Research Article

A Modified Pareto Ant Colony Optimization Approach to Solve Biobjective Weapon-Target Assignment Problem

You Li,1 Yingxin Kou,1 Zhanwu Li,1,2 An Xu,1,2 and Yizhe Chang1

1Aeronautics and Astronautics Engineering College, Air Force Engineering University, Xi’an, Shaanxi 710038, China
2College of Electronic Communication, Northwestern Polytechnical University, Xi’an 710072, China

CorrespondenceshouldbeaddressedtoYouLi;reason891020@126.com

Received 31 December 2016; Revised 24 February 2017; Accepted 28 February 2017; Published 16 March 2017

Copyright © 2017 You Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The weapon-target assignment (WTA) problem, known as an NP-complete problem, aims at seeking a proper assignment of weapons to targets. The biobjective WTA (BOWTA) optimization model which maximizes the expected damage of the enemy and minimizes the cost of missiles is designed in this paper. A modified Pareto ant colony optimization (MPACO) algorithm is used to solve the BOWTA problem. In order to avoid defects in traditional optimization algorithms and obtain a set of Pareto solutions efficiently, MPACO algorithm based on new designed operators is proposed, including a dynamic heuristic information calculation approach, an improved movement probability rule, a dynamic evaporation rate strategy, a global updating rule of pheromone, and a boundary symmetric mutation strategy. In order to simulate real air combat, the pilot operation factor is introduced into the BOWTA model. Finally, we apply the MPACO algorithm and other algorithms to the model and compare the data. Simulation results show that the proposed algorithm is successfully applied in the field of WTA which improves the performance of the traditional P-ACO algorithm effectively and produces better solutions than the two well-known multiobjective optimization algorithms NSGA-II and SPEA-II.

1. Introduction

Among air power, land power, and sea power, air power plays the most decisive role in the outcome of any war [1]. With the development of military air combat, the problem of WTA has received considerable attention [2–6]. Because the goal of air power is to maximize the military effectiveness, it becomes imperative to find a proper assignment of weapons to targets. From the perspective of the quantity of objective functions, the WTA problem can be classified into single-objective WTA problem and multiple-objective WTA problem. When taking the time factor into account, the WTA problem can be divided into dynamic WTA problem and static WTA problem. The current research status of various WTA is summarized in Table 1. In this paper, we mainly focus on the static biobjective WTA problem which aims at finding a proper static assignment of maximizing the expected damage of the enemy and minimizing the cost of missiles.

The WTA problem belongs to NP-complete problems (NP: nondeterministic polynomial), so the increment of weapons and targets will result in an explosive growth of the searching space. Although the enumeration method ensures a global solution, its computational requirement grows exponentially as the scale of WTA problem grows. Therefore, seeking better weapons for targets distribution scheme and improving the operational efficiency especially for the large-scale WTA problem are still an important issue in the command and control problem (C2).

The study of WTA problem can be traced back to the 1950s and 1960s when Manne [7] and Day [8] built the model of WTA problem. Hosein and Athans [9] classified the WTA problem into two classes: the single-objective WTA problem and the multiple-objective WTA problem. Genetic algorithm [10], ACO algorithm [11], auction algorithm [12], VLSN algorithm [13], Tabu search [14], and other hybrid algorithms [15–17] have been used to optimize single-objective WTA model by many scholars. In contrast to single-objective WTA, multiple-objective optimization can take different criterions into consideration and is more in line with the real combat
Table 1: Summary of variant metaheuristic algorithms and implementation of various WTA.

<table>
<thead>
<tr>
<th>Researchers</th>
<th>Year</th>
<th>Metaheuristic algorithm</th>
<th>Implementation (WTA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lee et al. [40]</td>
<td>2002</td>
<td>IS + ACO</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Lee et al. [41]</td>
<td>2002</td>
<td>GA</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Lee [13]</td>
<td>2010</td>
<td>VLSN</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Xin et al. [42]</td>
<td>2010</td>
<td>VP + Tabu</td>
<td>Dynamic single-objective</td>
</tr>
<tr>
<td>Li and Dong [16]</td>
<td>2010</td>
<td>DPSO + SA</td>
<td>Dynamic single-objective</td>
</tr>
<tr>
<td>Xin et al. [43]</td>
<td>2011</td>
<td>Rule-based heuristic</td>
<td>Dynamic multiobjective</td>
</tr>
<tr>
<td>Chen et al. [44]</td>
<td>2010</td>
<td>SA</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Bogdanowicz et al. [10]</td>
<td>2013</td>
<td>GA</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Fei et al. [12]</td>
<td>2012</td>
<td>Auction algorithm</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Liu et al. [18]</td>
<td>2013</td>
<td>MOEA/D</td>
<td>Static multiobjective</td>
</tr>
<tr>
<td>Zhang et al. [19]</td>
<td>2014</td>
<td>MOEA/D</td>
<td>Static multiobjective</td>
</tr>
<tr>
<td>Li et al. [20]</td>
<td>2015</td>
<td>NSGA-II, MOEA/D</td>
<td>Static multiobjective</td>
</tr>
<tr>
<td>Dirik et al. [46]</td>
<td>2015</td>
<td>MILP</td>
<td>Dynamic multiobjective</td>
</tr>
<tr>
<td>Liang and Kang [47]</td>
<td>2016</td>
<td>CSA</td>
<td>Static single-objective</td>
</tr>
<tr>
<td>Li et al. [48]</td>
<td>2016</td>
<td>MDE</td>
<td>Dynamic multiobjective</td>
</tr>
</tbody>
</table>

