Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(4): 7088–7112.
DOI: 10.3934/mbe.2023306
Received: 30 November 2022
Revised: 15 January 2023
Accepted: 28 January 2023
Published: 09 February 2023

Research article

A multi-objective optimization framework with rule-based initialization for multi-stage missile target allocation

Shiqi Zou[1,*], Xiaoping Shi[1] and Shenmin Song[2]

[1] Control and Simulation Center, Harbin Institute of Technology, Harbin 150080, China
[2] Center for Control Theory and Guidance Technology, Harbin Institute of Technology, Harbin 150080, China

* **Correspondence:** Email: shiqi_zou@163.com.

Abstract: This paper investigates a novel multi-objective optimization framework for the multi-stage missile target allocation (M-MTA) problem, which also widely exists in other real-world complex systems. Specifically, a constrained model of M-MTA is built with the trade-off between minimizing the survivability of targets and minimizing the cost consumption of missiles. Moreover, a multi-objective optimization algorithm (NSGA-MTA) is proposed for M-MTA, where the hybrid encoding mechanism establishes the expression of the model and algorithm. Furthermore, rule-based initialization is developed to enhance the quality and searchability of feasible solutions. An efficient non-dominated sorting method is introduced into the framework as an effective search strategy. Besides, the genetic operators with the greedy mechanism and random repair strategy are involved in handling the constraints with maintaining diversity. The results of numerical experiments demonstrate that NSGA-MTA performs better in diversity and convergence than the excellent current algorithms in metrics and Pareto front obtained in 15 scenarios. Taguchi method is also adopted to verify the contribution of proposed strategies, and the results show that these strategies are practical and promotive to performance improvement.

Keywords: weapon-target assignment; multi-objective optimization; rule-based initialization; non-dominated sorting; random repair strategy

1. Introduction

The missile target allocation (MTA) problem is an essential expression of the weapon target assignment (WTA) problem in the field of military operations research [1]. In essence, the MTA problem belongs to the resource allocation problem, which also widely exists in other real-world fields [2–4]. The issue addressed in this paper is that limited missiles need to intercept and strike targets, satisfying the expectations of minimizing the survivability of targets and minimizing the cost consumption of missiles. Recent studies of scholars [5,6] have reported that the WTA problem is NP-complete, which belongs to the typical constrained combinational optimization problem. Specifically, the WTA problem has been divided into two versions in more detail, static WTA (SWTA) problems and dynamic WTA (DWTA) problems [7]. Furthermore, scholars have focused on the research of DWTA and derived multi-stage WTA (MWTA) problems [8], and multi-objective WTA (MOWTA) problems [9] in order to describe the battlefield situation. In this paper, we further investigate a multi-stage MTA (M-MTA) problem of a multi-objective optimization version by combining MWTA and MOWTA problems.

The cooperation of multiple missiles can ensure the accomplishment of striking targets, where these missiles must be strictly independent of each other during this progress. What is more, more complex constraints should be taken into account in real-world problems. Meanwhile, using missiles during the allocation is another primary concern in practical applications. Therefore, MTA belongs to the multi-objective optimization problem (MOP) with constraints in real-world applications, whose general mathematical formulation is stated as follows.

$$
\begin{aligned}
&\min\ \boldsymbol{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})) \\
&s.t. \\
&g(\mathbf{x}) \leq 0 \\
&h(\mathbf{x}) = 0 \\
&\mathbf{x} \in \Omega
\end{aligned} \tag{1.1}
$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ denotes the candidate solutions, which belongs to the decision space Ω. $\boldsymbol{F}(\mathbf{x})$ is the conflicting objective function, where M means the number of the objective function. $g_i(\mathbf{x})$ and $h_e(\mathbf{x})$ express inequality constraints and equality constraints, respectively.

Thus, obtaining the optimal allocation of limited missiles to targets cooperatively is challenging. In the past decades, several attempts have been made to develop various efficient algorithms to optimize the WTA problem, which contains many proposed exact methods [1, 10], and heuristic algorithms [11, 12]. In terms of exact methods, many technologies have sprung up, such as the branch-and-bound, the maximum marginal return [13], dynamic programming, and so on. However, rapid changes in the computational scale have significantly affected applying heuristic algorithms to solve WTA problems. The swarm optimization algorithm [14, 15] and genetic algorithm with the greedy mechanism [16] are the breakthrough in solving the SWTA problem. Improved artificial bee colony algorithm [17] has also made remarkable achievements in DWTA problems. In terms of the research of MOWTA problems, scholars [18,19] have successfully applied and improved based on multi-objective optimization evolutionary algorithms (MOEAs) [20,21] to solve the problem.

Significantly, a fast and elitist multiobjective genetic algorithm (NSGA-II) [20] on the basis of NSGA [22] is an excellent multi-objective optimization algorithm, which is mainly composed of fast non-dominated sorting (fast-NS), crowding distance (CD), and the elite strategy. The operations of NSGA-II are driven by genetic mechanisms. The sorting method is designed to divide several subspaces containing feasible solutions. The advantage of fast-NS is that each solution γ is compared with all solutions in order by using two defined parameters: 1) n_γ: domination count, 2) S_γ: a set of α dominates other solutions, and filed into the corresponding dominant front finally. The CD is proposed to replace the traditional shared function for density estimation, and the elite strategy is combined with crowding distance to ensure the distribution of solutions. In the same dominant front, the CD of each solution γ is defined as the Manhattan distance of solution $\gamma+1$ and solution $\gamma-1$ in the objective space.

In order to make NSGA-II more generalized for complex combinatorial optimization problems, many scholars [23–25] have improved the algorithm from the perspective of feasibility solution generation and density estimation approach. As a basic framework, NSGA-II has been successfully involved in many complex optimization problems in science and engineering.

In this paper, a multi-objective mathematical model of M-MTA is formulated. Then, we propose a novel NSGA-II framework (NSGA-MTA) for solving this problem. The framework mainly constitutes the rule-based initialization, the hybrid encoding mechanism, an efficient non-dominated sorting [26] (efficient-NS), and genetic operators with the greedy mechanism and random repair strategy. Furthermore, the Taguchi method is adopted to verify the effectiveness of the proposed strategy. There are some main innovations and contributions in this article as follows:

1) A multi-objective mathematical model of M-MTA is formulated to describe the survivability of targets and the cost consumption of missiles simultaneously. Meanwhile, the constraints that can reflect the characteristics of multi-stage and combat are proposed.
2) NSGA-MTA derived from NSGA-II is proposed for solving the M-MTA, and it is further demonstrated that the performance is superior to that of other excellent algorithms.
    a) A hybrid encoding mechanism is built to express the allocation scheme, which combines integer coding with binary coding to involve the algorithm.
    b) A rule-based initialization is designed to generate feasible solutions in advance according to prior knowledge.
    b) Genetic operators with greedy mechanisms and random repair strategies are constructed to maintain the trade-off between exploration and exploitation and further handle the constraints.

The remainder of this paper is organized as follows. Section 2 defines the detailed mathematical model of the M-MTA. The proposed NSGA-MTA is presented in detail in Section 3. Preliminary numerical experiments are implemented in Section 4. Section 5 gives the experimental results and the discussion based on obtained results. Finally, Section 6 provides the conclusion of this paper.

2. Problem description and formulation

In this paper, the multi-stage MTA (M-MTA) occurs in this scenario [27] wherein R_n different defensive missile types are allocated to strike T_n offensive targets in S_n stages cooperatively, depicted in Figure 1. Meanwhile, we assume there will be cost consumption after the allocation, and each target has its threat value v_t. Without a doubt, each missile has a corresponding capability to destroy targets at each stage, wherein the capability is defined as the kill probability. In this paper, the kill probability is determined as a probability value $p_{rt}(s)(r \in [1, R_n], t \in [1, T_n], s \in [1, S_n])$, which is stochastic and independent. Last but not least, each target must match at least one missile for a successful interception.

The allocation scheme of M-MTA intuitively is expressed as mapping by $\boldsymbol{X} = [[x_{rt}(s)]_{R_n \times T_n}]_{1 \times S_n}$. Furthermore, $x_{rt}(s)$ is the decision variable, which means that r th missile is allocated to t target at s stage. This variable is defined as a binary variable. That is, if the allocation is determined, it will be 1. Otherwise, it is 0. Specifically, the notations adopted in this formulation will briefly be described in Table 1. Therefore, the objective function F_1, F_2 of M-MTA is established as follows.

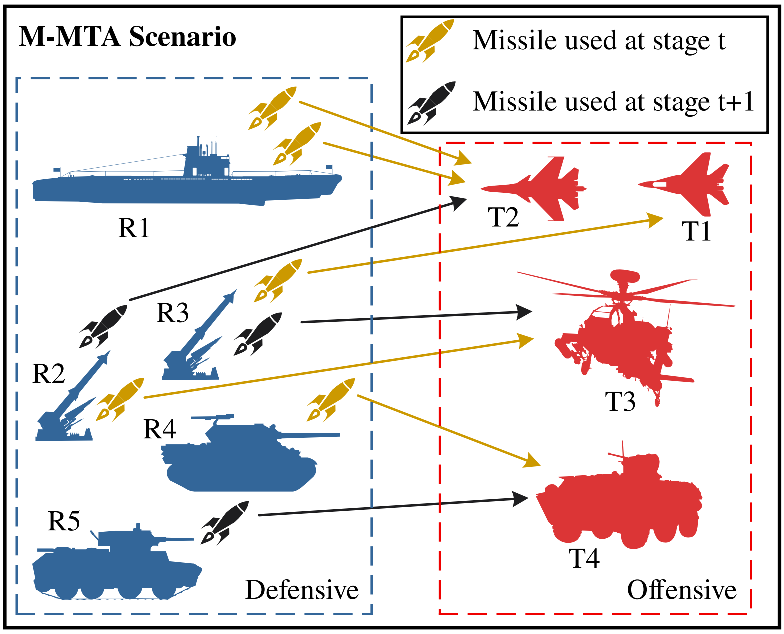


Figure 1. An description of M-MTA scenario.

Table 1. Notations and definition.

