Light Source Detection using Multirobot Systems with Particle Swarm Optimization Approach

Phát hiện nguồn sáng với hệ đa robot sử dụng thuật toán tìm kiếm bầy đàn

Anh-Quy Hoang, Minh-Trien Pham
University of Engineering and Technology – Vietnam National University, Hanoi
e-Mail: trienpm@vnu.edu.vn

Abstract
Exploration and searching in unknown or hazardous environments using Multirobot System (MRS) is among the principal topics in robotics. There have been numerous researches on searching and detection of odor, fire or pollution sources. In this paper, we present results on detecting light sources with MRS using modified Particle Swarm Optimization (PSO). The robot swarm makes use of Artificial Potential Field (APF) to avoid collision with obstacles or other robots while heading towards areas of highest illuminance. Simulation results on Matlab in various scenarios have proven the algorithm's reliability.
Keywords: PSO, MRS, APF, light source detection

Tóm tắt: Một trong số những vấn đề được quan tâm trong robotics là tìm kiếm trong các môi trường không biết trước hoặc nguy hiểm đối với con người bằng hệ đa robot. Đã có nhiều công trình nghiên cứu phương pháp tìm kiếm nhằm xác định nguồn phát tán mùi, phát hiện ô nhiễm hay đám cháy. Trong báo cáo này, chúng tôi trình bày kết quả nghiên cứu phát hiện nguồn sáng bằng hệ đa robot sử dụng thuật toán tối ưu bầy đàn (PSO) cải tiến. Được áp dụng một luật điều khiển có kết hợp phương pháp trường thế nhân tạo (APF), bầy đàn robot có khả năng tránh các vật cản trong quá trình di chuyển tới khu vực có độ rọi cao nhất. Thuật toán này đã được mô phỏng trên Matlab trong nhiều kịch bản và thể hiện độ tin cậy cao.
Từ khóa: PSO, MRS, APF, phát hiện nguồn sáng

Notation

Notation	Unit	Meaning
w		Inertial factor
a_1		Cognitive learning factor
a_2		Social learning factor
$\boldsymbol{v_i}(t)$		Velocity of particle i at time t
$\boldsymbol{v}_{PSOi}(t)$		Velocity component determined by PSO algorithm of particle i at time t
$\boldsymbol{v}_{APFi}(t)$		Velocity component determined by APF algorithm of particle i at time t
$\boldsymbol{x_i}(t)$		Position of particle i at time t
$\boldsymbol{g}(t)$		Position of leader (best particle) at time t
$\boldsymbol{p_i}(t)$		Best position of particle i up to time t
u_d, U_d		Uniform random number in [0, 1]
$\boldsymbol{F}_{APFij}$		The force exerted by potential field of a robot or a detected point on an obstacle, noted j, on robot i
$\boldsymbol{r}_{ij}$		Distance vector from a robot or a detected point on an obstacle, noted j, to robot i
r_{ij}		Distance between a robot or a detected point on obstacle, noted j, to robot i
r_{rep}		Radius of repulsive zone surrounding a robot
u(r)		Heaviside step function
Φ	lumen	Luminous flux
I	candela	Luminous intensity
L	lux	Illuminance
λ_2		Algebraic connectivity of a graph

Abbreviation

PSO	Particle Swarm Optimization
MRS	Multirobot System
APF	Artificial Potential Field

1. Introduction

As a result of their ability to cooperate and coordinate, MRSs are highly efficient in the application of searching in unknown environments. There have been numerous works on using MRS to detect fire, pollution and odor sources [1]. In these applications, MRSs are always preferable to single robots because of their robustness to local optima [2], widespread coverage and high degree of accuracy. Numerous algorithms have been developed or adapted for MRSs, and PSO is one of them.