Decision-making. So, it has aroused wide attention from scholars. Liu et al. [18] proposed an improved multiobjective particle swarm optimization (MOPSO) algorithm to solve multiple-objective WTA problem and apply it to a simple example including 7 platforms and 10 targets. However, they did not apply the proposed algorithm to different scales WTA problems for contrastive study. Zhang et al. [19] designed a WTA mathematic model and proposed a decomposition-based evolutionary multiobjective optimization algorithm. But the algorithm has not been tested on large-scale WTA problem and it has a low convergence speed. Li et al. [20] adopted NSGA-II (domination-based) and MOEA/D (decomposition-based) to solve the multiple-objective WTA problem and carried out tests on three different BOWTA problems. They only applied the proposed adaptive mechanism to the WTA problems, but they did not verify the behavior of the proposed adaptive mechanism on standard problems.

Different from other algorithms, ACO algorithm is a class of reactive search optimization (RSO) methods adopting the principle of “learning while optimizing” [21, 22]. Since it was introduced in 1992, many variant ACO algorithms have been presented, including ant colony system (ACS) [23] and MAX-MIN ant system (MMAS) [24]. Meanwhile, ACO algorithms have been intensively investigated and successfully applied to deal with multiobjective problems such as travelling salesmen problem (TSP) [25], scheduling problem [26], vehicle routing problem [27], portfolio selection problem [28], network optimization problem [29], and some others problems [30–32]. The main applications of ACO algorithms are shown in Figure 1. In 2004, Doerner et al. proposed the P-ACO algorithm [28] which combined traditional ant system with ant colony optimization. Because P-ACO has the ability to balance local search and global search, it will avoid premature convergence during the solution construction phase. The characteristic is especially suitable for solving portfolio selection problem. Therefore, we choose P-ACO as the approach for solving the BOWTA problem. However, to the best of our knowledge, there has not been any research conducted about P-ACO for BOWTA problem.

Basic multiobjective optimization algorithms have such shortcomings as slow convergence to the Pareto front, no efficient termination criterion, and a lack of theoretical convergence proof. In this paper, we have proposed a modified P-ACO for tackling the above problems. The proposed algorithm is used to seek better compromise solutions between maximizing the expected damage and minimizing the cost. Based on the framework of the traditional P-ACO, the proposed algorithm is devised with several attractive features to enhance the optimization performance, including a dynamic heuristic information calculation approach, an improved movement probability rule, a dynamic evaporation rate strategy, a global updating rule of pheromone, and a boundary symmetric mutation strategy.

Dynamic Heuristic Information Calculation Approach. Because heuristic information seriously affects the efficiency of ACO algorithm, the problem-specific heuristic information function is one of the most important problems that should be solve. Hence, we design a dynamic heuristic information calculation approach for providing necessary guidance for an efficient search.

Improved Movement Probability Rule. In order to avoid P-ACO plunge into the local optimum which is caused by the stagnation phenomenon, we proposed a dynamic movement probability rule to enhance local exploitation and alleviate the premature convergence.
Dynamic Evaporation Rate Strategy. When the algorithm is used to solve large-scale WTA problem, a static pheromone evaporation rate will reduce global searching ability and convergence speed. Therefore, a dynamic evaporation rate strategy is proposed in this paper.

Global Updating Rule of Pheromone. The traditional static pheromone updating rule makes the algorithm easy to fall into local optimum. Hence, a global pheromone updating rule is introduced to solve the conflict between expanding search and finding optimal solution.

Boundary Symmetric Mutation Strategy. When solving large-scale problem, the distributions of targets are complex. In order to avoid the search being stuck in local optimum and diversify the solutions, a boundary symmetric mutation strategy is proposed to improve the mutation efficiency and the quality of solutions.

The rest of this paper is organized as follows. In Section 2, a mathematical model of the BOWTA problem and a problem-specific assumption description are presented. Section 3 provides algorithmic descriptions of P-ACO algorithm and presents the application of ACO algorithm for solving WTA problem. Detailed improvements of the proposed algorithm are introduced in Section 4. In Section 5, experimental evaluation is carried out and the analysis of the results is presented. Firstly, in order to validate effectiveness and show performance, the proposed algorithm is applied in a specific instance. Secondly, the proposed algorithm and others, like NSGA-II [33], SPEA-II [34], and traditional P-ACO [28], are tested on several different scales WTA problems. Section 6 provides some conclusion of the paper.

2. Problem Formulation

The WTA problem focuses on how to allocate attack units to targets and can be illustrated in Figure 2. WTA assumptions and mathematical model are introduced as follows in this section.

2.1. Assumption Description. In order to establish a reasonable WTA mathematical model, the following assumptions can be defined.

Assumption 1. We assume that the mathematical model is composed of $F$ fighters, $M$ missiles, and $T$ targets. The fighters
and targets are not necessarily equal in quantity (each fighter is equivalent to one platform which takes different kinds and quantities of missiles).

