

基于出动效率最大的离场优化排序

唐鑫磊, 沈堤*, 张哲, 余付平

(空军工程大学空管领航学院, 西安, 710051)

摘要 飞机离场优化排序对于缓解跑道拥挤现象、提升作战体系出动效率具有重要意义。针对军航双跑道离场优化排序问题, 首先建立了基于飞机在预定空域形成“任务包”时间最小为目标函数的优化排序模型, 然后设计了一种基于精英反向学习策略和黏菌觅食行为的改进海洋捕食者算法进行求解, 最后通过一个仿真实验, 与其他算法的优化结果进行对比, 改进的海洋捕食者算法使得“任务包”形成时间明显缩短, 有效提升了作战飞机的出动效率, 可以应用于解决飞机离场优化排序问题。

关键词 离场优化排序; 海洋捕食者算法; 精英式反向学习; 黏菌觅食行为

DOI 10.3969/j.issn.2097-1915.2022.06.004

中图分类号 V355.2; V271.4 **文献标志码** A **文章编号** 2097-1915(2022)06-0025-08

Optimal Departure Sequencing Based on Maximum Dispatch Efficiency

TANG Xinlei, SHEN Di*, ZHANG Zhe, YU Fuping

(Air Traffic Control and Navigation School, Air Force Engineering University, Xi'an 710051, China)

Abstract Optimal sequencing is important for aircraft departure to reduce runway congestion and improve the efficiency in combat system. In view of such a problem at military aviation double runways, firstly, an optimal sequencing model is established based on the minimum time of aircraft forming “task package” in the predetermined airspace, and then an improved marine predator algorithm based on elite reverse learning strategy and slime fungus foraging behavior (ESMPA) is used to solve it. Finally, through the simulation experiment in comparison with the optimization results of other algorithms, the improved marine predator algorithm makes the “task package” formation time significantly shorter and effectively improves the combat aircraft sortie efficiency.

Key words departure optimization sequencing; marine predator algorithm; elite-style reverse learning; slime mould foraging behavior

在当前军航机场管制条件下, 管制员一般依据飞机执行任务的下达时间并结合个人经验, 给离场飞机安排离场时间和起飞次序, 这在飞机较少的日

常飞行训练中, 有较好的适用性, 但随着体系化作战强度的增加或者紧急进入战备状态, 大批次飞机需要快速从机场起飞到达预定空域执行任务时, 继续

收稿日期: 2022-05-27

基金项目: 国家自然科学基金(18XGL026)

作者简介: 唐鑫磊(1998-), 男, 湖南邵阳人, 硕士生, 研究方向为航空管制指挥与安全。E-mail: 15029908189@163.com

通信作者: 沈堤(1983-), 男, 湖南浏阳人, 副教授, 研究方向为航空管制指挥与安全。E-mail: 317193179@qq.com

引用格式: 唐鑫磊, 沈堤, 张哲, 等. 基于出动效率最大的离场优化排序[J]. 空军工程大学学报, 2022, 23(6): 25-32. TANG Xinlei, SHEN Di, ZHANG Zhe, et al. Optimal Departure Sequencing Based on Maximum Dispatch Efficiency[J]. Journal of Air Force Engineering University, 2022, 23(6): 25-32.

应用依靠管制员经验安排起飞次序这种方法难以解决机场跑道拥挤的问题。鉴于此,如何实现飞机快速高效离场,尽快到达预定空域并形成体系作战能力,已成为亟需解决的问题。

国内外学者为解决飞机离场优化排序问题应用了许多算法。尹嘉男等建立了基于独立运行模式的多跑道飞机离场调度模型,并用 NSGA-II 算法求解,均衡了各跑道之间的运行负荷^[1]。Ming 等将两阶段无等待混合流水车间模型应用于多机场飞机调度问题,并设计了一种 CPLEX 算法应用于多机场系统的最优起飞序列计算,使总加权延误成本最小^[2]。Zhong 等针对多机场中的低优先级机场的飞机离场优化排序问题,建立了基于机场总延误时间、单架飞机最大延误时间的双目标整数规划模型,并设计了禁忌搜索算法进行求解,实验结果证明,可以有效提高低优先级机场的运行效率^[3]。Ma 等考虑了滑行道的等待队列容量,并以到达跑道端口等待时间为决策变量,建立了基于时隙分配的多跑道飞机起飞排序模型,利用模拟退火算法求解^[4]。Sandamali 等针对离场优化排序问题,建立了基于飞机离场速度和离场时间的不确定性的优化模型,通过使用最佳飞行高度层分配策略来最小化燃料消耗,通过贪心算法求解该问题,有效降低了离场速度和离场时间偏差导致的延误等问题^[5]。马琳娜等利用了 Anylogic 建立了基于货运飞机离场时间最短的智能体仿真模型,对货运飞机的跑道分配问题进行了优化,降低了飞机在跑道端的等待时间^[6]。Wei 等针对综合飞机离场调度和飞行航线分配问题,分析了飞机离场调度、航线的时间窗约束、运输负载率之间的相互作用,采用基于蚁群算法的两阶段启发式方法,有效降低了航空公司的运营成本,并最大限度提高了运输效率^[7]。江灏等建立了基于交通状态的飞机离场优化排序模型,在机场拥挤和非拥挤两个场景下进行实验仿真,对比先到先服务方法,NSGA-II 算法得到的离场飞机总次序调整次数降低 80%,且总延误时间减少一半^[8]。