PSO was first introduced by Russel Ebenhart and James Kennedy in 1995 [3] and has rapidly gained popularity among bio-inspired heuristic algorithms since then. The algorithm has proven highly effective in optimization problems and is famous for its efficiency, intuitiveness and simplicity to implement. For the above-mentioned reasons, PSO emerges as a natural choice for MRSs in exploration mission. However, an MRS controlled by PSO algorithm is notoriously prone to collision as an intrinsic property. Recently, there have been numerous modified PSO algorithms proposed for mobile robots in real situations. Examples include MPSO with the introduction of odor-gated rheotaxis to solve real-world odor source localization problems [4] and VL-ALPSO to optimize the motion planning while the swarm mobile robots is searching for odor or light sources [5]. Both the modified PSO algorithms are applicable for mobile robots in obstacle environment. In this research, to avoid collisions with obstacles and mutual collisions within the swarm during their exploration, we integrate into PSO an algorithm originally proposed by Oussama Khatib in 1986 for single robot path planning, namely APF [6]. Nowadays, APF is widely used in researches on MRSs that demonstrate the interaction between robots and obstacles in their work space [7]. APF generates around each robot a virtual potential field containing repulsive field and attractive field. Attractive field directs each robot towards other robots in the system to remain system connectivity while repulsive field prevents mutual collisions or collisions with obstacles. The forms of potential field and control rule are customized in conformity with specific problems. In this research, the entire potential field around each particle is repulsive, since our objective is to dispose of collision in the swarm. The communication range of each robot is larger than search space's size.

In the simulations, we have successfully used an MRS to detect light source or the brightest area in the search space. In all scenarios, each robot (or particle as described in PSO algorithm) has to move towards the mutual target and meanwhile avoid obstacles. For the robot swarm to exhibit this behavior, we modified PSO algorithm by associating each particle with a potential field that exerts repulsive forces to any other particle if their distance is less than a predetermined value called repulsive radius (r_{rep}). With reliable and promising Matlab simulation results, this modified PSO algorithm is hopefully applicable in various further applications such as dynamic deployment of robotic systems, flame detection or optical wireless charging.

This paper presents theoretical background and experiments using modified PSO to solve the practical problem of detecting light sources (or searching for brightest regions in a search space). The methodology is discussed in detail in part 2, the simulations, results and discussions follow in part 3. Finally, part 4 concludes this paper with main conclusions and directions for further researches.

2. Methodology

2.1 APF and its application in the modified PSO algorithm

The APF model is inspired by Artificial Physics with quadratic functions, where the choice of coefficients is commensurate to the wireless sensor network of MRSs. Myriads of architectures for APF have been developed in accordance with the users' definitions and specific tasks. In any architecture, the magnitude of potential force existing around each robot is continuously updated based on information collected from its immediate surrounding environment and other robots via connection network. Therefore, artificial potential forces are used to regulate the relation between robots in term of position. Potential forces are categorized into two main groups: passive forces and active forces. Passive forces are generated when robots emit signals and determine distances to neighboring robots or obstacles by the magnitude of reflected signal to avoid obstacles or remain relative position with other robots. The signal used in the application could be infrared, ultrasound, laser or camera [10]. On the contrary, active forces are generated from signals from outside sources, usually by other robots and transmitted via the communication system [7]. In this research, APF is only utilized for the purpose of collision avoidance and only generates repulsive forces on other particles within repulsive region. The repulsive force between robot i and robot j is defined by this formula:

$$\boldsymbol{F}_{APFij} = G\frac{\boldsymbol{r_{ij}}}{r_{\boldsymbol{ij}}^3}\left[u\left(r_{\boldsymbol{ij}}\right) - \mathrm{u}\left(r_{\boldsymbol{ij}} - r_{rep}\right)\right] \tag{1}$$

Where G is a predetermined constant used to regulate the magnitude of potential forces. Distances between robots are measured from their centers. The magnitude of this force is zero if i = j.

