Optimal path planning for motion robots based on bees pollen optimization algorithm

Thi-Kien Dao a, Jeng-Shyang Pan b, Tien-Szu Pan a and Trong-The Nguyen c

aDepartment of Electronics Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan; bFujian Provincial Key Lab of Big Data Mining and Applications, Fujian University of Technology, Fuzhou, People's Republic of China; cDepartment of Information Technology, Hai Phong Private University, Hai Phong, Vietnam

ABSTRACT
Due to interference phenomena among unnatural dimensions of the motion robots’ operations space, optimal path planning of them has to satisfy not just one criterion, but rather multi-objects. In this paper, we propose a novel multi-object approach for optimal mobile robot path planning, based on bees pollen optimizer (BPO). We consider two objects of distance and smooth path of the special plan for motion robots for constructing a minimization one. In operation environment for action robots, the location of the target and the obstacles are set up for the solution of BPO. The selected sequence of the mobile robot is a set of the chosen global best settlement in each iteration, which updates its archived data throughout the movement for motion robots in order. A series of simulations are executed in some environments for the best pathway once the robot reaches its goal. The results indicate that the proposed approach offered the robot path to its target without touching the obstacles, and the proposed method may be an alternative approach to optimize the motion robot path planning.

ARTICLE HISTORY
Received 1 May 2017
Accepted 27 May 2017

KEYWORDS
Bees pollen optimizer; robot path planning; computing intelligence

1. Introduction
Mobile robots have been widely applied in many fields, including industry, agriculture, architecture, and military because of their abilities to function well in hazardous environments (Siegwart & Nourbakhsh, 2004). An efficient path planning based on the parameters (e.g. cost, energy, time, and distance) for a motion robot plays an important role in mobile robot navigation. Finding an optimal path from start point to target point without colliding with the obstacle in an environment of the robot working space is the most necessary task for efficient route planning (Samson, 1995). An optimization route of the motion robot has to satisfy goals such as shortest path, lowest energy consumption, or right time, without colliding with the obstacle on its paths (Raja & Pugazhenti, 2012).

The problem of effective route planning has attracted exceptional attention from many researchers, and many critical studies results had been obtained using both traditional and
metaheuristic methods. If the scale of a problem is large enough and has high degrees of freedom, the conventional methods involve high computational costs and a load of complexities (Volos, Kyprianidis, & Stouboulos, 2012). The robot route-planning problem is a fundamental NP-completeness problem. The proposed based on the inspiration of the swarm intelligent, the evolution genetic, or the natural phenomena, the metaheuristic algorithms can deal with the mentioned drawbacks of classical methods. The metaheuristic methods have been applied successfully to solve the problems in the robot path planning (Qu, Xing, & Alexander, 2013; Raja & Pugazhenthi, 2009; Samadi & Othman, 2013). Also due to NP completeness of the route-planning problem, metaheuristic algorithms have been created increasingly to face with heavy computational load, and complexities involved made the traditional methods difficult to use, with high degrees of freedom of route planning (Chu, Lee, & Sunwoo, 2012).

Artificial bees colony optimization (ABC) (Karaboga, 2005) and flower pollination algorithm (FPA) (Pan et al., 2017; Yang, 2012) are two powerful metaheuristic algorithms. These algorithms have been applied in many successful applications related to engineering and management fields (Chiroma et al., 2015; Karaboga, Gorkemli, Ozturk, & Karaboga, 2014). However, the disadvantages of these algorithms are the existence of the premature convergence in the later search period, which could be easy to converge to a local optimum, and loss of the accuracy of requirements sometimes. Diversity optimizations could overcome this issue because of difference jump out of local extrema. Improved diversity agents for optimization algorithm can be implemented by using communication between two algorithms and exchanging information among subpopulations of them.

In this paper, we extend our previous work (Pan, Dao, Nguyen, Chu, & Pan, 2016) by two further contributions that include implementing multi-objective optimization and, solving the mobile robot path planning based on the multi-objects. Because a robot needs to keep a safe space to avoid collisions with obstacles on its path, from a start point to the target point, the preventing trap local extrema has indeed been considered while planning pathways. Moreover, in practical situations, the environment of working space of robots has not just one criterion but more than one criterion. However, most of the optimal robot route-planning approaches have optimized this problem based on a single objective function, e.g. the path length. The obtained results from these methods are non-smooth, and robots might lose energy and time whenever moving abruptly or turning course. Many motion robot operations required a route planning that is efficient over several criteria.

