

# 空地多目标攻击武器-目标分配与制导序列优化

张安, 徐双飞, 毕文豪, 徐晗

(西北工业大学 航空学院, 西安 陕西 710072)

**摘要:** 武器-目标分配 (WTA) 与空地导弹的接力制导规划是远距离空地多目标攻击中亟需解决的难题, 具有参数复杂、约束多、非线性强等特点。为此, 建立多目标、多约束武器-目标分配与制导序列优化模型, 优化目标为目标综合生存概率最小和总用弹量最少, 约束条件涉及攻击机导弹配置、导弹毁伤能力、目标毁伤要求、制导站性能。对带精英策略的非支配排序遗传算法(NSGA-II)进行改进, 提出基于双序列编码的多种群 NSGA-II (DSMPNSGA-II), 通过优化 WTA 序列和制导站序列, 实现 WTA 方案及各导弹制导序列的优化。在 DSMPNSGA-II 中, 使用深度优先搜索-Dijkstra 算法搜索空地导弹制导序列, 改进交叉和变异操作, 以减少非可行解的产生, 引入多种群策略提升算法性能。仿真结果表明, DSMPNSGA-II 能够获得有效的 WTA 与空地导弹接力制导方案, 并且求解质量优于单种群 NSGA-II 和多目标粒子群优化算法。

**关键词:** 武器-目标分配; 接力制导; 多目标优化; 遗传算法

**中图分类号:** E926.1

**文献标志码:** A

**文章编号:** 1000-1093(202x)xx-xxxx-xx

**DOI:** 10.12382/bgxb.2022.0326

## Weapon-target Assignment and Guidance Sequence Optimization in Air-to-Ground Multi-target Attack

ZHANG An, XU Shuangfei, BI Wenhao, XU Han

(School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China)

**Abstract:** Weapon-target assignment (WTA) and relay guidance of air-to-ground missiles are urgently important in long-range air-to-ground multi-target attack, and the problem is of complex parameters, multiple constraints, and strong nonlinearity. A multi-objective and multi-constraint optimization model of WTA and guidance sequence is established, in which the objective functions include the minimum comprehensive survival probability of targets and the minimum number of used missiles, and the constraints involve missile configuration of the attack aircraft, damage capability of the missiles, damage requirements of the targets, performance of the guidance stations, etc. A double sequence coding multi-population non-dominated sorting genetic algorithm II (DSMPNSGA-II), an improved NSGA-II, is proposed, which optimizes the scheme of WTA and every missile's guidance sequence through optimizing the WTA sequence and guidance stations sequence. In DSMPNSGA-II, the depth-first-search-Dijkstra (DFS-DJ) algorithm is used to search for missiles' guidance sequences, crossover operation and mutation operation are improved to reduce the production of infeasible solutions, and the multi-population strategy is introduced to improve the performance of DSMPNSGA-II. Simulation results show that the DSMPNSGA-II algorithm can obtain effective schemes of WTA and air-to-ground missile relay guidance, and its solution quality is better than that of the single-population NSGA-II algorithm and the multi-objective discrete particle swarm optimization (MODPSO) algorithm.

**Keywords:** weapon-target assignment; relay guidance; multi-objective optimization; genetic algorithm

### 0 引言

远距离空地多目标攻击是现代战争中的一种新型作战形式, 我方攻击机编队携带并发射多枚、多

种型号的空地导弹<sup>[1]</sup>, 各空地导弹由多个制导站接力制导<sup>[2]</sup>, 对敌方多个地面目标实施精确打击, 具有打击范围更广、抗干扰能力更强的特点。为了提升打击成功率, 并减少资源消耗, 需要在一定的约束下, 对各攻击机的打击目标、使用的空地导弹类型

**收稿日期:** 2022-04-29

**基金项目:** 国家自然科学基金项目(62073267、61903305); 航空科学基金项目(201905053001); 西北工业大学博士论文创新基金项目(CX2022019)

**作者简介:** 张安(1962—), 男, 教授, 博士生导师。E-mail: zhangan@nwpu.edu.cn;

徐双飞(1996—), 男, 博士研究生。E-mail: xsf@mail.nwpu.edu.cn;

徐晗(1997—), 男, 博士研究生。E-mail: 744950146@qq.com

**通信作者:** 毕文豪(1986—), 男, 助理研究员。E-mail: biwenhao@nwpu.edu.cn

与数量, 以及各导弹的接力制导方案等进行规划, 即涉及武器-目标分配 (WTA) 和接力制导规划2类问题。

WTA将有限的武器弹药, 通过最优策略, 分配到相关目标, 实现作战效能最大化, 是一种多参数、多约束、非线性优化问题<sup>[3-5]</sup>。传统WTA方法以解析法为主, 包括分支定界法<sup>[6]</sup>、割平面法<sup>[7]</sup>、隐枚举法等, 这些方法对复杂WTA问题的适用性较差。WTA智能优化方法是近年来的主要研究方法: 夏维等<sup>[8]</sup>提出包含变量随机分解、合作协调进化和种群非支配排序的改进多目标粒子群算法, 实现大规模WTA问题的实时求解; 王然辉等<sup>[9]</sup>使用遗传算法求解对地打击WTA问题, 并通过控制种群规模和改进变异策略, 提升算法效率, 避免算法早熟; Chang等<sup>[10]</sup>针对多阶段动态WTA问题, 以最小化敌方目标的生存概率为优化目标, 采用改进的人工蜂群算法进行求解。其他WTA智能优化方法还包括神经网络法<sup>[11]</sup>、K近邻算法<sup>[12]</sup>、强化学习算法<sup>[13]</sup>、布谷鸟算法<sup>[14]</sup>、模拟退火算法<sup>[15]</sup>等。当前关于WTA的研究大多对作战场景的设计较为简单, 直接给定武器种类、数量以及对各目标的毁伤概率<sup>[16]</sup>, 忽略了作战环境及作战过程对打击效果的影响。在远距离空地多目标攻击中, WTA需要考虑目标位置、武器发射平台位置、不同武器发射平台的火力配置差异等因素, 并与导弹的接力制导过程紧密结合。

规划各空地导弹由哪些制导站进行接力制导并规划制导次序, 属于序列优化<sup>[17-18]</sup>问题。目前关于序列优化算法有一定的研究成果: Xu等<sup>[19]</sup>采用基于试飞任务序列的改进遗传算法, 优化多架试验机的试飞任务编排; 窦建平等<sup>[20]</sup>采用基于可行工序序列的遗传算法, 对零件的加工工序和各工序匹配的机床、刀具等资源进行优化, 这些研究方法对本研究具有指导意义。

本研究针对远距离空地多目标攻击作战任务, 设计更为复杂的作战场景: 对传统WTA问题中的武器种类、数量、毁伤概率等因素进行扩展, 考虑不同攻击机的导弹配置差异, 以及各攻击机与各目标在战场中的位置分布, 并在战场中加入制导单元, 将WTA与空地导弹的飞行过程及接力制导过程紧密结合。建立多目标、多约束WSA与制导序列优化模型, 目标函数为目标综合生存概率最小和总用弹量最少, 约束条件涉及攻击机导弹配置、导弹毁伤能力、目标毁伤要求、制导站性能等。提出基于双序列编码的多种群带精英策略的非支配排序遗传算法 (DSMPNSGA-II), 设计适用于WSA与制导序列优化模型的编码、解码、选择、交叉、变异等操作,

并通过多种群策略提升解的质量。通过算例仿真与算法对比, 验证DSMPNSGA-II求解WSA与制导序列优化问题的有效性。

## 1 WSA 与制导序列优化模型

### 1.1 作战场景构建

假设在未来某一远距离空地多目标攻击任务中, 我方派出  $M$  架攻击机, 每架攻击机携带一定数量的同种类型或不同类型的空地导弹, 对敌方  $N$  个地面固定目标进行打击。我方已在战场部署  $K$  个地面固定制导站, 各制导站的制导区域 (在二维平面的投影) 为以制导站为圆心的圆, 每个制导站可承担多枚空地导弹的制导任务。作战区域为二维平面矩形区域。以矩形西北顶点为原点  $O$ , 正东方向为  $X$  轴正方向, 正南方向为  $Y$  轴正方向, 建立平面直角坐标系。作战场景如图1所示:

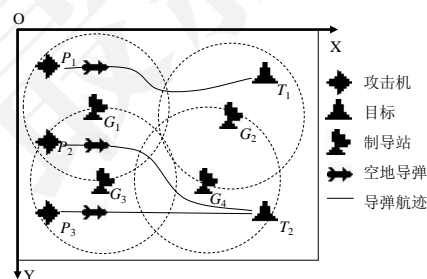


图1 作战场景示意图

Fig.1 Combat scene diagram

图1表示我方派出3架攻击机  $P_1 \sim P_3$ , 对敌方2个地面目标  $T_1$  和  $T_2$  进行打击, 由4个制导站  $G_1 \sim G_4$  对导弹进行制导。目标  $T_1$  由攻击机  $P_1$  发射导弹进行打击, 导弹由制导站  $G_1$  和  $G_2$  依次制导; 目标  $T_2$  由攻击机  $P_2$  和  $P_3$  发射导弹进行打击, 两枚导弹分别由  $G_1$  和  $G_4$ 、 $G_3$  和  $G_4$  依次制导。

本研究假设: 图1所示各攻击机位置即为该攻击机携带导弹的发射位置; 各空地导弹在巡航平飞阶段高度一致, 在该高度对应二维水平面内进行WSA与接力制导规划; 两个制导站的制导区域有交集即可进行制导交接。WSA与制导序列优化问题需要在一定的约束下, 对以下内容进行规划:

- 1) 规划每个目标由哪些攻击机进行打击;
- 2) 规划每架攻击机使用的导弹类型与数量;
- 3) 规划各导弹由哪些制导站依次接力制导。

优化目的为提升对目标的整体毁伤概率, 并减少资源消耗。

### 1.2 优化模型建立

#### 1.2.1 模型参数定义

模型中的参数定义为:

- 1) 攻击机参数。攻击机数量为  $M$ , 第  $m$  架攻击

机记为  $P_m$ ，位置为  $pp_m = (ppx_m, ppy_m)$ ，引入攻击机导弹配置矩阵  $PM$ ：

$$PM = \begin{pmatrix} pm_{11} & pm_{12} & \cdots & pm_{1H} \\ pm_{21} & pm_{22} & \cdots & pm_{2H} \\ \vdots & \vdots & \ddots & \vdots \\ pm_{M1} & pm_{M2} & \cdots & pm_{MH} \end{pmatrix} \quad (1)$$

式中： $pm_{mh}$  为第  $m$  行第  $h$  列元素，表示攻击机  $P_m$  装备第  $h$  种类型导弹的数量，取值范围为非负整数集， $h=1,2,\dots,H$ ， $H$  为导弹类型数量。

2) 目标参数。目标数量为  $N$ ，第  $n$  个目标记为  $T_n$ ，位置  $pt_n = (ptx_n, pty_n)$ ，价值系数为  $v_n$ ，各目标的最小毁伤概率均为  $dpm$ ，对各目标的最大毁伤次数均为  $dnm$ 。

3) 制导站参数。制导站数量为  $K$ ，第  $k$  个制导站记为  $G_k$ ，位置为  $pg_k = (pgx_k, pgy_k)$ ，制导区域为  $gxy_k$ ，各制导站制导半径均为  $RG$ ，制导能力均为  $CG$ 。

4) 导弹参数。空地导弹数量等于  $PM$  中所有元素之和，记为  $sum(PM)$ ，导弹类型数量为  $H$ ，第  $i$  枚导弹的类型为  $mt_{ie}$ （取值为不超过  $H$  的正整数），所属攻击机为  $mp_i$ （取值为不超过  $M$  的正整数），引入导弹杀伤概率矩阵  $MT$ ：

$$MT = \begin{pmatrix} mt_{11} & mt_{12} & \cdots & mt_{1N} \\ mt_{21} & mt_{22} & \cdots & mt_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ mt_{H1} & mt_{H2} & \cdots & mt_{HN} \end{pmatrix} \quad (2)$$

式中： $mt_{hn}$  为第  $h$  行第  $n$  列元素，表示第  $h$  种类型导弹对目标  $T_n$  的杀伤概率，取值范围为实数集  $(0,1)$ 。

5) 其他参数：引入 WSA 矩阵  $PT$  和制导矩阵  $MG$ ，作为本研究的决策变量：

$$PT = \begin{pmatrix} pt_{11} & pt_{12} & \cdots & pt_{1N} \\ pt_{21} & pt_{22} & \cdots & pt_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ pt_{sum(PM)1} & pt_{sum(PM)2} & \cdots & pt_{sum(PM)N} \end{pmatrix} \quad (3)$$

$$\begin{cases} MG = (MG_1 \quad MG_2 \quad \cdots \quad MG_{sum(PM)})^T \\ MG_i = [mg_{i1} \quad mg_{i2} \quad \cdots \quad mg_{ikmg(i)}] \\ i = 1, 2, \dots, sum(PM) \end{cases} \quad (4)$$

式中： $pt_{ij}$   $PT$  中的第  $i$  行第  $j$  列元素，取值为 0 或 1，

若取值为 1 则表示第  $i$  枚空地导弹攻击目标  $T_j$ ，1 在

$PT$  的每行元素中最多出现 1 次； $MG_i$  为第  $i$  枚导弹的制导序列，该导弹由制导站  $mg_{i1}$ 、 $mg_{i2}$ 、 $\dots$ 、 $mg_{ikmg(i)}$  依次接力制导， $kmg(i)$  表示为第  $i$  枚导弹进

行制导的制导站数量，若该导弹不攻击任何目标则  $MG_i = \emptyset$ ， $kmg(i)$  为 0。

目标综合生存概率为  $f_1$ ，总用弹量为  $f_2$ ， $f_1$  和  $f_2$  作为目标函数。

### 1.2.2 目标函数

本研究考虑目标综合毁伤概率最大和总用弹量最少两个优化目标。为了使各目标函数优化方向一致（即均为最小值优化），用目标综合生存概率最小代替目标综合毁伤概率最大的优化目标。两个目标函数计算公式如下：

1) 目标综合生存概率最小：

$$\begin{cases} \min f_1 = 1 - \sum_{n=1}^N v_n \left( 1 - \prod_{i=1}^{sum(PM)} (1 - mt_{type(i)n})^{PT(i,n)} \right) \\ type(i) = mt_{ie}, \quad PT(i,n) = pt_{in} \end{cases} \quad (5)$$

(5) 式以目标的价值系数为权重，对各目标的毁伤概率进行线性加权求和，获得目标综合毁伤概率，进而计算目标综合生存概率，(5) 式的推导过程见文献[8]。

2) 总用弹量最少：

$$\min f_2 = \sum_{i=1}^{sum(PM)} \sum_{j=1}^N pt_{ij} \quad (6)$$

(6) 式表示  $PT$  中所有元素之和为总用弹量。

### 1.2.3 约束条件

本研究考虑约束条件包括：决策变量取值范围约束、导弹攻击能力约束、目标最大毁伤次数约束、目标最小毁伤概率约束、制导区域约束、制导能力约束。

1) 决策变量取值范围约束：

$$pt_{ij} = 0, 1 \quad i = 1, 2, \dots, sum(PM), j = 1, 2, \dots, N \quad (7)$$

$$\begin{cases} mg_{ij} = 1, 2, \dots, K \\ mg_{i\alpha} \neq mg_{i\beta} \\ kmg(i) = 0, 1, \dots, K \\ j = 1, 2, \dots, kmg(i), i = 1, 2, \dots, sum(PM) \\ \alpha, \beta = 1, 2, \dots, kmg(i), \alpha \neq \beta \end{cases} \quad (8)$$