Notations	Definition
r	Index for missile types ($r \in [1, R_n]$);
t	Index for target types ($t \in [1, T_n]$);
s, h	Index for stages ($s, h \in [1, S_n]$);
R_n	The overall number of missile types;
$R_n(h)$	The remaining number of missile types at the stage h ($h \in [1, S_n]$);
T_n	The overall number of targets;
$T_n(h)$	The remaining number of targets at the stage h ($h \in [1, S_n]$);
S_n	The overall number of stages;
$\boldsymbol{V} = [v_t]_{1\times T_n}$	The threat value matrix for targets;
$\boldsymbol{P} = [[p_{rt}(s)]_{R_n\times T_n}]_{1\times S_n}$	The kill probability for missiles to targets at S_n stages;
$\boldsymbol{C} = [[c_{rt}(s)]_{R_n\times T_n}]_{1\times S_n}$	The cost of missiles to targets at S_n stages;
$\boldsymbol{Z} = [[z_{rt}(s)]_{R_n\times T_n}]_{1\times S_n}$	The feasibility allocation at S_n stages;
$\boldsymbol{Num} = [Num_r]_{1\times R_n}$	The maximum of missiles that can be allocated in each type;
$\boldsymbol{X} = [[x_{rt}(s)]_{R_n\times T_n}]_{1\times S_n}$	The decision matrix at S_n stages;
F_1	The total survival expectation of targets after all stages;
F_2	The total cost consumption of missiles after all stages.

$$\begin{cases} F_1(\mathbf{X}) = \sum_{t=1}^{T_n(h)} v_t(\prod_{s=h}^{S_n} \prod_{r=1}^{R_n(h)} (1 - p_{rt}(s))^{x_{rt}(s)}) \\ F_2(\mathbf{X}) = \sum_{s=1}^{S_n} \sum_{t=1}^{T_n} \sum_{r=1}^{R_n} c_{rt}(s) \cdot x_{rt}(s) \end{cases} \tag{2.1}$$

Specifically, $\prod_{s=h}^{S_n} \prod_{r=1}^{R_n(h)} (1 - p_{rt}(s))^{x_{rt}(s)}$ denotes the survive expectation of t th target after S_n stages. Thus, F_1 expresses the total survival expectation of targets after all stages. Meanwhile, F_2 represents the cost consumption of missiles during the allocation. Finally, it is also necessary to design reasonable constraints to solve the above problem. From the perspective of reliability and feasibility, the following constraints are built.

$$\sum_{s=1}^{S_n} \sum_{r=1}^{R_n} x_{rt}(s) \geq 1, \forall t \in [1, T_n] \tag{2.2}$$

$$\sum_{s=1}^{S_n} \sum_{t=1}^{T_n} x_{rt}(s) \leq Num_r, \forall r \in [1, R_n] \tag{2.3}$$

$$\boldsymbol{X}(s) \leq \boldsymbol{Z}(s), \forall s \in [1, S_n] \tag{2.4}$$

Constraints (2.2) mean that each target is strictly allocated to at least one missile type in terms of reliability. Constraints (2.3) indicate the maximum amount of each missile that can be allocated to strike the target. The same type of missile can destroy one target in all stages. Then the type can be regarded as several independent same missiles. What is more, this constraint also limits the cost of missiles. Constraints (2.4) represent the multi-stage feature, which reflects that the actual allocation of each stage must obey the feasibility allocation. Preliminary data and decision-makers often obtain feasibility allocation to achieve the expected allocation in the corresponding stage.

Furthermore, the definition of feasibility variables is similar to the decision variable. If the allocation is feasible, the variable $z_{rt}(s)$ is 1. Otherwise, it is 0, where the formulation is $z_{rt}(s) = \begin{cases} 1, \text{if the allocation is feasible at } s \text{ stage,} \\ 0, \text{otherwise.} \end{cases}$. Therefore, we can transform the M-MTA into a multi-objective optimization model as follows.

$$\begin{cases} \min\ F_1 = \sum_{t=1}^{T_n(h)} v_t(\prod_{s=h}^{S_n} \prod_{r=1}^{R_n(h)} (1 - p_{rt}(s))^{x_{rt}(s)}) \\ \min\ F_2 = \sum_{s=1}^{S_n} \sum_{t=1}^{T_n} \sum_{r=1}^{R_n} c_{rt}(s) \cdot x_{rt}(s) \end{cases}$$
$$s.t.$$
$$(2.2), (2.3), (2.4),$$
$$x_{rt}(s) = \begin{cases} 1, \text{if the allocation is determined at } s \text{ stage} \\ 0, \text{otherwise.} \end{cases} \tag{2.5}$$

3. Algorithm framework for M-MTA

This paper focuses on determining the optimal allocation scheme of M-MTA. Hence, we proposed a problem-specific framework (NSGA-MTA) based on the NSGA-II, shown in Algorithm 1. In the beginning, an initial population with feasible solutions will be obtained by the designed rule-based initialization. Then, the population will be selected through the binary tournament and genetic operators with the greedy mechanism and random repair strategy iteratively. Next, a novel Efficient-NS with low complexity provides a search strategy for non-dominated solutions, and CD plays a role in density estimation.

Algorithm 1: The framework of NSGA-MTA

Input: N, $MaxEva$, R_n, T_n, S_n, $\boldsymbol{V}$, $\boldsymbol{P}$, ρ
Output: $\boldsymbol{Pop}$, $\boldsymbol{X}$, F_1, F_2,

$D \leftarrow R_n \times S_n$;
`//D denotes the number of decision variables.`
$\boldsymbol{H} \leftarrow$ *Rule-Based Initialization*(N, D, ρ, $\boldsymbol{V}$, $\boldsymbol{P}$, T_n); `//Algorithm 3`
$\boldsymbol{FrontNo} \leftarrow$ *Efficient-NS*($\boldsymbol{H}$); `//Algorithm 4`
$\boldsymbol{CDNo} \leftarrow$ *Crowding Distance*($\boldsymbol{H}$, $\boldsymbol{FrontNo}$);
while *MaxEva* exceeds **do**
  $\widetilde{\boldsymbol{Pop}} \leftarrow$ *TournamentSelection*($2N$, $\boldsymbol{FrontNo}$, $\boldsymbol{CDNo}$);
  $\boldsymbol{O} \leftarrow$ *Variation*($\widetilde{\boldsymbol{Pop}}$); `//Algorithm 5`
  $\overline{\boldsymbol{Pop}} \leftarrow \boldsymbol{H} \cup \boldsymbol{O}$;
  $\boldsymbol{FrontNo} \leftarrow$ *Efficient-NS*($\overline{\boldsymbol{Pop}}$);
  $\boldsymbol{CDNo} \leftarrow$ *Crowding Distance*($\overline{\boldsymbol{Pop}}$, $\boldsymbol{FrontNo}$);
  $\zeta \leftarrow argmin|FrontNo| \geq N$;
  Remove $|FrontNo_1 \cup \cdots \cup FrontNo_\zeta| - N$ soultions from $FrontNo_\zeta$ with the smallest *CDNo*;
  $\boldsymbol{Pop} \leftarrow FrontNo_1 \cup \cdots \cup FrontNo_\zeta$;
  $\boldsymbol{X} \leftarrow$ *Hybrid Encoding*($\boldsymbol{Pop}$); `//Algorithm 2`
  F_1, $F_2 \leftarrow$ Best individuals in $\boldsymbol{Pop}$;
end while
return $\boldsymbol{Pop}$, $\boldsymbol{X}$, F_1, F_2

Meanwhile, the proposed hybrid encoding mechanism will transform the best individuals into the expected binary decision matrix. Finally, the corresponding objective function will also be evaluated. The detailed contributions of NSGA-MTA are depicted in the following subsections.

3.1. Hybrid encoding

The hybrid encoding mechanism that expresses integer encoding and binary encoding is developed for M-MTA in this paper. Binary encoding is a very intuitive representation of the problem and algorithm construction. However, a large number of redundant 0s and 1s will cause dimension explosion, especially when R_n, T_n, and S_n increase with the complexity of the problem. Therefore, integer encoding is implemented to express feasible solutions at runtime, and binary encoding is adopted to speed up the calculation at function evaluation. Moreover, integer encoding can effectively guarantee the

Constraints (2.2) of the proposed M-MTA, and it is more suitable than permutation encoding in terms of repeatable expression.

The encoding proposed in this subsection constructs a mapping relationship of decision variables, taking Figure 2 as an example. The value of each individual represents the target number that the index number weapon strikes at a particular stage. As shown in Figure 2, the value of the first position representing missile 1 is assigned to damage target 4 in the first stage.

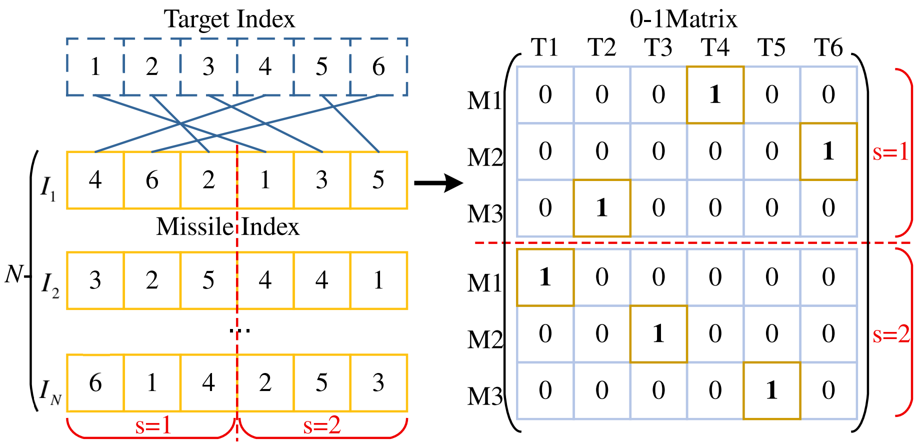


Figure 2. An example of the encoding mechanism.