Additionally, because of the previous work is a novel proposed algorithm of bees pollen optimizer (BPO) based on communicated ABC and FPA that is only for the single objective function in optimization. For the motion robot route planning with two objectives simultaneously, e.g. the distance and smooth criteria, the one objective function in the optimization of BPO cannot work out this problem of the path planning. This paper suggests multi-object BPO (MBPO), and figures out the motion robot route planning with two objectives simultaneously, based on MBPO.

2. Mobile robot path planning

The motion robot path planning is mainly modelled in an operation space of robot navigation as follows. Two-dimensional robot action area includes the obstacles, danger sources, and given robot (Sariff & Buniyamin, 2006). The primary task is to find out an
optimal collision-free path from start point to the target state. Optimal criteria for route planning are according to some preferred merits – e.g. the length, time, smoothness, and energy (Spong, Hutchinson, & Vidyasagar, 2006).

We model the environment of robot operations and the related objective functions of the robot path planning problem. Two considered phenomenon criteria of the path planning problem area for constructing objective functions include the shortness length and the smooth way.

2.1. Modelling of the robot working space

Because of moving points of the robot in two dimensions of robot workplace, coordinates $O-XY$ are used to describe the position of mobile robot and obstacles. The start point and the target position of the motion robot, the polygon entities, and the dashed circles are utilized to represent the constraints, respectively. To decrease the dimension of the decision variable, we use a transformed coordinate to locate the new $X'$-axis to coincide with the line Start-to-Target when Start-to-Target intersects to the $X$-axis. The formula of corresponding transformed coordinates is

$$
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = \begin{bmatrix}
  \cos \theta & -\sin \theta \\
  \sin \theta & \cos \theta
\end{bmatrix} \times \begin{bmatrix}
  x \\
  y
\end{bmatrix} + \begin{bmatrix}
  x_{\text{Start}} \\
  y_{\text{Targ}}
\end{bmatrix}
$$

where $\theta$ is angle of anti-clockwise rotation from the $X$-axis to the line Start–Target, $(x_{\text{Start}}, y_{\text{Targ}})$ is the point Start in the coordinates $O_{x,y}$, and $(x', y')$ is the point $(x, y)$ in the new coordinates $Start_{x_0,y_0}$. First, the line Start–Target is divided into $n + 1$ equal segments by $n$ points. After drawing $n$ vertical lines through these points in turn, a set of parallel lines were denoted $\{l_1, l_2, \ldots l_n\}$. A complete path $\{p_1, p_2, \ldots p_n\}$ can be constructed by sampling at random on vertical lines of $l_1, l_2, \ldots l_n$. Hence, the robot path planning problem is transformed into optimizing the following set of points $\{\text{Start}, p_1, p_2, \ldots p_n, \text{Target}\}$. This path is a collision-free constraint. It means obstacles do not cover each point on this route, and each line among the set does not intersect with obstacles. The generated decision vector of search agents is at the beginning of the robot’s initialized position and regarding its sensing range.

2.2. Modelling objectives of the robot path planning

Let $L$ be the length of supposing path for moving robot on working place. It is a total of the stepped lines through the points that robot could pass or go. The approximated length of a path is as follows:

Supposing that the start state and the target state are $p_0$ and $p_{n+1}$, the length of a path can be approximated by

$$
L(p) = \sum_{i=0}^{n} d(p_i, p_{i+1})
$$

where $p_0$ and $p_{n+1}$ are the start and target states, respectively; $d(p_i, p_{i+1})$ denotes the distance between $p_i$ and $p_{i+1}$. In the coordinates $Start_{x_0,y_0}$, since the line Start-to-Target is
divided into $n+1$ equal segments, the value of $d(p_i, p_{i+1})$ can be calculated as follows:

$$d(p_i, p_{i+1}) = \sqrt{(x'_{p_i} - x'_{p_{i+1}})^2 + (y'_{p_i} + y'_{p_{i+1}})^2} \quad (3)$$

The generated agents of the algorithm population are such that along each sensing direction, an agent is at a certain distance from the robot, determined by the range of the used sensor.

Two considered phenomenon criteria of the path planning problem area for constructing objective functions include the shortness length and the smooth way that paid attention in this paper. The shortness path considers as the Euclidean distance of the moving agent for the first objective function. This range can be defined from the starting point to the goal positions in working place over each iteration:

$$F_1(p) = \sum_{i=0}^{n-1} d_i. \quad (4)$$

If any point of the obstacle is within the sensing range in the direction of a robot, a position near the barriers’ border will be selected as the searching component at that direction. Smoothness function is modeled for fit the segments in the path of the robot move forward safe and sound. The smoothness path is metric as the second objective function. This objective is mathematically described as the angle between the two imaginary lines connecting the goal point to the robot’s two successive positions over each generation.