(7) 式表示  $PT$  中的元素取值为 0 或 1；(8) 式表示  $MG$  中元素的取值为制导站的编号，各制导站在同一导弹的制导序列中最多出现 1 次。

2) 导弹攻击能力约束：

$$\sum_{j=1}^N pt_{ij} \leq 1, \quad i = 1, 2, \dots, sum(PM) \quad (9)$$

(9) 式表示每一枚导弹最多攻击一个目标。

3) 目标最大毁伤次数约束：

$$\sum_{i=1}^{sum(PM)} pt_{ij} \leq dnm, \quad j = 1, 2, \dots, N \quad (10)$$

(10) 式表示攻击每个目标的目标数量最多为

$dnm$ 。

4) 目标最小毁伤概率约束:

$$\begin{cases} 1 - \prod_{i=1}^{sum(PM)} (1 - mt_{type(i)n})^{PT(i,n)} \geq dpm, n=1,2,\dots,N \\ type(i) = mtype_i, PT(i,n) = pt_{in} \end{cases} \quad (11)$$

(11)式表示每个目标的毁伤概率不小于  $dpm$ 。

5) 制导区域约束

$$\begin{cases} pp_{mp(i)} \in gxy_{MG(i,1)} \\ pt_n \in gxy_{MG(i,kmg(i))}, pt_{in} = 1 \\ gxy_{MG(i,\alpha)} \cap gxy_{MG(i,\alpha+1)} \neq \emptyset \\ gxy_k = \{(x,y) | (x - pgx_k)^2 + (y - pgy_k) \leq RG^2\} \\ mp(i) = mp_i, MG(i,j) = mg_{ij} \\ n=1,2,\dots,N, \alpha=1,2,\dots,kmg(i)-1, k=1,2,\dots,K \\ i=1,2,\dots,sum(PM), j=1,2,\dots,kmg(i) \end{cases} \quad (12)$$

(12)式表示每一枚导弹所属攻击机的位置应在其制导序列第一个制导站的制导范围内; 该导弹打击的目标应在其制导序列最后一个制导站的制导范围内; 每个制导序列中相邻制导站的制导范围应有交集, 以满足制导交接条件。

6) 制导能力约束:

$$\sum_{i=1}^{sum(PM)} fnum(k, MG_i) \leq CG, k=1,2,\dots,K \quad (13)$$

式中:  $fnum(x,y)$  表示  $x$  在向量  $y$  中出现的次数。(13)式表示每个制导站最多承担  $CG$  枚空地导弹的制导任务。

## 2 DSMPNSGA-II 原理

带精英策略的非支配排序遗传算法(NSGA-II)<sup>[21]</sup>是目前最流行的多目标优化算法之一, 具有运算速度较快、收敛性较好等优点, 但也存在算法早熟、最优解多样性差的问题。本文需要对WTA方案和接力制导方案进行规划, 求解空间以及约束条件都较为复杂。为将NSGA-II合理应用于WSA与制导序列优化模型的求解, 并克服传统NSGA-II的算法早熟问题, 本文提出基于双序列编码的多种群NSGA-II。本节内容对DSMPNSGA-II的关键步骤进行介绍, 其中涉及非支配排序、拥挤距离计算、精英保留等操作与传统NSGA-II相同, 具体流程见文献[22]。

### 2.1 编码与解码

#### 2.1.1 编码与解码基本步骤

设计种群中每个个体包含两条染色体, 编码分别为  $st$  和  $sg$ 。 $st$  为WTA序列,  $sg$  为制导站序列, 两序列编码分别为

$$\begin{cases} st = [st_1 \ st_2 \ \dots \ st_{sum(PM)}] \\ st_i \in \{0,1,\dots,N\} \ i=1,2,\dots,sum(PM) \end{cases} \quad (14)$$

$$\begin{cases} sg = [sg_1 \ sg_2 \ \dots \ sg_{K \times CG}] \\ sg_i \in \{1,2,\dots,K\} \ i=1,2,\dots,K \times CG \\ fnum(k, sg) = CG \ k=1,2,\dots,K \end{cases} \quad (15)$$

(14)式表示  $st$  中  $st_i$  的取值为第  $i$  枚导弹的打击目标, 若取值为0则该导弹不参与打击任务; (15)式表示每个制导站在  $sg$  中均出现  $CG$  次。

在编码时需初始化  $st$  中每个目标至少出现1次 (在目标最小毁伤概率约束下, 各目标最少需要被攻击1次), 最多出现  $dnm$  次。

解码过程如下所示:

1)对  $st$  进行解码, 初始化  $PT$  中所有元素均为0, 对于  $i=1,2,\dots,sum(PM)$ , 若  $st_i \neq 0$ , 则更新  $pt_{in} = 1$ , 其中  $n = st_i$ ;

2)根据(12)式和(13)式判断  $PT$  是否满足目标最大毁伤次数约束和最小毁伤概率约束, 若不满足约束则赋值  $f_1 \gg 1$ 、 $f_2 \gg sum(PM)$ , 解码结束, 否则继续解码  $sg$ , 初始化  $l=1$ , 执行步骤3;

3)根据  $PT$  确定第  $l$  枚导弹的攻击目标, 若该导弹无攻击目标则跳转至步骤5, 否则根据导弹发射位置 (即所属攻击机位置) 和目标位置, 对  $sg$  的剩余制导站进行搜索, 获得该导弹的制导序列  $MG_l$ , 若无可行制导序列, 则赋值  $f_1 \gg 1$ 、 $f_2 \gg sum(PM)$ , 解码结束, 否则执行步骤4;

4)将  $MG_l$  中的制导站从  $sg$  中删除, 每个制导站仅删除一次;

5)更新  $l=l+1$ ;

6)重复步骤3~5, 直到  $l > sum(PM)$ ;

7)计算目标函数值  $f_1$  和  $f_2$ ;

8)输出决策变量  $PT$  和  $MG$ , 目标函数值  $f_1$  和  $f_2$ , 解码结束。

在解码的步骤3中, 导弹制导序列搜索方法为DFS-DJ (Depth-First-Search- Dijkstra) 方法, 具体步骤在2.1.2节给出。

#### 2.1.2 空地导弹制导序列搜索方法

本研究提出DFS-DJ搜索方法, 用于空地导弹制导序列搜索, 实现制导站序列的解码。给定第  $l$  枚导弹所属攻击机位置、打击目标位置、剩余制导站序列  $sgl$ , 制导序列搜索过程为:

1)对  $sgl$  进行去重, 每个制导站仅保留第一次出现的位置, 去重后的制导站序列为  $sgl'$ ;

2)构建无向无权图  $Gdfs = \langle Vdfs, Edfs \rangle$ , 顶点集  $Vdfs$  中的元素依次为起始点 (对应攻击机位置)、中间点 (依次对应  $sgl'$  中的制导站位置)、目标点 (对应目标位置), 根据各制导站的制导区域是否覆盖攻击机与目标位置, 以及制导区域是否有交集, 构建边集  $Edfs$ ;

3)对  $Gdfs$  进行深度优先搜索 (DFS), 获得制导区域连通起始点到目标点的制导站序列  $sgl''$ ;

4)根据攻击机位置、目标位置和  $sgl''$  中的制导站位置重新构建无向无权图  $Gdj = \langle Vdj, Edj \rangle$ , 并采用Dijkstra算法<sup>[23]</sup>获得导弹的制导序列  $MG_i$ 。

Dijkstra算法能够获得导弹的最短制导序列, 减少制导站的占用, 提升后续导弹制导序列搜索的成功率。本研究在使用Dijkstra算法搜索之前先进行DFS, 而不是直接使用Dijkstra算法, 原因如下:

1)  $sgl$  中的制导站数量较多, 对应搜索节点较多, 导致Dijkstra算法运算速度较慢, 通过DFS可以减少制导站数量, 进而减少搜索节点数量, 提升运算速度。

2) DSMPNSGA-II算法旨在通过优化  $sg$  中制导站的次序来优化  $MG$  中各导弹的制导序列, Dijkstra算法在搜索过程中会遍历当前节点的所有相邻节点,  $sg$  中的制导站次序对Dijkstra算法结果影响较小, 而对DFS结果影响较大, 因此在Dijkstra算法前加入DFS算法, 有利于提升DSMPNSGA-II算法的搜索能力。

### 2.1.3 编码与解码示例

这里通过一个例子说明编码与解码的过程, 如图2所示, 本例包含3架攻击机、2个目标和4个制导站, 设每个攻击机携带1枚导弹,  $CG$  为2。

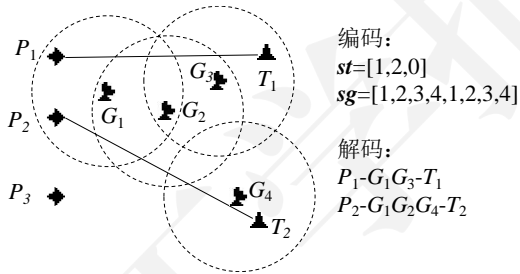


图2 编码和解码示意图

Fig.2 Encoding and decoding diagram

假设某一个体的两条染色体编码为  $st=[1,2,0]$ ,  $sg=[1,2,3,4,1,2,3,4]$ , 解码过程如下:

1) 根据  $st$  确定  $P_1$  发射导弹打击  $T_1$ ,  $P_2$  发射导弹打击  $T_2$ ,  $P_3$  携带的导弹不参与打击任务;

2) 设  $sgl = sg$ , 对  $sgl$  去重得到  $sgl'=[1,2,3,4]$ , 根据  $P_1$  和  $T_1$  的位置以及  $sgl'$ , 通过深度优先搜索获得  $sgl''=[1,2,3]$ , 再使用Dijkstra算法获得第一枚导弹的制导序列  $MG_1=[1,3]$ ;

3) 从  $sg$  中将1和3均删除一次, 更新制导站序列为  $sg=[2,4,1,2,3,4]$ ;

4) 根据  $P_2$  和  $T_2$  的位置以及  $sg$ , 使用DFS算法和Dijkstra算法, 获得  $MG_2=[1,2,4]$ ;

5) 由于  $P_3$  携带的导弹不参与打击任务, 故解码结束, 得到  $PT$  和  $MG$  如下所示:

$$PT = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \quad MG = \begin{pmatrix} 1 & 3 \\ 1 & 2 & 4 \\ \emptyset \end{pmatrix}$$

根据  $PT$  和  $MG$  计算  $f_1$  和  $f_2$ 。

可以看出, 本研究设计的编码与解码方式建立了从  $st$  和  $sg$  到  $PT$  和  $MG$  的映射, 后续通过遗传算法的选择、交叉、变异等操作, 实现对  $st$  和  $sg$  的优化, 进而实现对WTA方案及导弹制导序列的优化。

## 2.2 选择、交叉与变异

### 2.2.1 选择算子

选择算子采用二元锦标赛选择法<sup>[24]</sup>。在选择之前已对种群进行了非支配排序与拥挤距离计算。对于待选择的两个个体, 选择Pareto等级较小的个体进入新种群; 若Pareto等级相同, 则选择拥挤距离较大的个体进入新种群; 若拥挤距离也相同, 则随机选择一个个体进入新种群。

### 2.2.2 交叉算子

设计交叉算子基本步骤如下所示 (其中对  $sg$  的交叉参考文献[19]):

1) 设交叉概率为  $pc(0 \leq pc \leq 1)$ , 待交叉的两个个体的染色体分别为  $st1$  和  $sg1$ 、 $st2$  和  $sg2$ ;

2) 生成随机数  $0 \leq r \leq 1$ , 若  $r > pc$  则两个体不进行交叉, 否则交叉继续进行;

3)  $st1$  与  $st2$  进行交叉, 生成随机整数  $i$  和  $j$  满足  $1 \leq i < j \leq \text{sum}(PM)$ , 将  $st1$  和  $st2$  中从第  $i$  位到第  $j$  位的基因相互交换, 每完成一位交换后判断  $st1$  和  $st2$  是否满足目标毁伤次数约束 (最小为1, 最大为  $d_{nm}$ ), 若违反约束, 则将两基因再次交换 (即状态还原), 最终得到两条新的染色体  $st1'$  和  $st2'$ ;

4)  $sg1$  与  $sg2$  进行交叉, 生成随机整数  $i$  和  $j$  满足  $1 \leq i < j \leq K \times CG$ , 设  $sg1$  和  $sg2$  中从第  $i$  位到第  $j$  位的基因片段分别为  $s1$  和  $s2$ ;

5) 在  $sg2$  中寻找  $s1$  中的所有基因, 并记录基因相对位置为  $s1'$ , 在  $sg1$  中寻找  $s2$  中的所有基因, 并记录基因相对位置为  $s2'$  (这里对同一制导序列中多次出现的相同制导站进行严格区分, 确保序列中各基因的唯一性);

6) 在  $sg1$  中用  $s1'$  替换  $s1$ , 在  $sg2$  中用  $s2'$  替换  $s2$ , 得到两条新的染色体  $sg1'$  和  $sg2'$ 。

这里给出  $st$  与  $sg$  的交叉示例:

①假设  $st1=[1,2,2,3]$ ,  $st2=[3,1,1,2]$ , 交叉位置为第1位和第2位, 首先交换  $st1$  和  $st2$  的第1位基因, 交换后  $st1$  不满足目标1的毁伤次数约束 (无导弹攻击目标1, 必然违反目标最小毁伤概率约束), 故两

基因再次交换；然后交换  $st1$  和  $st2$  的第2位基因，不违反目标毁伤次数约束；交叉后的基因序列为  $st1'=[1,1,2,3]$ ， $st2'=[3,2,1,2]$ 。

②假设  $sg1=[1,2,3,4,5,6]$ ， $sg2=[1,3,5,2,4,6]$ ，若选取  $sg1$  中的片段  $s1=[2,3,4]$ ，则  $s1'=[3,2,4]$ （因为在  $sg2$  中，基因2、3、4的相对位置为3-2-4），即  $sg1$  变为  $sg1'=[1,3,2,4,5,6]$ ，对  $sg2$  执行类似的操作。

### 2.2.3 变异算子

设计变异算子基本步骤如下所示（其中对  $sg$  的变异参考文献[19]）：

1) 设变异概率为  $pm(0 \leq pm \leq 1)$ ，待变异个体的两条染色体为  $st$  和  $sg$ ；

2) 生成随机数  $0 \leq r \leq 1$ ，若  $r > pm$  则该个体不进行变异，否则变异继续进行；

3) 生成随机整数  $1 \leq i \leq \text{sum}(PM)$ ，设  $st$  的第  $i$  位基因为  $n0$ ，生成随机整数  $0 \leq n1 \leq N$ ，用  $n1$  替换  $n0$ ；

4) 若  $n0 \neq 0$ ，且  $\text{fnum}(n0, st) < 1$ ，则随机选择一位  $st$  中值为0的基因（非第  $i$  位基因），将其值变为  $n0$ ；若  $n1 \neq 0$ ，且  $\text{fnum}(n1, st) > dnm$ ，则随机选择一位  $st$  中值为  $n1$  的基因（非第  $i$  位基因），将其值变为0；得到新的染色体  $st'$ ；

5) 随机选取  $sg$  中的某一片段  $s$ ，将  $s$  随机打乱为  $s'$ ，用  $s'$  替换  $s$ ，得到新的染色体  $sg'$ 。

