].

– When the coverage rate of influence scopes of all targets is high (i.e. dense distribution). “stochastic laws governing *run and tumble* movement patterns come into play and have a clear impact on the search success, with Lévy-like features becoming beneficial [13].”

- When the coverage rate is low (i.e. sparse distribution), ballistic strategies are optimal for non-renewable targets while Lévy flights are optimal strategies for renewable targets.

In our problem model, the coverage rate is high and the targets are non-renewable, so a Lévy-like strategy is suitable and the leader in TFS strategy performs ballistic flights reoriented at exponentially distributed times, and the mean value of the distribution is set to $2 * MapLength$ (i.e. $2 * 1000$).

Gradient Estimation. The leader will integrate the fitness values and positions of the team, and calculate the approximate gradient direction based on the supposition that the fitness value varies almost linearly alone a line in local area. Our basic idea is to construct a vector perpendicular to the local contours, and various cases are presented as follows.

- Case I: all three members share the same fitness value, which means that the team is in the area without fitness, and the leader will search randomly.
- Case II: two robots share the same better fitness value, then the gradient vector equals the center of the two better positions minus the worse one.
- Case III: two robots share the same worse fitness value, then the gradient vector equals the better position minus the center of the two worse ones.
- Case IV: all robots have different fitness values, then the gradient is perpendicular to the local contour line (Fig. 2(a)). In Fig. 2(a), A, B and C stand for the robots in a team, and the fitness values satisfy the inequality $f(A) > f(B) > f(C)$. Based on the local linear variation, $f(B') = f(B)$ and the position of B' can be calculated from Eq. 1. The line BB' serves as a contour line, whose vertical vector $B'P$ is the gradient direction (Eq. 2).

$$\boldsymbol{BB'} = \boldsymbol{BC} + \boldsymbol{CA} \cdot \frac{\mathrm{f}(B) - \mathrm{f}(C)}{\mathrm{f}(A) - \mathrm{f}(C)}. \tag{1}$$

$$\begin{cases} \boldsymbol{B'P} \cdot \boldsymbol{BB'} = 0 \\ \boldsymbol{B'P} \cdot \boldsymbol{B'A} > 0 \end{cases}. \tag{2}$$

Role Switching. In order to avoid turning abruptly, to facilitate the maintenance of formation and the estimation of gradient direction, a role switching trick is introduced. At each iteration, the robot with the maximum fitness value serves as the leader while the other two determine their roles (i.e. left wing or right wing) according to their relative positions. As is shown in Fig. 2(b), robot A is the leader and A' is its next position. $\boldsymbol{AP}$ is the right vertical vector of $\boldsymbol{AA'}$. If $\boldsymbol{AB} \cdot \boldsymbol{AP} > \boldsymbol{AC} \cdot \boldsymbol{AP}$, then robot B serves as the right wing, else the left wing.

Formation Control. Each robot is assumed to be equipped with a compass. The leader will broadcast its next position within the team per iteration, and its members will determine their roles and next positions. As is shown in Fig. 2, given the positions of A and A', the positions of left wing and right wing (i.e. L and R) can be calculated. To maintain the formation, the leader will monitor the distances(D) from itself to its members, and slow down if the distance exceeds a certain threshold (T), otherwise accelerate for high efficiency (Eq. 3, where α and β are factors for deceleration and acceleration). Parameters α and β are set to 0.75 and 1.33 respectively while T is set to $0.8 * Lengh$, where $Length$ $(0.8 * 2r_t)$ is the ideal side length of the triangle.

$$V_{leader} = \begin{cases} V_{leader} \cdot \alpha, D > T \\ V_{leader} \cdot \beta, D \leq T \end{cases} . \tag{3}$$

4 Simulation Results and Discussions

4.1 Algorithms for Comparison

Four searching strategies are chosen as comparison algorithms, which are TFS, IS, RPSO and IGES. All parameters of algorithms are tuned under the same experimental conditions, where the map size is 1000*1000, containing 50 robots and 10 targets. Details and parameters values of TFS strategy are presented in section "Triangle Formation Search Strategy".

The Independent Search (IS) strategy, is a new random walk strategy combining linear ballistic motion with triangle estimation technology. In areas without fitness, robots will move along a straight line until perceiving fitness values. In areas with fitness, robots will estimate the gradient direction using history information and triangle estimation technology introduced in Section "Gradient Estimation". Current position, the best and worst positions in history serve as the vertices. In the TFS strategy, if a robot does not belong to any team, it will search alone according to the IS strategy.

In Robotic Particle Swarm Optimization (RPSO [6]), each robot acts as a particle and determines the gbest individual in a spacial-based topology. And IGES [8] is an improved version of GES [7], and the basic idea for intra-group cooperation is moving the group center towards the center of best positions in the group, or splitting the group when the group size exceeds some threshold value or members within the group share the same fitness value.