The second objective function is computed as follows:

$$F_2(p) = \sum_{i=0}^{n-1} \varphi_i + \delta \times L, \quad (5)$$

where $L$ is the length of a mobile robot path included line segments; the angle of $\varphi_i$ is angle between the two line segments $(0 \leq \varphi \leq \pi)$, connecting the point $p_i$; $\delta$ is a positive constant. The objective functions of shortness criterion and smoothness are referred as $F_1(p)$ and $F_2(p)$, respectively. The total cost of fitness or objective functions of the feasible path $p$ included $n$ points are obtained by optimization process in the later section.

3. Bees pollen optimizer

BPO (Pan et al., 2016) was constructed based on the advantages of both artificial bees colony optimization (ABC) and FPA – e.g. robustness, fast convergence, and high flexibility. Before reviewing in detail the deployed construction of BPO, we need reviewing both the ABC and FPA algorithms summary briefly as below subsection.

3.1. Artificial bee colony optimization

Artificial bee colony optimization (ABC) is inspired by simulating the bees’ behaviours on sharing the searched food and nectar in their hive (Karaboga, 2005). The description of optimization with ABC is as follows.
A randomly generated proportion of ABC population is spread out into a solution space that is called the employed bees. The fitness value of these candidates is referred as nectar amount. A probability candidate of the employed bees is given as follows.

\[ \theta_i = \frac{F(p_i)}{\sum_{k=1}^{K} F(p_k)}, \]  

where \( K \) is some employed bees; \( p_i \) is of the \( i \)-th employed bee position; \( \theta_i \) is the probability the \( i \)-th employed bee of a food source; \( F(p_i) \) is assigned to get fitness function value of \( p_i \).

A food source is selected to move for every Onlooker bees by using the roulette wheel is used to select Onlooker bees based on the food source. Onlooker’s positions are modelled as follows:

\[ x_{ij}(t + 1) = p_{ij}(t) + \varphi(p_{ij}(t) - p_{kj}(t)) \]  

de where \( \varphi \) is a random number with range \([-1, 1]\), \( t \) is the number of iteration, \( p \) is the chosen employed bee randomly, and \( j \) is index of the solution dimension. The best location based on found fitness values so far are updated for the recorded memory.

The moving Scouts: if the improved working bees have no values with a so-called Limit of the iterations continuously, the food sources will be exhausted, and these employed bees will be transformed to scouts. ‘Limit’ is a predetermined number of iterations of continuous. The scouts are transformed by Equation (8):

\[ p_{ij} = p_{j\min} + \varepsilon \times (p_{j\max} - p_{j\min}), \]  

where \( \varepsilon \) is random variable with range \([0, 1]\). Checking termination condition is whether maximized iteration satisfies the termination or not. If stopping is not true, the programme will go the onlooker searching step in Equation (7); otherwise, it will be terminated, and the results are output and recorded.

### 3.2. Flower pollination algorithm

FPA is drawn inspiration from two pollination processes of the flowering plant, including self-pollination and cross-pollination (Yang, 2012). In the flowering plant, pollen are transported by pollinators according to the rules of Lévy flights, and they can self-pollinate randomly. Two universal concepts that guide the optimal process in the population-based algorithm are for exploring and exploiting the search space. A self-pollination of the flowering plant viewed as local pollination has been expressed for exploitation in the search area. However, a cross-pollination considered as global pollination that has been represented for exploration of a promising area search.

Because the flower pollination processes can occur at both local and global, to imitate this feature, the proximity probability \( p \) (denoted \( p \in [0, 1] \)) is to switch between natural global pollination to intensive local pollination. It means to change between the exploring and exploiting phases in FPA to control characteristics of local and global pollination. Let \( x_i^j, x_k^j \) be solution vectors of the pollen, i.e. pollen in the same plant or the flowers. We could model for the local pollination as follows:

\[ x_i^{j+1} = x_i^j + u \times (x_i^j - x_k^j) \]  

where \( u \) is a random variable with a distributed uniform \([0, 1]\). In cross-pollination considered global, gametes of flower pollen are carried by pollinators, e.g. insects. Insects
can often fly and move longer range. A Lévy flight can express flying insects over long distances with various length steps, and be used to mimic this characteristic efficiently. Let $L$ be a Lévy distribution with active as drawn formula.

$$L = \frac{\lambda \Gamma(\lambda) \times \sin\left(\frac{\pi \lambda}{2}\right)}{\pi \times s^{\lambda+2}},$$

where $\Gamma(\lambda)$ is the gamma function, and this distribution is valid for large steps $s > 0$. $L(\lambda)$ is called the step size like the parameter that corresponds to the strength of the pollination. Updating solution vectors learned from the cross-pollination for global pollination are given as

$$x_{i}^{t+1} = x_{i}^{t} + \gamma \times L(\lambda) \times (x_{i}^{t} - g^*),$$

where $g^*$ is the current best solution found so far, $\gamma$ is a scaling factor to control the step size, and $t$ is the current generation or iteration. In the evaluation for updating pollination, according to the fitness function, if $x_{j}^{t}$ and $x_{k}^{t}$ come from the same plants or the same selected population, $u$ becomes a local random walk. A probability switching $p$ is set to 0.8 as a preliminary parametric might offer to perform some applications (Yang, 2012).

