

基于自适应禁忌搜索多目标鲸鱼算法的武器目标分配

宰光军^{1,2,3}, 徐旺旺^{1,3}, 钟李红^{1,2,3}, 田钊^{1,3}, 余维^{1,2,3}

(1. 郑州大学 网络空间安全学院 河南 郑州 450002; 2. 嵩山实验室 河南 郑州 450046;
3. 郑州市区块链与数据智能重点实验室 河南 郑州 450053)

摘要: 针对多目标鲸鱼优化算法在解决武器目标分配时存在参数设置经验化、种群多样性差以及空间搜索能力弱等问题,提出一种自适应禁忌搜索多目标鲸鱼优化算法。首先,通过自适应网格划分和外部存档调整策略,使网格和档案大小能够根据种群分布状态和多样性变化情况自动调整。其次,设计了动态轮盘赌选择方法来控制全局最优个体的生成,以提高种群分布的多样性和均匀性。此外,引入了禁忌搜索算法中的禁忌列表和邻域搜索策略,扩大种群对新区域的探索能力。仿真实验结果表明,所提算法在种群分布性和解集多样性方面表现更优,同时具有更快的求解效率,有效提高了解集的质量,能够较好地解决多目标武器分配优化问题。

关键词: 多目标鲸鱼优化算法; 武器目标分配; 自适应网格划分; 外部存档; 禁忌搜索算法

中图分类号: TP181

文献标志码: A

文章编号: 1671-6841(2026)02-0055-09

DOI: 10.13705/j.issn.1671-6841.2024156

Weapon Target Assignment Based on Adaptive Tabu Search Multi-objective Whale Optimization Algorithm

ZAI Guangjun^{1,2,3}, XU Wangwang^{1,3}, ZHONG Lihong^{1,2,3}, TIAN Zhao^{1,3}, SHE Wei^{1,2,3}

(1. School of Cyber Science and Engineering, Zhengzhou University, Zhengzhou 450002, China;
2. Songshan Laboratory, Zhengzhou 450046, China;
3. Zhengzhou Key Laboratory of Blockchain and Data Intelligence, Zhengzhou 450053, China)

Abstract: Aiming at the problems of empirical parameter setting, insufficient population diversity, and weak searchability of the multi-objective whale optimization algorithm for solving the weapon target allocation problem, an adaptive tabu search multi-objective whale optimization algorithm was proposed. Firstly, adaptive grid partitioning and external archiving adjustment strategies were adopted to allow the grid and archive sizes to be automatically adjusted based on population distribution and diversity changes. Secondly, a dynamic roulette wheel selection method was designed to control the generation of global optimal individuals, thereby enhancing diversity and uniformity of population distribution. Additionally, the tabu list and neighborhood search strategies from the tabu search algorithm were introduced to expand the population's exploration capability in new areas. Simulation experiment results demonstrated that the proposed algorithm exhibited superior population distribution and solution set diversity along with faster-solving efficiency, effectively improved quality of the solution set, and could better solve the multi-objective weapon allocation optimization problems.

Key words: multi-objective whale optimization algorithm; weapon target assignment; adaptive grid partitioning; external archiving; tabu search algorithm

收稿日期: 2024-09-15

基金项目: 国家重点研发计划项目(31703-3); 河南省重点研发与推广专项(212102310039); 郑州大学产学研用科技创新团队培育项目(32214110-23)

第一作者: 宰光军(1979—), 男, 副教授, 主要从事复杂系统建模与仿真、区块链和数据智能研究, E-mail: zaiguangjun@zzu.edu.cn。

通信作者: 余维(1977—), 男, 教授, 主要从事复杂系统建模与仿真、机器学习、区块链和数据智能研究, E-mail: wshe@zzu.edu.cn。

0 引言

武器目标分配(weapon target assignment, WTA)问题是作战指挥与控制决策领域的研究热点^[1],其核心目标在于如何合理地分配有限的武器资源,旨在以最小化弹药成本消耗实现对敌方目标的最大杀伤,是一种典型的组合优化问题^[2]。

目前,关于组合优化问题的求解主要包括基本数学算法和智能优化算法^[3]。基本数学算法有隐枚举法、分支定界法、动态规划法等,其可以求得问题的精确解,但应对大规模优化问题时计算复杂度较高;智能优化算法有遗传算法、蚁群算法、粒子群算法等,其具有较好的全局优化能力和并行处理优势,可以有效解决复杂的组合优化问题。因此,使用智能优化算法是一种有效解决 WTA 问题的主流方法。例如,文献[4]提出一种基于粒子群遗传禁忌的 WTA 优化算法,通过自适应调控种群多样性和改进遗传算法的交叉、变异算子及禁忌搜索的邻域动作对 WTA 问题进行求解。文献[5]提出一种双目标 WTA 问题的蚁群优化方法,并通过改进的蚁群算法获得 Pareto 最优解集对 WTA 问题进行求解。

鲸鱼优化算法作为一种智能优化算法,被广泛应用于 WTA 问题的求解^[6]。文献[7]采用改进多目标鲸鱼优化算法,通过加入交叉和变异机制增加种群多样性以提升算法性能。文献[8]提出一种非支配排序的鲸鱼优化算法,通过拥挤度计算策略和狩猎行为来更新非支配解集,并利用优化算法对 WTA 问题进行求解。文献[9]则引入反向精英保留和种群引导策略加快收敛速度,通过 Levy 变异策略提升算法全局搜索能力。

然而,这些改进的鲸鱼优化算法在实际求解 WTA 问题时存在以下不足:① 求解过程中需要依赖专家经验提前设置网格划分和外部存档大小,而设置不同参数会显著影响解的质量和计算时间。② 在选择 Pareto 最优解时过于随机,存在解集分布不均匀、种群多样性差以及容易陷入局部最优的问题。③ 在迭代更新中经常出现相同解被多次搜寻的情况,存在搜索效率低、空间探索能力较弱的问题。

针对上述问题,本文提出一种自适应禁忌搜索多目标鲸鱼优化算法(adaptive tabu search multi-objective whale optimization algorithm, A-TSMOWOA)用于求解多目标武器分配问题。该算法在以下三个方面进行了改进:

1) 通过自适应调整网格划分和外部存档大小,避免固定参数导致的人为影响。

2) 通过设计动态轮盘赌选择方法,提升种群的多样性和均匀性。

3) 通过引入禁忌列表和邻域搜索策略,增强算法的探索能力和避免陷入局部最优。

1 鲸鱼优化算法的改进

1.1 基本鲸鱼优化算法

鲸鱼优化算法是一种元启发式算法,其主要由搜索猎物、包围猎物和气泡网攻击三个阶段构成。

1) 搜索猎物

算法初期,鲸鱼个体根据自身的位置和周围环境的适应度来搜寻猎物的潜在位置。具体表示为

$$\begin{aligned} X(t+1) &= X_r(t) - A \cdot D, \\ D &= |C \cdot X_r(t) - X(t)|, \end{aligned} \quad (1)$$

其中: t 为当前迭代次数; $X_r(t)$ 为种群随机选择的鲸鱼位置; $X(t)$ 为当前鲸鱼位置; $X(t+1)$ 为更新后的鲸鱼位置; A 和 C 为系数向量; D 为鲸鱼个体与最优解之间的距离。

2) 包围猎物

算法中期,鲸鱼个体会向猎物位置(较优解)包围游去。在这个过程中,鲸鱼会在水下形成一个气泡网,然后快速游到水面将猎物捕获。具体表示为

$$\begin{aligned} X(t+1) &= X_b(t) - A \cdot D, \\ D &= |C \cdot X_b(t) - X(t)|, \end{aligned} \quad (2)$$

其中: $X_b(t)$ 为目前得到的最优鲸鱼位置。

3) 气泡网攻击

算法后期,鲸鱼会采取一种特定的觅食行为将猎物捕获,称为气泡网攻击。具体表示为

$$\begin{aligned} X(t+1) &= X_b(t) + D \cdot e^{bl} \cos(2\pi l), \\ D &= |X_b(t) - X(t)|, \end{aligned} \quad (3)$$

其中: b 是定义螺旋形状的常数; l 为区间 $(-1, 1)$ 上的随机浮点数。

1.2 改进鲸鱼优化算法

为解决多目标鲸鱼优化算法在求解 WTA 问题时遇到的参数设置经验化、种群多样性不足和空间搜索能力弱的问题,提出了三种算法改进策略。

1.2.1 自适应网格划分与外部存档调整策略 传统网格划分方法通常采用固定数值对目标维度进行划分,存在种群分布不均和多样性不足的问题^[10]。因此,设计了一种自适应网格划分策略,依据种群在目标空间的分布情况动态调整网格划分大小,以确保种群的多样性和分布均匀度。假设对有 r 个目标

的优化问题进行网格划分,网格的切分方式满足以下公式:

$$dist_i = \frac{F_{\max}^{(i)} - F_{\min}^{(i)}}{G_i}, \quad (4)$$

其中: $dist_i$ 为第 i 个目标空间的网格划分距离; $F_{\max}^{(i)}$ 和 $F_{\min}^{(i)}$ 分别为第 i 个目标空间的适应度函数的最大值和最小值; G_i 为网格划分数量。

为了解决固定网格划分的不足,结合经济学中的基尼系数^[11],根据基尼系数 ($Gini$) 的值将 $dist_i$ 的取值划分为

$$dist_i = \begin{cases} \text{mean}([dist]), Gini \leq 0.4, \\ \frac{\text{mean}([dist]) + \text{media}([dist])}{2}, 0.4 < Gini < 0.8, \\ \text{mean}([dist^*]), 0.8 \leq Gini, \end{cases} \quad (5)$$

其中: $[dist]$ 表示第 i 个目标空间上相邻两个个体距离组成的数组; mean 表示数组均值; media 表示数组中位数; $[dist^*]$ 为对 $[dist]$ 截取前 $\theta \in (0, 1)$ 取整后的距离数组。拟合数组 $[dist]$ 的洛伦兹曲线,求取曲线中切线值最接近 1 的切点,切点所在位置的 x 即为 θ 的取值。

此外,针对算法求解过程中外部存档规模需要依赖专家经验设置的问题,提出了一种自适应外部存档调整策略^[12]。在传统策略的基础上改进了存档的最优解保留机制,根据存档空间中解的质量和多样性动态调整存档大小,具体包括以下步骤。

1) 去重操作。对于外部存档中存在相同决策变量的解决方案进行去重,只保留一个解决方案。

2) 保留冗余方案。为了增加种群个体的探索能力,保留一定比例决策变量不同但目标函数值相同的解决方案。

3) 网格密度删减。根据外部存档中网格密度分布情况,对网格密度较高处的冗余解方案进行删减操作,以保持存档的多样性和均匀性。

4) 外部存档阈值的设置与更新。根据外部存档中解决方案的数量以及冗余方案的数量,调整更新外部存档阈值大小。

1.2.2 动态轮盘赌选择方法 轮盘赌选择法^[13]是算法求解过程中的一种个体随机选择方法。传统轮盘赌选择法容易受到种群分布密度的影响,导致最优解集均匀性较差。因此,设计了一种动态轮盘赌选择方法,通过动态调整个体被选中的概率,以确保解集的多样性和均匀性。在算法迭代初期,根据外部档案划分计算网格密度分布,采用动态轮盘赌从

外部档案中选择鲸鱼个体的最优解。在后期阶段,由于种群个体已具备一定的多样性,将全局网格密度的动态轮盘赌策略改为在个体邻域内进行筛选,这一过程可表示为

$$Pgd(i) = \begin{cases} \frac{1}{Grid_i} / \sum_j \frac{1}{Grid_j}, t = 0, \\ Pgd_{t-1}(i) + \frac{Pgd_{t-1}(k)}{2 \cdot N_{Grid}^t}, t > 0, i \neq k, \\ \frac{1}{2} Pgd_{t-1}(i) + \frac{Pgd_{t-1}(k)}{2 \cdot N_{Grid}^t}, t > 0, i = k, \end{cases} \quad (6)$$

$$p(i) = \frac{Pgd(i)}{Grid_i}, \quad (7)$$

其中: $Pgd(i)$ 表示网格 i 选中最优个体的概率; $Grid_i$ 表示网格 i 中个体数量; N_{Grid} 表示网格数目; k 表示第 $t-1$ 次选中最优个体的网格编号; $p(i)$ 表示网格 i 中个体被均匀选中的概率。

