中国科学: 技术科学 2024 年 第 54 卷 第 9 期: 1707 ~ 1719
SCIENTIA SINICA Technologica techcn.scichina.com
《中国科学》杂志社 SCIENCE CHINA PRESS
论 文 复杂系统协同优化控制专题

面向防空资源分配优化问题的自适应演化算法

邢立宁[1], 罗天羽[1], 李豪[1], 公茂果[1*], 王锐[2]

1. 西安电子科技大学协同智能系统教育部重点实验室, 西安 710071;
2. 国防科技大学系统工程学院, 长沙 410073
* E-mail: mggong@mail.xidian.edu.cn

收稿日期: 2023-10-25; 接受日期: 2024-01-17; 网络版发表日期: 2024-09-05
陕西省科技创新团队项目(编号: 2023-CX-TD-07)、陕西省重点研发计划项目(编号: 2024GH-ZDXM-48)和湖南省自然科学基金项目(编号: 2024JJ5109、2024JJ7098)资助

摘要 防空资源分配作为作战装备-目标分配问题的一种重要组成部分, 必须不断适应战场的动态变化. 如何在复杂战场环境中协调多作战装备单元、合理分配防空资源, 提出更加灵活和智能的分配策略, 以实现装备系统效能最大化, 成为了当下亟待解决的核心问题. 在综合考虑多部雷达接力跟踪、多枚导弹拦截同一目标和目标优先级等因素的基础上, 构建了防空资源分配数学模型. 将启发式规则的运算效率及定制化优势与演化算法的广域搜索优势相结合, 设计一种新的自适应演化算法对此问题分阶段求解. 为了提升算法的搜索能力与速度, 在迭代中为目标分配首枚导弹的分配方案, 并引入资源释放和重规划的方法提升现有方案质量. 然后, 在迭代后采用启发式规则对已发现的最优方案按需分配多枚导弹. 通过对不同资源与不同规模的算例进行数值仿真实验, 验证了该算法具有求解时间和求解精度上的优越性, 可有效解决防空资源分配问题.

关键词 防空资源分配, 重规划, 自适应演化算法

1 引言

目前, 关于作战装备-目标分配问题的研究得到了广泛的关注. Manne[1]最早提出作战装备-目标分配问题的概念, 并于1986年证明了该问题为NP-Complete问题[2]. 解决这一问题需要综合考虑多个因素, 包括来袭目标的性质、位置、作战装备单元的种类和数量以及各个作战装备单元的性能等. 根据典型作战样式的不同, 作战装备-目标分配问题可分为进攻作战和防御作战[3]. 本文所研究的防空资源分配问题则是一种结合地面资源(雷达和发射车)对空中来袭目标进行拦截的防御作战问题.

针对防空资源分配问题, Kline等人[4]提出了一种启发式和元启发式方法求解静态作战装备资源分配问题. 在Kline等人[4]的基础上, Silav等人[5]提出了一种以作战装备系统接战顺序的不泄密概率最大化和稳定性最大化的双目标模型, 并设计了一种通过交换作战装备交战顺序来最大限度地提高稳定性的解决方法. 为了将量子计算机的计算性能用于提升资源分配速度, Acar等人[6]将量子计算机在解决某些问题时具有指数加速的潜力用于求解作战装备资源分配问题. 随着作战装备的不断升级, 作战装备资源分配问题已发展为

引用格式: 邢立宁, 罗天羽, 李豪, 等. 面向防空资源分配优化问题的自适应演化算法. 中国科学: 技术科学, 2024, 54: 1707–1719
Xing L N, Luo T Y, Li H, et al. Adaptive evolutionary algorithm for air defense resource allocation optimization (in Chinese). Sci Sin Tech, 2024, 54: 1707–1719, doi: 10.1360/SST-2023-0337

© 2024 《中国科学》杂志社

多种作战装备资源的有机结合与优势互补. 因此, 许多学者研究了多种作战装备资源之间的相互依赖关系. Xin等人[7]构建了综合考虑作战装备与传感器之间的资源约束、能力约束以及策略约束的数学模型, 并提出一种基于边际收益的构造性启发式算法. Severson和Paley[8]研究了雷达对来袭导弹的分配问题, 其中雷达的主要功能在于跟踪来袭目标的飞行轨迹, 并且在跟踪过程中不允许多部雷达切换. Zhang等人[9]通过分析地空防御中"最远拦截、分步抵抗"的战术, 提出了传感器/异构作战装备一体化配置问题, 设计了一种包含三种编码方法的动态传感器/异构作战装备目标分配的进化算法.

随着来袭目标数量和作战装备种类的增加, 问题解空间也会呈现指数增长的趋势[10]. 因此, 为了能够在短时间内得到满意的资源分配方案, 许多学者进一步研究作战装备对作战效能的影响, 将这一影响转化为可实施的约束条件[11~16]. Yan等人[17]为了协调不同类型雷达的异构发射资源(发射功率、停留时间等), 以达到更好的资源利用效率, 提出了两种优化资源分配方案. Hocaoğlu[18]以最小生存概率为目标, 研究了防空导弹对来袭空中目标的分配问题, 建立了一种基于约束的非线性目标规划作战装备分配问题模型. Chang等人[19]与以往研究不同, 研究了一种多阶段作战装备-目标分配问题, 即在多个攻击阶段向所有目标分配有限的作战装备. 通过构建一种二元非线性整数规划模型来表述该问题, 并设计了一种新的改进的自适应大规模邻域搜索(ALNS)算法进行求解. 此外, 为了求解更为复杂的组合优化问题, 许多学者也设计了诸如遗传算法、粒子群算法和差分演化算法进行求解[20~23].

综上所述, 现阶段针对防空资源分配问题的研究大多假设雷达在跟踪过程中不允许切换, 且集中在对雷达信号、传感器等资源的分配, 缺乏对考虑接力跟踪制导的研究. 本文结合现有防空资源分配的发展现状, 在研究问题中综合考虑多部雷达连续接力跟踪和对目标连续发射多枚导弹两方面的因素开展了深入研究. 基于此, 本文的主要贡献如下.

(1) 研究一种多部雷达连续接力跟踪目标, 且对目标连续发射多枚导弹进行拦截的防空资源分配问题, 即多部雷达交替跟踪同一目标.

(2) 为了降低不可行解被搜索的次数, 本文将包含多枚导弹的防空资源分配问题分成两阶段进行求解, 即第一阶段为每一目标分配第一枚导弹, 第二阶段分配其他导弹.

(3) 为了有效求解此两阶段问题, 提出一种基于资源释放与重分配的自适应演化算法. 即在迭代中只考虑为每一目标分配一枚导弹, 并采用遗传算法与资源释放和重规划的方法进行优化. 在迭代后考虑多枚导弹, 采用启发式规则进行再优化.

2 问题描述及数学模型

2.1 问题描述

本文所研究的防空资源分配优化问题可描述为: 二维平面下的一定区域范围内, 为所有来袭目标分配防空资源(雷达和发射车)和发射车发射导弹时刻的组合方案. 其中, 在为特定来袭目标分配发射车和雷达后, 从发射车发射导弹的瞬间开始, 直至导弹成功命中目标的整个过程, 都需要该雷达进行不间断的持续跟踪, 如图1(b)所示. 此外, 雷达具有不同精度的探测范围, 以一定角度的扇形展开, 且雷达在同一时刻可跟踪的目标数量与其通道数有关. 发射车具有不同精度的拦截范围, 以一定角度的圆形展开, 且每辆发射车上装备的导弹类型唯一, 数量固定. 图1(a)给出了一种防空资源分配示例, 其中圆圈为雷达, 扇形为雷达探测范围; 方块为发射车, 虚线圆示意拦截范围. 以某个目标为例, 假设其直线飞行, 其预测轨迹为朝向箭头的虚线段, 飞行路线的实线段为雷达跟踪时间段, X为命中点.