Assumption 2. Each fighter can use different missiles to attack one target (each missile can only attack one target).

Assumption 3. All allocated missiles cannot exceed the number of missiles resources in military air operation.

Assumption 4. We assume the probability of kill ($q_{ij}$) between the missile (ith unit of $M$) and the target (jth unit of $T$).

Assumption 5. If the target is in work area, then a missile can be assigned effectively. On the other hand, the missile is not.

2.2. Mathematical Model. The multiobjective optimization is to make the number of weapons to targets reach the maximum and reduce missiles consumption as much as possible. The definitions about the optimization model are shown as follows.

Definition 6. The multiobjective functions of WTA are to maximize the total effectiveness of attack and minimize the cost of missiles. The optimization mathematical model is described as

\[ f^1 = \max \sum_{j=1}^{n} \left[ 1 - \prod_{i=1}^{m} \left( 1 - \rho_k q_{ij} \right)^{x_{ij}} \right], \]

\[ f^2 = \min \sum_{i=1}^{m} \sum_{j=1}^{n} c_i x_{ij}, \]

\[ f = \min \left( \frac{1}{f^1}, f^2 \right). \]

Definition 7. $q_{ij} \in (0,1)$ represents the effectiveness of the missile unit. It is known as the probability of kill for each $T$ target ($j = 1, 2, \ldots, n$) and $M$ missile ($i = 1, 2, \ldots, m$).

Definition 8. The decision table $X = [x_{ij}]_{M \times T}$ can be shown as Table 2.

<table>
<thead>
<tr>
<th>$T_1$</th>
<th>$T_2$</th>
<th>$T_3$</th>
<th>$T_4$</th>
<th>\ldots</th>
<th>$T_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>$x_{11}$</td>
<td>$x_{12}$</td>
<td>$x_{13}$</td>
<td>$x_{14}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_2$</td>
<td>$x_{21}$</td>
<td>$x_{22}$</td>
<td>$x_{23}$</td>
<td>$x_{24}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_3$</td>
<td>$x_{31}$</td>
<td>$x_{32}$</td>
<td>$x_{33}$</td>
<td>$x_{34}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_4$</td>
<td>$x_{41}$</td>
<td>$x_{42}$</td>
<td>$x_{43}$</td>
<td>$x_{44}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_5$</td>
<td>$x_{51}$</td>
<td>$x_{52}$</td>
<td>$x_{53}$</td>
<td>$x_{54}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_6$</td>
<td>$x_{61}$</td>
<td>$x_{62}$</td>
<td>$x_{63}$</td>
<td>$x_{64}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_7$</td>
<td>$x_{71}$</td>
<td>$x_{72}$</td>
<td>$x_{73}$</td>
<td>$x_{74}$</td>
<td>\ldots</td>
</tr>
<tr>
<td>$M_8$</td>
<td>$x_{81}$</td>
<td>$x_{82}$</td>
<td>$x_{83}$</td>
<td>$x_{84}$</td>
<td>\ldots</td>
</tr>
</tbody>
</table>

... (Table continues)

Definition 9. $c_i$ denotes the cost of $i$ missile and is given a constant value in this paper.

Definition 10. In combat, different fighters are equipped with different numbers of missiles, so one assumes that the number of missiles carried by each fighter is not necessarily equal (the number of missiles carried by each plane is no more than 4).

Definition 11. According to the talent, training time, and operation stability of the pilot, all the above factors may affect the attack performance of the missile. One assumes that $\rho_k$ is pilot operation factor $\rho_k \in (0,1)$.

Definition 12. There are some constraints that the above function variables must satisfy.

Missile Assignment Constraint. For simplicity, we assume that at least one missile unit is assigned to one target. This constraint can be expressed as

\[ \sum_{i=1}^{m} x_{ij} \geq 1 \quad \forall j = 1, 2, \ldots, n. \]  

(2)

Quantity Assignment Constraint. We assume that one missile can only attack one target. The corresponding constraint is

\[ \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall i = 1, 2, \ldots, m. \]  

(3)

Parameters Constraint. In this paper, we will use some parameters like $x_{ij}$. The corresponding constraint is defined as follows:

\[ \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} \leq m. \]  

(4)

3. Ant Colony Optimization Algorithm

3.1. Pareto Ant Colony Optimization Algorithm (P-ACO). The ACO is a heuristic algorithm which has emerged in recent years and has been applied to solve hard combinatorial optimization problems [35]. Scientists have developed ACO by imitating the behaviors of real ant colonies in the process of searching for food. When ants in the colony find a short path for gathering food by frequently travelling between their nest and the food source, they deposit a chemical called pheromone trail on the path which can be followed by other ants to search for the food source [36].