这里给出  $st$  和  $sg$  的变异示例：

①假设  $st=[2,1,1,1,0,3]$ ，变异位置为第1位， $st$  的第1位基因为  $n0=2$ ，生成随机数  $n1=1$ ，则  $st$  变为  $st=[1,1,1,1,0,3]$ ；判断攻击目标2的导弹数量小于1，需要将  $st$  的第5位基因由0变为2；假设  $dnm=3$ ，则判断攻击目标1的导弹数量超过目标最大毁伤次数，需要将  $st$  的第2、3、4位基因中的一位由1变为0；最终得到新的染色体  $st'$ 。

②假设  $sg=[1,2,3,2,1,3]$ ，选取  $s=[2,3,2]$ ，则  $s'$  的可能取值为  $[2,2,3]$ 、 $[2,3,2]$ 、 $[3,2,2]$ ，用  $s'$  替换  $s$  得到新的染色体  $sg'$ 。

将本文研究设计的交叉算子和变异算子作用于原本可行的个体（即解码结果不违反约束），操作后的个体解码结果必然满足1.2.3节的约束1、2、3、6，减少非可行解的产生。对于违反约束4或5的个体（即目标毁伤概率达不到最小毁伤概率，或找不到可行的制导序列），通过选择操作进行淘汰。

## 2.3 多种群 NSGA-II

多种群 NSGA-II 设置多个子种群，各子种群采用不同的控制参数，在迭代进化的过程中，通过移民算子，将各子种群中较为优秀的个体加入其它子种群，实现各子种群的协同进化，有利于克服传统遗传算法的算法早熟问题。

本研究设计多种群 NSGA-II 的子种群数量为  $nGA$ ，各子种群规模均为  $nscale$ ，迭代次数均为  $nev$ ，第  $i$  个子种群的初始交叉概率为  $pc0_i$ ，最终交叉概率为  $pc1_i$ ，初始变异概率为  $pm0_i$ ，最终变异概率为  $pm1_i$ ，第  $i$  个种群在第  $j$  轮迭代中的交叉概率为  $pc_{ij}$ ，变异概率为  $pm_{ij}$ ，计算公式为

$$\begin{cases} pc_{ij} = pc0_i - j \times (pc0_i - pc1_i) / nev \\ pm_{ij} = pm0_i - j \times (pm0_i - pm1_i) / nev \\ i = 1, 2, \dots, nGA, j = 1, 2, \dots, nev \end{cases} \quad (16)$$

在每一代进化中，各子种群均完成精英保留操作之后，进行移民操作，步骤如下：

1) 对于每个子种群，选出所有 Pareto 等级为 1 的个体，目标函数值相同的多个个体只保留一个，获得各子种群的移民个体；

2) 将第  $i$  个子种群的所有移民个体复制添加至第  $i+1$  个子种群（ $i=1, 2, \dots, nGA-1$ ），第  $nGA$  个子种群的所有移民个体复制添加至第 1 个子种群；

3) 各子种群重新进行非支配排序、拥挤距离计算和精英保留。

## 2.4 最终方案选择

决策者根据任务需求或自身偏好，从最终种群中选择某一个体，解码获得 WSA 方案与接力制导方案，作为最终方案。

## 2.5 算法流程

DSMPNSGA-II 流程如图3所示。

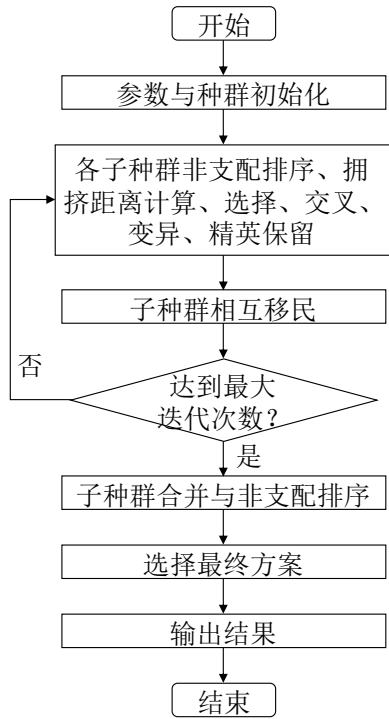


图3 DSMPNSGA-II 算法流程图

Fig.3 DSMPNSGA-II algorithm flow diagram

具体流程如下：

- 1)参数与种群初始化，初始化参数包括攻击机、目标、制导站、导弹参数及遗传算法相关参数，种群初始化随机生成各子种群各个体的染色体基因序列，并解码计算目标函数；
- 2)各子种群进行非支配排序、拥挤距离计算、选择、交叉、变异、精英保留等操作；
- 3)子种群进行移民操作；
- 4)重复步骤2和步骤3，直达到最大迭代次数；
- 5)将各子种群合并，并进行非支配排序；
- 6)从非支配个体中选择某一个体，解码获得最终WSA与制导序列结果；
- 7)输出最终方案。

### 3 仿真与算例分析

为了验证本研究提出的基于双序列编码的多种群NSGA-II算法求解空地多目标攻击WSA与制导序列优化问题的有效性，在Intel(R) Core(TM) i9-10900K 3.7GHz CPU、16.0 GB RAM、Windows 7实验平台上，在Python 3.6实验环境下进行仿真计算。

#### 3.1 参数说明

作战区域东西距离为400km，南北距离为200km；攻击机数量  $M$  为10；目标数量  $N$  为5，目标最小毁伤概率  $dpm$  为0.6，最大毁伤次数  $dnm$  为3，各目标价值系数  $v_1 \sim v_5$  分别为0.2、0.1、0.3、0.25、0.15；制导站数量  $K$  为25，制导半径  $RG$  为60km，制导能力  $CG$  为3。各攻击机位置、目标位置和制导站位置如

表1所示。

表1 攻击机、目标、制导站位置

Table 1 Locations of attack planes, targets and guidance stations

类别	位置坐标/km
攻击机	(24,13), (37,29), (29,54), (19,73), (29,91), (37,116), (29,135), (20,155), (20,177), (26,193)
目标	(387,27), (387,67), (387,107), (387,147), (387,187)
制导站	(50,21), (129,17), (203,16), (261,19), (343,27), (57,53), (125,53), (207,56), (279,64), (335,62), (51,100), (132,97), (190,103), (275,94), (335,93), (61,136), (134,142), (208,143), (265,146), (332,142), (69,177), (138,181), (209,181), (267,181), (332,179)

空地导弹类型数量  $H$  为3，导弹数量为25，攻击机导弹配置矩阵  $PM$  如下：

$$PM = \begin{pmatrix} 0 & 1 & 2 & 1 & 0 & 1 & 0 & 2 & 1 & 1 \\ 2 & 0 & 0 & 0 & 1 & 2 & 1 & 0 & 1 & 1 \\ 0 & 2 & 0 & 2 & 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}^T$$

导弹杀伤概率矩阵  $MT$  如下：

$$MT = \begin{pmatrix} 0.85 & 0.4 & 0.75 & 0.35 & 0.3 \\ 0.35 & 0.61 & 0.49 & 0.63 & 0.45 \\ 0.48 & 0.59 & 0.38 & 0.55 & 0.8 \end{pmatrix}$$

作战场景如图4所示，图4展示了本算例中各攻击机、目标、制导站的位置与编号，以及各制导站的制导区域。

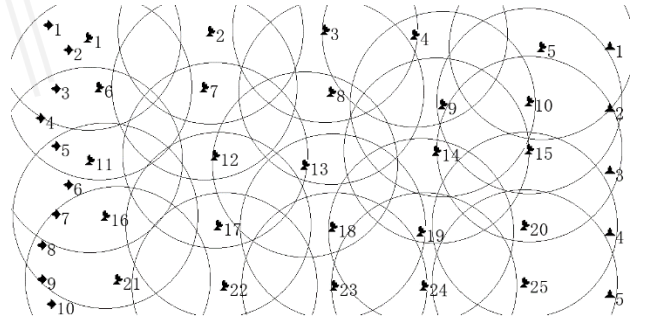


图4 算例作战场景示意图

Fig.4 Combat scene diagram of the case

遗传算法参数为：子种群数量  $nGA$  为3，种群规模  $n_{scale}$  为50，迭代次数  $n_{ev}$  为30，3个种群的初始交叉概率分别为0.5、0.8、0.5，最终交叉概率分别为0.1、0.3、0.1，初始变异概率分别为0.1、0.1、0.3，最终变异概率分别为0.02、0.02、0.1。