针对防空资源分配问题现状, 分析总结拦截目标的现实需求, 本文明确了防空资源分配的具体研究问题. 即在指定区域内, 以当前探测信息和轨迹预测结果为输入信息, 对当前探测到的每个来袭目标, 结合己方雷达与发射车的防空资源信息, 在指定时间内制定出一套拦截方案. 方案中具体表明针对每一来袭目标, 使用何种型号的发射车在何时发射导弹, 以及对该目标采用哪部雷达提供跟踪制导信息. 以此输出面向全局和未来的规划结果, 并为作战人员制定精确且高效的打击方案提供借鉴.

此外, 防空资源分配问题属于一种在连续时间区间内分配可行方案的连续优化问题. 然而, 考虑到计算资源和数值精度的限制, 为了实现可行的解决方案并保证问题的可解性, 本文通过设置固定的时间片段(时间离散化标准), 将该问题转化为离散形式. 基于

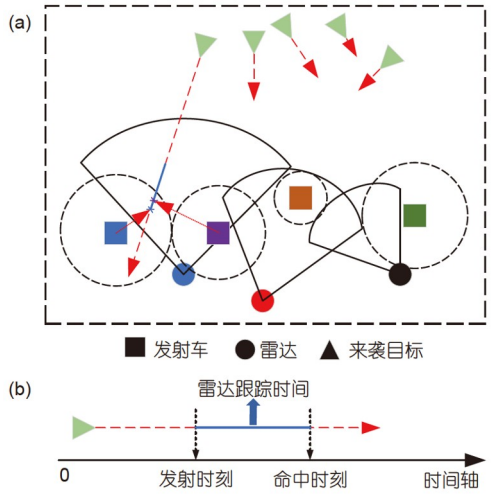

图 1　(网络版彩图)防空资源分配问题示意图. (a) 目标拦截示意图; (b) 目标拦截时间示意图

Figure 1　(Color online) Schematic diagram of air defense resource allocation problem. (a) Target interception schematic diagram; (b) target interception time diagram.

此, 本文对防空资源分配问题做出如下任务设定:

(1) 来袭目标飞行的角度固定, 即目标飞行轨迹为直线;

(2) 来袭目标和发射车发射的导弹的飞行速度恒定, 且匀速飞行;

(3) 每一发射车只配备同种类型的导弹;

(4) 对每一目标可发射导弹数量不超过2枚;

(5) 每枚导弹在飞行中的每个时间段, 至少有1部雷达在跟踪该导弹拦截的目标;

(6) 考虑导弹在飞行过程中, 由多部雷达接力跟踪探测目标, 即考虑雷达接力制导;

(7) 导弹在命中目标的前3个时间片段内禁止雷达切换.

根据上述问题描述与任务设定设置参数符号和变量, 具体内容如表1所示.

2.2　数学模型

在实际作战中, 由于来袭目标飞行时间长短不同和目标飞行目的地不同等因素的影响, 构建的数学模型需将上述因素转化为目标的拦截优先级来进行优化[24]. 因此, 该问题的优化目标是在考虑目标优先级的前提下最大化综合拦截概率, 以合理运用防空资源, 发挥整体优势.

表 1　参数符号及说明

Table 1　Parameter symbols and descriptions

符号	说明
I	打击目标集合, $I=\{1, 2, 3, \cdots, n_1\}$
L	发射车集合, $L=\{1, 2, 3, \cdots, n_2\}$
R	雷达集合, $R=\{1, 2, 3, \cdots, n_3\}$
S	发射车和雷达的组合集合, $S=\{1, 2, 3, \cdots, n_4\}, n_4 = n_2 \times n_3$
μ	时间离散化标准
$T_i^{\max}$	目标i的最大飞行时长
T_i	目标i的可拦截时间节点集合, $T_i=\{1, 2, 3, \cdots, t_i^{\max}\}, t_i^{\max} = \left\lfloor \frac{T_i^{\max}}{\mu} \right\rfloor$
$p_{i,l}^t$	目标i在时刻t被发射车l拦截的概率
P_i	对目标i的综合拦截概率
W_i	目标i的拦截优先级
θ_l	发射车l配备的导弹数量
$q_{\max}$	对目标发射导弹数量的最大值
m_i^t	向目标i在时刻t发射的导弹数量
$t_{i,p}^{\text{start}}$	第p枚导弹朝目标i发射的时刻
$t_{i,p}^{\text{end}}$	第p枚导弹命中目标i的时刻
η	朝目标发射多枚导弹命中时刻的最大间距
$c_{i,r}$	目标i占用雷达r的通道数
C_r	雷达r的最大通道数
$x_{i,l}^t$	决策变量, 为1表示目标i被发射车l发射的导弹在t时刻拦截, 否则为0
$y_{i,r}^t$	决策变量, 为1表示目标i在时刻t被雷达r跟踪, 否则为0

$$\max \sum_{i=1}^{n_1} W_i P_i, \tag{1}$$

$$P_i = 1 - \prod_{t=1}^{t_i^{\max}} \left[\prod_{l=1}^{n_2} (1 - x_{i,l}^t \times p_{i,l}^t)^{m_i^t} \right], \forall i \in I, \tag{2}$$

$$m_i^t = \sum_{l=1}^{n_2} x_{i,l}^t, \forall i \in I, t \in T_i, \tag{3}$$

$$0 \leqslant \sum_{t=1}^{t_i^{\max}} m_i^t \leqslant q_{\max}, \forall i \in I, \tag{4}$$

$$\sum_{i=1}^{n_1}\sum_{t=1}^{t_i^{\max}} x_{i,l}^{t}\times m_i^{t}\leqslant\theta_l,\ \forall l\in L, \tag{5}$$

$$0\leqslant\left|t_{i,(p+1)}^{\text{end}}-t_{i,p}^{\text{end}}\right|\leqslant\eta,\ \forall i\in I, \tag{6}$$

$$0\leqslant\sum_{i=1}^{n_1}y_{i,r}^{t}\times c_{i,r}\leqslant C_r,\quad \forall r\in R,t\in\left[\left\lfloor\frac{t_{i,p}^{\text{start}}}{\mu}\right\rfloor,\left\lceil\frac{t_{i,p}^{\text{end}}}{\mu}\right\rceil\right], \tag{7}$$

$$y_{i,r}^{t}=y_{i,r}^{t+1},\forall t\in\left[\max\left(\left\lfloor\frac{t_{i,p}^{\text{end}}}{\mu}\right\rfloor-3,\left\lfloor\frac{t_{i,p}^{\text{start}}}{\mu}\right\rfloor\right),\left\lceil\frac{t_{i,p}^{\text{end}}}{\mu}\right\rceil\right], \tag{8}$$

$r\in R,\ i\in I$.

式(1)表示在考虑目标优先级的前提下最大化综合拦截概率; 式(2)表示对目标i发射多发导弹的综合拦截概率; 式(3)表示对目标i在时刻t发射的导弹数量; 式(4)表示对目标i发射的导弹数量不超过约束上限; 式(5)表示发射车l发射的导弹数量不能超过上限; 式(6)表示对目标i发射的多发导弹之间的时间间隔约束; 式(7)表示雷达r在任一时刻t所使用的通道数不超过上限; 式(8)表示约束拦截目标前雷达不切换, 以避免对末段制导产生干扰.