Pareto ant colony optimization (P-ACO) algorithm is one of the multiple-objective ant colony optimization (MOACO) algorithms that was proposed by Doerner et al. [28] to be originally applied to optimize the multiobjective portfolio selection. The main objective of P-ACO algorithm is to find a set of solutions ordered by fronts under the concept of Pareto front. Although it is based on the classical ACS, the update process of global pheromone trail is implemented by using two different ants (the best and the second-best solutions) which are generated by each objective $g$ in current iteration.
During each iteration, \( \psi \) rulelike formula (5). Node heuristic information. In the route, an ant selects the next and is used to combine the pheromone information and \( \alpha \) and \( \beta \) belong to \( g = \{1, 2, \ldots, G\} \). When \( \xi \leq \xi_0 \), the available knowledge is exploited for selecting the best option through pheromone trail heuristic information. The formula is shown below:

\[
j = \arg \max_{j \in \Omega} \left( \sum_{g=1}^{G} w_g \cdot \tau_{ij}^g \right)^\alpha \cdot [\eta_{ij}]^\beta, \quad \text{if } \xi \leq \xi_0
\]

\[
\text{otherwise,}
\]

where \( g = \{1, 2, \ldots, G\} \) is a set of objectives, \( \eta_{ij} \) represents the heuristic information of edge \( e_{ij} \), and \( j \) is a probabilistic node. \( \alpha \) is the weight coefficient of pheromone information and \( \beta \) is the weight coefficient of heuristic information. \( \alpha \) and \( \beta \) belong to \( R \). \( \Omega \) is the current feasible neighborhood of the ant.

When \( \xi > \xi_0 \), \( p_{ij} \) is implemented for finding \( j \) and the formula is expressed as

\[
p_{ij} = \frac{\left( \sum_{g=1}^{G} w_g \cdot \tau_{ij}^g \right)^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{a \in \Omega} \left( \sum_{g=1}^{G} w_g \cdot \tau_{ia}^g \right)^\alpha \cdot [\eta_{ia}]^\beta}, \quad \text{if } j \in \Omega
\]

\[
\text{otherwise.}
\]

When an ant has travelled an edge \( e_{ij} \) in the algorithm, local pheromone update (online pheromone trail update) is implemented for each objective \( g \) by the rule

\[
\tau_{ij}^g = (1 - \rho) \cdot \tau_{ij}^g + \rho \cdot \tau_0,
\]

where \( \rho \) is the pheromone evaporation rate (\( \rho \in (0, 1) \)) and \( \tau_0 \) is the value of initial pheromone trail. In this paper, we use the same method as the literature [37] to set the value of \( \tau_0 \).

When each ant has constructed its solution, global pheromone update (offline pheromone trail update) is implemented for each objective \( g \). The update formula is shown below:

\[
\tau_{ij}^g = (1 - \rho) \cdot \tau_{ij}^g + \rho \cdot \Delta \tau_{ij}^g,
\]

where \( \Delta \tau_{ij}^g \) follows the problem-specific values:

\[
\Delta \tau_{ij}^g = \begin{cases} 
10 & \text{if edge } e_{ij} \text{ is best solution}, \\
5 & \text{if edge } e_{ij} \text{ is second-best solution}, \\
0 & \text{otherwise.}
\end{cases}
\]

In the process, the nondominated solutions are stored in an external set \( S \).

The pseudocode of the P-ACO algorithm can be demonstrated as shown in Algorithm 1.

**Algorithm 1**

1. Initialization: generate the number of ants \( A \), initialize pheromone vectors with \( \tau_0 \), set the values of \( \alpha \) and \( \beta \).
2. Repeat until Termination conditions has not been met {
   1. for Ant = 1 to \( A \) {
      1. determine the lifespan \( \Theta \) of the ant randomly of the interval \([1, 2, \ldots, N]\); create \( x = (0, \ldots, 0)/\) built empty assignment solution ‘/’;
      1. determine the objective weight \( w_g \) for each objective \( g \);
      1. indicates the number of targets to be selected \( \xi = \Theta \);
   }
   2. while \( \xi > 0 \) and \( \exists i \) with \( \eta(x) > 0 \) {
      1. select an assignment \( ij \) using formula (5) and add it to \( x \);
      1. update local pheromone information;
      1. decrement \( \xi \);
      1. check feasibility \( x \);
   }
   3. if \( x \) is feasible {
      1. check efficiency \( x \);
   }
   3. if \( x \) is efficient {
      1. store \( x \) and remove dominated ones;
   }
}
3. for each objective \( g \) {
   1. determine best and second-best solution \( \Delta \tau_{ij}^g \) for each objective \( g \);
   1. update global pheromone information using best and second-best solution;
   1. using formula (8);
}

3.2. Application of ACO Algorithm for Solving WTA Problem. At the beginning, each missile has a list of possible target assignments (Figure 3). The number of assignments
4. Improvements of the MPACO Algorithm


The heuristic information $\eta_{ij}$ denotes the probability of the ant moving from $i$ to $j$ and is an optional weighting function calculated by heuristic approach. Because $\eta_{ij}$ must be calculated for all ants in each iteration, it may seriously affect the efficiency of the ACO algorithm. How to design the problem-specific heuristic information function is one of the important problems that should be solved. There are two main calculation approaches of heuristic information at present: (1) $\eta_{ij}$ is calculated at first and remains unchanged in each iteration until the algorithm finishes [23] and (2) compared with the first approach $\eta_{ij}$ is given a dynamic value according to the current iteration [38]. In the first approach, $\eta_{ij}$ is more efficient but cannot completely indicate the probability. The second approach can satisfy precise estimation of the probability but will relatively reduce the efficiency of computation. In order to solve the WTA problem, an improved calculating approach for heuristic information which considers the efficiency of calculation and the quality of solution is proposed. The values of biobjective functions are considered in the approach. The formulas can be shown as follows:

$$\eta_{ij}^1 = \sum_{j=1}^{n} \left[ 1 - \prod_{i=1}^{m} (1 - \rho \xi_{ij})^{x_{ij}} \right],$$

$$\eta_{ij}^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} \xi_{ij} x_{ij},$$

$$\eta_{ij} = \theta \cdot \frac{\eta_{ij}^1}{\eta_{ij}^2},$$

where $\eta_{ij}^1$ and $\eta_{ij}^2$ dividedly denote the first objective function and the second one. $\eta_{ij}$ is the total probability heuristic information which is represented as $\eta_{ij}^1$ multiplied by $\eta_{ij}^2$. $\theta$ is a positive constant.
4.2. Improved Movement Probability Rule. In practice, ACO algorithm is easy to plunge into the local optimum which is caused by stagnation phenomenon. So, the dynamic movement probability rule based on essential selection \((q \leq q_0)\) and random selection \((q > q_0)\) is proposed to improve the strategy of global search in the process. The phenomenon \(\tau_{ij}\) of the paths which have been visited is enhanced. The selected probability of the paths will increase in the next iteration. Because few ants visit the path, some better solutions cannot be stored at the beginning iterations. If more ants can visit the path less traveled in the process, it will be beneficial to the solutions space and effectively perfect the P-ACO algorithm. The probability \(p_{ij}\) can be computed as

\[
p_{ij} = \begin{cases} \frac{\sum_{g=1}^{G} p_g \cdot \eta^\alpha_{ij} \cdot \tau_{ij}^\beta \cdot z_{ij}}{\sum_{s \in \Omega} \sum_{g=1}^{G} p_g \cdot \eta^\alpha_{su} \cdot \tau_{su}^\beta \cdot z_{su}}, & \text{if } j \in \Omega \\ 0, & \text{otherwise} \end{cases},
\]

where \(A\) denotes the number of ants according to the specific problem, \(I_c\) represents the current iteration, \(\delta\) is a constant, \(A_{sf}\) is the total number of ants that have visited the same node so far, and \(\eta_{max}\) is the maximum value of \(\eta_{ij}\). The value of \(z_{ij}\) takes into account \(A_{sf}\) and \(\eta\) simultaneously. If the current iteration tends to the sub-Pareto solution, the pheromone \(\tau_{ij}\) of sub-Pareto solution will be enhanced. But the number of \(A_{sf}\) increased and the value of \(z_{ij}\) decreased. This strategy can restrain the excessive growth of pheromone and escape from the local optimum.

4.3. Dynamic Evaporation Rate Strategy. In classical P-ACO algorithm, the pheromone evaporation rate \(\rho\) is a fixed parameter \((\rho \in (0, 1))\). The value of \(\rho\) is one of the main factors which have an influence on global search ability and convergence speed. When P-ACO algorithm is applied to solve large-scale problem, the value of pheromone \(\tau\) is close to 0 and global search ability declines. If the value of pheromone becomes very large, the selection probability of the passed nodes increases and global search ability is also directly affected. How to design the value of pheromone is crucial to the release and evaporation of pheromone. In order to improve the global search ability, the common strategy is to set a large value of \(\rho\) at the beginning of the algorithm. But the evaporation rate will also decay in the process, so that the P-ACO algorithm can quickly fall into local optimum. Two advantages are brought about by the dynamic evaporation rate strategy: (1) better global search ability and (2) faster convergence speed. The mathematical expression of decay model is shown below:

\[
\rho_{ct} = \frac{I_{max} \cdot I_c \cdot \left[ \sum_{g=1}^{G} w_g \left( \tau_{max}^g - \tau_{min}^g \right) \right]}{I_{max} - 1} + \frac{\sum_{g=1}^{G} w_g \left( I_{max}^g \cdot \tau_{min}^g - \tau_{max}^g \right)}{I_{max} - 1},
\]

where \(I_{max}\) is the maximum iteration and is set at the beginning and \(I_c\) is the current iteration. \(\tau_{max}^g\) denotes the maximum of pheromone and \(\tau_{min}^g\) denotes the minimum of pheromone.

4.4. Global Updating Rule of Pheromone. A fixed pheromone update rule is mainly used in the classical P-ACO algorithm. The traditional strategy may flung the algorithm into local optimum and has three disadvantages: (1) the distribution features of solution are ignored; (2) there is the trend of stagnation; (3) the convergence rate is slow. In addition to the pheromone local updating rule, the proposed MPACO adopts an adaptive adjustment global updating rule which can effectively deal with the conflict between expanding search and finding Pareto solution. In this rule, the pheromone information of each objective is updated by the best and second-best solution. The rule also provides some instructive guidance to improve the quality of the solutions. The pheromone \(\Delta \tau_{ij}^\alpha\) can be computed as

\[
\Delta \tau_{ij}^\alpha = \begin{cases} \frac{1}{f_b} & \text{if } x_{best,ij}^g = 1 \\ \frac{1}{f_{second-best}^g} & \text{if } x_{second-best,ij}^g = 1 \\ 0 & \text{otherwise} \end{cases},
\]

where \(x_{ij}\) is an assignment of a population best (or second-best) ant according to objective \(g\). The strategy maintains the Pareto exploitation and balances the random search and the function about pheromone evaporation or augmentation. In order to reduce the otherness of pheromone on the local Pareto solution and expand the search scope, the positive feedback of P-ACO algorithm is restrained by adding a little amount of negative feedback pheromone in the process. If the algorithm falls into a local Pareto point during a period of iterations, then the adaptive adjustment strategy will be adopted to decrease the amount of pheromone for escaping from local Pareto front. The two expressions are similar in format, but the objective function is different.