### 3.3. Bees pollen optimizer

The main inspiration of this algorithm is taken based on frameworks of ABC (Karaboga, 2005) and FPA (Yang, 2012). The original idea of the algorithm is communication strategies in parallel processing for exchanging information between the populations for forwarding promising area search. The self-pollination and cross-pollination considered as the local and the global search in FPA achievement. The agents are like flies, bees, bats, and birds that can fly the long distances for playing as cross-pollination, and the pollen of the same plant is for representing as the self-pollination. Moreover, inspiration from honeybees’ behaviours on finding nectar is to define for different roles in the optimization process in ABC achievement.

The parallel structure of the employed bees and pollen is to immigrate to the best area of the strong agents among them by swapping positions. The copied best bees in ABC fly to other subpopulations in FPA to replace the poorer pollens with the best value, in reverse, the best pollens FPA would migrate to the weaker bees in ABC, substitute them, and update the positions of all subpopulations of the optimization algorithms in every period of exchanging time. The parallel structure of the employed bees and pollen is to immigrate to the best area of the strong agents among them by swapping positions. The copied best bees in ABC fly to other subpopulations in FPA to replace the poorer pollens with the best value, and update the positions of all subpopulations of the optimization algorithms in every period of exchanging time. The flow of this exchange between the bees and pollens is called communication. Several communication strategies include: a pair of groups’ exchange, the best agent of a group replace with all imperfect solution in the other groups, and some best solutions replace the weak individuals.

Moreover, evolved subpopulations could be created by dividing the population of the populated-based algorithm into, then applied the communication strategy to these group for forming a parallel structure. The advantages of applied strategies of communication to
algorithms are to achieve diversity cooperation, diversity population, and alternative running. The summarized main steps of the proposed BPO are described as shown in Figure 1. Notations used in the steps of the scheme: \( R \) is a period of exchanging solution information between FPA and ABC. \( N \) is BPO population size. \( N_1 \) and \( N_2 \) are ABC and FPA population sizes respectively, and e.g. they can be set to \( N/2 \). \( k \) (a number of the best-fitted individuals of a group) will be copied for replacing the same number of worst individuals in another group.

3.4. Evaluated BPO performance

The experimental results of the proposed BPO are compared with those obtained results of ABC (Karaboga, 2005), FPA (Yang, 2012), and Genetic algorithm (GA) (Whitley, 1994) for a set of selected six benchmark tests (Suganthan et al., 2005; Yao, Liu, & Lin, 1999) in regard to their performance of the accuracy and execution time.

Table 1 illustrates the BPO quality optimization for a set of testing functions with the ABC, FPA algorithms. Apparently, BPO method provides the better performance than those obtained from the ABC and FPA algorithms.

Figures 2–4 show the results of experiments for the first three testing problems of the BPO, ABC, and FPA algorithms in the same condition. Obviously, the performance for the cases of testing functions of BPO offers the better accuracy and convergence than the ABC and FPA.

4. MBPO for motion robot path planning

As the above-mentioned optimizer, BPO is only for the single objective problem (Pan et al., 2016). To solve multi-objective functions of the motion robot path planning, BPO has to extend to multi-objective bees pollen optimizer (MBPO is for short). MBPO is constructed based on combining BPO and Pareto-optimal front. Then applied MBPO is for dealing with
the problem of motion robot path planning. We present constructed MBPO, built objective function, modelled robot working space, and discussing results as follows.