In general, the beginning will generate a population of N individuals ($I_i, i \in [1, N]$) with S_n subspaces, each of which represents an allocation scheme of the proposed problem. Precisely, each element of individuals in each subspace corresponds to the mapping of missile type r allocated to target t at stage s. R_n and S_n determine the length of individuals. Then, each mapping relationship will be transformed into a straightforward binary matrix $\boldsymbol{X}$ for function evaluation. Besides, the repeatable expression mentioned above is ubiquitous in the M-MTA. The execution process of this encoding with S_n stages is described in detail in Algorithm 2.

Algorithm 2: *Hybrid Encoding(**Pop**, R_n, T_n, S_n)*

Input: $\boldsymbol{Pop}$, R_n, T_n, S_n
Output: $\boldsymbol{X}$

```
N = sizes(Pop, 1);
X = zeros(R_n × S_n, T_n);
for all j ∈ [1, N] do
    V = Pop(j, :);
    for all k ∈ [1, R_n × S_n] do
        X(k, V(k)) = 1;
    end for
end for
return X
```

Appropriate encoding can provide the basis for evolution, but enhancing the diversity and convergence of the population also needs to design proper heuristic initialization and match genetic operators.

3.2. Rule-based initialization

In general, the initialization method of MOEAs usually adopts the random generation mechanism. Although this is effective for some benchmark problems, it is also necessary to design a rule-based initialization to construct feasible solutions for real-world problems. In other words, the search space exploration will only be efficient if we continue to apply the random generation mechanism. When a reasonable rule-based method is taken into account in the mechanism, this phenomenon will be alleviated to a great extent. Significantly, the whole evolutionary process will be accelerated toward the global optimal or near-optimal solutions. However, suppose the initial population is full of many rule-based solutions. In that case, the algorithm will fall into the local optimum, and the population also will need more diversity. Therefore, a rule-based initialization with priority is developed to generate the initial population in this paper.

The rule-based initialization for M-MTA is inspired by design in the DWTA problem [17] to generate solutions. The proposed initialization not only adopts the heuristic of missile choice priority and sequence calculation to increase feasible solutions but also applies the random generation to ensure diversity. The proposed method aims to get the allocation scheme by maximizing the expected damage effectiveness of missiles to targets based on prior knowledge. The prior damage effectiveness $p_{rt}^{prior}(s)$ of missile r to target t at the stage s is calculated as follows.

$$p_{rt}^{prior}(s) = p_{rt}(s) \times v_t \tag{3.1}$$

where definitions of $p_{rt}(s)$ and v_t have been given in Table 1.

The main framework of the proposed rule-based initialization with priority is described in Algorithm 3. Firstly, the random population $\boldsymbol{Pop}_{rnd}$ is constructed according to population size N and decision variable D. Meanwhile, prior knowledge $\boldsymbol{P}_{rt}^{prior}$ can also be obtained via Eq (3.1). Next, we assume that the heuristic mainly follows the rule of missile choice priority, which means that missiles can choose the most threat target based on damage ability. Although feasible solutions are generated under the given data, we also pay attention to the lack of diversity in the population. Synthetically, the sequence calculation is involved in constructing more heuristic solutions, whose purpose is to disorderly sort the given data and obtain more feasible solutions. Finally, the rule-based solutions replace $\boldsymbol{Pop}_{rnd}$ in the proportion of ρ in order to get the population $\boldsymbol{H}$, where ρ is defined as a dimensionless constant ($\rho \in (0, 1)$).

3.3. Efficient-NS

The non-dominated sorting method is essential in determining non-dominated fronts as a search strategy. The outstanding advantage of NSGA-II lies in the design of fast-NS, but there are still some redundant comparisons in finding non-dominated solutions. In brief, fast-NS shows that each solution needs to be compared with all solutions in the population to determine the non-dominated front. Then, there will inevitably be many redundant comparisons to aggravate the complexity of the whole algorithm. Hence, a novel efficient-NS is implemented to the improved framework in this paper, which has been proved to have lower complexity and higher efficiency than fast-NS [26]. The critical characteristic of efficient-NS is that the solution only needs to be compared once. A new front will be created when the compared solution belongs to something other than the existing non-dominated fronts. This operation only terminates once all the dominant fronts are determined.

Algorithm 3: *Rule-Based Initialization(N, D, ρ, **V**, **P**, T_n)*

Input: $N, D, \rho, \boldsymbol{V}, \boldsymbol{P}, T_n$
Output: $\boldsymbol{H}$

$\boldsymbol{Pop}_{rnd} \leftarrow randi([D.lower, D.upper], N, D)$;
Calculate $\boldsymbol{P}_{rt}^{prior}$ via Eq (3.1);
$U \leftarrow ceil(N \times \rho)$;
`//Missile Choice Priority`
$[\backsim, indr1] \leftarrow \max(\boldsymbol{P}_{rt}^{prior}, [], 2)$;
$\boldsymbol{Ind}_r(1, :) = indr1$;
`//Sequence Calculation`
for all $t \in [2, N \times \rho]$ **do**
    $\boldsymbol{P}_{seq} = \boldsymbol{P}_{rt}^{prior}(:, randperm(T_n))$;
    $[\backsim, indr2] = \max(\boldsymbol{P}_{seq}, [], 2)$;
    $\boldsymbol{Ind}_r(t, :) = indr2$;
end for
`//Population Generation`
$ind = randperm(N)$;
$\boldsymbol{Pop}_{rnd}(ind(1{:}N \times \rho), :) = \boldsymbol{Ind}_r$;
$\boldsymbol{H} \leftarrow \boldsymbol{Pop}_{rnd}$;
return $\boldsymbol{H}$

The M-MTA has been transformed into a constrained optimization problem above, so we made some problem-specific improvements. Infeasible solutions are repaired by combining the results of heuristic factors and the constraints. Meanwhile, any objective function can be sorted in ascending order. The efficient-NS is expressed in Algorithm 4 in detail.

3.4. Offspring Generation with random repair strategy

The genetic algorithm drives the main framework of NSGA-II, so selection, crossover, and mutation operations must occur during the offspring generation. The selection operator is a significant element in the whole generation, directly determining the parent individuals participating in the next step. Hence, we adopt the binary tournament select approach with low complexity and adjustable selection pressure to generate parent individuals. This approach aims at selecting two individuals randomly from the population and defining the winner with a better objective function value as the parent individual.

Furthermore, the crossover operator and the mutation operator are involved in Algorithm 5 (*Variation*) in this paper, both necessary conditions for evolution in maintaining diversity. Recently, various crossover and mutation operators have been conducted by many scholars, such as single-point crossover, order crossover, uniform crossover and simulated binary crossover (SBX), simple mutation, uniform mutation, and Gaussian mutation. Nevertheless, it is worth noting that the proposed encoding contains repeatable integer encoding, and the M-MTA belongs to the combinational optimization problem. Therefore, we design the greedy mechanism and the random repair strategy to add the partially mapped crossover (PMX) and inverse mutation in this paper, respectively.

Algorithm 4: *Efficient-NS(**Pop**)*

```
Input: Pop
Output: FrontNo
Find infeasible solution p_in in Pop;
Do infeasible repair on p_in;
FrontNo = [   ];
Γ ← Sort Pop in ascending order according to any objective function;
for all p_b ∈ Γ do
    l = size(FrontNo);
    n = 1;
    while true do
        Compare p_b with the solutions in FrontNo_n;
        //From the last to the first in FrontNo_n.
        if no solution in FrontNo_n dominates p_b then
            return n;
        else
            n++;
            if n > l then
                return l + 1;
            end if
        end if
    end while
end for
return FrontNo;
```

3.4.1. The design of greedy mechanism

Moreover, the greedy mechanism is vital in preserving better genes during the crossover. When the parents have the same gene in the same position, Eq (3.1) is applied to judge whether the current gene is good. If the selected genes are good, they will inherit from the parents to offspring. Finally, two positions are randomly selected for a crossover at other positions where good genes are removed. In particular, if no good genes exist, the crossover will be done according to the PMX operator. The primary process can be illustrated in Algorithm 6 (*Crossover with greedy mechanism*).

3.4.2. The design of random repair strategy

Last but not least, the random repair strategy is introduced in mutation, which will randomly select the generated offspring and repair the infeasible individuals via handling the Constraints (2.3) and (2.4). In detail, the strategy aims at querying unassigned targets, randomly reassigning them according to the damage capacity of missiles, and simultaneously repairing the assigned results according to the feasibility constraints. Finally, the repaired individuals rejoin the offspring to participate in the next evolution. Notably, the mutation follows the inverse mutation for individuals satisfying the constraints. The detailed process is described in Algorithm 7 (*Random repair strategy*).

Algorithm 5: *Variation*($\widetilde{\boldsymbol{Pop}}$)

Input: $\widetilde{\boldsymbol{Pop}}$
Output: $\boldsymbol{O}$

//Crossover Operation
NSel1 = size($\widetilde{\boldsymbol{Pop}}$,1);
for i = 1:2: *NSel1*-mod(*NSel1, 2*) **do**
  if $p_c \geq rand$ **then**
    if Find good genes **then**
      Oc ← *Crossover with greedy mechanism(**O**)*; //Algorithm 6
    else
      Oc ← PMX Crossover;
    end if
  end if
end for
//Mutation Operation
[*NSel2*, *L*] = size (***Oc***);
for i =1: *NSel2* **do**
  if $p_m \geq rand$ **then**
    if Current Individual Satisfy Constraints **then**
      Om ← Inverse Mutation;
    else
      Om ← *Random Repair Strategy(**Oc**)*; //Algorithm 7
    end if
  end if
end for
O ← ***Om***;
return ***O***;

3.5. Complexity analysis

In this subsection, we further complete the complexity analysis on the proposed NSGA-MTA, where m is denoted as the number of objectives. Firstly, the time complexity of initialization is $O(m \cdot N \cdot \rho)$. Then, the process of Efficient-NS is $O(m \cdot N \cdot \sqrt{N})$. Next, generating solutions based on proposed strategies is $O(m \cdot N)$. Finally, $O(m \cdot N \cdot \log N \cdot Iter)$. So, the time complexity of NSGA-MTA is $O[m \cdot N \cdot (1 + \rho + \sqrt{N} + \log N \cdot Iter)]$.