分析前人研究可知,已有飞机离场优化排序研究存在一些不足,主要有:①现有研究大多数只考虑了跑道端的约束让飞机快速离场,而军航飞机不仅有快速离场的需求,还有快速到达各自预定空域阵位,建立起“任务包”,形成体系作战能力,执行军事行动的需求,这就要求在建立问题模型时,将飞机从机场飞往预定空域的时间对飞机起飞次序的影响纳入考虑范围;②虽有诸如模拟退火算法等传统经典算法以及遗传算法、蚁群算法等群智能算法应用于飞机离场优化排序问题,但仍有许多优秀的新算法

尚未应用于该领域,如海洋捕食者算法、多元宇宙算法、天牛须搜索算法等。本文在现有研究的基础上,以飞机在预定空域形成“任务包”时间最短为优化目标,建立飞机离场优化排序模型,并针对海洋捕食者算法的寻优速度较慢、寻优精度不高等问题,引入精英反向学习策略提升初始解的质量,以提高其寻优速度,融入黏菌的觅食行为提高其寻优精度,最后利用案例仿真与其他算法优化结果进行对比,验证所提算法的性能。

1 问题定义

空中力量前出执行作战任务,通常以“任务包”形式组织实施。以空中突击作战为例,一个典型的任务包通常由扫荡编队、压制编队、对地突击编队构成。“任务包”中的战机首先从机场起飞,前往预定空域集结编组队形,形成完整的突击“任务包”,然后从前推线向前推进。预定空域集结队形通常如图 1 所示,扫荡编队、压制编队、对地突击编队三者队形之间保持一定距离(图 1)。

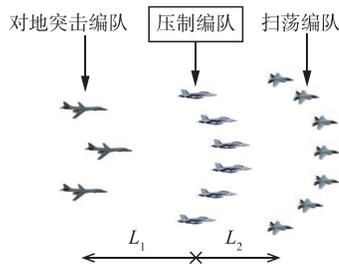


图 1 预定空域集结队形

离场优化排序作为本文研究内容,是飞机执行作战任务的关键一环,其出动效率高将直接影响后续前往预定阵位形成突击任务包的速度。这里,出动效率主要由飞机到达预定空域阵位的时间来体现,因此求解离场优化排序问题是核心。解决离场优化排序问题的关键是,如何在跑道离场端安排多个离场队列中的不同飞机进入不同的跑道,以使离场飞机前往预定空域形成完整“任务包”的时间最短。显然,这个问题能通过建立一个数学模型并应用一个优化算法来解决,这正是本文的研究重点。而多队列飞机离场优化排序问题是单队列离场的一种拓展情况,其流程如下图 2 所示。其中,蓝色虚线方框部分为本文的重点研究部分。

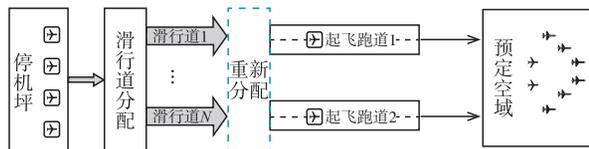


图 2 多队列飞机离场过程

2 模型构建

与民航单跑道离场优化问题不同的是,军航多跑道离场优化问题的难点在于,它不但需要确定作战飞机在起飞跑道上的最终起飞次序,还需要将这些作战飞机合理分配到预定的空域阵位上,以便使得执行作战任务的飞机到达预定空域形成完整“任务包”的总时间最短。本文假定每条滑行道可以进入任意一条跑道上,因此形成完整“任务包”总时间等于最后一架飞机到达预定空域阵位的时间,而飞机到达预定阵位的时间等于飞机的起飞时间加上跑道占用时间和从机场起飞到预定阵位的时间。

基于上述分析,建立多队列飞机离场优化排序问题的模型。

2.1 目标函数

成批次的作战飞机从跑道起飞到达预定空域阵位形成完整“任务包”执行作战任务这个过程,主要受飞机的起飞时间、跑道占用时间、从机场起飞后前往预定空域阵位的到达时间影响,表达式如式(1)所示:

$$\min(\max D_i) = \min\{\max(Q_i + P_i + F_i)\}, i = 1, 2, \dots, N \quad (1)$$

式(1)表示模型的最小化目标函数,目的是将飞机到达预定空域阵位的最大时间极小化,从而使得“任务包”的形成时间最短。 D_i 为飞机*i*到达预定空域阵位的时间, Q_i 为飞机*i*的起飞时间, P_i 为飞机*i*的跑道占用时间, F_i 为飞机*i*从机场起飞到达预定空域阵位时间, N 为飞机数量。

2.2 安全间隔约束

作战飞机作为高价值资产,保障其运行安全尤为重要,为避免发生前后飞机发生碰撞等事故,规定同一跑道上每次只允许一架飞机单独使用,与此同时,在一条跑道上连续起飞的飞机须满足一定的安全间隔。本文的安全间隔分为前机的发动机喷流等产生推力引起的纵向间隔和前机获得升力形成尾涡引起的尾流间隔^[9]。同一跑道上的前后飞机起飞时间间隔应大于前后飞机的纵向间隔、尾流间隔(距离间隔均转换为时间间隔)以及前机的跑道占用时间三者中的最大值,以此确保作战飞机的起飞安全,如式(2)、(3)所示:

$$Q_{i,j} - Q_{i-1,j} \geq \delta_{(i-1),j}, i = 1, 2, \dots, N_j, j = 1, 2, \dots, M \quad (2)$$

$$\delta_{(i-1),j} = \max(\phi_{(i-1),j}, \varphi_{(i-1),j}, P_{i-1,j}), i = 1, 2, \dots, N_j, j = 1, 2, \dots, M \quad (3)$$

式中: $Q_{i,j}$ 为飞机在跑道上的起飞时间; $\delta_{(i-1),j}$ 为在跑道上前后飞机的最大安全间隔; N_j 为跑道*j*的飞机数量;与*N*定义不同; M 为跑道的数量; j 为跑道的编