4.1. Constructed MBPO

The multi-objective optimization problem for a maximizing/minimizing problem with \(d\)-dimensional decision vectors, a set of components of decision vector as \((p_1, p_2, \ldots, p_d) \in P \subset R^d\) and \(h\)-objectives is a form as follows:

\[
\text{Maximize/Minimize } F(p) = (f_1(p), f_2(p), \ldots, f_h(p))
\]

Subject to \(p \in [p_L, p_U]\), \((12)\)

where \(p\) is a decision vector and \(F(p)\) is the objective function with the objective vector as a set of \((f_1, f_2, \ldots, f_q) \in Q \subset R^h\). The search component in BPO is decision vector \(p\) that is belonging to the decision space \(P\) with \(d\)-dimensional. The objective function, i.e. \(F(p)\), belongs to the \(h\)-dimensional objective space \(Q\), in which the decision vector space can map the functions to the actual space, i.e. the space of objective functions. Subscripts \(L\) and \(U\) in \(p_L\) and \(p_U\) are constraints of lower and upper bounds in search components range, respectively. All the search agents of the decision vector of optimizer have to

| Benchmark tests | ABC  | FPA  | BPO  | Comparison performance (%) |
|-----------------|------|------|------|---------------------------|
| 1               | 1.45E + 00 | 1.48E + 00 | 1.05E + 00 | 38 | 41 |
| 2               | -4.87E + 03 | -4.10E + 03 | -6.09E + 03 | 20 | 33 |
| 3               | 1.65E + 02 | 1.73E + 02 | 1.29E + 02 | 28 | 34 |
| 4               | 1.60E-03 | 1.60E-03 | 1.10E-03 | 44 | 45 |
| 5               | -2.94E + 00 | -3.04E + 00 | -3.32E + 00 | 8 | 5 |
| 6               | -7.15E + 00 | -8.15E + 00 | -9.72E + 00 | 26 | 16 |
| Avge            | -7.86E + 02 | -6.56E + 02 | -9.95E + 02 | 28 | 30 |

Figure 2. Comparison of the performance of BPO with ABC and FPA for function of

\[
F_1(x) = \sum_{i=1}^n \sin(x_i)(\sin(\frac{jx^2}{\pi}))^{2m}, \quad m = 10.
\]
meet the constraints that form the decision space with the possible set of feasible, i.e. \( V = \{ p \in \mathbb{R}^d \mid p \in [p_L, p_U] \} \). The multi-objective optimization can be solved by applying the principle of Pareto-optimal solution (Horn, Nafpliotis, & Goldberg, 1994).

In Pareto-optimal solution, not component of \( p \) is larger than the corresponding element of \( q \), and at least one of the elements of \( p \) is smaller than \( q \), which is called \( q \) dominates \( p \). Similarly, the dominance relationship can be represented as follows:

\[
\Rightarrow p < q \quad \forall \; p = q.
\] (13)

A vector of solution with the domination \( p = (p_1, p_2, \ldots, p_n)^T \) on a vector \( q = (q_1, q_2, \ldots, q_n)^T \) for a minimization problem if and only if \( p_i \leq q_i \) for \( \forall \; i \in \{1, \ldots, n\} \) and \( \exists \; j \in \{1, \ldots, n\} : p_j < q_j \). The defined domination for maximization problems is only replacing denotation of \( \prec \) with the symbol of \( \succ \). Therefore, a position of Pareto solution \( x_\ast \) is called a non-dominated solution if no solution can be found that dominates on it. Let \( S \) be a set of solutions. The set of non-dominated solutions is referred to a Pareto front of a multi-objective optimization. Denoted \( Pf \) is the Pareto front of a multi-
objective optimization. $P_f$ can be modelled as follows:

$$P_f = \{ s \in S | \exists s' \in S : s' < s \} \quad (14)$$

A good approximation could be obtained from the Pareto front if a diverse range of solutions should be generated as the optimized solution space in Pareto front $S$.

### 4.2. Built objective function

As mentioned in the robot path planning problem in section 2, two objective functions of $F_1$ and $F_2$ are defined in Equations (4) and (5) for supposing the goal of optimization in this paper. The goal of the optimization here is to find the Pareto-optimal solutions. Formed MBPO is for multi-objective functions, as shown in Equation (12). The decision space includes $h$ objective space and $d$ dimension. Thus, we can deal the robot path planning problem with MBPO by applying Equations (4), (5), and (12) to construct formulated optimum as follows:

$$\text{Minimize } F(p) = (f_1(\hat{x}_i, \hat{y}_i), f_2(\hat{x}_i, \hat{y}_i))$$

Subject to $(\hat{x}_i, \hat{y}_i) \in (\hat{x}_L, \hat{y}_L), (\hat{x}_U, \hat{y}_U)$

$$i = m + 1, \ldots n, \quad (15)$$

where $p$ is an estimated coordinate assigned to $p = (\hat{x}_i, \hat{y}_i)$, that is decision vector corresponding to search agents in BPO. The lower and upper bounds of constraint values are assigned $(\hat{x}_L, \hat{y}_L), (\hat{x}_U, \hat{y}_U)$. Two objective functions $f_1$ and $f_2$ are the length path constraint and the smoothness path constraint, respectively. Generated $N$ search agents of population size should randomly distribute among the search space as uniformly as possible. Distributed agents regularly can achieve by sampling through the uniform distributions (Figure 5).