4. Preliminary numerical experiments

This section is devoted to evaluating and comparing the proposed NSGA-MTA. Firstly, a test-case generator for the M-MTA is constructed to provide 15 different problems with various scenarios. Then, four prevailing MOEAs were selected for the performance comparison, and some parameters were determined from the relevant literature. Meanwhile, we introduce four metrics to measure performance.

Algorithm 6: *Crossover with greedy mechanism(**O**)*

Input: ***O***
Output: ***Oc***

Find the same location in Parents;
Do Eq (3.1) on the same position and compare, respectively;
if Good gene in these positions **then**
    Good gene inherit to the offspring;
    Select other point to crossover;
else
    Do PMX Crossover;
end if
return ***Oc***;

Algorithm 7: *Random repair strategy(**Oc**)*

Input: ***Oc***
Output: ***Om***

Find the infeasible individual o_{in} in ***Oc***;
Reshape o_{in} into the 0-1 encoding matrix $\mathbf{M}_{in}$;
if $\mathbf{M}_{in}$ not equal to $\mathbf{Z}$ based on Constraints (2.4) **then**
    Randomly select element in $\mathbf{Z}$ to assign to the element in $\mathbf{M}_{in}$;
    Calculate the sum $\mathbf{M}_r$ of all rows in new matrix;
    if The value $\mathbf{M}_r$ not satisfy Constraints (2.3) **then**
        Change 1s into 0s in unsatisfied rows according to the number of constraints;
    end if
end if
Oc ← Repaired individual;
Om ← ***Oc***
return ***Om***;

4.1. Test-case generator

In this subsection, we design the test-case generator to provide 15 different M-MTA scenarios, and each represents the corresponding problem scale to evaluate the algorithm's performance. Moreover, we refer to the settings in [9, 28] to construct the essential parameters in the model, which will be further generated within the interval based on the following rules.

1) *Generation of* v_t. The threat value of target t ($\forall t \in [1, T_n]$) is defined as a random value in a range between 1 and 100.
2) *Generation of* $p_{rt}(s)$*, and* $c_{rt}(s)$. The damage capability and cost of missile r to target t at s stage are described as follows, respectively.

$$p_{rt}(s) = p_l + (p_h - p_l) \times rand \tag{4.1}$$

$$c_{rt}(s) = c_l + (c_h - c_l) \times rand \tag{4.2}$$

where $\forall r \in [1, R_n]$, $\forall t \in [1, T_n]$, and $\forall s \in [1, S_n]$; *rand* denotes a uniformly distributed number between 0 and 1; p_h and p_l are the constants with $0 < p_l < p_h < 1$ which represent the limitation of damage capability; Similarly, c_h and c_l are the constants with $0 < c_l < c_h < 1$ which mean the available cost consumption.

The parameters can be referenced as $p_h = \{0.9, 0.95, 0.99\}$, $p_l = \{0.55, 0.6, 0.65\}$, $c_h = \{8, 10, 12\}$ and $c_l = \{5, 6, 7\}$, referred to the settings in [6, 9, 28]. Therefore, we further apply the Taguchi method to tune these parameters as follows, where the main effect plot for tuning parameters is shown in Figure 3. Referring to the data under the maximum SN ratios in Figure 3, the parameters are determined as $p_l = 0.55$, $p_h = 0.95$, $c_l = 5$, and $c_h = 10$.

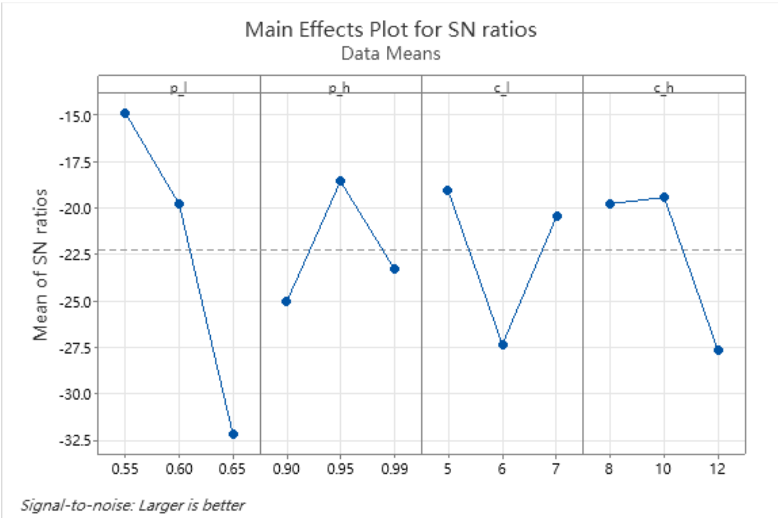


Figure 3. The main effect plot for SN ratios for tuning the parameters.

3) *Generation of* Num_r. The available number of missile r ($\forall r \in [1, R_n]$) will be discussed in line with different scenarios.

$$Num_r = \begin{cases} 2, 0 < v_t \leq 50; \\ 3, 50 < v_t \leq 90; \\ 4, 90 < v_t \leq 100. \end{cases} \tag{4.3}$$

4.2. Comparison algorithm and Performance metrics

4.2.1. Comparison algorithm

In order to verify the effectiveness of the improvements, we compare NSGA-MTA with four algorithms that have shown excellent performance in the constrained optimization problem. These algorithms include Pareto-based algorithm (NSGA-II [20], NSGA-III [29], C-TAEA [30]), and decomposition-based algorithm (C-MOEA/D [31]). More importantly, these algorithms had shown superior performance in solving the allocation problem [12, 32, 33] discussed in this paper.

4.2.2. Performance metrics

Furthermore, this paper introduces four state-and-art metrics to compare the performance among different algorithms.

Inverted generational distance (IGD) [34] The *IGD* metric can express the diversity and convergence of solutions simultaneously, whose characteristic is to obtain the average distance between known PF (P) and true PF (P^*). The algorithm with smaller $IGD(P, P^*)$ shows that the obtained P is closer to P^*, so the performance of this algorithm will be better. The detailed formulation is as follows.

$$IGD\,(P, P^*) = \frac{\sum_{\mathbf{k} \in P^*} d\,(\mathbf{k}, P)}{|P^*|} \tag{4.4}$$

Generally, P^* means a set of points distributed along the true PF or nearly true PF in the objective space. $d(\mathbf{k}, P)$ denotes the minimum Euclidean distance between $\mathbf{k}$ in P^* and elements in P. In this paper, the discussed problem belongs to a real-world problem, so its true PF is hard to obtain. We run different algorithms independently 31 times and sort all the obtained fronts via non-dominate sorting. Finally, the first front is determined to be nearly true PF (P^*).

Inverted generational distance plus (IGD+) [35] The *IGD*+ metric further measures the diversity and convergence by slightly improving the distance calculation of *IGD*. The improved distance $d^+(\mathbf{k}, \mathbf{a})$ between $\mathbf{k} = (k_1, k_2, \ldots, k_m)$ in P^* and the solution $\mathbf{a} = (a_1, a_2, \ldots, a_m)$ in the dominated region is defined as follows, where k_m and a_m denote the component of $\mathbf{k}$ and $\mathbf{a}$ in m-th objective space, respectively.

$$d^+(\mathbf{k}, \mathbf{a}) = \sqrt{\sum_{i=1}^{m} (\max\{a_i - k_i, 0\})^2} \tag{4.5}$$

Therefore, the detailed formulation of *IGD*+ is defined as follows.

$$IGD{+}\,(P, P^*) = \frac{\sum_{\mathbf{k} \in P^*} \min_{\mathbf{a} \in P} d^+\,(\mathbf{k}, \mathbf{a})}{|P^*|} \tag{4.6}$$

where the main form of *IGD*+ is similar to *IGD*, the difference lies in the distance calculation. The nearly true PF (P^*) is also obtained in the same way as in *IGD*.

Hypervolume (HV) [36] The *HV* metric mainly determines the volume of the objective space that PF weakly dominates to evaluate the performance of algorithms. The detailed definition is shown as follows.

$$HV(P) = VOL\left(\bigcup_{b \in P} [b_1, b_1^w] \times \cdots \times [b_n, b_n^w]\right) \tag{4.7}$$

where *VOL* means the Lebesgue method for measuring volume, and $[b_1^w, b_2^w, \ldots, b_n^w]$ denotes the worse points in the hyper-volume and is also defined as the reference point. It can be seen from Eq (15) that the larger *HV* formed by the reference point and the points in *PF*, the better the performance of the algorithm.

Therefore, the reference point is of great significance in measuring performance. In this paper, we obtain the worst point from the true *PF* evaluated above and define the value expanded by 1.1 times as the reference point.

Table 2. The M-MTA scenario definition.

Scenario	S_n	R_n	T_n
1	2	24 (9,15)	8
2	2	30 (12,18)	12
3	2	51 (23,18)	25
4	2	62 (32,30)	34
5	2	166 (70,96)	90
6	3	41 (13,10,19)	14
7	3	48 (18,20,10)	16
8	3	51 (18,15,18)	19
9	3	70 (21,23,26)	14
10	4	59 (18,11,13,17)	11
11	4	80 (15,28,26,11)	17
12	4	83 (14,23,25,21)	20
13	5	42 (8,6,10,9,9)	7
14	5	36 (7,7,7,5,10)	8
15	5	39 (6,10,6,8,9)	10

Metric for diversity (DM) [37] The metric for diversity (*DM*) mainly measures the spread of non-dominated solutions on the Pareto front. The detailed formulation is as follows.

$$DM = \sqrt{(\max f_i^1 - \min f_i^1)^2 - (\max f_i^2 - \min f_i^2)^2} \tag{4.8}$$

where f_i^1 and f_i^2 means the first and second objective values of the i-th non-dominated solution, respectively.

4.3. Experiment environment and parameters settings

All the experiments were implemented using Python 3.8.5 and run on a PC with a 5.0 GHz Core i7-12700KF CPU and 32.00 GB RAM, mainly conducted on the Pycharm and Minitab software platforms. Moreover, numerical results are derived from 31 independent runs of each algorithm on the 15 scenarios.