#### 3.2 仿真结果

经过仿真计算，得到10种非支配的WSA与制导序列结果，各结果的目标综合生存概率 ( $f_1$ )、总用弹量 ( $f_2$ ) 如表2所示。

表2 算例结果

Table 2 Results of the case

结果序号	$f_1$	$f_2$	结果序号	$f_1$	$f_2$
1	0.2665	5	6	0.0795	10
2	0.2082	6	7	0.0579	11
3	0.1520	7	8	0.0482	12



4	0.1265	8	9	0.0471	13
5	0.1025	9	10	0.0444	14

在表2中,完成空地多目标攻击任务所需的最少用弹量为5,此时每个目标仅被一枚空地导弹攻击,满足各目标最小毁伤次数与最小毁伤概率约束;随着用弹量的不断增加,目标的综合生存概率不断减小,即综合毁伤概率不断增大;最大用弹量为14,受到目标最大毁伤次数与制导站制导能力的限制,用弹量无法继续增加,综合毁伤概率无法继续提升。

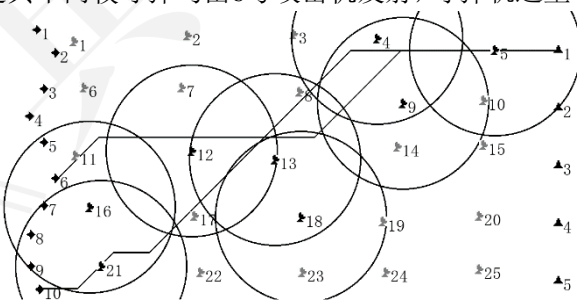
决策者根据任务需求或自身偏好,从表2中选择一种方案作为最终方案。例如打击任务要求目标的综合毁伤概率达到0.93,即目标综合生存概率不得超过0.07,则从结果7~10中选择用弹量最少的方案,即将结果7对应的WSA方案与接力制导方案作为最终方案,方案详情如表3所示。

表 3 最终方案详情

Table 3 Details of the final scheme

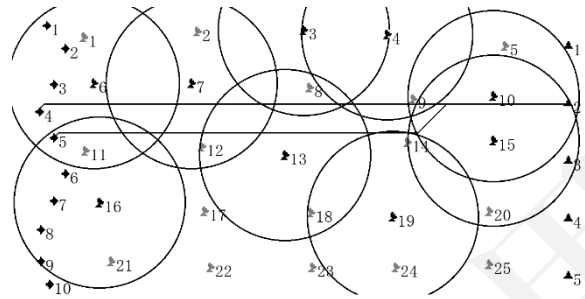
目标编号	导弹型号	导弹所属攻击机	制导序列
1	1	6	16-12-18-9-5
	1	10	21-12-13-4-5
2	3	4	6-7-3-4-15
	2	5	16-7-13-19-10
3	1	3	6-17-23-24-15
	1	4	11-7-8-4-15
	2	6	11-22-23-14-20
4	2	6	11-17-18-24-20
	2	10	21-12-8-19-20
5	3	2	1-16-22-18--24-25
	3	5	6-2-3-14-25

绘制最终WSA与接力制导方案如图5所示,图5(a)~图5(e)展示了参与打击任务的各导弹航迹以及为导弹进行接力制导的制导站,各导弹航迹使用A\*算法<sup>[25]</sup>规划得到。其中对于图5(d),打击目标4的空地导弹有3枚,而图中只显示了2条导弹轨迹,原因是其中两枚导弹均由6号攻击机发射,导弹轨迹重合。



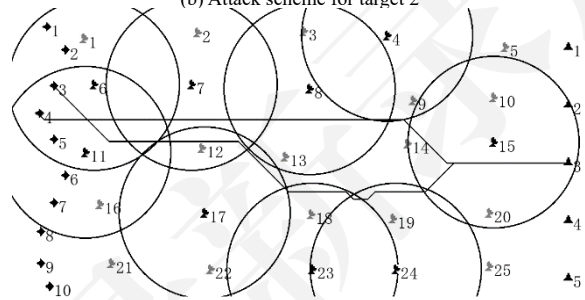
(a) 目标 1 打击方案

(a) Attack scheme for target 1



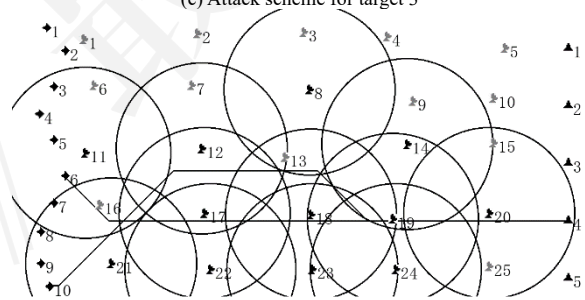
(b) 目标 2 打击方案

(b) Attack scheme for target 2



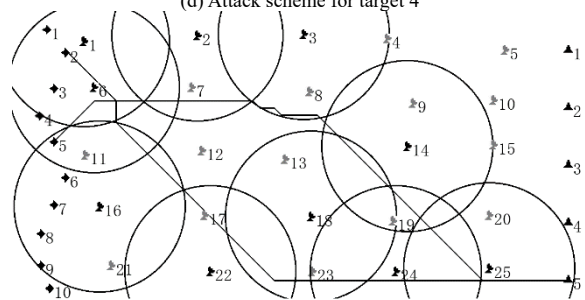
(c) 目标 3 打击方案

(c) Attack scheme for target 3



(d) 目标 4 打击方案

(d) Attack scheme for target 4



(e) 目标 5 打击方案

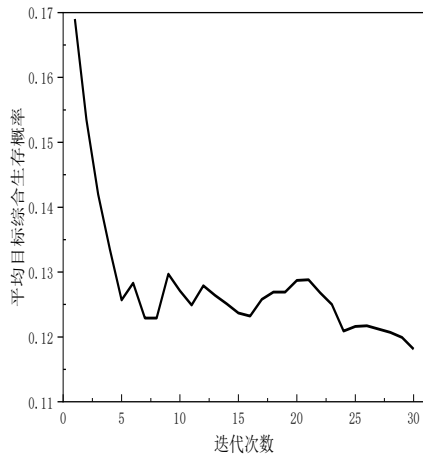
(e) Attack scheme for target 5

图 5 最终 WSA 与接力制导方案

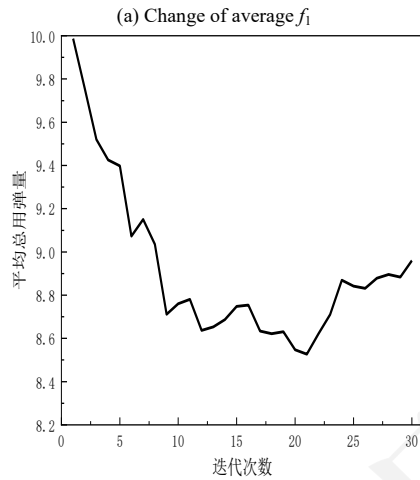
Fig.5 Final scheme of WTA and relay guidance

为充分展示DSMPNSGA-II算法的优化效果,将算法运行20次,绘制平均 $f_1$ 、 $f_2$ 的变化曲线如图6所示。平均目标函数值计算方法为:记录每次运行的每一轮迭代中,种群所有个体 $f_1$ 、 $f_2$ 的平均值,并计算相同迭代次数下,各目标函数平均值对算法运行次数的平均值。