1.2.3 禁忌搜索算法 禁忌搜索是一种启发式算法^[14],其通过维护禁忌列表避免重复搜索已经存在的解,可以防止算法陷入局部最优。因此,为增强算法的全局搜索能力,引入禁忌搜索算法中的禁忌列表和邻域搜索策略。禁忌列表用来记录已经搜索到的解,避免重复探索到相似解;邻域搜索则帮助算法在求解空间中寻找新解,避免算法陷入局部最优。在迭代过程中,算法会根据当前种群状态和目标函数值选择最优解,依据禁忌列表的记录决定是否接受新解,若接受则更新种群状态,反之会继续通过邻域搜索方法搜索新解。

2 A-TSMOWOA 算法求解多目标武器分配问题

2.1 多目标武器分配数学模型

假设某次军事作战行动中,我方现有 m 种可用弹药资源,其库存数量为 $Num_i (i = 1, 2, \dots, m)$, 成本为 $v_i (i = 1, 2, \dots, m)$ 。对敌方 n 个地面工事目标进行火力打击,每个目标的威胁度系数为 $\omega_j (j = 1, 2, \dots, n)$, 第 i 种弹药对第 j 个地面工事目标的毁伤概率为 $p_{ij} (i = 1, 2, \dots, m; j = 1, 2, \dots, n)$, 其中 $p_{ij} \in (0, 1)$, 则多目标武器分配的决策方案为

$$\mathbf{X} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}, \quad (8)$$

其中: x_{ij} 表示第 i 种弹药对第 j 个地面工事目标的弹药发射量; c_i 表示为第 i 种弹药的总耗弹量。在实际场景中, 弹药消耗总数不超过该弹药库存数量, 因此 x_{ij} 同 c_i 的关系及限制条件可表示为

$$c_i = \sum_{j=1}^n x_{ij}, x_{ij} \in \mathbf{N}, \quad (9)$$

$$0 \leq x_{ij} \leq c_i \leq \text{Num}_i, \forall i \in \{1, 2, \dots, m\}. \quad (10)$$

根据上述武器分配决策方案 \mathbf{X} , 以最小化弹药成本消耗实现对敌方工事造成最大杀伤为目标^[15], 优化目标函数设计为

$$\begin{cases} \min F_1(\mathbf{X}) = \sum_{j=1}^n \omega_j \cdot \prod_{i=1}^m (1 - p_{ij})^{x_{ij}}, \\ \min F_2(\mathbf{X}) = \sum_{i=1}^m \sum_{j=1}^n x_{ij} * v_i. \end{cases} \quad (11)$$

2.2 编码

为了直观反映武器目标火力分配方案, 采用十进制编码规则^[16]。多目标武器分配的决策方案集合是一个由式(8)表示的矩阵, 假设弹药种类为 3 种, 敌方地面工事目标为 4 个, 则某种群个体编码为

$$\mathbf{X} = \begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix}。$$

具体来说, 其火力决策方案为: 对于第 1 种弹药, 分配 1 枚给第 2 个地面工事, 分配 2 枚给第 3 个地面工事; 对于第 2 种弹药, 分配 1 枚给第 1 个地面工事, 分配 1 枚给第 3 个地面工事; 其他弹药分配以此类推。

2.3 种群初始化

算法初始阶段会随机选择每种弹药的消耗量, 并根据这些消耗量生成相应的目标分配方案, 然后将各种弹药的分配方案组合成鲸鱼个体的编码方案 \mathbf{X} 。其中, 鲸鱼个体初始化的编码方案 \mathbf{X} 必须同时满足式(9)和式(10)中规定的约束条件。任何不符合约束条件的编码方案都将被视为不合理, 需要重新生成, 直至满足所有约束条件为止。这一过程确保了算法在初始化阶段即具备有效性和合理性。

2.4 改进算法求解多目标武器分配问题

A-TSMOWOA 算法处理多目标武器分配问题的基本步骤如下。

Step 1 设置鲸鱼种群大小 N_p , 鲸鱼种群初始位置 Init_P , 迭代次数 t 和最大迭代次数 t_{\max} , 设置禁忌表存储容量 Tabu_size 和禁忌表粒子生存代数 RL 。

Step 2 计算种群 P_t 中鲸鱼个体的各个目标函

数值 (F_1, F_2) , 并进行非支配排序, 计算拥挤度距离。

Step 3 根据 Step 2 中的非支配排序等级, 挑选出种群 P_t 中的 Pareto 前沿个体 PF_t , 将其保存到外部存档空间 Archive 中, 使 PF_t 同 Archive_{t-1} 组合生成新的 Archive_t 。如果外部存档超出存储阈值, 采用自适应网格划分策略删除密度大的个体, 重新计算网格密度直至满足存储阈值。

Step 4 根据外部存档空间 Archive 中的鲸鱼种群网格密度分布, 使用动态轮盘赌选择策略为每个鲸鱼个体筛选一个最优位置。

Step 5 将种群 P_t 存储到禁忌表 Tabu_list 中, 设置新种群中鲸鱼个体在 Tabu_list 的生命周期 cy 为 RL 。对 Tabu_list 中所有个体生命周期减 1, 并移除 $cy \leq 0$ 的个体。检查禁忌表 Tabu_list 是否超出存储容量 Tabu_size , 必要时进行删除以满足容量要求。

Step 6 依据鲸鱼优化算法的基本思想, 控制每个鲸鱼个体向最优位置更新, 并生成新种群 $P_t (0 < t \leq t_{\max})$ 。若新产生的个体在禁忌表中, 则在其邻域内搜索并选取最优方案替换。

Step 7 判断是否达到最大迭代次数, 是则跳出循环并输出最优解值和最优鲸鱼个体的位置信息, 否则返回 Step 2 继续执行。

3 实验及分析

3.1 实验环境

实验环境描述如下: CPU 为 AMD Ryzen 7 5800H, GPU 为 NVIDIA GeForce RTX 3060 Laptop, 内存为 8 GB, 操作系统选用 Windows 10, 编译语言为 Python3.7。