Firstly, 15 different M-MTA scenarios built by the proposed test-case generator are shown in Table 2, where the value in brackets in the R_n column denotes the number of available missiles at each stage. Besides, we divide the problem scale into small ($0 < R_n < 50$), medium ($50 \leq R_n < 100$), and large ($R_n \geq 100$) according to the number of decision variables (R_n). Based on the definition of multi-stage, we also divide the scenarios into two and more than two stages.

Significantly, the population size (*PopNum*) and the number of iterations (*Iter*) need to be consistent for each algorithm during the process. According to preliminary experiments [9], the effect of *PopNum* was negligible. Furthermore, we define *Iter* based on the number of decision variables (R_n) as follows,

Table 3. The parameters in the experiment.

Parameters	Value or range
R_n	Given by Table 2
T_n	Given by Table 2
S_n	Given by Table 2
v_t	[1, 100]
$p_{rt}(s)$	[0.55, 0.95]
$c_{rt}(s)$	[5, 10]
Num_r	{2, 3, 4}
$PopNum$	100
$Iter$	{100, 150, 200}
NE	{10000, 15000, 20000}
p_c	0.9
p_m	0.01
ρ	0.1

where $NE = PopNum \times Iter$.

$$Iter = \begin{cases} 100, 0 < R_n < 50, \\ 150, 50 \le R_n < 100, \\ 200, R_n \ge 100. \end{cases} \tag{4.9}$$

The other parameters mainly include the crossover rate (p_c), mutation rate (p_m), and the ratio of rule-based individuals to initial individuals (ρ). Their settings adopted mainly refer to those claimed in [6, 17, 19, 32, 38, 39], shown in Table 3.

5. Results and discussion

5.1. Results

Experimental results on the 15 scenarios are involved in this section, which mainly consists of two parts. Regarding the multi-stage nature, we group experiments into scenarios with $S_n = 2$ and $S_n \ge 2$. Besides, the effect of the proposed strategies is further verified by the Taguchi method.

The statistical results (the median and IQR of IGD, IGD+, HV, and DM) about the 31 independent runs are provided in Tables 4 and 5, wherein the best results are highlighted in bold, and IQR is revealed in the corresponding brackets. Furthermore, the Wilcoxon rank-sum test results are added to the table.

5.1.1. Experiments on effect of proposed NSGA-MTA

A. Results of the scenarios with $S_n = 2$. Table 4 expresses the median and IQR of metrics for each algorithm under the scenarios with $S_n = 2$. The table shows that the proposed NSGA-MTA is statistically better than other algorithms in all metrics. It is worth mentioning that the results obtained by NSGA-MTA under IGD and IGD+ have more significant advantages than other algorithms, significantly as the scale of the problem increases, and the numerical difference is even more significant.

Table 4. The Median (IQR) of metrics ($S_n = 2$).

No.	Metic	NSGA-II	NSGA-III	C-MOEA/D	C-TAEA	NSGA-MTA
1	IGD	2.9177e+1 (7.36e+0) -	2.8351e+1 (8.06e+0) -	2.7856e+1 (4.25e+0) -	1.9384e+1 (4.75e+0) -	**1.1953e+1 (3.36e+0)**
	IGD+	2.3332e+1 (6.92e+0) -	2.4221e+1 (8.82e+0) -	1.9714e+1 (6.73e+0) -	1.7884e+1 (4.63e+0) -	**4.6839e+0 (2.99e+0)**
	HV	1.4367e-1 (1.15e-2) -	1.4900e-1 (1.77e-2) -	1.4648e-1 (1.63e-2) -	1.6793e-1 (8.73e-3) -	**1.9277e-1 (8.35e-3)**
	DM	1.5215e-1 (8.83e-2) -	1.6966e-1 (1.31e-1) -	**4.8517e-1 (1.12e-1) =**	2.9702e-1 (1.76e-1) -	**5.2298e-1 (1.60e-1)**
2	IGD	7.4856e+1 (2.22e+1) -	5.4460e+1 (1.70e+1) -	5.6604e+1 (1.70e+1) -	6.3017e+1 (1.91e+1) -	**1.5127e+1 (1.21e+1)**
	IGD+	7.4856e+1 (2.22e+1) -	5.4460e+1 (1.71e+1) -	5.6604e+1 (1.71e+1) -	6.3008e+1 (1.92e+1) -	**1.4448e+1 (1.30e+1)**
	HV	4.0943e-2 (1.38e-2) -	5.8727e-2 (1.32e-2) -	5.0377e-2 (9.90e-3) -	5.5959e-2 (1.17e-2) -	**1.0530e-1 (1.34e-2)**
	DM	5.0582e-2 (3.09e-2) -	5.9880e-2 (4.80e-2) -	1.9029e-1 (1.74e-1) -	8.8518e-2 (4.17e-2) -	**4.3001e-1 (1.49e-1)**
3	IGD	7.9908e+1 (1.73e+1) -	6.1677e+1 (2.32e+1) -	1.0243e+2 (1.78e+1) -	9.3906e+1 (1.41e+1) -	**1.9546e+1 (1.83e+1)**
	IGD+	7.9574e+1 (1.73e+1) -	6.1290e+1 (2.32e+1) -	1.0242e+2 (1.78e+1) -	9.3841e+1 (1.41e+1) -	**1.7597e+1 (2.00e+1)**
	HV	5.0485e-2 (6.87e-3) -	5.6421e-2 (8.97e-3) -	3.6003e-2 (5.96e-3) -	5.0997e-2 (6.78e-3) -	**8.7212e-2 (1.40e-2)**
	DM	1.7515e-1 (1.05e-1) -	3.3614e-1 (1.75e-1) -	3.0654e-1 (2.03e-1) -	1.4524e-1 (7.72e-2) -	**7.2879e-1 (1.99e-1)**
4	IGD	1.2507e+2 (3.31e+1) -	1.0778e+2 (2.45e+1) -	1.9538e+2 (3.99e+1) -	1.7821e+2 (5.20e+1) -	**4.6037e+1 (1.98e+1)**
	IGD+	1.2507e+2 (3.59e+1) -	1.0679e+2 (2.48e+1) -	1.9538e+2 (3.99e+1) -	1.7819e+2 (5.18e+1) -	**4.5957e+1 (2.00e+1)**
	HV	5.2203e-2 (6.35e-3) -	5.7042e-2 (8.76e-3) -	3.0573e-2 (5.03e-3) -	4.7962e-2 (7.90e-3) -	**8.7531e-2 (9.42e-3)**
	DM	2.1893e-1 (1.75e-1) -	3.5587e-1 (2.11e-1) -	2.6741e-1 (1.09e-1) -	1.0531e-1 (7.76e-2) -	**6.0446e-1 (1.42e-1)**
5	IGD	1.5022e+2 (3.56e+1) -	1.2336e+2 (3.09e+1) -	3.4064e+2 (4.07e+1) -	3.2947e+2 (4.27e+1) -	**3.8313e+1 (1.72e+1)**
	IGD+	1.5022e+2 (3.56e+1) -	1.2302e+2 (3.10e+1) -	3.4064e+2 (4.07e+1) -	3.2947e+2 (4.27e+1) -	**3.7206e+1 (2.12e+1)**
	HV	4.9373e-2 (4.65e-3) -	5.1198e-2 (4.22e-3) -	2.8566e-2 (2.21e-3) -	3.5044e-2 (2.86e-3) -	**6.9763e-2 (3.56e-3)**
	DM	3.5965e-1 (1.57e-1) -	4.2934e-1 (1.69e-1) -	1.6667e-1 (6.38e-2) -	6.0132e-2 (5.05e-3) -	**5.6335e-1 (1.76e-1)**
+/-/=		0/20/0	0/20/0	0/19/1	0/20/0	

Note: "+/-/=" indicates that the algorithm is statistically better, worse, and equal to NSGA-MTA under the Wilcoxon rank-sum test, respectively.

However, in Scenario 1, C-MOEA/D is statistically equal to NSGA-MTA in DM by about 7.2%, which means that C-MOEA/D also has certain advantages in obtaining better diversity of PF distribution in small-scale problems.

B. Results of the scenarios with $S_n \geq 2$. In this group, scenarios with $S_n \geq 2$ are experimentally demonstrated individually, whose results are shown in Table 5. Intuitively, it can be argued that the overall performance of MTA is statistically slightly better than other algorithms. In particular, C-TAEA performs better and slightly equal to NSGA-MTA on IGD for Scenarios 8 and 15, respectively. Furthermore, in Scenario 8, NSGA-MTA fails to show the superiority in IGD so that it is only statistically equal to NSGA-II and NSGA-III by about 2%, and 5.2%, and worse than C-TAEA by about 14.2%.

C. Results of all scenarios. The PF of each algorithm in all scenarios under the best IGD+ is visually shown in Figure 4. In each subfigure, the true PF is composed of points (solutions) in the shape of red squares, the first non-dominated layer obtained by all algorithms. Moreover, the PF made of green is obtained by NSGA-MTA, which is the closest to the true PF in all scenarios. Combined with the comprehensive metrics (IGD, IGD+, and HV) in Tables 4 and 5, the results in Figure 4 further prove that the PF obtained by MTA has better convergence and diversity in distribution than others.