4.5. Boundary Symmetric Mutation Strategy. Based on the mathematical statistics theory, the distribution of the samples will be close to the normal distribution when the number of samples tends to infinity. In solving WTA, targets coordinates according to the values are sorted. When a larger scale problem is solved, the distributions of targets are complex and a majority of targets are on the central region. The path of ants is boundary-center-boundary in the initial stage, so that all ants will follow the original track mode.

In order to solve the above problem, the mutation strategy is used for boundary mutation which can improve the mutation efficiency and the quality of solutions. Based on the quartile idea in probability theory, the mutation strategy is applied to mutate 1/4 solutions in both sides of the path. Due to the symmetry of the boundary, crossover and mutation only take place in the selected boundary which did not happen among others.
According to the above improvements, a modified Pareto ant colony optimization (MPACO) algorithm based on dynamic calculation approach of heuristic information, improved movement probability rules, improvement strategy of evaporation factor, improved global updating rules of pheromone, and boundary symmetric mutation strategy is proposed in this paper.

4.6. The Structure of MPACO Algorithm. According to the idea of the proposed MPACO algorithm, the structure of MPACO algorithm in solving BOWTA problem can be shown as follows.

Begin

Step 0 (preprocessing). These nodes of decision table are divided into center nodes and boundary nodes according to $q_{ij}$ which represents the effectiveness of the missile. The classification result is saved and sorted in order to prepare for the subsequent run of the MPACO algorithm.

Step 1 (initialization of parameters for the proposed MPACO). These parameters include the colony of ants $A$, the maximum iteration $I_{\text{max}}$, the pheromone information factor $\alpha$, heuristic information factor $\beta$, and initial uniform probability $\zeta_0$.

While (Termination conditions has not been met) do

For $I = 1, 2, \ldots, I_{\text{max}}$.

For $A = 1, 2, \ldots, A_{\text{max}}$.

Step 2. The $A$ ants randomly visit targets of each missile by constraints (2), (3), and (4), and these assignments are added into the decision list of the ant $X_a = \{x_{ij}^1, x_{ij}^2, \ldots, x_{ij}^m\}$.

Step 3. Due to the dynamic calculation approach of heuristic information, formulas (10) are used to calculate coupling $\eta_{ij}$ before Step 4.

Step 4. According to (11) of improved movement probability rules, the probability that the ant selects the next node is calculated (Section 4.2). Then, this node is added into the decision list, and the pheromone is locally updated.

Step 5. After the ants have completed a selection, the fitness function $f$ will be calculated. Then, the decision list is modified. Repeat Step 4 until the ant completed the tour (namely, all missiles are allocated). The current best and second-best solutions are saved and the global best solution is updated for the current iteration.

End for

Step 6. Based on (8) and (13), the pheromone is globally updated.


Step 8. Set the iterative counter $I = I + 1$ for the iteration control. If $I < I_{\text{max}}$, return to Step 2.

End while

Step 9. The proposed MPACO algorithm is terminated; output Pareto solutions.

5. Experiment Results and Analysis

5.1. Parameter Settings. In order to select the initial value of pheromone $\tau_0$, we use the formula in the literature [39], which can be shown as follows:

$$\tau_0 = \frac{G}{\sum_{g=1}^{G} f^g(X_g)},$$

where $X_g$ is a solution of the $g$th objective computed by a greedy algorithm, $G$ is the number of objectives in problem-specific function, and $f^g(X_g)$ is the fitness function of the solution $X_g$ according to the $g$th objective.

Based on the numbers of targets and missiles in a different BOWTA scale, the maximum number of iterations $I_{\text{max}}$ is set. Some parameters of MPACO algorithm are shown in Table 3.

We can see from Table 4 that the numerical experiments have been performed on a PC with Intel(R) Core(TM) i5-4460T CPU @ 1.90 GHz, with 8.00 GB of RAM, running Windows 7 64 bits as the operating system. The codes of all algorithms and optimization models have been written in C++ under Microsoft Visual C++ 6.0.

5.2. Numerical Experiments and Analysis

5.2.1. Computation Results on Specific Example. A specific example is designed to verify performances of the algorithm. Assume that there are 4 fighters that carry different numbers of missiles (12 missile units in total) and 10 targets in the simulation. Missiles damage probability $p_{ij}$ ($i = 1, \ldots, 12; j = 1, \ldots, 10$), pilot operation factor $\rho_k$ ($k = 1, \ldots, 4$), and cost of missiles $c_i$ are shown in Tables 5–7.

According to the literature [1], we present the missiles cost, as shown in Table 7.
Table 5: The table of missiles damage probability.