Robots in the swarm cannot see an obstacle as a whole, however, they can detect points on obstacles via on-board sensors. To the robots, obstacles are a set of points from which they receive repulsive forces. The force exerted on a robot by each detected point is determined by an equation similar to equation (1):

$$\boldsymbol{F}_{APFik} = G\frac{\boldsymbol{r_{ik}}}{r_{\boldsymbol{ij}}^3}\left[u\left(r_{ik}\right) - \mathrm{u}\left(r_{ik} - r_{rep}\right)\right] \tag{2}$$

Total force exerted on i-th robot of the system is:

$$\boldsymbol{F}_{APFi} = \sum_{i=1}^{\boldsymbol{N}} \boldsymbol{F}_{APFij} + \sum_{k=1}^{M} \boldsymbol{F}_{APFik} \tag{3}$$

Where N is swarm population, M is the number of detected points on obstacles.

$\boldsymbol{v}_{APFi}(t)$ is proportional to this total APF force. The impact of $\boldsymbol{v}_{APFi}(t)$ on overall velocity is controlled by G. As G increases, the particle is less likely to approach obstacles.

The next section gives more details on the modified PSO algorithms and shows how APF is used to update particle's velocity.

2.2. PSO and the modified PSO algorithm

Motivated by social behavior of natural swarms in search of food or other commodities, PSO is especially effective when applied to robotic swarms to find local and global optima. The swarms consist of homogeneous particles capable of exploring the search space collectively. At each step of the exploration, the movement of each particle is controlled by a velocity comprised of three components: inertial, cognitive and social velocity. Inertial velocity guides the particle towards its previous direction and thus keeps particles' movement smooth, cognitive velocity leads the particle towards its personal best position and social velocity leads the particle towards the global best position [2]. The social learning factor should be increased and cognitive factor should be decreased throughout the exploration in order to enlarge the swarm's coverage at initial steps. The searching process using PSO is implemented in four steps as described below:

1. Initialize: Generate the population (including setting initial position and velocity of every particle and evaluating their objective (fitness) function).

2. Update leader and best positions: Check the swarm for new global best position and for new personal best positions. The fitness of a position is determined by the value returned by objective function. The personal best position of a particle is updated if it is worse than the particle's current position, or otherwise remains the same.

If personal best position of any particle is better than global best position and all other personal best positions, the particle becomes new leader and its personal best position becomes global best position.

3. Update velocity and position: The position and velocity of each particle are updated according to the following equations:

New velocity = inertial velocity + cognitive velocity + social velocity

$$\boldsymbol{v}_{PSOi}(t+1) = w\boldsymbol{v}_i(t) + a_1 u_d\left(\boldsymbol{p}_i(t) - \boldsymbol{x}_i(t)\right) + a_2 U_d\left(\boldsymbol{g}(t) - \boldsymbol{x}_i(t)\right) \quad (4)$$

New position = current position + new velocity × time interval

$$\boldsymbol{x}_i(t+1) = \boldsymbol{x}_i(t) + \boldsymbol{v}_i(t+1)\Delta t \quad (5)$$

4. Check stopping criteria: The process is terminated if at least one of the following criteria is satisfied:

- The current step is equal to the last step
- The swarm has converged

Otherwise, the whole process is repeated from step 2.

To apply PSO to an MRS, each robot is modelled as a particle of the swarm and their movements in the search space are basically in the same manner as those of ideal particles described above. Actual implement of PSO for MRSs involves additional techniques to solve problems which are not covered in its conventional version (because of the ideal nature of this algorithm), such as connectivity maintenance and collision avoidance. In these simulations, the communication range of a robot is assumed to be sufficient for connectivity of the MRS and APF is used to avoid collision. The actual velocity of a particle is determined by:

$$\boldsymbol{v}_i(t) = \boldsymbol{v}_{PSOi}(t) + \boldsymbol{v}_{APFi}(t) \quad (6)$$

We claim that the exploration is successful if the swarm converges and the point of optimum illuminance is within its region of convergence. The criterion for convergence of the swarm will be discussed in the coming section.