4.2 Experimental Setup and Results

The map size is 1000*1000 and 10 targets are generated randomly, covering about 70 % of the map, while other setup information is stated in Section "An Idealized Model". In the experiment, six tests are carried out with 25, 50, 75, 100, 150, 200 robots in turn. In each test, 20 random maps are generated and each strategy is repeated for 20 times, and the results in this section are the average value of these 400 runs.

The criterion for measuring the efficiency of searching strategies, is the number of iterations required for a swarm of robots to collect all 10 targets. Another criterion indicating the computational load is the CPU time used by the swarm in simulation. The experimental results are presented in Table 1 and Fig. 3.

Table 1. Iterations and time costed by each strategy at various population sizes.

Population	RPSO		IGES		IS		TFS	
	Iterations	Time	Iterations	Time	Iterations	Time	Iterations	Time
25	587.77	23.91	294.68	15.39	**275.35**	**9.47**	312.54	14.74
50	417.94	38.90	240.00	20.13	229.74	**7.87**	**211.07**	20.14
75	374.13	64.78	217.62	41.87	208.47	**12.25**	**178.35**	25.06
100	334.41	94.95	205.07	46.09	195.72	**14.52**	**166.21**	38.21
150	295.97	196.71	189.36	81.48	176.76	**20.22**	**147.13**	64.33
200	269.84	347.61	180.45	151.41	167.71	**27.34**	**138.99**	104.85

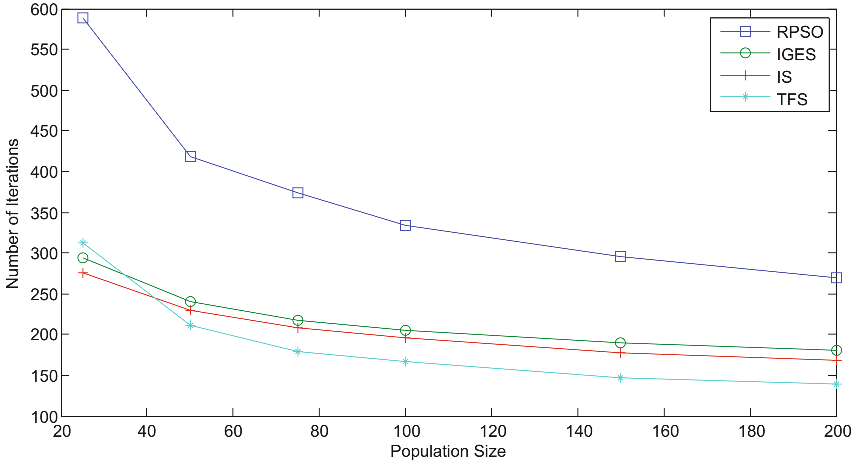

Fig. 3. Iterations costed by each strategy at various population sizes.

4.3 Discussions

As the results show, the efficiency of RPSO is the lowest among four strategies, though we have introduced a random vector to improve its performance by avoid robots' vibrating. We can infer that traditional heuristic algorithms for high dimensional optimization may not apply to swam robotics, which focuses on two or three-dimensional problem scenarios, and the key distinction is the simplicity of gradient estimation in low dimensional cases even with interference. It is worth pointing out that there are few local extrema in the fitness landscape, which plays a critical role to such results.

In the figure, there is a stable difference between IS and IGES. Since IGES has diffusion mechanism and IS works as random walk in areas without fitness, both of them possess excellent exploration ability. So the difference mainly results from the triangle estimation technology, which provides a more accurate direction to approximate the local gradient. Actually, additional experiments show that the IS integrated with independent strategies from IGES instead of triangle estimation technology, shares similar performance with IGES.

When the population size is small, the exploration ability is important for a search strategy, for IS shows the best performance while TFS performs poorly when the number of robots is 25. Under such circumstances, some mechanisms for maintaining connectivity, such as mutual attraction or formation control, may limit the exploration range of the swarm.

As the population size becomes larger (such as 50 or larger), the potential merits of the TFS strategy emerge and it outperforms other three strategies. When the population size is 75, compared with IS and IGES, the search efficiency of TFS increases 14.45 % and 18.04 % respectively, which means the triangle formation improves the local exploitation ability of the swarm and leads to a more accurate gradient direction than that estimated in IS. In addition, the downtrends of the curves in Fig. 3 demonstrate that both IS and TFS possess excellent scalability like IGES and RPSO. As we can see, three members are enough to construct a proper gradient direction while small team size tends to high exploration performance, so we adopt triangle formation technology.