<table>
<thead>
<tr>
<th>Missile unit</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.82</td>
<td>0.85</td>
<td>0.78</td>
<td>0.75</td>
<td>0.52</td>
<td>0.88</td>
<td>0.44</td>
<td>0.76</td>
<td>0.72</td>
<td>0.56</td>
</tr>
<tr>
<td>2</td>
<td>0.56</td>
<td>0.72</td>
<td>0.88</td>
<td>0.46</td>
<td>0.64</td>
<td>0.47</td>
<td>0.68</td>
<td>0.45</td>
<td>0.48</td>
<td>0.75</td>
</tr>
<tr>
<td>3</td>
<td>0.45</td>
<td>0.62</td>
<td>0.54</td>
<td>0.73</td>
<td>0.84</td>
<td>0.76</td>
<td>0.78</td>
<td>0.42</td>
<td>0.53</td>
<td>0.65</td>
</tr>
<tr>
<td>4</td>
<td>0.56</td>
<td>0.42</td>
<td>0.76</td>
<td>0.84</td>
<td>0.73</td>
<td>0.83</td>
<td>0.86</td>
<td>0.62</td>
<td>0.78</td>
<td>0.82</td>
</tr>
<tr>
<td>5</td>
<td>0.45</td>
<td>0.58</td>
<td>0.81</td>
<td>0.44</td>
<td>0.63</td>
<td>0.59</td>
<td>0.78</td>
<td>0.77</td>
<td>0.65</td>
<td>0.70</td>
</tr>
<tr>
<td>6</td>
<td>0.46</td>
<td>0.61</td>
<td>0.55</td>
<td>0.68</td>
<td>0.75</td>
<td>0.83</td>
<td>0.73</td>
<td>0.66</td>
<td>0.82</td>
<td>0.48</td>
</tr>
<tr>
<td>7</td>
<td>0.66</td>
<td>0.71</td>
<td>0.65</td>
<td>0.44</td>
<td>0.86</td>
<td>0.79</td>
<td>0.59</td>
<td>0.85</td>
<td>0.53</td>
<td>0.56</td>
</tr>
<tr>
<td>8</td>
<td>0.56</td>
<td>0.42</td>
<td>0.76</td>
<td>0.84</td>
<td>0.73</td>
<td>0.72</td>
<td>0.44</td>
<td>0.75</td>
<td>0.48</td>
<td>0.47</td>
</tr>
<tr>
<td>9</td>
<td>0.88</td>
<td>0.78</td>
<td>0.44</td>
<td>0.67</td>
<td>0.56</td>
<td>0.86</td>
<td>0.58</td>
<td>0.65</td>
<td>0.73</td>
<td>0.42</td>
</tr>
<tr>
<td>10</td>
<td>0.56</td>
<td>0.88</td>
<td>0.68</td>
<td>0.45</td>
<td>0.75</td>
<td>0.73</td>
<td>0.61</td>
<td>0.76</td>
<td>0.84</td>
<td>0.78</td>
</tr>
<tr>
<td>11</td>
<td>0.84</td>
<td>0.54</td>
<td>0.44</td>
<td>0.42</td>
<td>0.65</td>
<td>0.56</td>
<td>0.71</td>
<td>0.55</td>
<td>0.45</td>
<td>0.88</td>
</tr>
<tr>
<td>12</td>
<td>0.83</td>
<td>0.76</td>
<td>0.84</td>
<td>0.62</td>
<td>0.82</td>
<td>0.75</td>
<td>0.42</td>
<td>0.68</td>
<td>0.57</td>
<td>0.54</td>
</tr>
</tbody>
</table>

Table 6: Pilot operation factor table.

<table>
<thead>
<tr>
<th>Fighter</th>
<th>Missile unit</th>
<th>$\rho_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_1$</td>
<td>$M_1$</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>$M_2$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$M_3$</td>
<td></td>
</tr>
<tr>
<td>$F_2$</td>
<td>$M_4$</td>
<td>0.95</td>
</tr>
<tr>
<td></td>
<td>$M_5$</td>
<td></td>
</tr>
<tr>
<td>$F_3$</td>
<td>$M_6$</td>
<td>0.98</td>
</tr>
<tr>
<td></td>
<td>$M_7$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$M_8$</td>
<td></td>
</tr>
<tr>
<td>$F_4$</td>
<td>$M_{10}$</td>
<td>0.93</td>
</tr>
<tr>
<td></td>
<td>$M_{11}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$M_{12}$</td>
<td></td>
</tr>
</tbody>
</table>

Firstly, an enumeration method is employed to the specific example instance and a set of evenly distributed true Pareto solutions (PSs) are obtained. Secondly, the proposed algorithm is used to find PSs.

Figure 4 presents the distribution of the true PSs solved by enumeration method and final PSs obtained by the MPACO in the objective space. We can see from the figure that the MPACO can find close and even PSs in the objective space. It is evident that MPACO can guarantee the quality of solutions. So, the MPACO for optimizing the BOWTA is well verified to be feasible.

We can get some results from Figure 5 and Table 8 as follows: when funds of national defense are enough, we can choose Scheme 1 which cost the most money and obtain the greatest expected damage of the enemy. When funds are insufficient, we should choose one scheme among Schemes 34, 35, and 36 on the basis of threat coefficient of each target. Considering that all targets must be allocated and the cost of missiles should be minimized, we can only choose Scheme 4.