2.3 Criterion for convergence

2.3.1 Convergence of a swarm in comparison with connectivity of a simple graph

In this simulation, we use the concept of connectivity in a simple graph to define the convergence of the swarm. A simple graph is said to be connected if and only if there is at least a path from an arbitrary vertex to any other vertex, i.e. any two vertices are connected. The robot system is modelled as a graph in which each robot is a vertex. There is a path between two vertices in this model when the distance between two corresponding robots does not exceed two times of repulsive radius (the radius within which the potential repulsive force is applied). The swarm converges if its graph model is connected.

2.3.2 Connectivity of simple graph

In this work, the swarm is said to be convergent if the algebraic connectivity of its graph model is positive, as the state of the graph's connectivity is used as criterion for convergence of the swarm. In this section, fundamental knowledge of graph algebraic connectivity is presented.

The Laplacian matrix is the representation of a graph in term of connectivity. It is defined as the subtraction of degree matrix and adjacency matrix.

An adjacency matrix is the means of representing which vertices (or nodes) of a graph are adjacent to which other vertices. Specifically, the adjacency matrix of a finite graph G on n vertices is the n×n matrix where the non-diagonal entry a_{ij} is the number of edges from vertex i to vertex j, and the diagonal entry a_{ii} equals to zero. The relationship between a graph and the eigenvalues and eigenvectors of its adjacency matrix is studied in spectral graph theory.

Degree matrix is a diagonal matrix containing information about the degree of each vertex. A degree matrix $D = \mathrm{diag}(d_1,...,d_n)$ is corresponding to a graph in which i-th vertex has the degree of d_i.

The Laplacian matrix has an important property: all of its eigenvalues are non-negative. Its second smallest eigenvalue λ_2 is positive if and only if the graph is connected. Then, λ_2 is defined as the algebraic connectivity of the graph [8]. Recently λ_2 has been considered a crucial parameter in network-related problems. In some works, λ_2 has been observed as a measure of system stability and robustness [9]. In our simulations, the value of λ_2 determines whether the swarm is convergent or not.

2.4 Introductory concepts in photometry

2.4.1 Measurement of light

- Luminous flux (Φ) is the measure of the power of light perceived by the human eye. The unit of luminous flux is lumen (lm). This quantity should not be confused with radiant flux – the total power emitted by a source of electromagnetic radiation. Luminous flux is derived from the product of the radiant power in Watts and the Visual response characteristic of the eye (luminosity function).
- Luminous intensity (I) is a measure of the power emitted by a light source in a particular direction per unit solid angle, which is perceived by the human eye. To put it simply, intensity is a measure of the strength of visible light emitted in a given direction. The SI unit of luminous intensity is candela (cd).
- Illuminance (L) is the total luminous flux received by a surface, per unit area. It measures how much incident light illuminates the surface. Illuminance is wavelength-weighted by the luminosity function to correlate with human brightness perception. In our simulation, light sensors equipped with the robots can detect illuminance.

2.4.2 Isotropic light source

An isotropic light source is a theoretical point source of light which radiates uniformly in all directions, i.e. its intensity is independent of radiating direction. In the simulations, light sources are Lambertian isotropic sources. Formulae for principal quantities of the sources can easily be derived from definitions:

$$I = \frac{\Phi}{4p} \tag{7}$$

$$L = \frac{Icosq}{r^2} \tag{8}$$

Where r is the distance to the source and θ is the incident angle.

3. Results and Discussions

3.1 Simulation setup and MRS configuration

In this research, we implement the modified algorithm on a homogeneous MRS in a Matlab environment. The radius of each robot (r) is set as unit of length. The system has direct communication, the communication range is unlimited (beyond the limit of search space). r_1 is 5×r, i.e. a robot can detect obstacles at the distance of 5×r from it. Population size varies between 5, 10 and 15. Maximum velocity is 1.5×r/step. Each robot is able to acquire the illuminance at its position via a light sensor on top.

If we set r = 1, the search space size is 100×100. In the Cartesian coordinate system, the ranges of x and y coordinates are both [-50, 50]. We evaluate and test the effectiveness of the modified PSO algorithm in four scenarios.