3 算法设计

本文所研究的防空资源分配问题属于一种NP-hard问题, 存在高维、非线性、强约束共存等复杂性, 对在短时间内从复杂解空间内获取较优方案提出了极大挑战. 而在求解大规模问题中, 精确算法会消耗过多的算力和存储空间来寻优, 导致其运行时间过长[25]. 因此, 本文提出了一种基于资源释放与重分配的自适应演化算法(adaptive evolutionary algorithm based on resource release and redistribution, AEA-RSRLRD)来求解此复杂问题. AEA-RSRLRD是一种元启发式算法, 它在演化算法的基础上结合求解问题的特性来最大化搜索性能. 其中, 在演化算法迭代过程中对染色体部分基因占用的资源进行释放, 并基于规则进行重分配; 在演化算法迭代结束后引入启发式规则来对最优个体进行深度搜索, 从而在合理的时间内最大化资源利用率, 进而得到更优解.

AEA-RSRLRD流程如图2所示. 首先, 根据输入的算例参数信息(包括雷达信息、发射车信息和来袭目标信息)和算法迭代参数进行种群初始化, 产生初始父代种群, 且种群初始化只为目标分配第一枚导弹的方案, 剩余导弹方案用启发式规则进行分配. 然后进入循环迭代寻优, 其中依次采用锦标赛选择法选择进行交叉变异的个体, 然后从交叉变异产生子代种群中随机选择某一个体进行释放与重分配优化, 并更新父代种群. 当循环结束后, 采用启发式规则对当前最优解进行局部搜索, 以获取更优解.

3.1 生成初始解

考虑到防空资源分配问题需要获取的方案为对每一来袭目标分配的雷达、发射车和导弹发射时间的组合方案. 因此, 本文在生成初始解前需进行如下操作.

首先, 通过输入的雷达信息(包括坐标、类型、探测范围与探测角度)和发射车信息(包括坐标、类型、拦截范围与导弹飞行速度)来分析对当前区域的可探测范围与拦截范围信息.

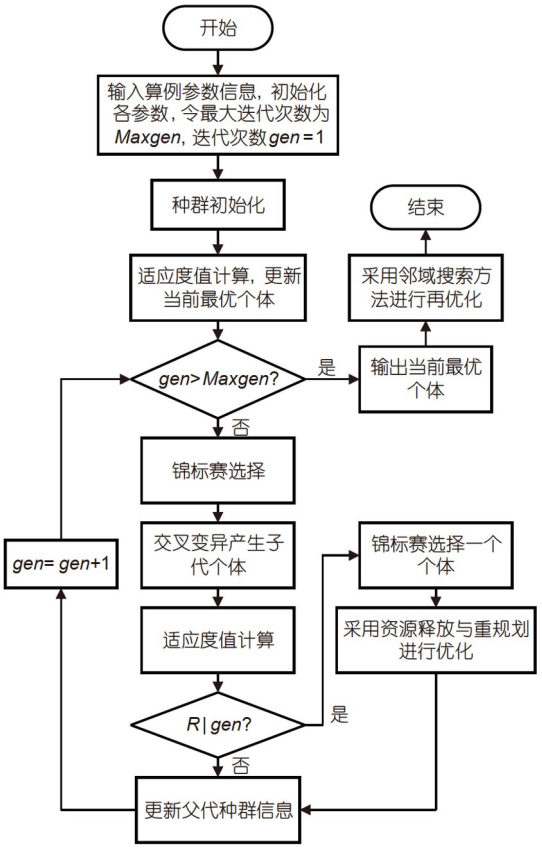

图 2 算法流程图
Figure 2 Algorithm flow chart.

其次, 结合来袭目标信息(包括坐标、种类、速度和角度)预测其飞行轨迹, 并通过固定的时间片段将飞行轨迹离散化, 得到每个目标的飞行时刻集合.

最后, 结合上述信息, 依次判断每个目标在飞行过程中可被拦截的时刻与对应的发射车编号、雷达编号(此雷达编号只负责$\left[\max\left\{t_{i,p}^{\text{end}}-3\mu, t_{i,p}^{\text{start}}\right\}, t_{i,p}^{\text{end}}\right]$区间内的跟踪任务), 同时根据发射车的拦截远近来计算在当前时刻被拦截的概率.

针对每一来袭目标, 按照导弹发射时间的先后顺序可得出其可被拦截的方案信息集合$TS_i=\left\{ts_1, ts_2, \cdots, ts_{n_i}\right\}$, n_i表示目标i可被拦截方案编号的序号上限值. 其中, 每一可拦截方案中包括发射车编号l、雷达编号r、导弹发射时刻t_i^{start}、导弹命中时刻t_i^{end}与拦截概率$p_{i,l}^{t}$.

综上, 初始解的生成如算法1所示.

在算法1中只考虑对每一目标分配一枚导弹的拦截方案信息, 其他导弹的方案采用启发式规则从候选方案中进行挑选. 具体原因如下.

(1) 对来袭目标发射多发导弹会增加问题解空间的维度, 从而提升AEA-RSRLRD的寻优难度, 进而降低算法的搜索性能, 增加算法运行时长.

(2) 由于本文考虑连续两枚导弹的命中时刻之间的差值需在$[0,\eta]$内, 因此对同一目标发射的多枚导弹之间存在一定的关联. 此外, 借鉴两阶段问题的处理方法[26], 将此复杂问题分阶段处理, 以降低算法搜索到不可行解的概率.

传统遗传算法的交叉变异操作可描述为: 在交叉变异过程中对指定长度的片段, 在满足大于交叉概率的前提下对两个个体的部分片段进行交叉与修复. 而AEA-RSRLRD构造的染色体基因片段为给定范围内的正整数值. 其编码示意图如图3所示.

基于此, 为了降低修复操作对算法运行时长的影响, 本文使用多点交叉变异方法来提升算法的性能. 具体操作示意图如图4所示.

算法1: 种群初始化

```
输入: 算例参数Parameter; 种群大小PopSize;
输出: 父代种群Population;
Scheme = Parameter.data; // 提取拦截方案集合信息
NTarget = numel(Parameter.Target); // 获取目标个数
for i = 1: PopSize
  for j = 1: NTarget
    k = 1;
    Pick = TournamentSelection(2, 1, Scheme, j);
    decs(j, k) = pick;
    decs1 = Scheme(pick);
    decs1{j}(k) = sort(decs1{j}(k));
  end for
  Population(i).decs = decs;
  Population(i).decs1 = decs1;
end for
//计算适应度
Population = CalObj(Population, parameter, PopSize);
return Population;
```

3.2　适应度值计算

结合第2.1节对防空资源分配问题的定义, 相比于传统不切换雷达的方式, 本文所研究的防空资源分配问题为了提升雷达资源的利用率, 考虑了多部雷达持续交替切换跟踪目标的方式. 此外, 根据第3.1节中染色体的构造原则可知, AEA-RSRLRD在初始解构造过程中, 染色体个体编号对应的拦截方案中雷达只负责$\left[\max\left\{t_{i,p}^{\text{end}}-3\mu, t_{i,p}^{\text{start}}\right\}, t_{i,p}^{\text{end}}\right]$内的跟踪任务, 而未对$\left(t_{i,p}^{\text{start}}, \max\left\{t_{i,p}^{\text{end}}-3\mu, t_{i,p}^{\text{start}}\right\}\right)$内的雷达跟踪信息进行分配. 因此, 本文在适应度值计算的过程中以一定规则对该部分时间片段分配雷达信息. 适应度值计算伪代码如算法2所示.

如算法2所示, 适应度计算的具体流程如下所示.

步骤1: 根据输入种群信息和相关参数信息, 提取雷达数量、发射车数量等信息. 令$i=1$.