We obtained the results of the optimization from multi-objective functions based on Pareto-optimal solution. It is the ultimate goal of building a multi-objective optimal model for the mobile robot planning issues. Criteria constraints have to meet both the shortest path and the smoothest path constraints. The major essence of MBPO for the mobile robot path planning can describe as determining the dominant relationship according to the decision space possible set $\Omega$ and the Pareto front $F(p^*)$ saving Pareto-optimal solution set $S$ in an archive by Equation (15) and updating the best solution of multi-objective. The basic steps of the optimization are described in Figure 6.

### 4.3. Modelled robot working space and discussing results

The two-dimension robot working space is setting up with a mobile robot and the obstacles by points and shapes related to coordinates $XY$. The shapes of the barriers represent based on the identified centre, polygon entities, and the dashed circles. Initialized parameters of the proposed method are as follows. The population size $N$ is set to 200. The maximum number of iterations is 500. 2-D of environment map is set to resolution points of $w \times l$, e.g. $w = 300$, and $l = 500$, respectively. The obstacle shapes can be rectangle, square, or circle, and the generated position randomly for obstacles are set in the environment of the robot working space. Figure 8 shows a graphical user
Step 1: Modeling robot workspace including obstacles’ positions and shapes, and the robot’s start and target positions.

Step 2: Parsing solution is as mapping search agents to a model of robot planning during optimization.

Step 3: Implementing the proposed MBPO to find optimal paths of the above model.

Step 3.1: WHILE not maximum iterations is true; i.e., $t \leq T_{max}$, DO

Run BPO scheme as Fig. 1.

Take the best position of agent;

Calculate the objective values and the constraint-violated degree of each agent by Eq. (15)

Step 3.2: Store all non-dominated feasible particles into the feasible archive, and non-dominated infeasible agents into the infeasible archive; Update the feasible archive and the infeasible archive;

Increase the loop counter, $t = t + 1$

Step 3.4: Output optimal results.

Step 4: Guide the robot forward to the target location by the obtained optimal path.

Figure 5. MBPO-Pareto-optimal solution for two objective functions (F1) and (F2) as Equations (4) and (5).

Figure 6. Described steps of MBPO for the robot path planning.

interface scheme of MBPO for motion robot route planning with setting two dimensions up of its environments Figure 7.

The coordinates of the starting and targeting points, the number of obstacles, and the coordinate of the robot could be set or reset as shown in Figure 8. Combination of two
objective functions $f_1$ and $f_2$ as in Equations (4) and (5) with Pareto evolution solution Equation (12) could make MBPO up as in Equation (15). We applied MBPO to execute the constraint problem in Equation (15). With setting the limitations of working space boundaries, we generate multiple feasible paths randomly that the robot can select in the running path planning. A parameter of error rate (ER) is used to evaluate the performance metric of the multi-objective optimization algorithms (Deb, Agrawal, Pratap, & Meyarivan, 2000). A measured ER is the probability of whether the obtained non-dominated solution is the real Pareto frontier or not. Calculation for ER is as follows:

$$\text{ER} = \frac{\sum_{i=1}^{n'} x_i}{n'}.$$  \tag{16}

Let $x$ be a set of solutions that is obtained from Pareto frontier. A number of the optimal points obtained in Pareto frontier is set to $n'$. If any obtained solutions to be actual Pareto frontier elements is true then $x_i$ is set to 0; otherwise, 1 is assigned $x_i$.

Conducted some testing the proposed method in the experiments have done with various obstacles density based on two objective functions e.g. the path length, and smoothness. Figures 7 and 8 show the robot path planning results in different barrier shapes and density. The set concave for the obstacle shape is to verify the paths effectiveness in the

**Figure 7.** Setting environments of robot workspace. (a) Two-dimensions of robot workplace. (b) Setting centre id of sensing area of obstacles.

**Figure 8.** Optimization of robot path planning with various segments. (a) Simulation of robot planning optimization with setting line segments among obstacles. (b) The robot workspace configuration with different setting sparse solutions.
complex environments. Observedly, the motion robot could well avoid the concave obstacle and find out the shortest path from the starting position to the target.