3.2 Detection of light sources in different scenarios

To evaluate the reliability and effectiveness of this modified algorithm, we run simulations in four scenarios, where number and luminous intensities of light sources alter. Throughout the scenarios, obstacles vary in position, size and shape. Positions of obstacles are referred as their centers.

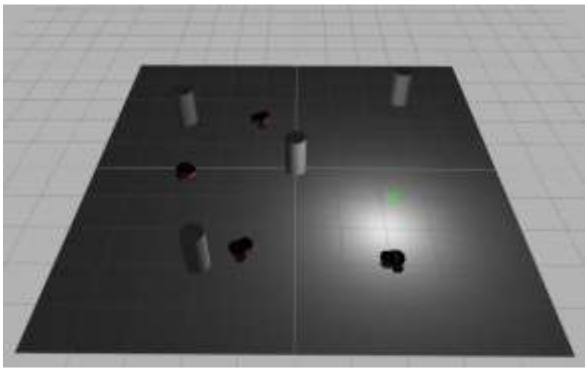

Figure 1. *Scenario 1 – Single isotropic source*

In the first scenario, a single light source is placed above the search space at (20, -20). There are four static cylindrical obstacles placed at (-30, -30), (-20, 30), (0, 0) and (30, 20) as illustrated in Figure 1. The radii of cylindrical obstacles used in all scenarios are 4.

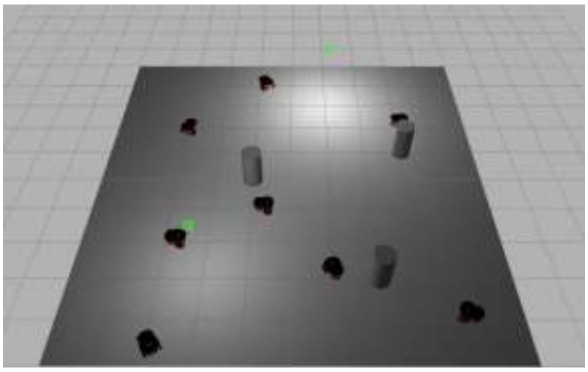

Figure 2. *Scenario 2 – Double isotropic sources*

In the second scenario, two light sources S1 and S2 are placed at (-20, -30) and (10, 30), with intensity of S2 being three times that of S1. There are three static cylindrical obstacles at (-10, 0), (20, -30), and (30, 10) (Figure 2).

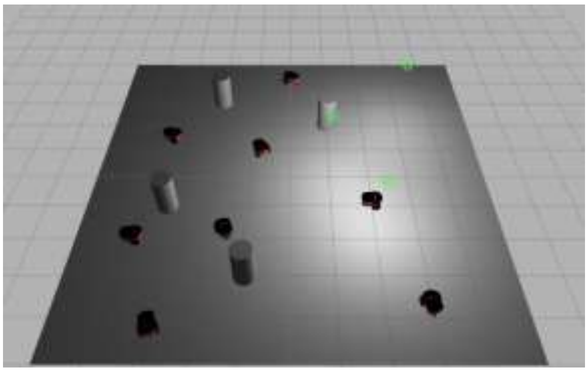

Figure 3. *Scenario 3 – Triple isotropic sources*

In the third scenario, the algorithm is tested in a search space with three isotropic sources S1, S2 and S3 at corresponding positions of (-10, 0), (30, 20) and (20 -20). The intensity of S2 is a half that of S1 and a third that of S3. Four cylindrical obstacles are placed at (-30, -10), (-10, -30), (-20, 30) and (10, 20) (Figure 3).