号; $\phi_{(i-1),j}, \varphi_{(i-1),j}$ 分别为跑道*j*上前后飞机的纵向间隔和尾流间隔; $P_{i-1,j}$ 为跑道*j*上前机的跑道占用时间。

2.3 滑行道约束

依据先到先服务原则,作战飞机进入滑行道形成队列后,假设该飞机队列后续从某一跑道起飞,那么起飞的前后顺序应保持相对不变。从该跑道起飞的飞机应满足如下约束:

$$Q_{i,j}^k - Q_{i-1,j}^k \geq 0, i = 1, 2, \dots, N_k, j = 1, 2, \dots, M, k = 1, 2, \dots, K \quad (4)$$

式中: $Q_{i,j}^k$ 为从滑行道*k*滑入跑道*j*的飞机*i*的起飞时间; k 为滑行道编号; K 为滑行道的数量; N_k 表示滑行道*k*上的飞机数量,与*N*, N_j 定义不同。

2.4 空域资源约束

在一个完整的“任务包”体系内,各种类型飞机各司其职,例如:电子战飞机执行压制干扰任务,歼击机执行扫荡任务等,故作战飞机与空域阵位必须相吻合,这就要求执行某种任务的一架飞机有且只能占用一个与之相对应的空域阵位,同样的,某种类型的每个空域阵位必须且只能分配给执行与之相对应任务的作战飞机使用,即:

$$\sum_{\tau \in F_g} \gamma_{i,\tau} = 1, i = 1, 2, \dots, N_g, g = 1, 2, \dots, G \quad (5)$$

$$\sum_{i \in N_g} \gamma_{i,\tau} = 1, \tau = 1, 2, \dots, F_g, g = 1, 2, \dots, G \quad (6)$$

式中: N_g 为*g*类型作战飞机数量; F_g 为发挥*g*类型作战飞机作用相对应的空域阵位数量; $\gamma_{i,\tau}$ 为0~1离散变量,若飞机*i*占用的空域阵位为*τ*,则 $\gamma_{i,\tau} = 1$,否则为0; G 为作战飞机的种类数。

2.5 变量约束

为用*N*表示所有的作战飞机数量以及与之相同的空域阵位数量,有下式成立:

$$\sum_{j=1}^M N_j = \sum_{k=1}^K N_k = \sum_{g=1}^G N_g = \sum_{g=1}^G F_g = N \quad (7)$$

飞机的起飞时间、跑道占用时间、从机场起飞到达预定空域阵位时间、以及2种安全间隔等均均为非负数,有下式成立:

$$Q_i, P_i, F_i, \phi_{(i-1),j}, \varphi_{(i-1),j}, \delta_{(i-1),j} \geq 0 \quad (8)$$

3 基于精英反向学习策略和黏菌觅食行为的改进海洋捕食者算法

2020年,Faramarzi等^[10]通过解析海洋捕食者的捕食行为,提出了海洋捕食者算法(marine predators algorithm, MPA)。ESMPA继承了MPA的所有捕食过程,并对其进行了改进。主要改进如下。

1) MPA算法的初始种群个体基于随机规则产

生的,这将导致初始解具有随机性和不确定性,而算法的计算时间与初始解的优劣直接相关,初始解与最优解的距离越短,MPA 算法计算时间则越短,反之则越长,故随机产生的初始解可能导致 MPA 算法性能不佳。根据 Tizhoosh 教授研究^[11]发现,初始解的反向解比初始解更接近最优解的概率多 50%,并通过实验证明,与随机产生的初始种群的原算法相比,加入反向学习策略的改进算法具有更快的收敛速度;而 ESMPA 引入了精英反向学习策略^[12],则是考虑到精英个体比其他个体具备更多接近最优解的可能性,通过对种群中的精英个体构建出反向精英种群,增加种群的多样性同时减少了构建普通个体的反向种群的运算时间,进一步提升 ESMPA 算法的收敛速度。

2)在 MPA 算法寻优的第 3 阶段中,捕食者通过莱维飞行方式进行捕食,而应用莱维飞行方式会在随机行走过程中出现较大概率的大跨步,使得 MPA 算法存在错过最优解的可能。黏菌算法(slime mould algorithm, SMA)的觅食行为则有较好的局部寻优能力^[13],任丽莉等将黏菌算法的觅食行为引入多元宇宙算法中^[14],与原多元宇宙算法在测试函数上的测试结果相比,具有更好的求解能力。本文将黏菌算法的觅食行为融入算法的 MPA 算法第三阶段,与应用莱维飞行方式的捕食行为相结合,以提升 ESMPA 算法的寻优能力。

ESMPA 算法详细过程如下所示,算法流程图 3。

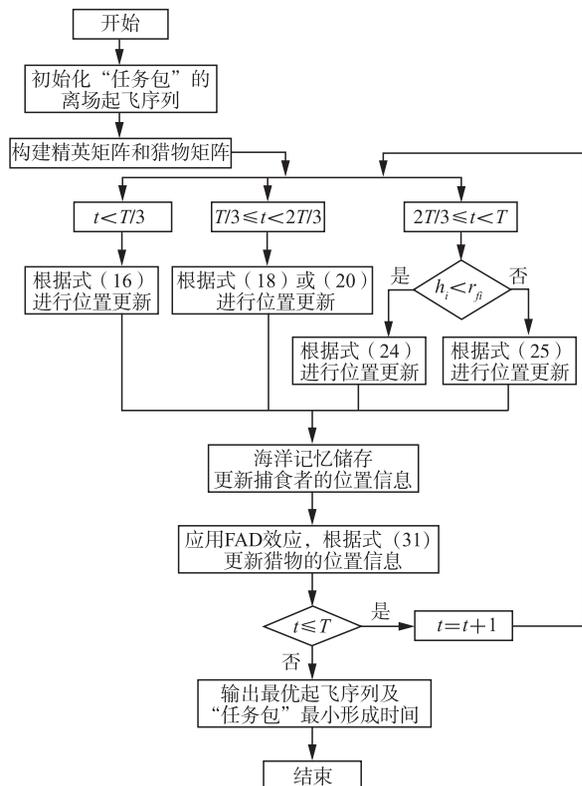


图 3 改进的海洋捕食者算法流程图

3.1 ESMPA 的种群初始化

ESMPA 先通过随机规则初始化“任务包”的离场起飞序列,其表达式如式(9)所示:

$$X_{i,j} = X_{\min,j} + r_1 (X_{\max,j} - X_{\min,j}), i = 1, 2, \dots, n, j = 1, 2, \dots, D \quad (9)$$

式中: $X_{i,j}$ 为第 i 种起飞序列中在第 j 架飞机的进入预定空域阵位时间; n 为种群数量(即为生成的起飞方案数量); D 为搜索维度(即为飞机数量); $r_1 \in (0, 1)$ 内的随机数; $X_{\max,j}$ 、 $X_{\min,j}$ 分别为飞机到达空域阵位时间的上下限。

而后选出种群中所有个体的极值点视为种群的精英个体,假设为 $X_{i,j}^e = (X_{i,1}^e, X_{i,2}^e, \dots, X_{i,D}^e)$, $i = 1, 2, \dots, n, j = 1, 2, \dots, D$, 则其反向解 $\overline{X_{i,j}^e} = (\overline{X_{i,1}^e}, \overline{X_{i,2}^e}, \dots, \overline{X_{i,D}^e})$, $i = 1, 2, \dots, n, j = 1, 2, \dots, D$ 定义为:

$$\overline{X_{i,j}^e} = r(b_{l,j} + b_{u,j}) - X_{i,j}^e, i = 1, 2, \dots, n, j = 1, 2, \dots, D \quad (10)$$

式中: $r \in (0, 1)$ 内的随机数; $X_{i,j}^e \in [b_{l,j}, b_{u,j}]$, $b_{l,j} = \min(X_{i,j}^e)$, $b_{u,j} = \max(X_{i,j}^e)$, $b_{l,j}$, $b_{u,j}$ 分别是动态边界的上下界,动态边界与固定边界相比,可以有效减少算法寻优时间。如果 $\overline{X_{i,j}^e}$ 超出了动态边界,则用式(11)重置。

$$\overline{X_{i,j}^e} = r(b_{l,j} + b_{u,j}), i = 1, 2, \dots, n, j = 1, 2, \dots, D \quad (11)$$

通过精英式反向学习后,计算个体的适应度函数,本文所求为最小值问题,故认为适应度函数值低的个体更优,将原种群与反向种群的适应度值进行对比,将适应度值好的个体选为初始种群个体,采用式(12)进行更新:

$$X_{i,j}^e = \begin{cases} \overline{X_{i,j}^e}, & f(\overline{X_{i,j}^e}) < f(X_{i,j}^e) \\ X_{i,j}^e, & \text{其他} \end{cases} \quad (12)$$

精英矩阵 E 和猎物矩阵 P 定义如下:

$$E = \begin{bmatrix} X_{11}^e & X_{12}^e & \dots & X_{1D}^e \\ X_{21}^e & X_{22}^e & \dots & X_{2D}^e \\ \vdots & \vdots & & \vdots \\ X_{n1}^e & X_{n2}^e & \dots & X_{nD}^e \end{bmatrix} \quad (13)$$

$$P = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1D} \\ X_{21} & X_{22} & \dots & X_{2D} \\ \vdots & \vdots & & \vdots \\ X_{n1} & X_{n2} & \dots & X_{nD} \end{bmatrix} \quad (14)$$

式中: n 为种群数量; D 为搜索维度。初始种群的所有个体的位置信息构成了猎物矩阵,将初始种群中的最优解命名为顶级捕食者,将其位置向量作为 E 矩阵的第 1 行,将其复制 n 次,构成了初始精英矩

阵。在寻优过程中,捕食者和猎物均为搜索者,因为每个个体都有捕食者与被捕食者两种属性。在每轮迭代中,如果捕食者被适应度值更优的捕食者取代,精英矩阵 E 的信息则需要更新。

3.2 ESMPA 的寻优过程

ESMPA 的寻优过程主要模拟了海洋捕食者追逐猎物的过程,依据两者之间的速度比变化,将整个寻优过程划分为3个阶段。

1)高速比阶段(迭代前期):该阶段发生在 $t < T/3$ 时(t 为当前迭代次数, T 为最大迭代次数),在这个阶段,假定猎物的移动速度比捕食者大,且猎物采取布朗运动进行全局搜索,捕食者不移动,猎物的位置变化如式(15)、(16)所示:

$$S_i = R_B \times (E_i - R_B \times P_i), i = 1, 2, \dots, n \quad (15)$$

$$P_i = P_i + p \cdot R \times S_i \quad (16)$$

式中: S_i 为移动步长; R_B 为布朗运动^[15] 向量; R 为 $[0, 1]$ 的随机均匀分布的向量; p 为一个常数,一般取 0.5。

2)均速比阶段(迭代中期):该阶段发生在 $T/3 \leq t < 2T/3$ 时,假设在此阶段猎物的移动速度比捕食者相当,此时算法的寻优过程由全局搜索向局部搜索转化,种群的前一半个体采用莱维飞行方式进行局部搜索,而后一半个体则继续采用布朗运动进行全局搜索,主要表现为猎物负责局部搜索,捕食者负责全局搜索,前一半种群个体的位置变化如式(17)、(18)所示:

$$S_i = R_L \times (E_i - R_L \times P_i), i = 1, 2, \dots, n/2 \quad (17)$$

$$P_i = P_i + p \cdot R \times S_i \quad (18)$$

后一半个体的位置变化如式(19)、(20)、(21)所示:

$$S_i = R_B \times (R_B \times E_i - P_i), i = n/2, \dots, n \quad (19)$$

$$P_i = E_i + p \cdot C \cdot S_i \quad (20)$$

$$C = (1 - t/T)^{(2t/T)} \quad (21)$$

式中: R_L 为莱维飞行^[16] 向量; C 为自适应参数,用以控制捕食者的移动步长;其他参数同上。

3)低速比阶段:该阶段发生在 $t \geq 2T/3$ 时,在这个阶段,猎物的移动速度比捕食者小,算法处于局部搜索阶段,每个搜索个体首先计算出自身适应度值 $f(X_i)$ 与种群适应度值 $\sum_{i=1}^n f(X_i)$ 的比值 h_i ,而后与适应度函数 r_{fi} 对比,来决定其以何种方式进行局部搜索,若 $h_i < r_{fi}$,则捕食者通过莱维飞行方式对猎物进行捕食,如式(24)所示,否则,捕食者模拟黏菌的觅食行为进行捕食,如式(25)所示:

$$h_i = \frac{f(X_i)}{\sum_{i=1}^n f(X_i)} \quad (22)$$

$$r_{fi} = \min(h_i) + \text{rand}(\max(h_i) - \min(h_i)) \quad (23)$$

$$\begin{cases} S_i = R_L \times (R_L \times E_i - P_i), i = 1, 2, \dots, n \\ P_i = E_i + p \cdot C \cdot S_i \end{cases}, h_i < r_{fi} \quad (24)$$

$$P_i = \begin{cases} P_b + v_b \cdot (W \times P_A - P_B), r_2 < g \\ v_c \cdot P_i, r_2 \geq g \end{cases}, h_i \geq r_{fi} \quad (25)$$

式中: P_b 为当前最优个体位置, P_A 、 P_B 为两个随机个体的位置; v_b 、 v_c 均为控制参数, $v_b \in [-a, a]$ 的一个随机值, v_c 从 1 线性递减到 0; a 、 v_c 的数学表达式分别如式(26)、(27)所示; $r_2 \in (0, 1)$ 内的随机数; g 为一个控制变量,其表达式如式(28)所示; W 为种群个体的适应度值权重,表达式如(29)所示:

$$a = \text{arctanh}\left(-\frac{t}{T}\right) + 1 \quad (26)$$

$$v_c = 1 - \frac{t}{T} \quad (27)$$

$$g = \tanh |f(X_i) - F| \quad (28)$$

$$W(k(i)) = \begin{cases} 1 + r_3 \log\left(\frac{f_b - f(X_i)}{f_b - f_w} + 1\right), i = m \\ 1 - r_3 \log\left(\frac{f_b - f(X_i)}{f_b - f_w} + 1\right), i = n \end{cases} \quad (29)$$

$$k(i) = \text{sort}(f(X_i)) \quad (30)$$

式中: $f(X_i)$ 为当前个体的适应度值; F 表示已迭代轮次中的个体最优适应度值; $r_3 \in (0, 1)$ 的随机数; f_b 、 f_w 分别表示本轮迭代的个体最优适应度值和最劣适应度值, $i = m$ 表示该个体的适应度值处于种群的前一半, $i = n$ 则表示其适应度值位于种群的后一半; $k(i)$ 为适应度值排序,本文所求为最小值问题,则使用升序排序法对种群个体进行排序。

3.3 ESMPA 的位置更新

此外,ESMPA 此外还考虑了 2 种效应对算法的影响,具体如下所示:

1)涡流(eddy formation, EF)和鱼群聚集(fish aggregation devices, FAD)效应。Filmlalter 等研究发现涡流的形成以及鱼群聚集效应会导致算法陷入局部最优^[17],因此 ESMPA 采用式(31)所示的跳跃方式来避免局部最优。

$$P_i = \begin{cases} P_i + C \cdot [X_{\min} + R \times (X_{\max} - X_{\min})] \times U, r_4 \leq V_{\text{FAD}} \\ P_i + [V_{\text{FAD}} \cdot (1 - r) + r] \cdot (P_{r_1} - P_{r_2}), r_4 > V_{\text{FAD}} \end{cases} \quad (31)$$

式中: $r_4 \in (0, 1)$ 的随机数; V_{FAD} 为涡流效应或鱼群聚集效应对算法寻优的影响概率,一般取 0.2; P_{r_1} 、 P_{r_2} 为 2 个随机猎物的位置; U 为只包含 0 和 1 的二进制数组向量。

2)海洋记忆效应^[18]。海洋捕食者存在对其曾经捕食成功的位置有一定记忆,这在 MPA 中可以表示对最优解的储存,即将上一代的最优解与新解进行比较,如果新解适应度更好,则用当前的新解替代原最优解,这个过程会在迭代过程持续进行,以提

高最优解的质量。

4 仿真实验

4.1 仿真准备

本文以某实施独立离场运行模式的军用远距双跑道机场为例,对某次执行空中突击任务的飞机编队进行离场优化排序,其在 3 条滑行道上排队等待进入跑道端的数据如表 1 所示,其中, A、B、C 分别为 3 条滑行道的编号, H 为实施对地打击的轰炸机, M 为实施电子压制的电子战飞机, S 为实施空中扫荡的歼击机。