步骤2: 判断i是否大于种群大小$PopSize$, 是则进入步骤14, 反之进入下一步.

步骤3: 记录当前雷达通道占用信息与发射车剩余导弹信息. 令$j=1$.

步骤4: 判断j是否大于来袭目标总数, 是则令$i=i+1$, 并进入步骤2, 反之令$p=1$, 进入下一步.

步骤5: 判断p是否大于$q_{\max}$. 是则令$j=j+1$, 进入步骤4, 反之则提取当前目标的方案信息, 包括雷达编号、发射车编号和目标种类信息, 并进入下一步.

步骤6: 提取当前目标的第p枚导弹的信息, 判断p是否等于1, 如果是则进入下一步, 否则进入步骤8.

步骤7: 检查发射车剩余弹量约束、雷达通道约束. 如果满足则进入步骤9, 反之令$p=p+1$, 进入步骤5.

算法 2: 适应度值计算

输入: 算例参数*Parameter*; 种群信息*Population*; 种群大小*PopSize*;

输出: 种群信息*Population*;

```
Radar_num = Parameter.radar; //提取雷达数量
Launch_num = Parameter.missile; //提取发射车数量
NTarget = Parameter. NTarget; //提取目标总数
Scheme = Parameter.data; // 提取拦截方案集合信息
Time_scatter = Parameter.scatter; //时间离散化标准
//提取对一个目标发射的导弹数量最大值
MaxMissile = Parameter.maxmissile;
for i = 1:PopSize
  curRadar_channel = []; // 记录雷达通道占用信息
  curLaunch_missile = []; // 记录发射车弹量信息
  for j = 1:NTarget
    //提取当前方案对应的资源占用信息
    curScheme = GainScheme(Population(i), Parameter);
    for n = 1:MaxMissile
      ConstraintJudge(); //检查约束
      //按剩余资源信息依次填充雷达信息
      //获取对目标j发射的第n枚导弹的总时长
      curTotoal_time = curScheme.time(n);
      if ceil(curTotoal_time/Time_scatter) <= 3
        //依次更新每个时间片段的雷达通道占用信息
        UpdateRadar_Channel();
      else
        //依次更新后3个时间片段的雷达通道占用信息
        UpdateRadar_Channel();
        //按从后往前原则依次填充每个时间片段的雷达
        FillRadar_Channel();
      end if
    //满足雷达通道约束则更新雷达占用信息
    UpdateRadar_Channel();
    //不满足雷达通道约束则释放对应方案占用资源信息
    ReleaseRadar_Channel();
    end for
    UpdatePopulation(i, j); //更新个体i第j个目标的信息
  end for
  UpdatePopulation(i); //更新个体i的方案信息
  //更新个体i的资源占用信息
  curRadar_channel←UpdateResource(i);
  curLaunch_missile←UpdateResource(i);
end for
```

步骤8: 判断第p枚导弹的命中时间与第$p-1$枚导弹的命中时间之间的差值是否满足约束, 是则进入下一步, 反之令$p=p+1$, 进入步骤5.

步骤9: 提取当前导弹方案中雷达编号r_i, 构建雷达填充集合$Sr=\{r_i,1,2,\cdots,r_{i-1},r_{i+1},\cdots,r_n\}$.

步骤10: 判断当前导弹的飞行时间是否大于3μ, 如果是则令$p=p+1$, 进入步骤5. 反之进入下一步.

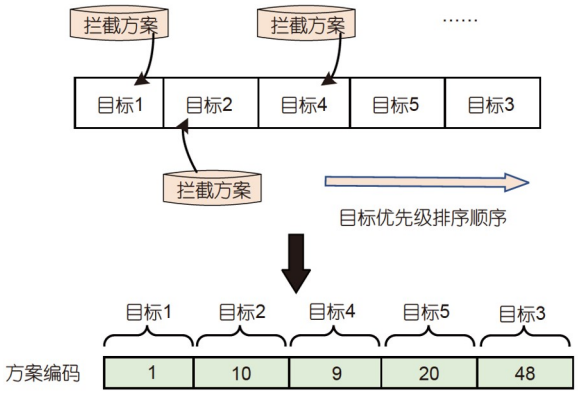

图 3 (网络版彩图)染色体编码示意图

Figure 3 (Color online) Chromosome coding schematic diagram.

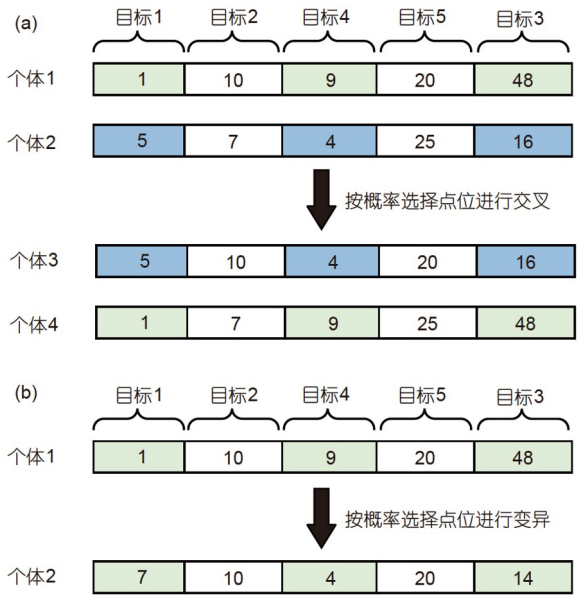

图 4 (网络版彩图)染色体交叉变异示意图. (a) 交叉示意图; (b) 变异示意图

Figure 4 (Color online) Schematic diagrams of chromosome crossover and mutation. (a) Crossover operation diagram; (b) mutation operation diagram.

步骤11: 从后往前分别对$\left[t_{i,p}^{\text{start}}, t_{i,p}^{\text{end}}-3\mu\right]$中各个时间片段按先后顺序从$Sr=\{r_i,1,2,\cdots,r_{i-1},r_{i+1},\cdots,r_n\}$选出满足雷达通道约束的雷达编号, 如果所有时间片段都能被雷达跟踪, 则记录当前雷达分配信息, 并更新雷达通道占用信息和发射车剩余弹量信息, 进入下一步. 反之则令$p=p+1$, 进入步骤5.

步骤12: 将雷达接力跟踪信息记录在当前个体的方案信息中, 令$j=j+1$, 并进入步骤4.

步骤13: 计算当前个体的适应度值, 令$i=i+1$, 进入步骤2.

步骤14: 循环结束, 输出种群信息.

3.3 启发式规则设计

为了在演化算法的基础上进一步提升算法的性能, 本文在AEA-RSRLRD的迭代过程中与迭代结束后分别设计了两种规则. 其中迭代过程中的规则如算法3所示, 迭代过程结束后的规则如算法4所示.

由算法3可知, 本文为了提升AEA-RSRLRD的局部搜索能力, 采用启发式规则优化的方式对可行解中第一枚导弹的部分基因片段进行资源释放与重分配操作(第8~17行).

在算法4中, 本文不仅对目标的第一枚导弹资源分配方案进行了再优化, 同时也采用规则的方法对多枚导弹的方案进行了分配. 即通过AEA-RSRLRD在循环迭代过程中找到的最优解信息, 采用规则的方法从满足多枚导弹命中时刻约束的可行方案集合中挑选拦截概率最大的资源分配方案, 以此输出更优解.