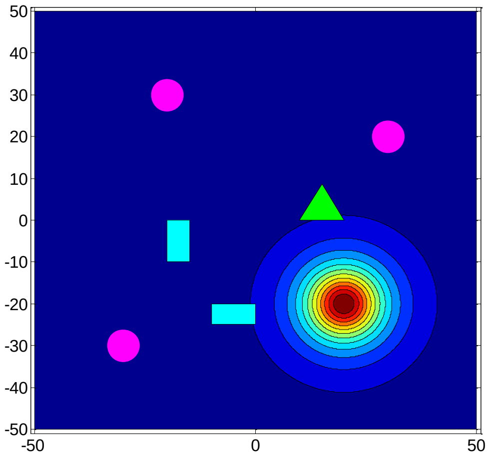

Figure 4. *Scenario 4 – Obstacles of varied shapes and sizes*

For the last scenario, we evaluate the modified algorithm's ability to avoid obstacles of diverse shapes and sizes. Light source used in this scenario is a single isotropic source. Three cylindrical obstacles are placed at (-30, -30), (-20, 30), and (30, 20). There are also three prismatic obstacles in the search space, including a triangular prism and two rectangular prisms. The triangular obstacle is placed vertically at (15, 4.33). Its base is an equilateral triangle with each size being 10. Two rectangular obstacles are placed at (-17.5, -5) and (-5, -22.5), their base dimensions are 5×10 and 10×5, respectively.

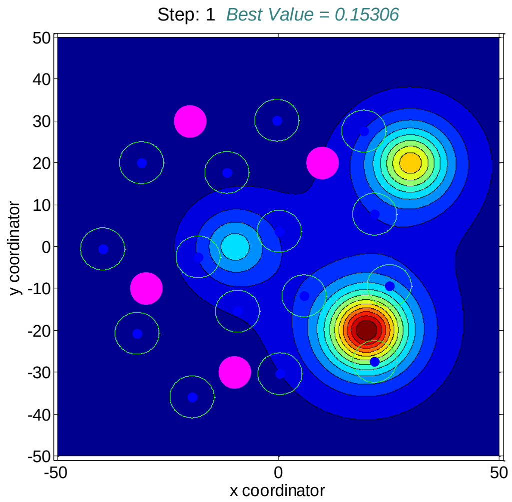

Figure 5. *Initial Deployment with 15 robots*

A completely arbitrary initial deployment of the swarm may allow for the possibility of premature convergence or local optima stagnation. To avoid such situations, the swarm is arranged uniformly in the initial step, so that population density just varies slightly across search space. Such commencing deployments yield a success rate of 100% in the simulations. The deployment demonstrated in Figure 5 has the desired property, with the distances from a robot to its neighbors differ slightly.

In Figure 5, the violet circles represent static obstacles. Each robot is a blue circle with a green circle outside delineating its repulsive region. At final steps, the swarm converges around the global maximum. The contours demonstrate illuminance on the search space. Figure 6 shows the typical final arrangement of the swarm in a simulation:

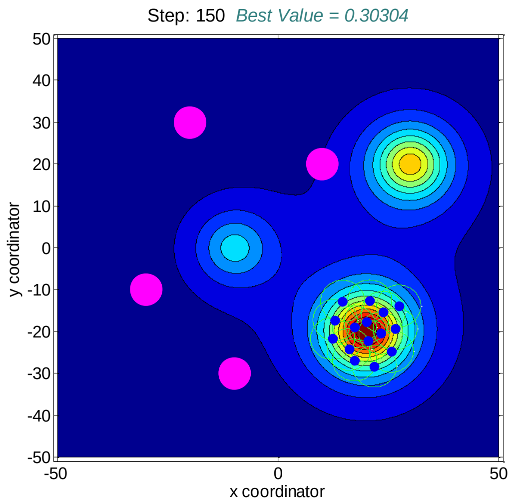

Figure 6. *Typical final deployment of MRS*

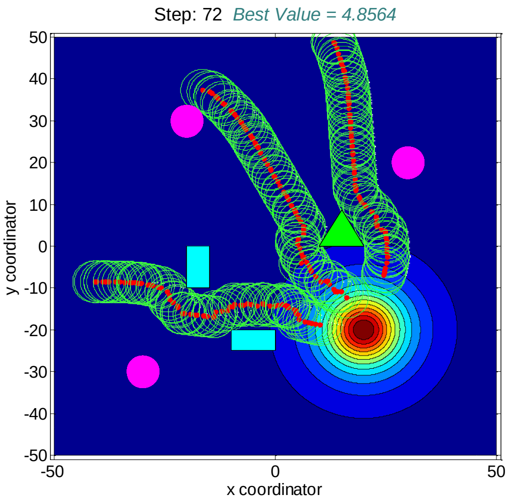

Figure 7. *Trajectories of swarm robots*

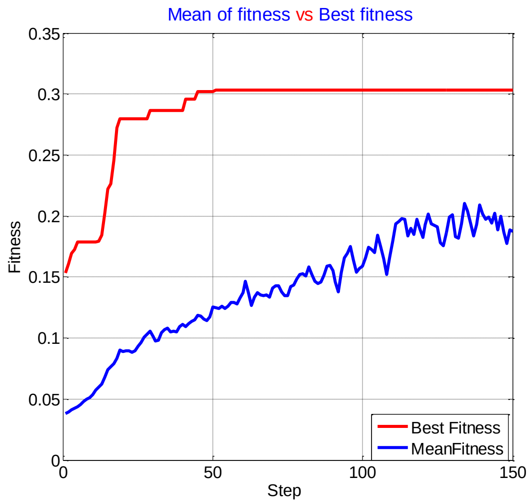

Figure 8. *Searching Performance*

The modified PSO algorithm presented in this research is a combination of APF and PSO, thus it has the advantages and also the disadvantages of APF. The robots controlled by the algorithm are incapable of escaping from local minima entrapment. Such situations are likely to occur in the presence of specially-shaped obstacles (e.g. U-shaped or V-shaped obstacles) or when an obstacle is too large, compared to the dimensions of the search space and sensing range of a robot.

The applicability of this modified algorithm is confined to the situations in which obstacles are simple (i.e. perceived by robots as polygons or circles) and their sizes are not too large. In such situations, simulations show that the swarm robots controlled by the algorithm can successfully avoid obstacles while heading towards their target. Figure 7 depicts trajectories of three robots during the exploration.

As the exploration of this swarm progresses, the robots move towards global best position. Thus, average value of illuminance at positions of the robots (the blue line in Figure 8) increases. Although the best position is discovered at as early as step 50, the average value does not approach highest value even in last steps. This is because after the swarm converges, some robots are in the repulsive region of others and unable to get closer to the global best position, while the objective function is quite steep.

Since the modified PSO is a stochastic algorithm, we run each simulation for numerous times and present the relevant statistics in the following tables. The data of interest are the steps at which the swarm converges in each run (step of convergence – SC). Each specific case is run 100 times.

It can be deduced from the tables that average step of convergence and population size do not stand in a monotonic relationship. Thus, an increase in the number of robots does not always result in a decrease in average step of convergence, as normally expected. However, a common pattern can be observed in these data sets: average step of convergence increases and then decreases as the number of robots increases from 5 to 15. In future researches, we will go deeper into this point and build a model for the relationship between these two quantities to figure out the optimal number of robots for specific explorations.

In general, more optima requires more iterations for the swarm to converge. By comparing results given in Table 1 and Table 4, we can deduce that the impact of obstacles' shape on the overall performance of an MRS in the process of exploring and detecting light sources is not considerable. The standard deviation of step of convergence fluctuates around ten percent of average step of convergence throughout specific cases. This high standard deviation substantiates the impact of random factors in this algorithm. However, despite of its stochastic nature, this algorithm guarantees the swarm's ability to avoid obstacles and claims perfect success rate in the simulations for detection of ideal light sources using MRS.