Moreover, to verify the effectiveness of the proposed MBPO for the motion robot path planning, the experimental results are also compared with the same kind of approach, e.g. multi-object genetic algorithm (MOGA) (Castillo, Trujillo, & Melin, 2007). Figure 9 shows the comparison of the obtained results by Pareto-optimal solutions of applied MBPO for the motion robot path planning with the obtained results from MOGA. Further, the probability of the obtained non-dominated solution on Pareto frontier is calculated according to Equation (16), for MBPO the ER has received is 0.1; however, this figure for MOGA for robot planning problem RE is only 0.2. Visibly, the optimal solutions of the MBPO are closer to the Pareto frontier than the obtained of MOGA method. It means that the proposed method of MBPO shows a better performance than that of the MOGA method (Figure 10).

Figure 9. Optimization of robot planning by different density obstacles. (a) Random shapes with 60 obstacles in optimization of robot planning. (b) Random shapes with 50 obstacles in optimization of robot planning.

Figure 10. The Pareto solutions of the MOGA (Castillo et al., 2007) and the proposed MBPO methods for robot planning problem.
5. Conclusion

This study proposed an MBPO based on a novel BPO and figured the mobile robot path planning out with two objectives simultaneously, e.g. the shortest distance and smooth criteria. MBPO has taken into account the excellent strength points of the Artificial bee’s colony (ABC) and Flower pollination algorithm (FPA) as a cooperation of optimization algorithms to make the diversity agents and parallel computation for dealing with a trapped local optimal issue of the optimization algorithms. The mobile robot path planning is the NP-complete problem; therefore dropping into trap local extrema could happen during optimization.

Moreover, due to interference phenomena among unnatural dimensions of the motion robot’s operations space, optimal path planning of them has to satisfy not just one criterion, but rather the multi-objects. MBPO can be applied to work out two objectives of the shortness path and smoothness path. The operation environment of robot consists of the positions and the shapes of obstacles, and the robot’s start and target positions that were set up for the agents of MBPO that mapped to a parsing solution based on Pareto solution.

A series of simulations are executed in some environments for the best pathway once the robot reaches its goal. The results indicate that the proposed approach offered the robot a path to its target without touching the obstacles and efficiently completed the robot path planning task with a convincing performance. Comparing the results with those of other methods, it is clear that the proposed method provides a better performance and the less ER than others.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by Fujian University of Technology; Hai Phong Private University, Vietnam; KUAS.

Notes on contributors

Thi-Kien Dao obtained her bachelor and master degrees from Faculty of Information Technology, College of Technology of Vietnam National University, Hanoi in 2001 and 2007, respectively. She is currently a Ph. D. student in the Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Taiwan. Her research interests include Computational Intelligence and Robot Sensor.

Jeng-Shyang Pan obtained his B.S. degree in Electronic Engineering from the National Taiwan University of Science and Technology in 1986, the M.S. degree in Communication Engineering from the National Chiao Tung University, Taiwan in 1988, and the Ph.D. degree in Electrical Engineering from the University of Edinburgh, U.K. in 1996. Currently, he is a Dean for College of Information Science and Engineering, Fujian University of Technology, also a professor of Fuzhou University of International Studies and Trade and a director of Innovative Information Industry Research Center, Harbin Institute of Technology Shenzhen Graduate School, China. He joined the editorial board of LNCS Transactions on Data Hiding and Multimedia Security, Journal of Computers, Journal of
Information Hiding and Multimedia Signal Processing, etc. His current research interests include soft computing, information security and signal processing.

**Tien-Szu Pan** obtained his Doctorate Degree from the University of New Orleans, USA. Currently, he is an Associate Professor with the Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Taiwan. His current research interests include Computational Intelligence, Robot Design, Mechatronics, and Decentralized Control System.

**Trong-The Nguyen** received his Ph.D. degree in Communication Engineering from the National Kaohsiung University of Applied Sciences, Taiwan in 2016. He is currently a lecturer with the Department of Information Technology, Haiphong Private University, Vietnam. His research interests include Computational Intelligence and Sensor Networks.

**ORCID**

Thi-Kien Dao [http://orcid.org/0000-0002-2805-652X](http://orcid.org/0000-0002-2805-652X)
Jeng-Shyang Pan [http://orcid.org/0000-0002-2354-5834](http://orcid.org/0000-0002-2354-5834)
Tien-Szu Pan [http://orcid.org/0000-0002-6610-7527](http://orcid.org/0000-0002-6610-7527)
Trong-The Nguyen [http://orcid.org/0000-0002-6963-2626](http://orcid.org/0000-0002-6963-2626)

**References**

Castillo, O., Trujillo, L., & Melin, P. (2007). Multiple objective genetic algorithms for path-planning optimization in autonomous mobile robots. *Soft computing, 11*, 269–279. doi:10.1007/s00500-006-0068-4