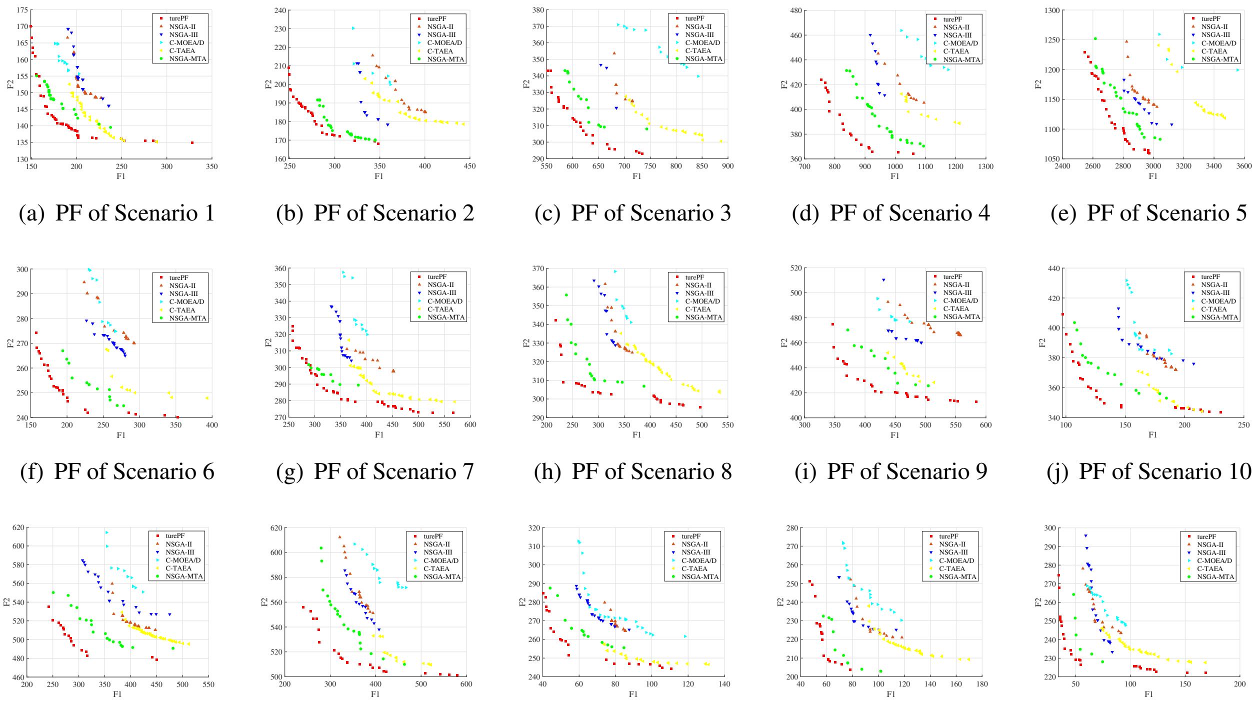
5.1.2. Experiments on effect of proposed strategies

Last but not least, the Taguchi method is adopted to verify the effect of the novel strategies in NSGA-MTA. The detailed descriptions of the levels and the factors in the Taguchi method are shown in Table

Table 5. The Median (IQR) of metrics. ($S_n \geq 2$)

No.	Metic	NSGA-II	NSGA-III	C-MOEA/D	C-TAEA	NSGA-MTA
6	IGD	5.6432e+1 (1.26e+1) -	5.1051e+1 (1.02e+1) -	6.3299e+1 (7.36e+0) -	6.2871e+1 (2.23e+1) -	**2.5320e+1 (1.09e+1)**
	IGD+	5.3454e+1 (1.19e+1) -	4.7640e+1 (1.17e+1) -	6.0347e+1 (7.15e+0) -	6.2657e+1 (2.35e+1) -	**1.7870e+1 (1.16e+1)**
	HV	9.5060e-2 (1.28e-2) -	9.8271e-2 (1.20e-2) -	7.6126e-2 (8.87e-3) -	9.8336e-2 (1.62e-2) -	**1.4041e-1 (1.30e-2)**
	DM	1.4669e-1 (1.51e-1) -	3.7083e-1 (2.42e-1) -	4.6656e-1 (2.51e-1) -	1.5944e-1 (6.85e-2) -	**6.6096e-1 (1.38e-1)**
7	IGD	6.0598e+1 (1.03e+1) -	5.5858e+1 (8.30e+0) -	7.5676e+1 (8.56e+0) -	4.6572e+1 (1.44e+1) -	**3.4339e+1 (1.00e+1)**
	IGD+	5.0268e+1 (9.86e+0) -	4.2000e+1 (1.32e+1) -	6.6641e+1 (8.52e+0) -	4.6037e+1 (1.32e+1) -	**1.1715e+1 (9.86e+0)**
	HV	1.0939e-1 (9.39e-3) -	1.1568e-1 (1.08e-2) -	8.6449e-2 (1.04e-2) -	1.2700e-1 (9.25e-3) -	**1.5414e-1 (1.15e-2)**
	DM	2.2031e-1 (2.06e-1) -	3.9455e-1 (1.72e-1) -	4.1077e-1 (1.64e-1) -	2.6224e-1 (1.21e-1) -	**5.1646e-1 (7.35e-2)**
8	IGD	**6.5007e+1 (1.72e+1) =**	**6.7191e+1 (1.36e+1) =**	7.8833e+1 (1.28e+1) -	**5.4587e+1 (1.05e+1) +**	6.3682e+1 (2.08e+1)
	IGD+	4.9890e+1 (1.09e+1) -	4.5073e+1 (9.92e+0) -	6.6777e+1 (1.34e+1) -	5.3543e+1 (1.20e+1) -	**1.2063e+1 (7.15e+0)**
	HV	1.0008e-1 (1.45e-2) -	1.0292e-1 (1.06e-2) -	7.9902e-2 (1.15e-2) -	1.0584e-1 (9.66e-3) -	**1.5104e-1 (1.11e-2)**
	DM	2.6201e-1 (1.32e-1) -	3.4053e-1 (1.48e-1) -	3.5489e-1 (3.27e-1) -	2.7508e-1 (1.11e-1) -	**4.8910e-1 (1.16e-1)**
9	IGD	7.0250e+1 (1.68e+1) -	6.0221e+1 (1.18e+1) -	7.2213e+1 (5.78e+0) -	3.9279e+1 (2.24e+1) -	**2.2585e+1 (8.88e+0)**
	IGD+	6.9857e+1 (1.91e+1) -	5.7224e+1 (1.16e+1) -	6.5276e+1 (8.81e+0) -	3.8533e+1 (2.41e+1) -	**1.7483e+1 (1.14e+1)**
	HV	7.1801e-2 (1.11e-2) -	7.9382e-2 (8.29e-3) -	7.1955e-2 (6.78e-3) -	9.5921e-2 (1.81e-2) -	**1.1956e-1 (1.01e-2)**
	DM	1.6774e-1 (1.09e-1) -	2.2762e-1 (1.39e-1) -	5.2833e-1 (1.81e-1) -	3.6249e-1 (1.59e-1) -	**7.1583e-1 (1.52e-1)**
10	IGD	4.6968e+1 (7.59e+0) -	4.3531e+1 (5.78e+0) -	4.7618e+1 (7.48e+0) -	3.0960e+1 (7.96e+0) -	**2.1657e+1 (8.06e+0)**
	IGD+	4.3965e+1 (9.35e+0) -	3.8651e+1 (1.05e+1) -	4.5588e+1 (8.09e+0) -	2.9215e+1 (8.20e+0) -	**1.0819e+1 (5.70e+0)**
	HV	1.0488e-1 (1.41e-2) -	1.1217e-1 (1.53e-2) -	1.0890e-1 (1.26e-2) -	1.2521e-1 (1.29e-2) -	**1.6508e-1 (1.03e-2)**
	DM	1.0972e-1 (9.01e-2) -	1.8784e-1 (2.13e-1) -	3.7083e-1 (2.10e-1) -	2.3107e-1 (9.07e-2) -	**5.4062e-1 (1.42e-1)**
11	IGD	8.6064e+1 (1.96e+1) -	7.4373e+1 (1.45e+1) -	1.0784e+2 (1.83e+1) -	9.7834e+1 (1.99e+1) -	**2.8988e+1 (1.08e+1)**
	IGD+	8.5054e+1 (2.09e+1) -	7.4370e+1 (1.67e+1) -	1.0741e+2 (1.87e+1) -	9.7824e+1 (1.99e+1) -	**2.6147e+1 (1.43e+1)**
	HV	6.5029e-2 (1.04e-2) -	7.1581e-2 (8.37e-3) -	4.9274e-2 (9.52e-3) -	6.5356e-2 (9.48e-3) -	**1.1370e-1 (1.72e-2)**
	DM	1.3624e-1 (9.21e-2) -	1.8727e-1 (1.82e-1) -	2.0599e-1 (1.15e-1) -	1.7465e-1 (7.73e-2) -	**8.0284e-1 (2.32e-1)**
12	IGD	7.5151e+1 (1.02e+1) -	7.1658e+1 (8.96e+0) -	9.6683e+1 (1.32e+1) -	6.7619e+1 (1.45e+1) -	**4.0364e+1 (9.91e+0)**
	IGD+	6.0605e+1 (1.13e+1) -	5.2996e+1 (1.52e+1) -	8.8487e+1 (1.58e+1) -	6.2974e+1 (1.69e+1) -	**2.1302e+1 (9.88e+0)**
	HV	9.3362e-2 (7.42e-3) -	9.8286e-2 (1.28e-2) -	7.3219e-2 (8.23e-3) -	1.0223e-1 (1.10e-2) -	**1.3488e-1 (7.89e-3)**
	DM	4.0736e-1 (1.24e-1) -	5.1820e-1 (1.36e-1) -	4.4795e-1 (1.36e-1) -	2.7553e-1 (1.26e-1) -	**6.4766e-1 (6.61e-2)**
13	IGD	2.7184e+1 (5.94e+0) -	2.2102e+1 (2.59e+0) -	2.3179e+1 (6.09e+0) -	1.8793e+1 (5.66e+0) -	**1.3466e+1 (2.73e+0)**
	IGD+	2.5983e+1 (7.12e+0) -	2.0271e+1 (3.33e+0) -	2.1705e+1 (7.36e+0) -	1.7581e+1 (6.10e+0) -	**8.8513e+0 (3.75e+0)**
	HV	1.0651e-1 (1.43e-2) -	1.2347e-1 (8.60e-3) -	1.2109e-1 (1.77e-2) -	1.2438e-1 (1.94e-2) -	**1.5754e-1 (1.40e-2)**
	DM	1.4368e-1 (6.56e-2) -	1.3099e-1 (1.78e-1) -	4.7120e-1 (1.82e-1) -	1.8713e-1 (1.25e-1) -	**5.7443e-1 (2.25e-1)**
14	IGD	3.0557e+1 (7.20e+0) -	2.5375e+1 (9.64e+0) -	3.6858e+1 (5.56e+0) -	3.6917e+1 (7.46e+0) -	**8.7787e+0 (3.31e+0)**
	IGD+	3.0557e+1 (7.28e+0) -	2.5373e+1 (9.72e+0) -	3.6858e+1 (5.56e+0) -	3.6711e+1 (7.96e+0) -	**6.9575e+0 (4.50e+0)**
	HV	8.8502e-2 (2.03e-2) -	1.0497e-1 (2.46e-2) -	8.0714e-2 (1.55e-2) -	6.7834e-2 (2.13e-2) -	**1.6195e-1 (1.36e-2)**
	DM	1.2361e-1 (6.57e-2) -	2.3733e-1 (1.53e-1) -	1.9212e-1 (1.02e-1) -	8.8731e-2 (1.27e-2) -	**7.6158e-1 (1.16e-1)**
15	IGD	3.3637e+1 (5.56e+0) -	3.2997e+1 (7.18e+0) -	3.6020e+1 (4.19e+0) -	**2.6125e+1 (5.03e+0) =**	2.6943e+1 (4.98e+0)
	IGD+	2.4314e+1 (4.80e+0) -	2.0852e+1 (4.93e+0) -	2.9354e+1 (6.49e+0) -	2.3475e+1 (4.47e+0) -	**7.9636e+0 (3.36e+0)**
	HV	1.8624e-1 (1.50e-2) -	1.9362e-1 (1.41e-2) -	1.7137e-1 (1.75e-2) -	1.9259e-1 (1.40e-2) -	**2.3607e-1 (1.10e-2)**
	DM	1.4970e-1 (2.52e-1) -	3.0283e-1 (1.88e-1) -	2.6690e-1 (2.42e-1) -	2.5841e-1 (1.39e-1) -	**4.6368e-1 (2.07e-2)**
+/-/=		0/39/1	0/39/1	0/40/0	1/38/1	