表 1 滑行道飞机序列初始数据

A 滑行道	B 滑行道	C 滑行道
H	S	M
M	H	S
M	S	H
S	H	S
H	M	M
H	M	S
	S	M
	S	M

轰炸机、电子战飞机、歼击机依据其最大起飞重量分别属于重型(H)、中型(M)、轻型(S)作战飞机。不同类型飞机对跑道的占用时间不同,本文采用的时间如表 2 所示。

表 2 跑道占用时间

飞机类型	跑道占用时间/s
H	70
M	60
S	50

由于飞机的发动机喷流等产生推力引起的尾迹寿命较短,对后机的影响小,而飞机获得升力形成尾涡形成的时间长,对后机的影响大,故本文计算时只考虑尾流间隔,飞机离场的尾流间隔标准^[19]如表 3 所示(表中军航间隔标准参考民航标准进行计算)。

表 3 尾流间隔标准 单位:s

前机	后机		
	H	M	S
H	60	90	120
M	60	60	90
S	60	60	60

飞机从机场起飞后,前往不同的空域阵位点消耗的时间也不同,例如,一架歼击机离开机场后可以选择前往 F₁、F₂、F₃、F₄ 空域阵位点进行盘旋等待,待到形成完整突击“任务包”之后整体前推执行空中突击任务,空域阵位到达时间如表 4 所示,空域阵位

点如图 4 所示。

表 4 空域阵位点到达时间 单位:min

阵位点 编号	到达 时间	阵位点 编号	到达 时间	阵位点 编号	到达 时间
F ₁	15	E ₁	16	B ₁	17
F ₂	16	E ₂	17	B ₂	19
F ₃	17	E ₃	18	B ₃	21
F ₄	18	E ₄	19		

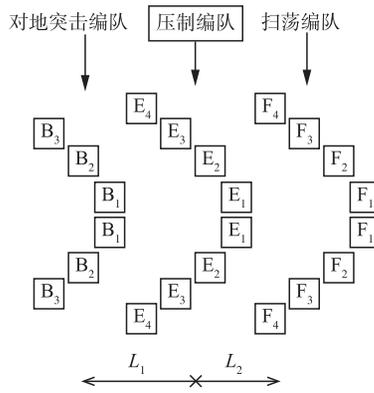


图 4 预定空域阵位点

4.2 结果分析

为了验证 ESMPA 算法应用在进场优化排序领域的性能,选择“任务包”形成最小时间 $\min(\max D_i)$ 为验证指标,与蚁群算法 (ant colony optimization, ACO)^[20]、遗传算法 (genetic algorithm, GA)^[21]、粒子群算法 (particle swarm optimization, PSO)^[22]、海洋捕食者算法进行对比验证,算法的参数设置如表 5 所示,所有算法的最大迭代次数设为 200 次,种群数量设为 100,运行环境为 MATLAB R2019a。

表 5 算法参数取值

算法	参数取值
ACO	信息素重要程度 $\alpha=1$,启发因子重要程度 $\beta=3$, 信息素蒸发系数 $\rho=0.4$,信息素区间 $[0.01,10]$, 信息素系数 $Q=0.001$
GA	交叉概率 $P_c=0.6$,变异概率 $P_m=0.01$
PSO	惯性权重 $\omega=0.8$,学习因子 $c_1=c_2=1.5$,个体速度 $[-0.5,0.5]$
MPA	鱼群聚集效应 $V_{FAD}=0.2$,常量 $p=0.5$
ESMPA	鱼群聚集效应 $V_{FAD}=0.2$,常量 $p=0.5$

各算法获得的仿真结果如表 6~10 所示,其中,符号(M)C₁, X₁, E₄ 表示位于滑行道 C 的等待队列中的第 1 架飞机为电子战飞机,安排其以第 1 顺位从跑道 X 起飞,前往压制编队所在的空域阵位点 E₄ (由于空域阵位点呈轴对称分布,默认先到达的飞机飞往左翼阵位点),其余符号以此类推; t_x 、 t_y 分别表示进入跑道 X、Y 的时间, t_0 表示到达空域阵位点的时间。寻优结果对比总时间如表 11 所示,基于

ACO、GA、PSO、MPA、ESMPA 算法求得的“任务包”形成最小时间分别为 1 700 s、1 700 s、1 720 s、1 690 s、1 670 s, ESMPA 算法较其他 4 种算法减少了 30 s、30 s、50 s、20 s,对于分秒必争的军事打击行动来说,ESMPA 算法有效提高了“任务包”的出动效率。

表 6 ACO 的仿真结果 单位:s

跑道 X	t_x	t_o	跑道 Y	t_y	t_o
(M)C1, X1, E4	0	1 200	(S)C2, Y1, F4	0	1 130
(H)A1, X2, B3	60	1 390	(H)C3, Y2, B3	60	1 390
(M)A2, X3, E3	150	1 290	(S)B1, Y3, F4	180	1 310
(M)A3, X4, E3	210	1 350	(S)C4, Y4, F3	240	1 310
(S)C6, X5, F3	300	1 370	(M)C5, Y5, E4	300	1 500
(S)A4, X6, F2	360	1 370	(M)C7, Y6, E2	360	1 440
(H)A5, X7, B2	420	1 630	(H)B2, Y7, B2	420	1 630
(S)B3, X8, F2	540	1 550	(H)A6, Y8, B1	490	1 580
(M)C8, X9, E2	600	1 680	(H)B4, Y9, B1	560	1 650
(M)B5, X10, E1	660	1 680	(M)B6, Y10, E1	660	1 680
(S)B7, X11, F1	750	1 700	(S)B8, Y11, F1	750	1 700