(a) 平均 $f_1$ 变化



(b) 平均 $f_2$ 变化

(b) Change of average  $f_2$

图6 平均目标函数值变化曲线

Fig.6 Change of average objective function values diagram

从图6可以看出,在迭代初期, $f_1$ 与 $f_2$ 均呈下降趋势,故DSMPNSGA-II能够有效实现空地多目标攻击WSA与制导序列的优化;而在迭代后期,两目标函数值均出现波动,平均总用弹量甚至略有上升,原因是两个目标函数存在耦合,不会同时达到最小,并且迭代后期,很多个体已成为非支配最优个体,继续迭代只会将一种非支配最优个体转化为另一种非支配最优个体,若出现大量个体向 $f_1$ 减小、 $f_2$ 增加方向转化的情况,则平均用弹量上升。

### 3.3 算法对比

对本研究提出的DSMPNSGA-II,与单种群NSGA-II以及文献[26]提出的多目标离散粒子群优化(MODPSO)算法分别进行对比,对比算法参数设置如下:

1) 单种群NSGA-II: 种群规模为150(与多种群NSGA-II的子种群规模之和相同),初始交叉概率为0.5,最终交叉概率为0.1,初始变异概率为0.1,最终变异概率为0.02;

2) MODPSO算法: 种群规模为150,初始变异概率为0.1,最终变异概率为0.02, Pareto外部档案最大规模为50。

通过3个算例进行对比,各算例的主要区别为制导站的数量 $K$ 与制导能力 $CG$ ,参数设置如下:

各算例参数设置如下:

- 1) 算例1: 算例参数与3.1节所述参数相同;
- 2) 算例2:  $CG$ 为2,其余参数与算例1相同;
- 3) 算例3:  $K$ 为17(从图4中删去编号为2、4、6、8、16、17、23、24的制导站),其余参数与算例1相同。

在每个算例中,三种算法均运行20次,算法性能的评价指标如下:

- 1) 平均计算时长: 算法每次运行时长的平均值,反映算法的计算速度;
- 2) 平均非支配解数量: 算法每次运行所得非支配解种类数的平均值,反映算法优化结果的多样性;
- 3) 平均非支配比率: 将不同算法的非支配解合并,并进行非支配排序,获得总非支配解集,各算法原非支配解在总非支配解集中所占比例即为各算法的非支配比率<sup>[27]</sup>,该指标反映算法所得非支配解的质量。

各算例对应的算法对比结果如表4~表6所示。

表4 算法对比结果表(算例1)

Table 4 Algorithm comparison results table for Case 1

算法	平均计算时长	平均非支配解数量	平均非支配比率
DSMPNSGA-II	25.88s	8.10	46.08%
单种群 NSGA-II	24.27s	7.05	20.08%
MODPSO 算法	104.89s	7.65	33.84%

注: 算例1中,制导站数量为25,制导能力为3

表5 算法对比结果表(算例2)

Table 5 Algorithm comparison results table for Case 2

算法	平均计算时长	平均非支配解数量	平均非支配比率
DSMPNSGA-II	21.46s	3.55	45.17%
单种群 NSGA-II	20.38s	2.95	14.23%
MODPSO 算法	75.83s	3.35	40.60%

注: 算例2中,制导站数量为25,制导能力为2

表6 算法对比结果表(算例3)

Table 6 Algorithm comparison results table for Case 3

算法	平均计算时长	平均非支配解数量	平均非支配比率
DSMPNSGA-II	18.01s	3.85	47.61%
单种群 NSGA-II	17.46s	3.00	10.83%
MODPSO 算法	66.11s	3.80	41.56%

注: 算例3中,制导站数量为17,制导能力为3

从表4~表6可以看出,对于3种多目标优化算法,

在算例1中,平均计算时长更长、平均非支配解数量更多。与算例1相比,算例2降低了制导站的制导能力,算例3减少了制导站的数量,两者都会造成可行WSA与接力制导方案数量的下降,即平均非支配解数量的下降,但也使得算法搜索空间变小、染色体长度缩短,算法操作得以简化,因此算法速度有所提升,即平均计算时长下降。

三种算法对比结果如下:

1) 计算速度方面,DSMPNSGA-II算法略慢于单种群NSGA-II算法(平均计算时长在算例1~算例3中分别落后1.61s、1.08s、0.55s),而明显快于MODPSO算法(平均计算时长在算例1~3中分别提升79.01s、54.37s、48.10s)。算法的计算量主要在对个体的解码并计算目标函数值上,一旦个体基因发生了改变,则需要重新计算目标函数值。DSMPNSGA-II的第2和第3种群分别拥有更大的交叉概率与变异概率,与单种群NSGA-II相比,个体交叉或变异次数更多,因此DSMPNSGA-II的计算速度略慢于单种群NSGA-II;而MODPSO算法的每个个体在每轮迭代中都会进行交叉操作,并重新解码计算目标函数值,因此计算速度最慢;

2) 优化结果多样性方面,DSMPNSGA-II显著优于单种群NSGA-II(平均非支配解数量在算例1~3中分别提升1.05、0.60、0.85),略优于MODPSO算法(平均非支配解数量在算例1~3中分别提升0.45、0.20、0.05)。可以看出,采用了多种群策略的DSMPNSGA-II算法能够获得更为多样的WSA与接力制导方案。

3) 非支配解质量方面,与单种群NSGA-II和MODPSO算法相比,DSMPNSGA-II的非支配比率更高(在算例1~3中的平均非支配比率比单种群NSGA-II算法分别高26.00%、30.94%、36.78%,比MODPSO算法分别高12.24%、4.57%、6.05%),因此DSMPNSGA-II算法的求解质量更高。

综上,根据不同制导站数量与制导能力下的算例仿真与算法对比结果,DSMPNSGA-II在计算速度上略慢于单种群NSGA-II,但明显快于MODPSO算法;在优化结果多样性以及求解质量方面,均优于单种群NSGA-II和MODPSO算法。在优化问题规模较大、可行WSA与接力制导方案数量较多时,DSMPNSGA-II算法的优势更为明显。

## 4 结论

本文对远距离空地多目标攻击中亟需解决的WSA与空地导弹接力制导规划问题进行了研究,主要工作如下:

1) 构建远距离空地多目标攻击作战场景,并建

立多目标、多约束WSA与制导序列优化模型,以目标综合生存概率最小和总用弹量最少为优化目标,综合考虑攻击机导弹配置、导弹毁伤能力、目标毁伤要求、制导站制导性能等约束条件。

2) 提出基于双序列编码的多种群NSGA-II算法,对WSA与制导序列优化模型进行求解。建立WSA序列和制导站序列到WSA方案与接力制导方案的映射,通过DFS-DJ搜索方法、改进的染色体交叉与变异操作、多种群策略等,提升算法性能。

3) 通过算例仿真,验证了DSMPNSGA-II算法求解WSA与制导序列优化问题的有效性,并将DSMPNSGA-II算法与单种群NSGA-II算法以及MODPSO算法进行比较,验证了DSMPNSGA-II算法在计算速度、优化结果多样性、求解质量等方面的优越性。

本研究有助于提升远距离空地多目标攻击的成功率、减少资源消耗。未来研究拟考虑更为复杂的战场环境,加入地形、禁飞区、敌方威胁等约束条件,并提升算法计算速度。

## 参考文献 (References)

- [1] 徐宏伟,李鹏,王玄. 无人机载空地导弹关键技术研究[J]. 弹箭与制导学报, 2018, 38(6):64-67.  
XU H W, LI P, WANG X. Research on technology of UAV-borne air-to-ground missile[J]. Journal of Projectiles, Rockets, Missiles and Guidance, 2018, 38(6):64-67. (in Chinese)
- [2] 石章松,吴玲,吴中红,等. 多平台协同制导技术及应用[M]. 北京: 电子工业出版社, 2019:4-18.  
SHI Z S, WU L, WU Z H, et al. Technology and application of multi-platform cooperative guidance[M]. Beijing: Publishing House of Electronics Industry, 2019:4-18. (in Chinese)
- [3] 张进,郭浩,陈统. 基于可适应匈牙利算法的武器-目标分配问题[J]. 兵工学报, 2021, 42(6):1339-1344.  
ZHANG J, GUO H, CHEN T. Weapon-target assignment based on adaptable hungarian algorithm[J]. Acta Armamentarii, 2021, 42(6):1339-1344. (in Chinese)
- [4] KLINE A, AHNER D, HILL R. The weapon-target assignment problem[J]. Computers & Operations Research, 2019, 105: 226-236.
- [5] BOGDANOWICZ Z R. Advanced input generating algorithm for effect-based weapon-target pairing optimization[J]. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 2011, 42(1):276-280.
- [6] 苗李达,姜青山. 基于毁伤积累的武装直升机对地火力分配[J]. 火力与指挥控制, 2018, 43(5):167-171.