3.2 实验设置

为了验证本文算法 (A-TSMOWOA) 在求解 WTA 问题中的优化性能, 根据武器和目标的数量设置了不同规模的战斗场景, 包括 4W-4T、4W-5T、6W-5T、6W-6T、7W-7T 和 8W-8T, 并针对每种情况生成毁伤概率矩阵、目标威胁系数、弹药成本和弹药库存, 其中矩阵元素为随机生成。此外, 将每种组合方案进行了 20 次独立重复实验。不同 WTA 场景下的实验数值见表 1。

3.3 评价指标

多目标优化问题涉及多个决策变量和目标函数, 目的是找到一组解, 使其在各个目标函数方面达到最优或接近最优。通过以下三项评价指标对模

表 1 不同 WTA 场景下的实验数值
Table 1 Experimental data with different WTA scenarios

场景	毁伤概率矩阵	目标威胁系数	弹药成本	弹药库存
4W-4T	$\begin{bmatrix} 0.701 & 0.405 & 0.812 & 0.569 \\ 0.625 & 0.210 & 0.717 & 0.634 \\ 0.510 & 0.725 & 0.682 & 0.569 \\ 0.482 & 0.717 & 0.224 & 0.701 \end{bmatrix}$	$\begin{bmatrix} 0.114 \\ 0.354 \\ 0.207 \\ 0.325 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \end{bmatrix}$
4W-5T	$\begin{bmatrix} 0.80 & 0.75 & 0.82 & 0.69 & 0.75 \\ 0.75 & 0.70 & 0.77 & 0.64 & 0.30 \\ 0.50 & 0.75 & 0.82 & 0.69 & 0.75 \\ 0.82 & 0.77 & 0.24 & 0.71 & 0.77 \end{bmatrix}$	$\begin{bmatrix} 0.186 \\ 0.296 \\ 0.164 \\ 0.179 \\ 0.175 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \end{bmatrix}$
6W-5T	$\begin{bmatrix} 0.327 & 0.653 & 0.276 & 0.198 & 0.753 \\ 0.565 & 0.410 & 0.417 & 0.324 & 0.540 \\ 0.610 & 0.505 & 0.182 & 0.569 & 0.175 \\ 0.282 & 0.677 & 0.424 & 0.481 & 0.617 \\ 0.520 & 0.355 & 0.102 & 0.869 & 0.075 \\ 0.322 & 0.417 & 0.584 & 0.401 & 0.275 \end{bmatrix}$	$\begin{bmatrix} 0.170 \\ 0.188 \\ 0.150 \\ 0.330 \\ 0.312 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \\ 0.60 \\ 0.28 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \\ 2 \\ 7 \end{bmatrix}$
6W-6T	$\begin{bmatrix} 0.447 & 0.753 & 0.476 & 0.698 & 0.753 & 0.770 \\ 0.550 & 0.470 & 0.717 & 0.624 & 0.340 & 0.580 \\ 0.510 & 0.705 & 0.682 & 0.369 & 0.675 & 0.365 \\ 0.482 & 0.677 & 0.324 & 0.471 & 0.677 & 0.248 \\ 0.580 & 0.305 & 0.602 & 0.569 & 0.675 & 0.465 \\ 0.512 & 0.677 & 0.384 & 0.401 & 0.475 & 0.398 \end{bmatrix}$	$\begin{bmatrix} 0.156 \\ 0.171 \\ 0.213 \\ 0.081 \\ 0.145 \\ 0.234 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \\ 0.60 \\ 0.28 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \\ 2 \\ 7 \end{bmatrix}$
7W-7T	$\begin{bmatrix} 0.447 & 0.753 & 0.476 & 0.698 & 0.753 & 0.770 & 0.216 \\ 0.550 & 0.470 & 0.717 & 0.624 & 0.340 & 0.580 & 0.126 \\ 0.510 & 0.705 & 0.682 & 0.369 & 0.675 & 0.365 & 0.761 \\ 0.482 & 0.677 & 0.324 & 0.471 & 0.677 & 0.248 & 0.381 \\ 0.580 & 0.305 & 0.602 & 0.569 & 0.675 & 0.465 & 0.102 \\ 0.512 & 0.677 & 0.384 & 0.401 & 0.475 & 0.398 & 0.124 \\ 0.272 & 0.413 & 0.681 & 0.741 & 0.376 & 0.518 & 0.354 \end{bmatrix}$	$\begin{bmatrix} 0.140 \\ 0.154 \\ 0.123 \\ 0.134 \\ 0.130 \\ 0.208 \\ 0.111 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \\ 0.90 \\ 0.20 \\ 0.80 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \\ 7 \\ 8 \\ 9 \end{bmatrix}$
8W-8T	$\begin{bmatrix} 0.447 & 0.753 & 0.476 & 0.698 & 0.753 & 0.770 & 0.216 & 0.641 \\ 0.550 & 0.470 & 0.717 & 0.624 & 0.340 & 0.580 & 0.126 & 0.144 \\ 0.510 & 0.705 & 0.682 & 0.369 & 0.675 & 0.365 & 0.761 & 0.264 \\ 0.482 & 0.677 & 0.324 & 0.471 & 0.677 & 0.248 & 0.381 & 0.714 \\ 0.580 & 0.305 & 0.602 & 0.569 & 0.675 & 0.465 & 0.102 & 0.218 \\ 0.512 & 0.677 & 0.384 & 0.401 & 0.475 & 0.398 & 0.124 & 0.556 \\ 0.272 & 0.413 & 0.681 & 0.741 & 0.376 & 0.518 & 0.354 & 0.453 \\ 0.712 & 0.277 & 0.304 & 0.491 & 0.536 & 0.420 & 0.324 & 0.251 \end{bmatrix}$	$\begin{bmatrix} 0.102 \\ 0.113 \\ 0.090 \\ 0.098 \\ 0.095 \\ 0.152 \\ 0.201 \\ 0.149 \end{bmatrix}$	$\begin{bmatrix} 0.50 \\ 0.70 \\ 0.30 \\ 0.40 \\ 0.90 \\ 0.20 \\ 0.80 \\ 0.48 \end{bmatrix}$	$\begin{bmatrix} 5 \\ 7 \\ 4 \\ 8 \\ 7 \\ 8 \\ 9 \\ 12 \end{bmatrix}$