5.2.2. Comparison Study. In order to verify the applicability and feasibility of the proposed algorithm, we test MPACO, P-ACO, NSGA-II, and SPEA-II on Scale 1, Scale 2, and Scale 3 instances. Three examples represent, respectively, small-, medium-, and large-scale BOWTA instances under the same condition. The quantitative indexes, such as Pareto figure and time performance, are chosen to evaluate the performance on C++. The value of three instances can be seen from Table 9.

In what follows, Scale 1 instance is firstly taken as an example to illustrate the performance of MPACO, P-ACO, NSGA-II, and SPEA-II. Then, Scale 2 and Scale 3 instances are introduced to further demonstrate the performances of both algorithms. The statistical results are shown in Figures 5–7.

Figures 5–7 present the distribution of PSs obtained in 30 runs with the best performance of each algorithm for each test instance in objective space. When especially tested on Scale 2 and Scale 3, the MPACO can obtain the best solutions among the four algorithms. Due to increments of dynamic calculation approach of heuristic information and improved movement probability rules, the calculation accuracy of MPACO algorithm is improved. Boundary symmetric mutation strategy is also good for the algorithm to solve large-scale BOWTA problem.

The primary aim of solving BOWTA problems is to lay the foundation for the DWTA problems whose goal is to provide a set of near-optimal or acceptable real-time decisions in air combat. So, in practice, the time performance of algorithms is also an important index. In the end, we test four algorithms on three instances in 30 runs and record the iteration time of each algorithm. The statistical results of time performances are shown in Figures 8–10.

Although MPACO does not have time advantage in solving Scale 1 and Scale 2 compared with other algorithms, it runs as fast as SPEA-II with the same number of function evaluations on Scale 3. Boundary symmetric mutation strategy improves the mutation efficiency and the quality of solutions and new evaporation factor is good for solving large-scale problems, but dynamic heuristic information and
Table 7: The cost of each missile table.

<table>
<thead>
<tr>
<th>Missile unit</th>
<th>$c_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>0.62</td>
</tr>
<tr>
<td>$M_2$</td>
<td>0.63</td>
</tr>
<tr>
<td>$M_3$</td>
<td>0.69</td>
</tr>
<tr>
<td>$M_4$</td>
<td>0.80</td>
</tr>
<tr>
<td>$M_5$</td>
<td>0.72</td>
</tr>
<tr>
<td>$M_6$</td>
<td>0.90</td>
</tr>
<tr>
<td>$M_7$</td>
<td>0.96</td>
</tr>
<tr>
<td>$M_8$</td>
<td>0.68</td>
</tr>
<tr>
<td>$M_9$</td>
<td>0.72</td>
</tr>
<tr>
<td>$M_{10}$</td>
<td>0.65</td>
</tr>
<tr>
<td>$M_{11}$</td>
<td>0.66</td>
</tr>
<tr>
<td>$M_{12}$</td>
<td>0.65</td>
</tr>
</tbody>
</table>

In this section, we do some work as follows. Firstly, we test the MPACO on a specific example. We verify the feasibility after comparing two kinds of PSs and show the corresponding distribution results. Secondly, to give a comparison study among the four algorithms, three different scale instances are introduced to evaluate each algorithm. Plots of the best PSs in 30 runs of each algorithm are presented to have an intuitive comparison. Lastly, we show the time performance of the four algorithms on each instance. To summarize, MPACO has been proven to be an effective technique for BOWTA optimization problem and obviously is the best among the four algorithms.

6. Conclusion

We apply ACO to WTA problem and propose the MPACO to solve BOWTA in this paper. The main contributions of
The values of Scale 1, Scale 2, and Scale 3 instances.

<table>
<thead>
<tr>
<th></th>
<th>Scale 1</th>
<th>Scale 2</th>
<th>Scale 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target</td>
<td>10</td>
<td>48</td>
<td>98</td>
</tr>
<tr>
<td>Missile</td>
<td>10</td>
<td>53</td>
<td>100</td>
</tr>
</tbody>
</table>

the thesis are summarized as follows. On the one hand, the expected damage of enemy and the cost of missiles are taken into account from a practical viewpoint; on the other hand, pilot operation factor and missiles constraints are introduced to make the model in line with the real air combat. In this aspect, five strategies are proposed and embedded into traditional P-ACO algorithm to improve performances of MPACO algorithm. The experiments have shown that MPACO can find better Pareto solutions than the other three algorithms for all instances. More importantly, MPACO is more suitable for solving large-scale problems from the time performance viewpoint.

However, we mainly have studied the static WTA problem; few researches focused on dynamic problems which are
Figure 6: Plots of PSs best performance in 30 runs of four algorithms on Scale 2.

Figure 7: Plots of PSs best performance in 30 runs of four algorithms on Scale 3.

Figure 8: Boxplot of time performance on Scale 1 (in secs).

Figure 9: Boxplot of time performance on Scale 2 (in secs).

Figure 10: Boxplot of time performance on Scale 3 (in secs).

more instructive to real combat. In recent years, more and more researches began to pay attention to DWTA problems. A further study on this topic is one of our future tasks.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References


Submit your manuscripts at https://www.hindawi.com