表 7 GA 的仿真结果 单位:s

跑道 X	t_x	t_o	跑道 Y	t_y	t_o
(H)A1, X1, B3	0	1 330	(M)A2, Y1, E4	0	1 200
(M)C1, X2, E2	90	1 230	(M)A3, Y2, E4	60	1 260
(S)B1, X3, F4	180	1 310	(S)C2, Y3, F4	150	1 280
(H)C3, X4, B3	240	1 570	(S)A4, Y4, F3	210	1 280
(H)B2, X5, B2	310	1 520	(S)C4, Y5, F3	270	1 340
(H)A5, X6, B2	380	1 590	(S)B3, Y6, F2	330	1 340
(H)A6, X7, B1	450	1 540	(H)B4, Y7, B1	390	1 480
(M)B5, X8, E3	540	1 680	(M)C5, Y8, E2	480	1 560
(M)C7, X9, E2	600	1 680	(S)C6, Y9, F2	570	1 580
(M)C8, X10, E1	660	1 680	(M)B6, Y10, E1	630	1 650
(S)B8, X11, F1	750	1 700	(S)B7, Y11, F1	720	1 670

表 8 PSO 的仿真结果 单位:s

跑道 X	t_x	t_o	跑道 Y	t_y	t_o
(H)A1, X1, B3	0	1 330	(S)B1, Y1, F4	0	1 130
(H)B2, X2, B3	70	1 400	(S)B3, Y2, F4	70	1 200
(H)B4, X3, B2	140	1 350	(M)A2, Y3, E4	130	1 330
(M)B6, X4, E2	250	1 330	(M)C1, Y4, E4	190	1 390
(S)A4, X5, F3	340	1 410	(M)B5, Y5, E3	250	1 390
(H)A5, X6, B2	400	1 610	(M)A3, Y6, E3	310	1 450
(H)A6, X7, B1	470	1 560	(S)C2, Y7, F3	400	1 470
(H)C3, X8, B1	540	1 630	(S)B7, Y8, F2	460	1 470
(M)C5, X9, E2	630	1 710	(S)B8, Y9, F2	520	1 530
(M)C7, X10, E1	690	1 710	(S)C4, Y10, F1	580	1 530
			(S)C6, Y11, F1	640	1 590
			(M)C8, Y12, E1	700	1 720

表 9 MPA 的仿真结果 单位:s

跑道 X	t_x	t_o	跑道 Y	t_y	t_o
(M)C1, X1, E4	0	1 200	(S)C2, Y1, F4	0	1 130
(H)C3, X2, B3	60	1 390	(S)B1, Y2, F4	60	1 190
(H)A1, X3, B3	130	1 460	(S)C4, Y3, F3	120	1 190
(H)B2, X4, B2	200	1 410	(M)A2, Y4, E4	180	1 380
(M)A3, X5, E3	290	1 430	(M)C5, Y5, E3	240	1 380
(S)A4, X6, F2	380	1 390	(S)C6, Y6, F3	330	1 400
(M)C7, X7, E2	440	1 520	(H)A5, Y7, B2	390	1 600
(S)B3, X8, F2	530	1 540	(H)A6, Y8, B1	460	1 550
(M)B5, X9, E2	590	1 670	(H)B4, Y9, B1	530	1 620
(M)B6, X10, E1	650	1 670	(M)C8, Y10, E1	620	1 640
(S)B7, X11, F1	740	1 690	(S)B8, Y11, F1	740	1 690

表 10 ESMPA 的仿真结果 单位:s

跑道 X	t_x	t_o	跑道 Y	t_y	t_o
(H)A1, X1, B3	0	1 330	(S)B1, Y1, F4	0	1 130
(H)B2, X2, B3	70	1 400	(S)B3, Y2, F4	70	1 200
(M)A2, X3, E4	160	1 360	(M)C1, Y3, E4	130	1 330
(S)C2, X4, F3	250	1 320	(M)A3, Y4, E3	190	1 330
(S)A4, X5, F3	310	1 380	(H)B4, Y5, B2	250	1 460
(S)C4, X6, F2	370	1 380	(H)C3, Y6, B2	320	1 530
(S)C6, X7, F2	430	1 440	(M)C5, Y7, E3	410	1 550
(M)C7, X8, E2	490	1 570	(H)A5, Y8, B1	470	1 560
(M)C8, X9, E2	550	1 630	(H)A6, Y9, B1	540	1 630
(M)B6, X10, E1	630	1 650	(M)B5, Y10, E1	630	1 650
(S)B8, X11, F1	720	1 670	(S)B7, Y11, F1	720	1 670

表 11 寻优结果对比 单位:s

算法	“任务包”形成总时间	运算时间
ACO	1 700	3.27
GA	1 700	3.07
PSO	1 720	5.36
MPA	1 690	4.84
ESMPA	1 670	4.75

算法对比见图 5。

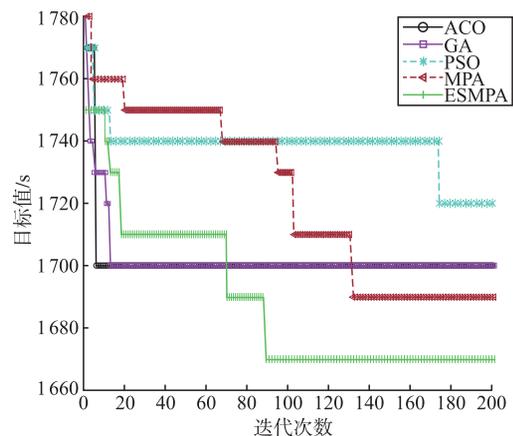


图 5 算法迭代曲线对比图

对图 5 的算法迭代曲线对比图进行分析可知, GA、ACO 算法寻优速度较快, 分别第 13 代、第 6 代获得其最优值, 但陷入了局部最优值, 过早收敛;