Note: ”+/-/=” indicates that the algorithm is statistically better, worse, and equal to NSGA-MTA under the Wilcoxon rank-sum test, respectively.



(a) PF of Scenario 1 (b) PF of Scenario 2 (c) PF of Scenario 3 (d) PF of Scenario 4 (e) PF of Scenario 5

(f) PF of Scenario 6 (g) PF of Scenario 7 (h) PF of Scenario 8 (i) PF of Scenario 9 (j) PF of Scenario 10

(k) PF of Scenario 11 (l) PF of Scenario 12 (m) PF of Scenario 13 (n) PF of Scenario 14 (o) PF of Scenario 15

Figure 4. The PF of 15 Scenarios under best *IGD*+.

6. We select the $L8(2^3)$ orthogonal table for the number of factors and levels. The signal-to-noise (SN) ratio is the measure of robustness, where the larger SN ratio means better algorithm performance.

$$\begin{cases} SN = -10 \times \log(\frac{\sum(1/Y^2)}{N_o}) \\ Y = (\sum_{i=1}^{N_t} (\frac{HV}{IGD})_i)/N_t, i \in [1, N_t] \end{cases} \tag{5.1}$$

where Y and N_o represent the response value and the number of rows in the orthogonal table, respectively. Meanwhile, Y is the combination of HV and IGD, where $\frac{HV}{IGD}$ expresses that the larger value is better, and HV and IGD are the medians for 31 independent runs about the ith test scenario. Y represents the mean of this ratio under N_t test scenarios. All scenarios are chosen to perform the sensitivity verification, so the value of N_t equals 15. Therefore, the results of the orthogonal table are shown in Table 7, and the main effects plot of SN ratios is shown in Figure 5. The SN of No. 6–8 in Table 7 and results in Figure 5 denote that the proposed strategies all significantly contribute to the algorithm's performance and efficiency.

5.2. Discussion

The numerical results have revealed some significant findings. The first finding is that for multi-stage in the M-MTA problem, the performance exhibited by the NSGA-MTA is enormous compared to those of the algorithms proposed in the literature [6, 39]. Tables 4 and 5 can intuitively reflect that IGD, IGD+, and HV of NSGA-MTA are statistically improved by about 55, 73, and 36% compared with NSGA-II, respectively. Besides, it is worth noting that NSGA-MTA is prominently superior to C-

Table 6. Factors and levels used in the Taguchi method.

Factors	Strategies	Level 1	Level 2
A	Initialization	Random	Rule-based
B	Crossover	PMX	Hybrid Crossover [1]
C	Mutation	Inverse Mutation	Hybrid Mutation [2]

[1] Proposed crossover with greedy mechanism.
[2] Proposed mutation with random repair strategy.

Table 7. Results using the Taguchi method.

No.	A	B	C	Y	SN
1	1	1	1	0.00197	-54.1107
2	1	1	2	0.00701	-43.0856
3	1	2	1	0.00726	-42.7813
4	1	2	2	0.00695	-43.1603
5	2	1	1	0.00690	-43.2230
6	2	1	2	0.00714	-42.9260
7	2	2	1	0.00701	-43.0856
8	2	2	2	0.00685	-43.2862

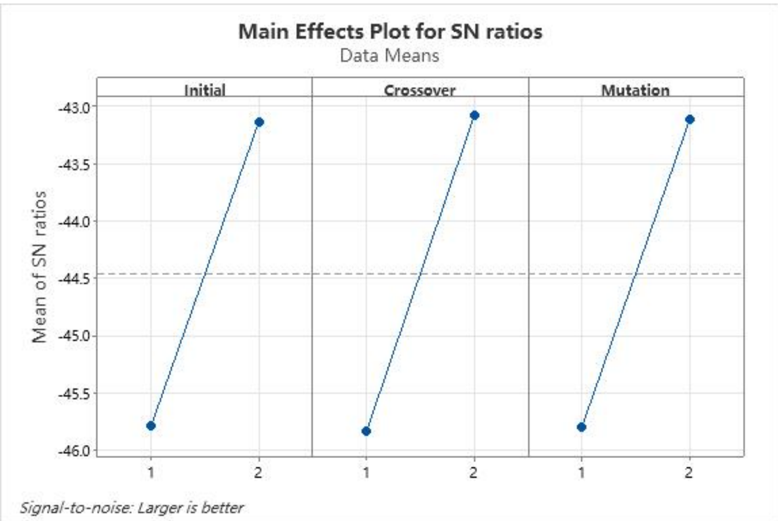


Figure 5. Main effect plot (data means) for SN ratios.

MOEA/D in IGD+ and HV by about 77% and 45%. These results may indicate that the improvements of NSGA-MTA are practical and have a considerable impact on overall performance.

Further analysis, the second finding is that convergence and diversity of NSGA-MTA are well documented via different experimental groups. In the first group ($S_n = 2$), NSGA-MTA is tested for all scale problems, whose metrics and obtained PF show that the proposed greedy mechanism and random repair strategy can play an essential role in obtaining better convergence and diversity of PF. Due to the small search space, rule-based initialization fails to show benefit. However, the initialization has gradually demonstrated its ability to solve large search spaces, with the increasing number of stages in the second group ($S_n \geq 2$). The conclusions from this group found are similar to the literature [15].

In particular, it can be found that the PF obtained by NSGA-MTA in all scenarios has the characteristics of uniform distribution, good convergence, and guaranteed solution diversity compared with others in Figure 4. However, C-TAEA still has a better effect in dealing with medium-scale scenarios, which benefits from the dual archiving mechanism. Besides, C-MOEA/D has a slight advantage in small-scale scenarios depending on the decomposition mechanism.

Therefore, another finding combined with the Taguchi method is that the proposed strategies have meaningful contributions to solving 15 scenarios and are superior to traditional mechanisms. In detail, the proposed initialization mechanism and the random repair strategy provide the direction for maintaining the diversity of the population in medium and large-scale scenarios. This kind of verification has also been applied in the literature [28], and similar results are obtained in this paper.

Lastly, this paper still has some technical areas for improvement. Although some parameters have been fine-tuned, there will still be a phenomenon in which parameter perturbation affects the algorithm's robustness. With the development of data-driven techniques, surrogate models [40] and deep-learning [41] have been successfully applied to real-world combinatorial optimization problems. The limitation is that the M-MTA studied in this paper is challenging to combine with these technologies in terms of coding form and multiple constraints. Therefore, we will work on the fusion of new technologies to solve the constrained M-MTA in the future.

6. Conclusions

In order to describe the conflict between minimizing the survival of targets and minimizing the cost consumption of missiles, a constrained multi-objective optimization version of the M-MTA model is formulated. Furthermore, a multi-objective optimization framework (NSGA-MTA) is proposed to solve this model based on NSGA-II. Specifically, rule-based initialization is developed to provide feasible solutions in advance through prior knowledge to enhance the ability to search for solutions. Based on the characteristics of the M-MTA, genetic operators with the greedy mechanism and random repair strategy not only guarantee the constraints but also maintain the balance between exploration and exploitation in the whole algorithm. Besides, a hybrid encoding mechanism is built as the expression between the model and the algorithm, ensuring that generated solutions meet the proposed constraints. The results of numerical experiments under the Wilcoxon rank-sum test demonstrate that the proposed NSGA-MTA can determine a better optimal allocation scheme than other excellent algorithms to ensure convergence and diversity. Moreover, the results of the Taguchi method further show that the proposed strategies have a promoting and influential role in improving the algorithm's performance.

In this paper, the built model of M-MTA still needs to take the uncertainty into account, resulting in the need for a complete representation of the actual combat environment. Besides, there are still too many parameter settings in the algorithm, which dramatically affects the robustness algorithm. In future work, reinforcement learning based on deep Q-network will be further used to replace the excessive parameter settings in the algorithm to enhance the algorithm's generalization ability. Besides, uncertainty will also be added in building the model.

Acknowledgments

This work was supported by the Major Program of Natural Science Foundation of China (61690210).

Conflict of interest

The authors declare that they have no conflict of interest.

Supplementary

The data generated in the experiment can be available at https://github.com/shiqi0404/M-MTA.