算法 3: 资源释放与重规划

```
输入: 算例参数Parameter; 个体信息Solution;
输出: 个体信息Solution;
1  Radar_channel = Solution.radar; //提取雷达通道占用信息
2  Launch_missile = Solution.missile; //提取发射车剩余弹量信息
3  NTarget = Parameter. NTarget; //提取目标总数
4  Scheme = Parameter.data; // 提取拦截方案集合信息
5  if sum(Launch_missile)>0
6     n = 1;
7     for i = 1:NTarget
8        if Solution.cons(n,i) != 0
9           //找到当前目标拦截最高概率
10          maxP = max(Scheme.p(i));
11          curP = Solution.cons(n,i);
12          if curP < maxP
13             //释放当前方案占用雷达与发射车资源信息
14             Release(Radar_channel, Launch_missile);
15          else
16             Continue
17          end if
18       end if
19       CurScheme = Scheme(i);
20       CurScheme = sort(CurScheme);
21       for j = 1:numel(CurScheme)
22          ConstraintJudge(); //检查约束
23          RadarFill(); //按规则填充雷达编号
24          //更新当前个体方案信息
25          NewSolution = UpdateScheme();
26       end for
27    end for
28 end if
29 if NewSolution.objs>Solution.objs
30    Solution = NewSolution;
31 end if
```

4 实验分析

为了评估AEA-RSRLRD的性能, 将其与传统遗传算法(GA)[27]、自适应大邻域搜索算法(ALNS)[19]和同时优化多发导弹的改进遗传算法(IGA-SOM)[28]进行对比. 所有的实验在配置为AMD Ryzen 3700x 4.2GHz(16 GB RAM), 操作系统为Win10的环境下进行, 上述所有算法均在MATLAB2021b上实现.

4.1 算例及参数设置

为了检验AEA-RSRLRD的适应性, 通过设置不同规模的来袭目标的算例(小规模、中规模和大规模)来进行仿真试验分析. 此外, 考虑问题解空间的复杂程度不仅与来袭目标数量相关, 也与防空资源数量相关, 因此本文根据不同目标规模来配备不同防空资源(资源不足、资源充足和资源过剩), 从而进一步验证AEA-RSRLRD的适应性. 所构建算例信息如表2所示. 其中雷达通道数上限表示来袭目标在某一时刻被一部雷达跟踪所占用的通道资源. 此外, 在后续实验中, 每一目标占用的通道数为1; 时间离散化标准μ为5; 对目标发射导弹数量的最大值$q_{\max}$为2.

所有对比算法采用的雷达填充逻辑均与AEA-

算法 4: 邻域搜索优化

输入: 算例参数*Parameter*; 最优个体信息*BestSol*;
输出: 最优个体信息*BestSol*;

```
Radar_channel = Solution.radar; //提取雷达通道占用信息
Launch_missile = Solution.missile; //提取发射车剩余弹量信息
NTarget = Parameter. NTarget; //提取目标总数
Scheme = Parameter.data; // 提取拦截方案集合信息
//提取对一个目标发射的导弹数量最大值
MaxMissile = Parameter.maxmissile;
if sum(Launch_missile)>0
  for n = 1:MaxMissile
    for i = 1:NTarget
      if BestSol.cons(n,i) ! = 0
        //根据第1枚导弹方案从备选方案中挑选满足多发弹命中时间间距的所有方案索引
        feasibleInd = Selectind(Scheme,BestSol,i);
        CurScheme = Scheme{i}(feasibleInd);
      else
        CurScheme = Scheme{i};
      end if
      CurScheme = sort(CurScheme);
      for j = 1:numel(CurScheme)
        ConstraintJudge(); //检查约束
        RadarFill(); //按规则填充雷达编号
        //更新当前个体方案信息
        NewSolution = UpdateScheme( );
      end for
    end for
  end for
end if
if NewSolution.objs >BestSol.objs
  BestSol = NewSolution;
end if
```

RSRLRD相同, 其他参数分别为: 最大迭代次数50, 种群大小为20, 交叉概率为0.5, 变异概率为0.2.

4.2 算法性能验证

本文在三种不同规模的算例下对AEA-RSRLRD求解所得到的解与其他3种算法得到的解进行了对比. 其中, 所有算法在不同算例下重复运行20次, 取20次结果的平均值(*Avg*)、最优值(*Best*)和平均求解时间(*T*)进行分析. 此外, 为了更加清晰对比算法间的差异, 引入了GAP值[29]表示AEA-RSRLRD与对比算法之间的差距, GAP的具体计算公式如式(9)所示. 相关实验结果如表3所示.

$$\mathrm{GAP}=\frac{C_1-C_0}{\min\{C_0,C_1\}},\tag{9}$$

其中, C_1表示AEA-RSRLRD的最优目标值, C_0表示其他3种算法的最优目标值.

由表3可知, 相较于GA, ALNS和IGA-SOM, AEA-RSRLRD的优势主要体现在以下两方面.

(1) 在求解质量方面: AEA-RSRLRD在所有不同防空资源和不同规模的算例中运行所得到的解都明显优于3种对比算法. 其中, 当防空资源充足和过剩时, AEA-RSRLRD的优势较为显著. 与GA相比, AEA-RSRLRD求得解的质量提升13.18%~64.05%, 特别是在中规模和大规模算例中, 可提升27.64%以上; 与ALNS相比, AEA-RSRLRD求得解的质量提升7.87%~28.10%; 与IGA-SOM相比, AEA-RSRLRD求得解的质量提升8.40%~29.04%. 因此, 通过将考虑多部雷达接力跟踪和同时发射多发弹的复杂问题分解成两阶段分别求解, AEA-RSRLRD相比其他算法能够有效减少搜索到不可行解的次数, 在局部搜索能力和全局搜索能力中实现了较好的平衡.

(2) 在求解时间方面: AEA-RSRLRD在所有算例中求解所耗时间都优于3种对比算法. 其中, 在所有算例中, 相较于其他算法, AEA-RSRLRD能减少50%左右的求解时间.

基于表3中的实验结果, 为了进一步验证AEA-RSRLRD在仿真场景中对防空资源有效利用, 分别绘制如图5所示的4种算法在INFO-S0中的最优解甘特图. 其中, 横坐标表示时间区间, 纵坐标表示各来袭目标, 每一线段表示对应的雷达跟踪区间, 线段右端表示对应发射车发射导弹的命中时刻.

如图5所示, 与其他3种算法相比, AEA-RSRLRD的优势包括以下两个方面.

(1) 考虑到本文所研究的防空资源分配优化问题存在高维、强约束、离散共存等复杂性, 在对目标发射多枚导弹后, 由于规定多枚导弹之间的命中时间需满足一定约束, 因此复杂了整个问题解空间, 导致GA, ALNS和IGA-SOM无法在有限时间内求得较优分配方案, 即无法获取对同一目标发射多枚导弹的资源分配方案. 而AEA-RSRLRD通过将此复杂问题分阶段求解, 有效避免该情况的发生, 同时也能在较短的时间内得出更优的资源分配方案.