型进行综合性评估。

1) 解集覆盖率

解集覆盖率 (*C-metric*) 用于衡量近似 Pareto 前沿与真实 Pareto 前沿之间的覆盖程度。其值越大, 表示最优解数量越多。具体表达式为

$$C\text{-metric}(P, P^*) = \frac{|\{x \in P^* \mid \exists y \in P: y < x\}|}{|P^*|}, \quad (12)$$

其中: P^* 表示真实 Pareto 前沿的解集; P 表示近似 Pareto 前沿的解集; $|\cdot|$ 表示其解集满足条件的数目; $y < x$ 表示 y 在求解目标维度上支配 x 。

2) 反世代距离评价指标

反世代距离评价指标 (*IGD*) 是计算真实 Pareto 前沿上的每个点到算法解集的最小距离和。其值越小, 表示算法的收敛性能和分布性能越好。具体表达式为

$$IGD(X, P^*) = \frac{\sum_{i=1}^{X_number} \min d(x_i, X)}{X_number}, \quad (13)$$

其中: P^* 为真实 Pareto 前沿分布解集; X 为求解得到的 Pareto 前沿分布解集; x_i 表示 X 上的一个解; $\min d(x_i, X)$ 是 x_i 到 P^* 上所有点的最小欧氏距离;

X_number 为 X 的解集数量。

3) 间距指标

间距指标 (*Spacing*) 用于计算最优解集的分布均匀性,度量解集中每个解到其他解的最小距离的标准差。其值越小,表示解分布越均匀。具体表达式为

$$Spacing(P) = \sqrt{\frac{1}{|P| - 1} \sum_{i=1}^{|P|} (\bar{d} - d_i)^2}, \quad (14)$$

其中: $|P|$ 表示种群数目; d_i 表示种群中第 i 个解

到其他解的最小距离; \bar{d} 表示所有 d_i 的均值。

3.4 实验结果对比分析

1) 不同网格划分数结果分析

为验证基本鲸鱼优化算法在解决多目标优化问题时网格划分数对求解 WTA 问题的影响,针对不同武器目标规模和外部存档规模下的网格划分数进行独立实验。设置种群大小为 100, 网格划分数为 10, 20, 30, ..., 290, 300。不同网格划分数的真实前沿覆盖率对比如图 1 所示。

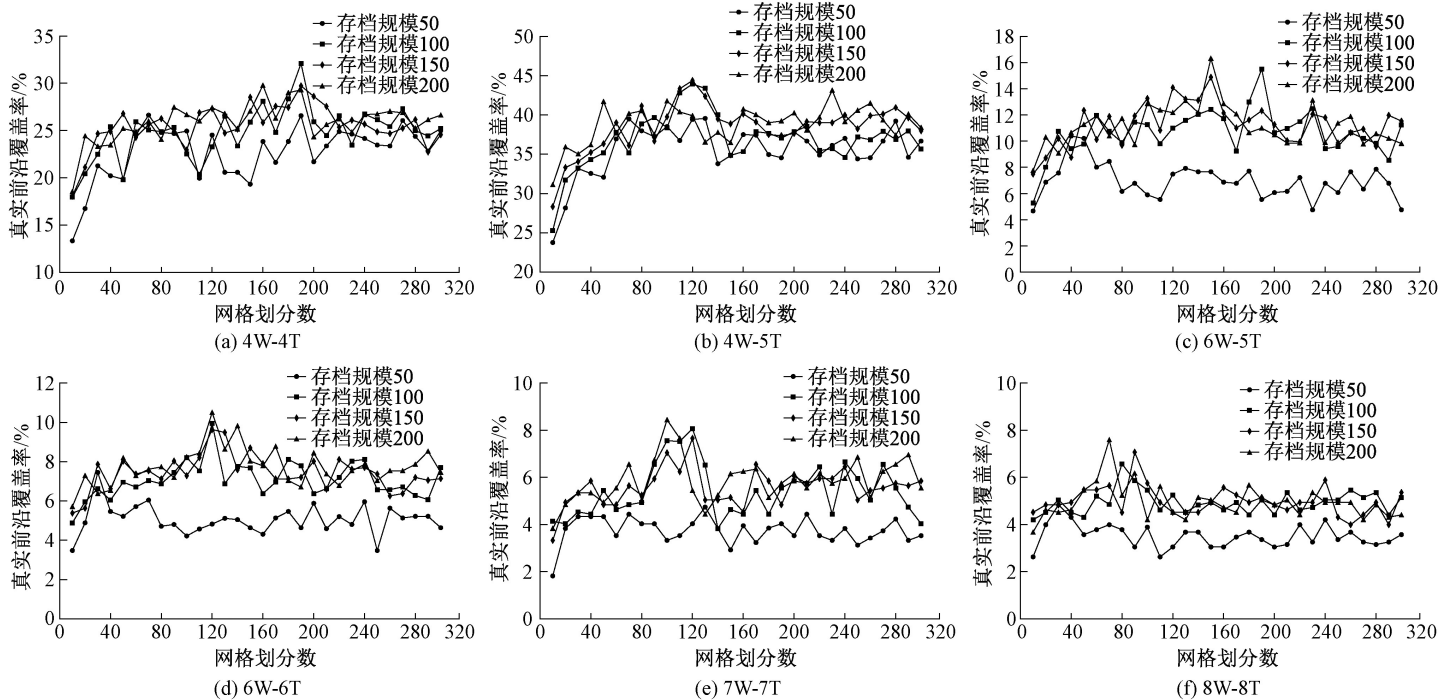


图 1 不同网格划分数的真实前沿覆盖率对比

Figure 1 Comparison of true front coverage rate of different meshing numbers

图 1 显示,网格划分数确实对多目标鲸鱼优化算法求解 WTA 问题的性能存在影响,求解精度会随着网格划分数的变化而发生改变。其中,较小的网格划分数可能导致解空间细粒度不足,使得算法难以在局部空间中深入搜索,从而影响全局最优解的发现。相反,较大的网格划分数可能会增加搜索空间的维度,使得算法的搜索范围过于广泛,从而导致搜索效率下降,甚至陷入局部最优解。