- MIAO L D, JIANG Q S. Armed helicopter-to-ground attack firepower distribution based on damage accumulation[J]. Fire Control & Command Control, 2018, 43(5):167-171. (in Chinese)
- [7] AHUJA R K, KUMAR A, JHA K. Exact and heuristic algorithms for the weapon-target assignment problem[J]. Operations Research, 2007, 55(6):1136-1146.
- [8] 夏维, 刘新学, 范阳涛, 等. 基于改进型多目标粒子群优化算法的武器-目标分配[J]. 兵工学报, 2016, 37(11):2085-2093.  
XIA W, LIU X X, FAN Y T, et al. Weapon-target assignment with an improved multi-objective particle swarm optimization algorithm[J]. Acta Armamentarii, 2016, 37(11):2085-2093. (in Chinese)
- [9] 王然辉, 王超. 面向对地打击武器-目标分配问题的遗传算法变量取值控制技术[J]. 兵工学报, 2016, 37(10):1889-1895.  
WANG R H, WANG C. Variable value control technology of genetic algorithm for WTA of ground target attacking[J]. Acta Armamentarii, 2016, 37(10):1889-1895. (in Chinese)
- [10] CHANG T Q, KONG D, HAO N, et al. Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization[J]. Applied Soft Computing, 2018, 70: 845-863.
- [11] 龙腾, 刘震宇, 史人赫, 等. 基于神经网络的防空武器目标智能分配方法[J]. 空天防御, 2021, 4(1):1-7.  
LONG T, LIU Z Y, SHI R H, et al. Neural network based air defense weapon target intelligent assignment method[J]. Air & Space Defense, 2021, 4(1):1-7. (in Chinese)
- [12] 王邑, 孙金标, 肖明清, 等. 基于类型 2 区间模糊 K 近邻分类器的动态武器-目标分配方法研究[J]. 系统工程与电子技术, 2016, 38(6):1314-1319.  
WANG Y, SUN J B, XIAO M Q, et al. Research of dynamic weapon-target assignment problem based on type-2 interval fuzzy K-nearest neighbors classifier[J]. Systems Engineering and Electronics, 2016, 38(6):1314-1319. (in Chinese)
- [13] 阎栋, 苏航, 朱军. 基于 DQN 的反舰导弹火力分配方法研究[J]. 导航定位与授时, 2019, 6(5):18-24.  
YAN D, SU H, ZHU J. Research on fire distribution method of anti-ship missile based on DQN[J]. Navigation Positioning and Timing, 2019, 6(5):18-24. (in Chinese)
- [14] 孙海文, 谢晓方, 庞威, 等. 基于改进火力分配模型的综合防空火力智能优化分配[J]. 控制与决策, 2020, 35(5):1102-1112.  
SUN H W, XIE X F, PANG W, et al. Integrated air defense firepower intelligence optimal assignment based on improved firepower assignment model[J]. Control and Decision, 2020, 35(5):1102-1112. (in Chinese)
- [15] SONUC E, SEN B, BAYIR S. A Parallel Simulated Annealing Algorithm for Weapon-Target Assignment Problem[J]. International Journal of Advanced Computer Science and Applications, 2017, 8(4):87-92.
- [16] BOGDANOWICZ Z R, TOLANO A, PATEL K, et al. Optimization of weapon-target pairings based on kill probabilities[J]. IEEE Transactions on Cybernetics, 2013, 43(6):1835-1844.
- [17] 刘海鸥, 晋磊, 董诗瑾, 等. 重型自动机械变速车辆换挡序列优化[J]. 兵工学报, 2015, 36(8):1377-1383.  
LIU H O, JIN L, DONG S J, et al. Shift sequence optimization of heavy-duty vehicles with automated manual transmission[J]. Acta Armamentarii, 2015, 36(8):1377-1383. (in Chinese)
- [18] 赵晶, 苏东楠. 多目标混合装配线平衡方法优化[J]. 计算机仿真, 2019, 36(2):145-148.  
ZHAO J, SU D N. Optimization of balancing method for multi-objective hybrid assembly line[J]. Computer Simulation, 2019, 36(2):145-148. (in Chinese)
- [19] XU S F, BI W H, ZHANG A, et al. Optimization of flight test tasks allocation and sequencing using genetic algorithm[J]. Applied Soft Computing, 2022, 115:1-16.
- [20] 窦建平, 李俊, 苏春. 基于可行工序序列遗传算法的工序排序优化[J]. 计算机集成制造系统, 2019, 25(8):1981-1990.  
DOU J P, LI J, SU C. Optimization of operation sequencing based on feasible operation sequence oriented genetic algorithm[J]. Computer Integrated Manufacturing Systems, 2019, 25(8):1981-1990. (in Chinese)
- [21] 冯睢睢, 张发平, 王武宏, 等. 基于非支配排序遗传算法的涡轮增压发动机转子系统装配参数优化[J]. 兵工学报, 2021, 42(5):1092-1100.  
FENG K K, ZHANG F P, WANG W H, et al. Assembly process parameters optimization of turbine engine rotor system based on non-dominated sorting genetic algorithm [J]. Acta Armamentarii, 2021, 42(5):1092-1100. (in Chinese)
- [22] 任禹谋. 高速铁路大型客运站到发线分配优化策略及方法研究[D]. 北京:中国铁道科学研究院, 2021.  
REN Y M. Research on strategies and methods of platform assignment optimization for high-speed railway large-scale passenger station[D]. Beijing:China Academy of Railway Sciences, 2021. (in Chinese)
- [23] ABDULKADIR S I, FADZLI S A, JAMAL A A, et al. Indoor global path planning based on critical cells using Dijkstra

algorithm[J]. Journal of Theoretical & Applied Information Technology, 2015, 79(1):115-121.

[24] ANTON V. On proportions of fit individuals in population of mutation-based evolutionary algorithm with tournament selection[J]. Evolutionary Computation, 2017, 26(2):269-297.

[25] ZHANG A, LI C, BI W H. Rectangle expansion A\* pathfinding for grid maps[J]. Chinese Journal of Aeronautics, 2016, 29(5): 1385–1396.

[26] 喻明让, 陈云, 张志刚. 离散粒子群优化算法求解多目标柔性作业车间调度问题[J]. 制造技术与机床,

2019(1):159-165.

YU M Z, CHEN Y, ZHANG Z G. A discrete version of particle swarm optimization for multi-objective flexible job-shop scheduling problems[J]. Manufacturing Technology & Machine Tool, 2019(1):159-165. (in Chinese)

[27] 王丽萍, 任宇, 邱启仓, 等. 多目标进化算法性能评价指标研究综述[J]. 计算机学报, 2021, 44(8):1590-1619.

WANG L P, REN Y, QIU Q C, et al. Survey on Performance Indicator for Multi-Objective Evolutionary Algorithms[J]. Chinese Journal of Computers, 2021, 44(8):1590-1619. (in Chinese)