表 2　算例信息

Table 2　Example information

规模	算例	目标数	发射车		雷达	
			数量	导弹数量	数量	通道上限
小规模	INFO-S0	10	4	20	2	1
	INFO-S1	24	4	20	4	1
	INFO-S2	24	4	40	4	1
	INFO-S3	24	6	24	5	2
	INFO-S4	24	6	48	5	2
	INFO-S5	24	8	32	5	3
	INFO-S6	24	8	64	5	3
中规模	INFO-M1	60	8	56	7	1
	INFO-M2	60	8	112	7	1
	INFO-M3	60	10	60	8	2
	INFO-M4	60	10	120	8	2
	INFO-M5	60	12	72	8	3
	INFO-M6	60	12	132	8	3
大规模	INFO-L1	100	14	98	14	1
	INFO-L2	100	14	168	14	1
	INFO-L3	100	16	112	16	2
	INFO-L4	100	16	208	16	2
	INFO-L5	100	18	144	18	3
	INFO-L6	100	18	198	18	3

表 3　算法对比结果

Table 3　Algorithm comparison results

算例	GA				ALNS				IGA-SOM				AEA-RSRLRD		
	Avg	*Best*	*T* (s)	GAP (%)	*Avg*	*Best*	*T* (s)	GAP (%)	*Avg*	*Best*	*T* (s)	GAP (%)	*Avg*	*Best*	*T* (s)
INFO-S0	8.15	8.96	0.66	19.67	8.61	9.19	0.58	16.70	8.84	9.42	0.69	13.82	10.27	10.73	0.55
INFO-S1	15.59	16.51	2.23	21.36	16.91	18.24	1.69	9.87	16.44	17.80	2.39	12.60	18.45	20.04	1.13
INFO-S2	22.33	22.95	2.26	31.84	25.41	27.00	1.74	12.05	24.51	26.30	2.29	15.07	29.03	30.26	1.03
INFO-S3	19.56	20.54	1.78	13.18	20.79	21.55	1.42	7.87	20.18	21.27	1.81	9.28	22.72	23.24	0.91
INFO-S4	23.94	25.59	2.18	18.16	24.97	26.01	1.67	16.25	24.39	25.81	2.19	17.16	29.50	30.23	0.97
INFO-S5	23.48	25.05	2.47	13.92	25.51	26.46	1.94	7.87	24.77	26.33	2.49	8.40	27.63	28.54	1.08
INFO-S6	26.31	27.59	2.44	27.14	27.47	28.28	1.92	24.01	26.65	28.36	2.42	23.68	34.66	35.07	1.06
INFO-M1	29.43	31.85	5.24	33.01	35.45	37.88	4.48	11.82	34.83	36.90	5.58	14.81	41.45	42.36	3.52
INFO-M2	34.60	35.88	5.85	62.88	45.61	47.02	4.82	24.30	44.27	47.37	6.08	23.37	56.64	58.44	3.22
INFO-M3	31.97	33.45	5.33	27.64	36.91	38.37	4.34	11.28	35.59	36.89	5.39	15.75	41.69	42.69	3.07
INFO-M4	38.30	39.57	5.93	48.74	45.25	48.98	4.78	20.15	44.36	46.28	5.99	27.16	57.97	58.85	2.97
INFO-M5	40.54	41.86	5.16	31.49	44.91	46.30	4.11	18.87	43.93	45.72	5.18	20.37	53.65	55.04	2.52
INFO-M6	47.07	49.40	5.77	38.11	51.86	53.29	4.47	28.04	50.69	52.87	5.78	29.04	66.65	68.23	2.59
INFO-L1	37.59	39.03	9.53	42.50	46.95	48.19	10.61	15.40	45.97	48.13	11.75	15.57	54.56	55.62	6.16
INFO-L2	38.78	41.78	9.87	64.05	51.89	54.15	12.95	26.56	50.44	53.35	12.41	28.47	67.12	68.54	6.00
INFO-L3	45.33	46.42	9.65	28.50	51.20	52.37	10.18	13.90	50.67	52.33	10.02	13.98	58.20	59.65	6.74
INFO-L4	50.92	53.44	10.83	39.69	57.84	60.64	10.39	23.10	57.41	60.37	11.00	23.65	73.08	74.65	6.05
INFO-L5	50.79	52.13	10.43	28.30	56.06	57.76	10.04	15.79	54.77	55.99	10.71	19.45	65.33	66.88	5.87
INFO-L6	52.49	53.26	11.10	41.11	56.79	58.66	10.06	28.10	56.73	59.97	11.12	25.32	73.07	75.15	5.77

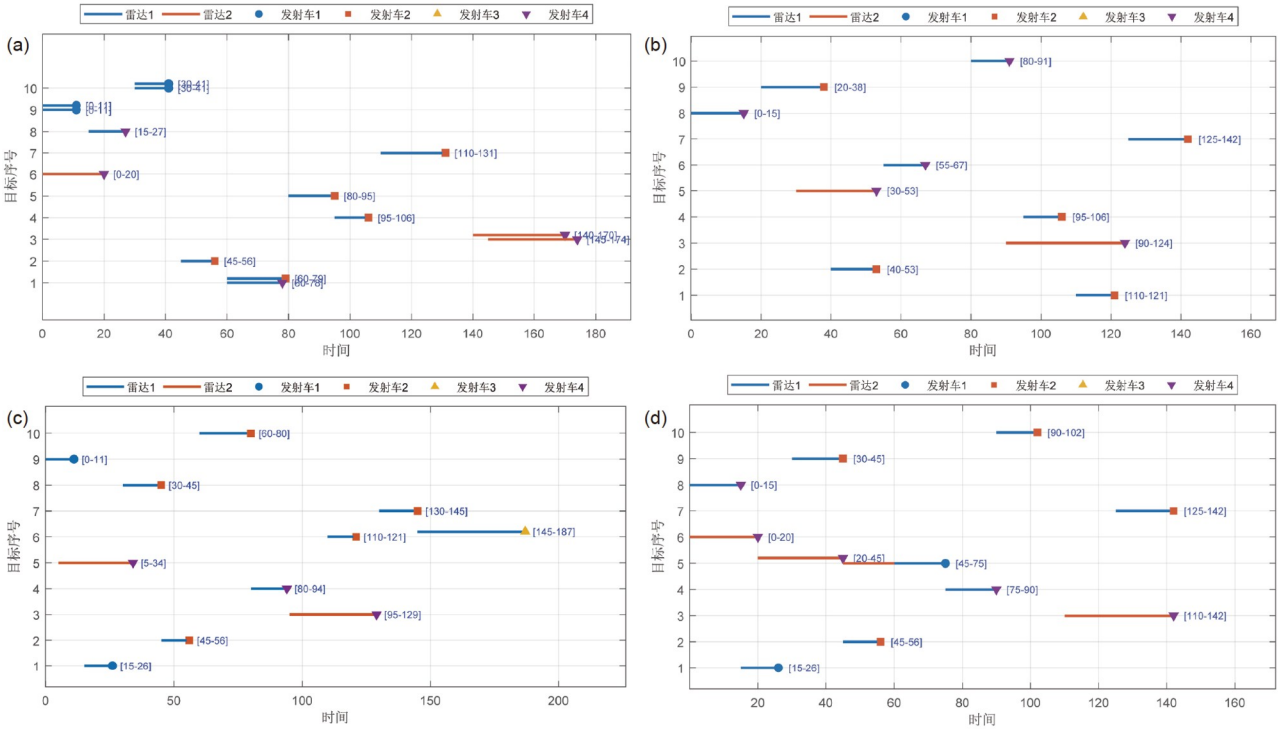

图 5 (网络版彩图) INFO-S0算例下4种算法最优甘特图. (a) AEA-RSRLRD; (b) GA; (c) ALNS; (d) IGA-SOM

Figure 5 (Color online) Optimal Gantt charts of four algorithms under the INFO-S0 example. (a) AEA-RSRLRD; (b) GA; (c) ALNS; (d) IGA-SOM.

(2) 本文所提出的AEA-RSRLRD以遗传算法为主循环, 在迭代过程中引入资源释放和重规划的规则, 在保证全局搜索能力的同时充分发挥规则在解空间的快速寻优与收敛能力. 此外, 基于已获取最优方案, 借助启发式规则快速获取更高质量的分配方案. 与其他3种算法相比, 能够表现出最佳的优化效果. 其中, 基于AEA-RSRLRD所得出的最优方案目标值见表3加粗部分.