2) 不同存档规模结果分析

为验证基本鲸鱼优化算法在解决多目标优化问题时存档规模对求解 WTA 问题的影响,针对不同武器目标规模下的外部存档规模大小进行独立实验,设置种群大小为 100, 外部存档规模为 50, 100, 150, 200。在实验中对不同外部存档规模所求解到的最优方案采用 $MOWOA_T$ 表示,其中 T 表示外部存档规模大小 ($T = 50, 100, 150, 200$)。不同存档规模的实验结果对比见表 2。

表 2 显示,存档空间的大小对求解结果具有显

著影响。对于较大规模的 WTA 问题,如果存档空间较小,则会由于存档空间限制无法保存更多优质解,算法难以获取覆盖真实前沿的解集。相反,对于小规模 WTA 问题,如果设置的存档空间较大(如 150 和 200)时,算法的求解能力并未明显提升,反而会因为非支配算法的特性而增加求解时间。因此,人为设置外部存档空间的大小对鲸鱼优化算法的求解质量具有较大影响。

3) 消融实验及与其他优化算法对比分析

为验证各部分策略的有效性,对所提出的改进策略采用逐步累加的方式进行对比实验。在不同策略组合方案中,使用 $MOWOA$ 表示基本鲸鱼优化算法,使用 $MOWOA_A$ 表示采用自适应网格划分策略和自适应外部存档调整策略的多目标鲸鱼算法,使用 $MOWOA_{A+B}$ 表示采用自适应网格划分策略、自适应外部存档调整策略和动态轮盘赌选择策略的多目标鲸鱼算法。

首先,为验证自适应网格划分策略和自适应外

表 2 不同存档规模的实验结果对比
Table 2 Comparison of experimental results for different archive sizes

场景	类别	<i>C-metric</i>		<i>IGD</i>		<i>Spacing</i>		运行时间/s
		均值	标准差	均值	标准差	均值	标准差	
4W-4T	MOWOA ₅₀	2.663E-01	1.260E-02	9.852E-02	7.150E-04	9.069E+00	7.337E-02	5.07
	MOWOA ₁₀₀	3.210E-01	1.854E-02	8.917E-02	6.629E-04	8.602E+00	1.029E-01	7.91
	MOWOA ₁₅₀	2.852E-01	1.170E-02	8.691E-02	4.520E-04	8.716E+00	1.190E-01	10.76
	MOWOA ₂₀₀	2.978E-01	1.573E-02	9.492E-02	8.040E-04	8.710E+00	7.512E-02	13.58
4W-5T	MOWOA ₅₀	3.956E-01	9.477E-03	9.170E-02	6.260E-04	9.763E+00	1.315E-01	5.11
	MOWOA ₁₀₀	4.334E-01	1.203E-02	8.316E-02	4.280E-04	9.641E+00	9.088E-02	7.96
	MOWOA ₁₅₀	4.258E-01	1.308E-02	9.784E-02	1.772E-03	9.648E+00	1.060E-01	10.74
	MOWOA ₂₀₀	4.169E-01	1.601E-02	8.981E-02	1.093E-03	9.741E+00	1.431E-01	13.65
6W-5T	MOWOA ₅₀	1.039E-01	1.966E-03	1.856E-01	6.802E-03	1.255E+01	8.334E-02	5.41
	MOWOA ₁₀₀	1.549E-01	4.325E-03	1.537E-01	7.447E-03	1.180E+01	1.823E-01	8.28
	MOWOA ₁₅₀	1.488E-01	1.091E-03	1.978E-01	3.659E-03	1.165E+01	2.025E-01	11.10
	MOWOA ₂₀₀	1.632E-01	2.289E-03	1.476E-01	2.916E-03	1.168E+01	2.996E-01	13.77
6W-6T	MOWOA ₅₀	7.450E-02	1.230E-04	2.545E-01	2.063E-03	1.266E+01	1.072E-01	5.53
	MOWOA ₁₀₀	9.947E-02	7.280E-04	2.253E-01	2.243E-03	1.206E+01	1.531E-01	8.40
	MOWOA ₁₅₀	9.633E-02	4.842E-04	2.284E-01	1.878E-03	1.198E+01	2.227E-01	11.21
	MOWOA ₂₀₀	1.050E-01	4.719E-04	2.148E-01	4.180E-03	1.194E+01	1.747E-01	13.89
7W-7T	MOWOA ₅₀	4.738E-02	3.960E-04	5.441E-01	8.778E-03	1.972E+01	1.627E+00	5.81
	MOWOA ₁₀₀	8.060E-02	4.845E-04	4.297E-01	8.246E-03	1.979E+01	1.059E+00	8.60
	MOWOA ₁₅₀	7.655E-02	6.172E-04	3.919E-01	7.643E-03	1.976E+01	8.804E-01	11.26
	MOWOA ₂₀₀	8.444E-02	3.701E-04	3.645E-01	6.824E-03	1.938E+01	8.603E-01	13.68
8W-8T	MOWOA ₅₀	4.832E-02	2.150E-04	6.191E-01	5.143E-03	2.299E+01	9.233E-01	6.08
	MOWOA ₁₀₀	6.573E-02	1.441E-04	4.542E-01	1.473E-02	2.362E+01	4.877E-01	8.87
	MOWOA ₁₅₀	7.067E-02	7.344E-05	4.385E-01	6.767E-03	2.338E+01	4.304E-01	11.46
	MOWOA ₂₀₀	7.594E-02	9.110E-05	4.255E-01	1.001E-02	2.244E+01	6.841E-01	13.85

注:黑体表示最优值。

部存档调整策略的有效性,对 MOWOA 设置存档规模为 100,采用不同的网格划分数,并从中选取最优情况与 MOWOA_A 进行实验结果对比。其次,为验证动态轮盘赌选择策略的有效性,将 MOWOA_A 和 MOWOA_{A+B} 进行实验结果对比。另外,为验证引入禁忌列表和邻域搜索策略的有效性,将 MOWOA_{A+B} 和本文算法(A-TSMOWOA)进行实验结果对比。最后,为验证所提算法的优越性,与 NSGA-II 算法^[17]和 MOPSO 算法^[18]进行对比实验。消融实验及与其他优化算法的实验结果对比见表 3。