Chiroma, H., Shuib, N. L. M., Muaz, S. A., Abubakar, A. I., Ila, L. B., & Maitama, J. Z. (2015). A review of the applications of bio-inspired flower pollination algorithm. *Procedia Computer Science, 62*, 435–441. doi:10.1016/j.procs.2015.08.438

Chu, K., Lee, M., & Sunwoo, M. (2012). Local path planning for off-road autonomous driving with avoidance of static obstacles. *IEEE Transactions on Intelligent Transportation Systems, 13*(4), 1599–1616. doi:10.1109/TITS.2012.2198214

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Parallel problem solving from nature PPSN VI* (Vol. 1917, pp. 849–858). doi:10.1007/3-540-45356-3_83

Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994). A niched Pareto genetic algorithm for multiobjective optimization. proceedings of the first IEEE conference on Evolutionary Computation. IEEE world congress on Computational Intelligence, Vol. 1, pp. 82–87. doi:10.1109/ICEC.1994.350037

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization* (Technical Report TR06, pp. 10). Erciyes University. doi:10.1109/ICEC.1994.350037

Karaboga, D., Gorkemli, B., Ozturk, C., & Karaboga, N. (2014). A comprehensive survey: Artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review, 42*(1), 21–57. doi:10.1007/s10462-012-9328-0

Pan, T.-S., Dao, T.-K., Nguyen, T.-T., Chu, S.-C., & Pan, J.-S. (2016, March 14–16). Bees and pollens with communication strategy for optimization. In T. N. Nguyen, B. Trawinski, H. Fujita, & T.-P. Hong (Eds.), *Intelligent information and database systems: 8th Asian conference, ACIIDS 2016, Da Nang, Vietnam, proceedings, part II* (pp. 651–660). Berlin: Springer. doi:10.1007/978-3-662-49390-8_63

Pan, J.-S., Dao, T.-K., Pan, T.-S., Nguyen, T.-T., Chu, S.-C., & Roddick, J. F. (2017). An improvement of flower pollination algorithm for node localization optimization in WSN. *Journal of Information Hiding and Multimedia Signal Processing, 8*(2), 500–509.

Qu, H., Xing, K., & Alexander, T. (2013). An improved genetic algorithm with co-evolutionary strategy for global path planning of multiple mobile robots. *Neurocomputing, 120*, 509–517. doi:10.1016/j.neucom.2013.04.020
Raja, P., & Pugazhenthi, S. (2009). Path planning for mobile robots in dynamic environments using particle swarm optimization. International conference on Advances in Recent Technologies in Communication and Computing, Kottayam, Kerala, India. doi:10.1109/ARTCom.2009.24

Raja, P., & Pugazhenthi, S. (2012). Optimal path planning of mobile robots: A review. International Journal of the Physical Sciences, 7(9), 1314–1320. doi:10.5897/IJPS11.1745

Samadi, M., & Othman, M. F. (2013). Global path planning for autonomous mobile robot using genetic algorithm. International conference on Signal-Image Technology & Internet-Based Systems, Kyoto, Japan, pp. 726–730. doi:10.1109/SITIS.2013.118

Samson, C. (1995). Control of chained systems application to path following and time-varying point-stabilization of mobile robots. IEEE Transactions on Automatic Control, 40(1), 64–77. doi:10.1109/9.362899

Sariff, N., & Buniyamin, N. (2006, June). An overview of autonomous mobile robot path planning algorithms. 4th student conference on Research and Development, Selangor, Malaysia, pp. 183–188. doi:10.1109/SCORED.2006.4339335

Siegwart, R., & Nourbakhsh, I. R. (2004). Introduction to autonomous mobile robots. Robotica, 23. doi:10.1109/ROBOT.2010.5509725

Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2006). Robot modeling and control. Control, 141(1), 419. doi:10.1109/TAC.2006.890316

Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y., Auger, A., & Tiwari, S. (2005, May). Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Natural Computing, 1–50.

Volos, C. K., Kyprianidis, I. M., & Stouboulos, I. N. (2012). A chaotic path planning generator for autonomous mobile robots. Robotics and Autonomous Systems, 60(4), 651–656. doi:10.1016/j.robot.2012.01.001

Whitley, D. (1994). A genetic algorithm tutorial. Statistics and Computing, 4(2), 65–85. doi:10.1007/BF00175354

Yang, X. S. (2012). Flower pollination algorithm for global optimization. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 7445 LNCS, pp. 240–249). doi:10.1007/978-3-642-32894-7_27

Yao, X., Liu, Y., & Lin, G. (1999). Evolutionary programming made faster. IEEE Transactions on Evolutionary Computation, 3(2), 82–102. doi:10.1109/4235.771163