4.3 算法稳定性分析

为了进一步验证AEA-RSRLRD的稳定性, 运用AEA-RSRLRD对19套算例重复求解20次, 并计算各算例的平均值、标准差和平均值与最优解之间的相对差. 实验结果如表4所示.

由表4可知, AEA-RSRLRD在20次求解中所得到的平均值与当前最优解差距较小, 在三种规模下的平均相对差分别为1.91%, 1.33%和1.25%. 其中, 最高相对差仅为INFO-S1算例中的2.88%. 此外, 从表4中的标准差一栏可知, AEA-RSRLRD求解每个算例的结果的标准差相较于平均值来说都偏小, 即该算法在多次运行过程中所得到的最优解的离散程度较低. 因此, 对防空资源分配问题来说, AEA-RSRLRD具有较强的稳定性, 在不同规模的算例中求解性能较高.

4.4 算法收敛性分析

为了验证AEA-RSRLRD的收敛性, 本文从三种不同规模的算例中分别选出INFO-S4, INFO-M4和INFO-L4来绘制4种算法的迭代图, 如图6所示.

根据图6可知, 对比其他3种算法, AEA-RSRLRD以较快的速度收敛, 这是因为AEA-RSRLRD将启发式规则与演化算法进行优势互补, 在迭代中采用规则的方法对部分个体进行资源释放与再分配, 提升了AEA-RSRLRD对局部空间的搜索能力; 在迭代后采用启发式规则来分配其他导弹的同时, 再次对当前最优个体进行资源释放与再分, 因此能够在减少在复杂解空间中搜索到较劣解的次数, 从而提升解的质量.

此外, 考虑到AEA-RSRLRD将防空资源分配问题分阶段处理, 即在迭代过程中只对目标的第一枚导弹进行迭代优化, 在迭代结束后采用启发式规则来分配其他导弹, 因此迭代图中只展示了AEA-RSRLRD对所

表 4　算例稳定性实验结果

Table 4　Experimental results of the stability of the example

	算例	平均值	最优解	相对差 (%)	标准差
小规模	INFO-S0	10.27	10.73	2.77	0.27
	INFO-S1	18.45	20.04	2.88	0.52
	INFO-S2	29.03	30.26	2.28	0.65
	INFO-S3	22.72	23.24	1.44	0.32
	INFO-S4	29.50	30.23	1.44	0.42
	INFO-S5	27.63	28.54	1.46	0.39
	INFO-S6	34.66	35.07	1.08	0.37
中规模	INFO-M1	41.45	42.36	1.09	0.44
	INFO-M2	56.64	58.44	1.62	0.89
	INFO-M3	41.69	42.69	1.37	0.55
	INFO-M4	57.97	58.85	1.07	0.60
	INFO-M5	53.65	55.04	1.40	0.73
	INFO-M6	66.65	68.23	1.41	0.92
大规模	INFO-L1	54.56	55.62	0.81	0.43
	INFO-L2	67.12	68.54	1.66	1.09
	INFO-L3	58.20	59.65	1.07	0.61
	INFO-L4	73.08	74.65	1.25	0.89
	INFO-L5	65.33	66.88	1.28	0.82
	INFO-L6	73.07	75.15	1.42	1.01

有目标第一枚导弹的迭代搜索过程.

5　结论

本文研究了一种考虑多部雷达接力跟踪探测目标的防空资源分配问题, 且考虑对同一目标发射多发导弹. 针对此问题, 在考虑雷达通道上限、导弹命中前一段时间禁止雷达切换等约束的前提下, 建立了防空资源分配数学优化模型, 并提出了基于资源释放与重分配的自适应演化算法(AEA-RSRLRD). 其中, AEA-RSRLRD将此复杂问题分阶段处理, 以降低算法搜索到不可行解的概率. 通过在3种不同规模和多种防空资源算例下进行仿真试验可知, 相比遗传算法、自适应大邻域搜索算法和同时优化多发导弹的改进遗传算法, AEA-RSRLRD的优化效果平均提升了33.98%, 17.51%和19.06%, 且在大部分算例下的求解时间优于其他3种算法, 表明了AEA-RSRLRD能够针对不同实例以较少的时间得到更优的资源分配方案. 本文提出的自适应演化算法在防空资源分配优化问题求解中取得了较好的结果, 但未考虑复杂战场下的扰动事件, 因此仍有进一步优化的空间.

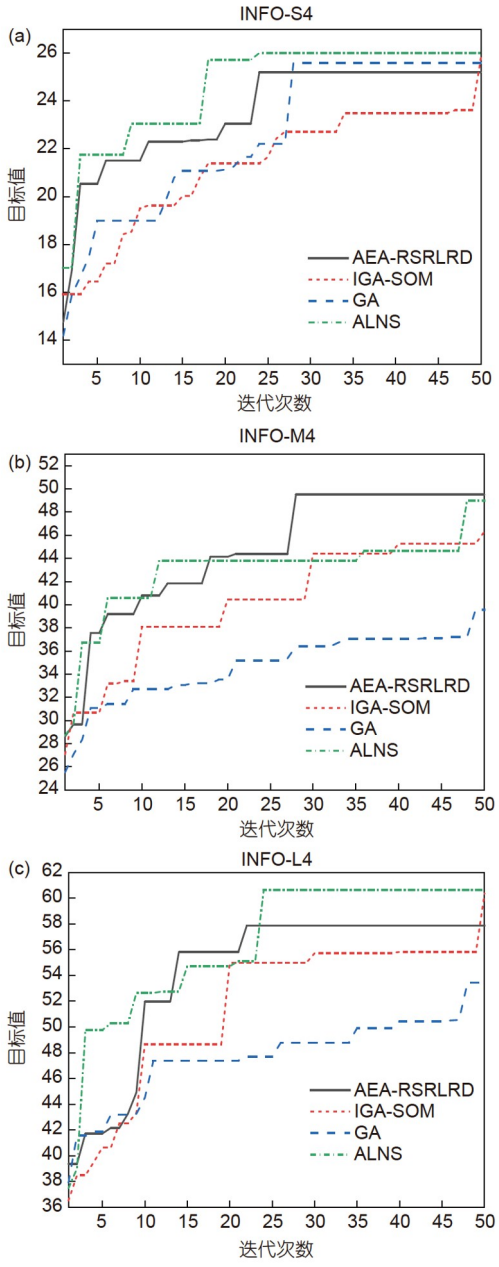

图 6　(网络版彩图)代表算例下4种算法迭代图. (a) INFO-S4算例下的算法迭代图; (b) INFO-M4算例下的算法迭代图; (c) INFO-L4算例下的算法迭代图

Figure 6　(Color online) Iteration diagrams of four algorithms under representative examples. (a) Algorithm iteration diagram under INFO-S4 example; (b) algorithm iteration diagram under INFO-M4 example; (c) algorithm iteration diagram under INFO-L4 example.

在未来, 由于空天威胁日趋复杂多样, 各类突发事件对已有资源分配方案的冲击愈发突出. 因此, 如何针对目标轨迹突变、目标速度可变和资源受干扰等典型突发事件做出及时的动态响应值得进一步研究.