表 3 给出了不同武器目标规模下 MOWOA、MOWOA_A、MOWOA_{A+B}、NSGA-II 和 MOPSO 算法与 A-TSMOWOA 的实验结果对比。结果表明,MOWOA_A 与 MOWOA 相比,其在 *C-metric*、*IGD* 和 *Spacing* 指标上基本相当,但在求解时间上有显著提升。MOWOA_{A+B} 与 MOWOA_A 相比,其在 *IGD* 和 *Spacing* 指标上具有更好的种群分布多样性和均匀性。而 A-TSMOWOA 与 MOWOA_{A+B} 相比,其在 *C-metric* 指标和运行时间上均有明显提升,展现出了

更高的求解质量和效率。具体而言,在相同的迭代次数下,在 4W-4T 问题中,A-TSMOWOA 的 *C-metric* 和 *IGD* 指标分别提升了 103.69%和 43.49%;在 4W-5T 问题中,*C-metric* 和 *IGD* 指标分别提升了 49.15%和 18.61%;在 6W-5T 问题中,*C-metric* 和 *IGD* 指标分别提升了 48.65%和 27.57%;在 6W-6T 问题中,*C-metric* 和 *IGD* 指标分别提升了 51.62%和 36.78%;在 7W-7T 问题中,*C-metric* 和 *IGD* 指标分别提升了 31.81%和 38.79%;在 8W-8T 问题中,*C-metric* 和 *IGD* 指标分别提升了 17.69%和 32.08%。

此外,相较于 NSGA-II 和 MOPSO 算法,A-TSMOWOA 在求解小规模 WTA 问题时,在 *C-metric* 和 *Spacing* 方面略有差异,但在 *IGD* 评价指标上表现优越;对于较大规模 WTA 问题,其在 *C-metric*、*IGD* 和 *Spacing* 指标上均取得了更优结果,具有明显优势。

综合表 3 的实验结果,A-TSMOWOA 算法在解决不同规模 WTA 问题时,在收敛性、多样性和均匀性方面均表现出较高的质量,能够更有效地应对火力分配问题。

表 3 消融实验及与其他优化算法的实验结果对比

Table 3 Comparison of experimental results of ablation experiments and other optimization algorithms

场景	类别	<i>C-metric</i>		<i>IGD</i>		<i>Spacing</i>		运行 时间/s
		均值	标准差	均值	标准差	均值	标准差	
4W-4T	MOWOA	2.978E-01	1.573E-02	9.492E-02	8.040E-04	8.710E+00	7.512E-02	13.58
	MOWOA _A	3.057E-01	1.853E-02	9.018E-02	3.905E-04	8.784E+00	9.769E-02	5.02
	MOWOA _{A+B}	4.608E-01	2.687E-02	6.629E-02	4.191E-04	8.540E+00	2.095E-01	5.12
	NSGA-II	6.312E-01	5.190E-05	6.027E-01	9.250E-04	7.337E+00	3.872E-01	8.95
	MOPSO	6.462E-01	8.488E-03	7.522E-02	1.187E-03	7.701E+00	1.975E-01	9.33
	A-TSMOWOA	6.066E-01	3.271E-02	5.364E-02	5.060E-04	8.470E+00	2.028E-01	4.86
4W-5T	MOWOA	4.169E-01	1.601E-02	8.981E-02	1.093E-03	9.741E+00	1.431E-01	13.65
	MOWOA _A	4.707E-01	1.109E-02	8.774E-02	8.494E-04	9.675E+00	1.347E-01	5.62
	MOWOA _{A+B}	5.307E-01	1.109E-02	5.724E-02	3.674E-04	9.528E+00	1.731E-01	5.75
	NSGA-II	6.154E-01	5.423E-05	6.924E-01	4.290E-04	8.111E+00	1.474E-01	9.60
	MOPSO	6.378E-01	1.537E-02	9.311E-02	3.922E-03	9.013E+00	4.205E-01	10.34
	A-TSMOWOA	6.218E-01	1.899E-02	5.316E-02	3.750E-04	9.591E+00	1.466E-01	4.70
6W-5T	MOWOA	1.632E-01	2.289E-03	1.476E-01	2.916E-03	1.168E+01	2.996E-01	13.77
	MOWOA _A	1.541E-01	2.587E-03	1.471E-01	5.926E-03	1.181E+01	3.437E-01	7.11
	MOWOA _{A+B}	1.581E-01	2.587E-03	1.100E-01	2.468E-03	1.161E+01	2.202E-01	7.26
	NSGA-II	2.053E-01	2.857E-04	1.283E+00	5.760E-03	8.232E+00	6.349E-01	10.02
	MOPSO	1.346E-01	3.456E-03	1.681E-01	1.410E-03	1.058E+01	3.655E-01	11.59
	A-TSMOWOA	2.426E-01	1.395E-02	1.069E-01	2.330E-03	1.159E+01	3.347E-01	6.27
6W-6T	MOWOA	1.050E-01	4.719E-04	2.148E-01	4.180E-03	1.194E+01	1.747E-01	13.89
	MOWOA _A	1.192E-01	7.232E-04	2.313E-01	1.844E-03	1.236E+01	1.524E-01	7.39
	MOWOA _{A+B}	1.247E-01	1.781E-03	1.242E-01	1.949E-03	1.160E+01	1.474E-01	7.63
	NSGA-II	1.535E-01	3.315E-04	7.072E-01	6.322E-03	9.441E+00	1.088E+00	10.57
	MOPSO	8.205E-02	2.964E-03	1.846E-01	3.414E-03	1.085E+01	5.832E-01	11.22
	A-TSMOWOA	1.592E-01	7.230E-04	1.358E-01	1.772E-03	1.154E+01	5.423E-02	6.50
7W-7T	MOWOA	8.444E-02	3.701E-04	3.645E-01	6.824E-03	1.938E+01	8.603E-01	13.68
	MOWOA _A	7.554E-02	2.862E-04	3.891E-01	6.137E-03	1.902E+01	4.138E-01	7.06
	MOWOA _{A+B}	9.048E-02	4.522E-03	2.431E-01	4.881E-03	1.867E+01	7.355E-01	7.32
	NSGA-II	1.613E-02	2.340E-05	1.396E+00	5.532E-03	1.315E+01	1.156E+00	11.67
	MOPSO	3.853E-02	1.368E-03	3.723E-01	8.977E-03	1.705E+01	2.065E+00	12.40
	A-TSMOWOA	1.113E-01	4.522E-03	2.231E-01	4.881E-03	1.864E+01	7.355E-01	6.79
8W-8T	MOWOA	7.594E-02	9.110E-05	4.255E-01	1.001E-02	2.244E+01	6.841E-01	13.85
	MOWOA _A	6.984E-02	6.450E-05	4.237E-01	5.140E-03	2.281E+01	9.956E-01	7.68
	MOWOA _{A+B}	7.110E-02	4.230E-04	3.001E-01	2.079E-03	2.190E+01	7.217E-01	8.19
	NSGA-II	0.000E+00	0.000E+00	2.785E+00	3.091E-02	2.342E+01	4.302E+00	14.15
	MOPSO	2.474E-02	3.038E-03	6.128E-01	2.886E-02	2.487E+01	2.854E+00	15.26
	A-TSMOWOA	8.937E-02	8.140E-04	2.890E-01	2.276E-03	2.188E+01	9.956E-01	7.46