As to the CPU time costed by each strategy, the IS has the overwhelming superiority, benefiting from its simplicity, while the TFS surpasses the other two strategies. Although the algorithm flow of TFS is kind of complex, its computation load is not heavy, which is a nice property for swarm robotics.

The performance of the IS strategy is qualified as a benchmark for such problem scenario in terms of efficiency and energy conservation.

5 Conclusion

In this paper, a triangle formation search (TFS) strategy and an independent search (IS) strategy were proposed, both of which bear high search efficiency and light computational load compared with RPSO and IGES. Among all four strategies, the TFS is the most efficient while the IS is the most energy-efficient, demonstrating the validity of the triangle estimation technology. In addition, the IFS and IS also show great scalability like RPSO and IGES.

As far as we know, technologies for formation control and rand walk have not been applied to the multiple targets search task in swarm robotics, and the IS strategy is qualified as a benchmark for its simplicity and performance.

Acknowledgements. This work was supported by the Natural Science Foundation of China (NSFC) under grant no. 61375119 and Supported by Beijing Natural Science Foundation (4162029), and partially supported by National Key Basic Research Development Plan (973 Plan) Project of China under grant no. 2015CB352302.

References

1. Tan, Y.: A Survey on Swarm Robotics. In: Li, J. (ed.) Handbook of Research on Design, Control, and Modeling of Swarm Robotics, 1. IGI Global, Hershey (2015)
2. Şahin, E.: Swarm robotics: from sources of inspiration to domains of application. In: Şahin, E., Spears, W.M. (eds.) Swarm Robotics 2004. LNCS, vol. 3342, pp. 10–20. Springer, Heidelberg (2005)
3. Li, J., Tan, Y.: The multi-target search problem with environmental restrictions in swarm robotics. In: 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 2685–2690. IEEE, December 2014
4. Gazi, V., Passino, K.M.: Stability analysis of social foraging swarms. IEEE Trans. Syst. Man Cybern. Part B Cybern. **34**(1), 539–557 (2004)
5. Krishnanand, K.N., Ghose, D.: A glowworm swarm optimization based multi-robot system for signal source localization. In: Liu, D., Wang, L., Tan, K.C. (eds.) Design and Control of Intelligent Robotic Systems. SCI, vol. 177, pp. 49–68. Springer, Heidelberg (2009)
6. Couceiro, M.S., Rocha, R.P., Ferreira, N.M.: A novel multi-robot exploration approach based on particle swarm optimization algorithms. In: 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 327–332. IEEE, November 2011
7. Zheng, Z., Tan, Y.: Group explosion strategy for searching multiple targets using swarm robotic. In: 2013 IEEE Congress on Evolutionary Computation (CEC), pp. 821–828. IEEE, June 2013
8. Zheng, Z., Li, J., Li, J., Tan, Y.: Improved group explosion strategy for searching multiple targets using swarm robotics. In: 2014 IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 246–251. IEEE, October 2014
9. Tan, Y., Zhu, Y.: Fireworks algorithm for optimization. In: Tan, Y., Shi, Y., Tan, K.C. (eds.) ICSI 2010, Part I. LNCS, vol. 6145, pp. 355–364. Springer, Heidelberg (2010)
10. Zheng, Z., Li, J., Li, J., Tan, Y.: Avoiding decoys in multiple targets searching problems using swarm robotics. In: 2014 IEEE Congress on Evolutionary Computation (CEC), pp. 784–791. IEEE, July 2014
11. Shlesinger, M.F.: Mathematical physics: search research. Nature **443**(7109), 281–282 (2006)
12. Viswanathan, G.M., Buldyrev, S.V., Havlin, S., Da Luz, M.G.E., Raposo, E.P., Stanley, H.E.: Optimizing the success of random searches. Nature **401**(6756), 911–914 (1999)
13. Bartumeus, F., Raposo, E.P., Viswanathan, G.M., da Luz, M.G.: Stochastic optimal foraging theory. In: Lewis, M.A., Maini, P.K., Petrovskii, S.V. (eds.) Dispersal, Individual Movement and Spatial Ecology. LNM, vol. 2071, pp. 3–32. Springer, Berlin, Heidelberg (2013)
14. Balch, T., Arkin, R.C.: Behavior-based formation control for multirobot teams. IEEE Trans. Robot. Autom. **14**(6), 926–939 (1998)
15. Amory, A., Meyer, B., Osterloh, C., Tosik, T., Maehle, E.: Towards fault-tolerant and energy-efficient swarms of underwater robots. In: 2013 IEEE 27th International Parallel and Distributed Processing Symposium Workshops and Ph.D. Forum (IPDPSW), pp. 1550–1553. IEEE, May 2013