References

1. R. K. Ahuja, A. Kumar, K. C. Jha, J. B. Orlin, Exact and heuristic algorithms for the weapon-target assignment problem, *Oper. Res.*, **55** (2007), 1136–1146. https://doi.org/10.1287/opre.1070.0440
2. W. Wei, R. Yang, H. Gu, W. Zhao, C. Chen, S. Wan, Multi-objective optimization for resource allocation in vehicular cloud computing networks, *IEEE Trans. Intell. Transp. Syst.*, **23** (2021), 25536–25545. https://doi.org/10.1109/TITS.2021.3091321
3. X. Hao, N. Yao, L. Wang, J. Wang, Joint resource allocation algorithm based on multi-objective optimization for wireless sensor networks, *Appl. Soft Comput.*, **94** (2020), 106470. https://doi.org/10.1016/j.asoc.2020.106470
4. O. A. Ogbolumani, N. I. Nwulu, Multi-objective optimisation of constrained food-energy-water-nexus systems for sustainable resource allocation, *Sustainable Energy Technol. Assess.*, **44** (2021), 100967. https://doi.org/10.1016/j.seta.2020.100967
5. S. P. Lloyd, H. S. Witsenhausen, Weapons allocation is np-complete, in *1986 Summer Computer Simulation Conference*, (1986), 1054–1058.
6. J. Li, B. Xin, P. M. Pardalos, J. Chen, Solving bi-objective uncertain stochastic resource allocation problems by the CVaR-based risk measure and decomposition-based multi-objective evolutionary algorithms, *Ann. Oper. Res.*, **296** (2021), 639–666. https://doi.org/10.1007/s10479-019-03435-4
7. P. A. Hosein, M. Athans, *Preferential Defense Strategies*, 1990.
8. B. Xin, J. Chen, J. Zhang, L. Dou, Z. Peng, Efficient decision makings for dynamic weapon-target assignment by virtual permutation and tabu search heuristics, *IEEE Trans. Syst. Man Cyber. C*, **40** (2010), 649–662. https://doi.org/10.1109/TSMCC.2010.2049261

9. X. Shi, S. Zou, S. Song, R. Guo, A multi-objective sparse evolutionary framework for large-scale weapon target assignment based on a reward strategy, *J. Intell. Fuzzy Syst.*, **40** (2021), 10043–10061. https://doi.org/10.3233/JIFS-202679

10. O. Karasakal, Air defense missile-target allocation models for a naval task group, *Comput. Oper. Res.*, **35** (2008), 1759–1770. https://doi.org/10.1016/j.cor.2006.09.011

11. A. G. Kline, D. K. Ahner, B. J. Lunday, Real-time heuristic algorithms for the static weapon target assignment problem, *J. Heuristics*, **25** (2019), 377–397. https://doi.org/10.1007/s10732-018-9401-1

12. A. G. Kline, D. K. Ahner, B. J. Lunday, A heuristic and metaheuristic approach to the static weapon target assignment problem, *J. Global Optim.*, **78** (2020), 791–812. https://doi.org/10.1007/s10898-020-00938-4

13. G. denBroeder Jr, R. Ellison, L. Emerling, On optimum target assignments, *Oper. Res.*, **7** (1959), 322–326. https://doi.org/10.1287/opre.7.3.322

14. Z. J. Lee, C. Y. Lee, S. F. Su, An immunity-based ant colony optimization algorithm for solving weapon–target assignment problem, *Appl. Soft Comput.*, **2** (2002), 39–47. https://doi.org/10.1016/S1568-4946(02)00027-3

15. C. M. Lai, T. H. Wu, Simplified swarm optimization with initialization scheme for dynamic weapon–target assignment problem, *Appl. Soft Comput.*, **82** (2019), 105542. https://doi.org/10.1016/j.asoc.2019.105542

16. Z. J. Lee, S. F. Su, C. Y. Lee, Efficiently solving general weapon-target assignment problem by genetic algorithms with greedy eugenics, *IEEE Trans. Syst. Man Cyber. B*, **33** (2003), 113–121. https://doi.org/10.1109/TSMCB.2003.808174

17. T. Chang, D. Kong, N. Hao, K. Xu, G. Yang, Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization, *Appl. Soft Comput.*, **70** (2018), 845–863. https://doi.org/10.1016/j.asoc.2018.06.014

18. L. Juan, C. Jie, X. Bin, Efficiently solving multi-objective dynamic weapon-target assignment problems by NSGA-II, in *2015 34th Chinese Control Conference (CCC)*, (2015), 2556–2561. https://doi.org/10.1109/ChiCC.2015.7260033

19. J. Li, J. Chen, B. Xin, L. Dou, Solving multi-objective multi-stage weapon target assignment problem via adaptive NSGA-II and adaptive MOEA/D: A comparison study, in *2015 IEEE Congress on Evolutionary Computation (CEC)*, (2015), 3132–3139. https://doi.org/10.1109/CEC.2015.7257280

20. K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, **6** (2002), 182–197. https://doi.org/10.1109/4235.996017

21. Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, *IEEE Trans. Evol. Comput.*, **11** (2007), 712–731. https://doi.org/10.1109/TEVC.2007.892759

22. N. Srinivas, K. Deb, Muiltiobjective optimization using nondominated sorting in genetic algorithms, *Evol. Comput.*, **2** (1994), 221–248. https://doi.org/10.1162/evco.1994.2.3.221

23. F. Sarro, F. Ferrucci, M. Harman, A. Manna, J. Ren, Adaptive multi-objective evolutionary algorithms for overtime planning in software projects, *IEEE Trans. Software Eng.*, **43** (2017), 898–917. https://doi.org/10.1109/TSE.2017.2650914

24. P. Arumugam, E. Amankwah, A. Walker, C. Gerada, Design optimization of a short-term duty electrical machine for extreme environment, *IEEE Trans. Ind. Electron.*, **64** (2017), 9784–9794. https://doi.org/10.1109/TIE.2017.2711555

25. J. Zhou, J. Sun, X. Zhou, T. Wei, M. Chen, S. Hu, et al., Resource management for improving soft-error and lifetime reliability of real-time mpsocs, *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, **38** (2019), 2215–2228. https://doi.org/10.1109/TCAD.2018.2883993

26. X. Zhang, Y. Tian, R. Cheng, Y. Jin, An efficient approach to nondominated sorting for evolutionary multiobjective optimization, *IEEE Trans. Evol. Comput.*, **19** (2014), 201–213. https://doi.org/10.1109/TEVC.2014.2308305

27. J. Chen, B. Xin, Z. Peng, L. Dou, J. Zhang, Evolutionary decision-makings for the dynamic weapon-target assignment problem, *Sci. China Ser. F Inf. Sci.*, **52** (2009), 2006. https://doi.org/10.1007/s11432-009-0190-x

28. Y. Wang, B. Xin, J. Chen, An adaptive memetic algorithm for the joint allocation of heterogeneous stochastic resources, *IEEE Trans. Cybern.*, **52** (2021), 11526–11538. https://doi.org/10.1109/TCYB.2021.3087363

29. K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints, *IEEE Trans. Evol. Comput.*, **18** (2013), 577–601. https://doi.org/10.1109/TEVC.2013.2281535

30. K. Li, R. Chen, G. Fu, X. Yao, Two-archive evolutionary algorithm for constrained multiobjective optimization, *IEEE Trans. Evol. Comput.*, **23** (2018), 303–315. https://doi.org/10.1109/TEVC.2018.2855411

31. H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach, *IEEE Trans. Evol. Comput.*, **18** (2013), 602–622. https://doi.org/10.1109/TEVC.2013.2281534

32. A. A. Ghanbari, H. Alaei, Meta-heuristic algorithms for resource management in crisis based on OWA approach, *Appl. Intell.*, **51** (2021), 646–657. https://doi.org/10.1007/s10489-020-01808-y

33. X. Wu, C. Chen, S. Ding, A modified moea/d algorithm for solving bi-objective multi-stage weapon-target assignment problem, *IEEE Access*, **9** (2021), 71832–71848. https://doi.org/10.1109/ACCESS.2021.3079152

34. C. A. C. Coello, N. C. Cortés, Solving multiobjective optimization problems using an artificial immune system, *Genet. Program. Evol. Mach.*, **6** (2005), 163–190. https://doi.org/10.1007/s10710-005-6164-x

35. H. Ishibuchi, H. Masuda, Y. Nojima, Sensitivity of performance evaluation results by inverted generational distance to reference points, in *2016 IEEE Congress on Evolutionary Computation (CEC)*, (2016), 1107–1114. https://doi.org/10.1109/CEC.2016.7743912

36. E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach, *IEEE Trans. Evol. Comput.*, **3** (1999), 257–271. https://doi.org/10.1109/4235.797969

37. K. Deb, S. Jain, Running performance metrics for evolutionary multi-objective optimization, in *Proceedings of the Fourth Asia-Pacific Conference on Simulated Evolution and Learning (SEAL'02),(Singapore)*, (2002), 13–20. https://doi.org/10.1016/S1350-4789(02)80143-X

38. M. Srinivas, L. M. Patnaik, Adaptive probabilities of crossover and mutation in genetic algorithms, *IEEE Trans. Syst. Man Cybern.*, **24** (1994), 656–667. https://doi.org/10.1109/21.286385

39. W. Xu, C. Chen, S. Ding, P. M. Pardalos, A bi-objective dynamic collaborative task assignment under uncertainty using modified MOEA/D with heuristic initialization, *Expert Syst. Appl.*, **140** (2020), 112844. https://doi.org/10.1016/j.eswa.2019.112844

40. I. Voutchkov, A. Keane, A. Bhaskar, T. M. Olsen, Weld sequence optimization: The use of surrogate models for solving sequential combinatorial problems, *Comput. Methods Appl. Mech. Eng.*, **194** (2005), 3535–3551. https://doi.org/10.1155/IMRN.2005.3551

41. W. Luo, J. Lü, K. Liu, L. Chen, Learning-based policy optimization for adversarial missile-target assignment, *IEEE Trans. Syst. Man Cybern. Syst.*, **52** (2021), 4426–4437. https://doi.org/10.1109/TSMC.2021.3096997