注:黑体表示最优值。

4 结语

本文提出的自适应禁忌搜索多目标鲸鱼优化算法有效解决了传统多目标鲸鱼优化算法在 WTA 中的一些关键问题,包括参数设置经验化、种群多样性不足和空间搜索能力有限等。通过引入自适应网格划分和外部存档调整策略,将动态轮盘赌选择方法

与禁忌搜索算法相结合,所提算法在种群分布、解集多样性和求解效率方面表现出了显著优势,为多目标武器分配优化提供了更为合理的解决方案。然而,当前研究仍局限于静态 WTA 问题,未充分考虑时间因素对分配方案和毁伤打击效果的影响。因此,未来的研究方向将集中在动态 WTA 问题上,以进一步完善多目标武器分配优化算法,从而为实际应用提供更精确有效的解决方案。

参考文献:

- [1] HOCAOĞLU M F. Weapon target assignment optimization for land based multi-air defense systems: a goal programming approach [J]. *Computers & industrial engineering*, 2019, 128: 681–689.
- [2] 李梦杰, 常雪凝, 石建迈, 等. 武器目标分配问题研究进展: 模型、算法与应用 [J]. *系统工程与电子技术*, 2023, 45(4): 1049–1071.
- LI M J, CHANG X N, SHI J M, et al. Developments of weapon target assignment: models, algorithms, and applications [J]. *Systems engineering and electronics*, 2023, 45(4): 1049–1071.
- [3] WANG C H, GAO J Q, LV N B, et al. Multi-objective optimization of weapon target assignment based on genetic algorithm [C]//International Conference on Computer, Internet of Things and Control Engineering. Piscataway: IEEE Press, 2021: 29–34.
- [4] 余维, 牛文涛, 孔德锋, 等. 基于粒子群遗传禁忌的武器目标分配优化算法 [J]. *郑州大学学报(理学版)*, 2023, 55(5): 1–10.
- SHE W, NIU W T, KONG D F, et al. Weapon target assignment optimization algorithm based on particle swarm genetic taboo [J]. *Journal of Zhengzhou university (natural science edition)*, 2023, 55(5): 1–10.
- [5] LI Y, KOU Y X, LI Z W, et al. A modified Pareto ant colony optimization approach to solve biobjective weapon-target assignment problem [J]. *International journal of aerospace engineering*, 2017: 1746124.
- [6] MIRJALILI S, LEWIS A. The whale optimization algorithm [J]. *Advances in engineering software*, 2016, 95: 51–67.
- [7] 邱少明, 刘良成, 杜秀丽, 等. 改进多目标鲸鱼优化算法的武器目标分配研究 [J]. *计算机应用与软件*, 2023, 40(12): 248–254, 276.
- QIU S M, LIU L C, DU X L, et al. Weapon target allocation based on multi-objective whale optimization algorithm [J]. *Computer applications and software*, 2023, 40(12): 248–254, 276.
- [8] JANGIR P. Non-dominated sorting moth flame optimizer: a novel multi-objective optimization algorithm for solving engineering design problems [J]. *Engineering technology open access journal*, 2018, 2(1): 555579.
- [9] 梁倩. 基于反向精英保留和 Levy 变异的多目标鲸鱼优化算法 [J]. *现代计算机*, 2021, 27(18): 25–31.
- LIANG Q. Multi-objective whale optimization algorithm based on reverse elite retention and Levy mutation [J]. *Modern computer*, 2021, 27(18): 25–31.
- [10] LIANG J, QIAO K J, YUE C T, et al. A clustering-based differential evolution algorithm for solving multimodal multi-objective optimization problems [J]. *Swarm and evolutionary computation*, 2021, 60: 100788.
- [11] PETERSSON J. Using the Gini coefficient for assessing heterogeneity within classes and schools [J]. *SN social sciences*, 2023, 3(11): 186.
- [12] KNOWLES J D, CORNE D W. Approximating the non-dominated front using the Pareto archived evolution strategy [J]. *Evolutionary computation*, 2000, 8(2): 149–172.
- [13] PHAM V H S, DANG N T N, NGUYEN V N. Enhancing engineering optimization using hybrid sine cosine algorithm with Roulette wheel selection and opposition-based learning [J]. *Scientific reports*, 2024, 14(1): 694.
- [14] LEE K, OZSEN L. Tabu search heuristic for the network design model with lead time and safety stock considerations [J]. *Computers & industrial engineering*, 2020, 148: 106717.
- [15] GUNANTARA N. A review of multi-objective optimization: methods and its applications [J]. *Cogent engineering*, 2018, 5(1): 1502242.
- [16] LIU C, LI J, WANG Y, et al. A time-driven dynamic weapon target assignment method [J]. *IEEE access*, 2023, 11: 129623–129639.
- [17] KATOCH S, CHAUHAN S S, KUMAR V. A review on genetic algorithm: past, present, and future [J]. *Multi-media tools and applications*, 2021, 80: 8091–8126.
- [18] WANG D S, TAN D P, LIU L. Particle swarm optimization algorithm: an overview [J]. *Soft computing*, 2018, 22(2): 387–408.