Number of robots	Average SC	Standard deviation of SC
5	69.19	5.19
10	74.32	5.62
15	55.14	6.32

Table 1. *Searching performance in scenario 1*

Number of robots	Average SC	Standard deviation of SC
5	116.82	9.08
10	162.47	13.62
15	155.35	16.67

Table 2. *Searching performance in scenario 2*

Number of robots	Average SC	Standard deviation of SC
5	136.91	16.32
10	151.72	18.33
15	121.43	16.87

Table 3: *Searching performance in scenario 3*

Number of robots	Average SC	Standard deviation of SC
5	69.92	5.12
10	75.23	5.68
15	62.12	8.09

Table 4: *Searching performance in scenario 4*

4. Conclusions and future work

In this paper a modified PSO algorithm with the integration of APF for an MRS to explore unknown environments and detect light sources is developed and evaluated. The features presented and discussed in previous sections have been implemented in Matlab and experimental results showed good performance of the algorithm. With the success rate being 100% and ability to avoid obstacles, this modified algorithm is a promising resolution for some practical problems involving the utilization of MRS and optical applications, such as dynamic deployment of robotic systems, flame detection or optical wireless charging.

Nevertheless, there exist some drawbacks in this algorithm that we need to overcome to make it a pragmatic approach to real-world problems, such as the inability to detect multiple sources (currently this algorithm is only effective to find global optimum) or the dependence upon random factors (that leads to high standard deviation of SC). This algorithm also needs to be modified to cope with the problem of local optima entrapment.

In future researches, apart from improving the overall performance of this algorithm, we will focus on determining optimal swarm population for specific types of problems, detecting multiple sources, carrying out simulation with real light sources and finally applying the algorithm on real MRSs.

References

[1] Marques, L., de Almeida, A. T., *Finding odours across large search spaces: A particle swarm-based approach*. Climbing and Walking Robots, pp. 419-426. Springer Berlin Heidelberg 2005.

[2] Couceiro, M. S., Rocha, R. P., Ferreira, N. M., *A novel multi-robot exploration approach based on particle swarm optimization algorithms.* Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on, pp. 327-332. IEEE, 2011.

[3] Kennedy, J., Eberhart, R., *Particle swarm optimization*. IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, IV, pp. 1942-1948, 1995.

[4] Jatmiko, W., Sekiyama, K., Fukuda, T., *A pso-based mobile robot for odor source localization in dynamic advection-diffusion with obstacles environment: theory, simulation and measurement*. Computational Intelligence Magazine, IEEE, 2(2), 37-51, 2007.

[5] Tang, Q., Eberhard, P., *A PSO-based algorithm designed for a swarm of mobile robots.* Structural and Multidisciplinary Optimization. 44(4), 483-498, 2011.

[6] Khatib, O., *Real-time obstacle avoidance for manipulators and mobile robots.* Autonomous robot vehicles (pp. 396-404). Springer New York, 1990.

[7] Ngo, T. D., *LinkMind: Link optimization in swarming mobile sensor networks.* Sensor (2011), Volume 11, Issue 8, Publisher: MDPI (Basel, Switzerland), pages: 8180-8202, 2011.

[8] Sabattini, L., Secchi, C., Chopra, N., Gasparri, A., *Distributed control of multirobot systems with global connectivity maintenance.* Robotics, IEEE Transactions on, 29(5), 1326-1332, 2013.

[9] Fax, J. A., Murray, R. M., *Information flow and cooperative control of vehicle formations*. Automatic Control, IEEE Transactions on, 49(9), 1465-1476, 2004.

[10] Lee, K. Y., Park, J. B., *Application of particle swarm optimization to economic dispatch problem: advantages and disadvantages.* Power Systems Conference and Exposition, 2006. PSCE'06. 2006 IEEE PES (pp. 188-192). IEEE, Oct. 2006.

Hoang Anh Quy graduated from Vietnam National University (VNU) in 2015 and is currently working as a research assistant at University of Engineering and Technology, VNU. His research interests involve biomimetic methods and Wireless Sensor Network. He has conducted researches on PSO algorithm and Multi-robot Systems since 2013.

Pham Minh Trien received his B.S. and M.S. degrees at the Department of Electronics of Vietnam National University in 2003 and 2007, respectively. He received PhD degree in Electrical Engineering from Chungbuk National University, South Korea in 2011. He is currently a lecturer at University of Engineering and Technology, VNUH. His research interests include optimization algorithm, multi-agents, and surrogate model applying in Multi-robot system, Networks and Electromagnetic design.