参考文献

1 Manne A S. A target-assignment problem. Oper Res, 1958, 6: 346–351

2 Lloyd S P, Witsenhausen H S. Weapons allocation is NP-complete. In: 1986 Summer Computer Simulation Conference. 1986. 1054–1058

3 Li M J, Chang X N, Shi J M, et al. Developments of weapon target assignment: Models, algorithms, and applications (in Chinese). Syst Eng Electron, 2023, 45: 1049–1071 [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. 系统工程与电子技术, 2023, 45: 1049–1071]

4 Kline A G, Ahner D K, Lunday B J. A heuristic and metaheuristic approach to the static weapon target assignment problem. J Glob Optim, 2020, 78: 791–812

5 Silav A, Karasakal E, Karasakal O. Bi-objective dynamic weapon-target assignment problem with stability measure. Ann Oper Res, 2022, 311: 1229–1247

6 Acar E, Hatipoğlu S, Yılmaz İ. A quantum algorithm for solving weapon target assignment problem. Eng Appl Artif Intell, 2023, 125: 106668

7 Xin B, Wang Y, Chen J. An efficient marginal-return-based constructive heuristic to solve the sensor-weapon-target assignment problem. IEEE Trans Syst Man Cybern Syst, 2018, 49: 2536–2547

8 Severson T A, Paley D A. Distributed multitarget search and track assignment with consensus-based coordination. IEEE Sens J, 2014, 15: 864–875

9 Zhang K, Zhou D, Yang Z, et al. A novel heterogeneous sensor-weapon-target cooperative assignment for ground-to-air defense by efficient evolutionary approaches. IEEE Access, 2020, 8: 227373–227398

10 Li J Y, Du K J, Zhan Z H, et al. Distributed differential evolution with adaptive resource allocation. IEEE Trans Cybern, 2023, 53: 2791–2804

11 Qiu S M, Bai C C, Lyu Y N, et al. Dynamic weapon target allocation based on the multi-objective whale optimization algorithm (in Chinese). Syst Eng Electron, 2023, 44: 153–159 [邱少明, 柏陈城, 吕亚娜, 等. 基于多目标鲸鱼优化算法的动态武器目标分配. 兵器装备工程学报, 2023, 44: 153–159]

12 Ma Y Y, Wang G Q, Hu X X, et al. Weapon target assignment method for multiple UAVs in beyond-visual-range air combat (in Chinese). Chin J Manag Sci, 2022, 30: 248–257 [马滢滢, 王国强, 胡笑旋, 等. 超视距空战中的多无人机武器目标分配方法. 中国管理科学, 2022, 30: 248–257]

13 Yu B W, Lv M. Optimization method for multi-stage collaborative weapon firepower distribution based on D-NSGA-GKM algorithm (in Chinese). Control Decis, 2022, 37: 605–615 [于博文, 吕明. 基于D-NSGA-GKM算法的多阶段武器协同火力分配方法. 控制与决策, 2022, 37: 605–615]

14 Zhan Z H, Wang Z J, Jin H, et al. Adaptive distributed differential evolution. IEEE Trans Cybern, 2020, 50: 4633–4647

15 Wu W H, Guo X F, Zhou S Y, et al. Improved differential evolution algorithm for solving weapon-target assignment problem (in Chinese). Syst Eng Electron, 2021, 43: 1012–1021 [吴文海, 郭晓峰, 周思羽, 等. 改进差分进化算法求解武器目标分配问题. 系统工程与电子技术, 2021, 43: 1012–1021]

16 Zhan Z H, Zhang J, Li Y, et al. Adaptive particle swarm optimization. IEEE Trans Syst Man Cybern B, 2009, 39: 1362–1381

17 Yan J K, Pu W Q, Zhou S H, et al. Optimal resource allocation for asynchronous multiple targets tracking in heterogeneous radar networks. IEEE Trans Signal Process, 2020, 68: 4055–4068

18 Hocaoğlu M F. Weapon target assignment optimization for land based multi-air defense systems: A goal programming approach. Comput Ind Eng, 2019, 128: 681–689

19 Chang X, Shi J, Luo Z, et al. Adaptive large neighborhood search algorithm for multi-stage weapon target assignment problem. Comput Ind Eng, 2023, 181: 109303

20 Kropp I, Nejadhashemi A P, Deb K. Improved evolutionary operators for sparse large-scale multiobjective optimization problems. IEEE Trans Evol Computat, 2024, 28: 460–473

21 Liu M L, Zhou Y L, Wang S Y, et al. Multiresource constrained flexible job shop intelligent scheduling considering fixture-pallet combinatorial

optimization (in Chinese). Sci Sin Tech, 2023, 53: 1114–1126 [刘墨林,周雨露, 王思阳, 等. 考虑夹具-托盘组合优化的多资源约束柔性作业车间智能调度. 中国科学: 技术科学, 2023, 53: 1114–1126]

22 Wang L, Wang J J. A cooperative memetic algorithm for the distributed green flexible jobshop with transportation time (in Chinese). Sci Sin Tech, 2023, 53: 243–257 [王凌, 王晶晶. 考虑运输时间的分布式绿色柔性作业车间调度协同群智能优化. 中国科学: 技术科学, 2023, 53: 243–257]

23 Lu H C, Tseng H Y, Lin S W. Double-track particle swarm optimizer for nonlinear constrained optimization problems. Inf Sci, 2023, 622: 587–628

24 Fu L, Liu J C. Resource allocation of target and fire of manned/unmanned cooperative air combat (in Chinese). Fire Control Command Control, 2017, 42: 67–70 [傅莉, 刘骏驰. 多机协同目标与火力资源分配. 火力与指挥控制, 2017, 42: 67–70]

25 Costa L, Contardo C, Desaulniers G. Exact branch-price-and-cut algorithms for vehicle routing. Transp Sci, 2019, 53: 946–985

26 Chen Y, Hao J K. Two phased hybrid local search for the periodic capacitated arc routing problem. Eur J Oper Res, 2018, 264: 55–65

27 Arram A, Ayob M. A novel multi-parent order crossover in genetic algorithm for combinatorial optimization problems. Comput Ind Eng, 2019, 133: 267–274

28 Ren T, Xu H B, Jin K N, et al. Optimisation of takeaway delivery routes considering the mutual satisfactions of merchants and customers. Comput Ind Eng, 2021, 162: 107728

29 Liu Y, Luo Z, Liu Z, et al. Cooperative routing problem for ground vehicle and unmanned aerial vehicle: The application on intelligence, surveillance, and reconnaissance missions. IEEE Access, 2019, 7: 63504–63518

Adaptive evolutionary algorithm for air defense resource allocation optimization

XING LiNing[1], LUO TianYu[1], LI Hao[1], GONG MaoGuo[1] & WANG Rui[2]

[1] *Key Laboratory of Collaborative Intelligence Systems, Ministry of Education, Xidian University, Xi'an 710071, China;*
[2] *College of Systems Engineering, National University of Defense Science and Technology, Changsha 410073, China*

As an important issue of combat equipment-target allocation problem, the air defense resource allocation must continue to adapt to the dynamic changes in the battlefield. How to coordinate multi-combat equipment units in the complex battlefield environment and reasonably allocate air defense resources, and propose the more flexible and intelligent allocation strategy to maximize the equipment system efficiency and become the core issue that needs to be solved at the moment. A mathematical model for allocating air defense resources was developed. This model factors in different elements, such as tracking through multiple radar relays, using multiple missiles to intercept the same target, and prioritizing targets. By joining together the computational efficiency and customization benefits of heuristic rules with the wide-area search advantages of evolutionary algorithms, a new adaptive evolutionary algorithm has been created to address this problem in steps. To enhance the speed and capacity of the algorithm's search, the first missile allocation plan is assigned to the target during the iteration phase, and the method of resource release and replanning has been introduced to enhance the quality of the current scheme. Then, after the iteration, heuristic rules are used to allocate multiple missiles to the optimal scheme based on demand. Through numerical simulation experiments on examples of different resources and different scales, it is verified that the algorithm has the advantages of solving time and solving accuracy. It can effectively solve the problem of air defense resource allocation.

air defense resource allocation, re-planning, adaptive evolutionary algorithm

doi: 10.1360/SST-2023-0337