PSO、MPA 则搜索速度与精度均较差,分别在第 174 代、第 132 代获得其最优值;ESMPA 算法搜索速度适中,在第 89 代获得了最优值,寻优能力最强,这也证明了本文所提出的改进措施有效且实用。

5 结语

本文通过引入精英反向学习策略和融入黏菌算法的觅食行为对 MPA 算法进行改进,并将其应用到解决“任务包”的离场优化排序问题,得出了“任务包”的最短形成时间与最优起飞序列。与 ACO、GA、PSO、MPA 算法求得的优化结果相比,应用 ESMPA 算法能更好地处理成批次飞机离场优化排序问题,有效提升作战飞机的出动效率,为提升军航机场的保障能力提供理论基础。下一步研究将以其他军事行动为背景,进一步提高 ESMPA 算法的实用性。

参考文献

- [1] 尹嘉男,胡明华,张洪海,等. 独立离场模式下多跑道时空资源优化调度方法[J]. 航空学报, 2015, 36(5): 1574-1584.
- [2] MING L, SUN Z, ZHANG X, et al. A Two-Stage No-Wait Hybrid Flow-Shop Model for the Flight Departure Scheduling in a Multi-Airport System[C]// IEEE International Conference on Networking, Zhuhai, China: IEEE, 2018:495-500.
- [3] ZHONG H, GUAN W, ZHANG W, et al. A Bi-Objective Integer Programming Model for Partly-Restricted Flight Departure Scheduling[J]. PLoS One, 2018, 13(5):1-18.
- [4] MA J, SBIHI M, DELAHAYE D. Optimization of Departure Runway Scheduling Incorporating Arrival Crossings[J]. International Transactions in Operational Research, 2021, 28(2):615-637.
- [5] SANDAMALI G G N, SU R, ZHANG Y. Flight Routing and Scheduling Under Departure and En Route Speed Uncertainty[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(5):1915-1928.
- [6] 马琳娜,鲁晓春,杨彦欢. 双跑道混合起降的货运飞机起飞排序问题优化[J]. 计算机仿真, 2020, 37(7): 44-47.
- [7] WEI M, ZHAO L, JING B, et al. An Integrated Optimization Mode for Multi-Type Aircraft Flight Scheduling and Routing Problem[J]. Mathematical Biosciences and Engineering: MBE, 2020, 17(5): 4990-5004.
- [8] 江灏,刘继新,董欣放. 基于交通状态的离场航班动态协同排序方法[J]. 北京航空航天大学学报, 2022, 48(10):2048-2060.
- [9] 魏志强. 尾涡流场及安全间隔的建模与仿真计算研究[D]. 天津:中国民航大学, 2008:38-41.
- [10] FARAMARZI A, HEIDARINEJAD M, MIRJALILI S, et al. Marine Predators Algorithm: A Nature-Inspired Metaheuristic[J]. Expert Systems with Applications, 2020, 152:113377-113405.
- [11] TIZHOOSH H R. Opposition-Based Learning: A New Scheme for Machine Intelligence[C]// International Conference on International Conference on Computational Intelligence for Modelling, Control & Automation, Vienna, Australia: IEEE, 2005:695-701.
- [12] 何小龙,张刚,陈跃华,等. 融合 Lévy 飞行和精英反向学习的 WOA-SVM 多分类算法[J]. 计算机应用研究, 2021, 38(12):3640-3645.
- [13] LI S, CHEN H, WANG M, et al. Slime Mould Algorithm: A New Method for Stochastic Optimization [J]. Future Generation Computer Systems, 2020, 111:300-323.
- [14] 任丽莉,王志军,闫冬梅. 结合黏菌觅食行为的改进多元宇宙算法[J]. 吉林大学学报(工学版), 2021, 51(6):2190-2197.
- [15] EINSTEIN A. Investigations on the Theory of the Brownian Movement [M]. New York: Dover Publications, 1956:51-56.
- [16] MANTEGNA R N. Fast, Accurate Algorithm for Numerical Simulation of Lévy Stable Stochastic Processes[J]. Physical Review E Statistical Physics Plasmas Fluids & Related Interdisciplinary Topics, 1994, 49(5):4677-4680.
- [17] FILMALTER J D, DAGORN L, COWLEY P D, et al. First Descriptions of the Behavior of Silky Sharks, *Carcharhinus Falciformis*, Around Drifting Fish Aggregating Devices in the Indian Ocean[J]. Bulletin of Marine Science, 2011, 87(3):325-337.
- [18] RAGHAV P P, KEDAR N D. A Memory Based Differential Evolution Algorithm for Unconstrained Optimization[J]. Applied Soft Computing, 2016, 38: 501-507.
- [19] HANCERLIOGULLARI G, RABADI G, ALSALEM A, et al. Greedy Algorithms and Metaheuristics for A Multiple Runway Combined Arrival-Departure Aircraft Sequencing Problem[J]. Journal of Air Transport Management, 2013, 32:39-48.
- [20] 杨剑峰. 蚁群算法及其应用研究[D]. 杭州:浙江大学, 2007:40-48.
- [21] 徐磊. 基于遗传算法的多目标优化问题的研究与应用[D]. 长沙:中南大学, 2007:20-24.
- [22] 邵晴. 粒子群算法研究及其工程应用案例[D]. 长春:吉林大学, 2017:33-37.

(编辑:姚树峰)