人工智能与产业融合

多跑道运行场景下机场抵离港航班动态协同调度优化

李 佳，王旻轲，翟 果

（中国民用航空飞行学院机场学院，广汉 618307）

摘要：针对多跑道枢纽机场资源分配不均、航班延误累积问题，构建总延误与跑道空闲时间最小化的双目标混合整数线性规划模型，纳入跑道功能匹配、尾流安全间隔等约束。采用改进的快速非支配排序遗传算法（NSGA-II）求解，分析帕累托最优解关系，确定航班跑道分配与执行次序。利用广州白云机场高峰航班数据仿真，结果表明：改进算法总延误降低 30%，跑道空闲时间优化超 47%，实现跑道负载均衡，性能优于传统算法与先到先服务策略，可为相关调度提供理论与方法支撑。

关键词：混合整数线性规划模型；航班执行次序；尾流安全间隔

中图分类号：[U8] **文献标志码：**A **文章编号：**1671－1807(2026)13－0001－07

随着航空运输业的高速发展，大型枢纽机场面临的运行压力日益凸显，多跑道系统的协同调度成为提升空域资源利用效率的关键环节。多跑道机场抵离港航班协同调度优化，其现实问题在于当前大型机场普遍存在跑道资源分配不均衡、航班延误累积、跨跑道运行冲突等挑战。特别是在高峰运行时段，传统先到先服务（first come, first served, FCFS）策略难以应对复杂跑道配置下的动态调度需求，导致跑道利用率波动和延误连锁反应。因此，以该热点问题作为研究对象，聚焦航班优化调度这一核心问题，开展数学优化建模与求解研究。

抵港航班和离港航班在运行过程中对必要资源（如机场跑道资源和空域资源）的使用存在明显的竞争关系。因此，在对终端区时空资源进行优化配置时，需要考虑不同航班的相互影响，对抵港航班和离港航班进行联合优化调度，以保证系统整体运行的高效和调度方案的有效性，实现时空资源利用最大化。

马晓萱等[1]考虑了离场运行条件下的放行间隔和港湾区运行间隔，构建了复杂多跑道机场离港航班动态协同调度模型，并模拟了实际运行中不同走廊口间隔标准对离场运行的影响。张军峰等[2]提出以最小化着陆跑道的延误为目标并适用于多跑道混合运行模式的进场航班动态鲁棒优化调度方法。Jiang 和 Zhang[3]提出了一种综合的混合整数规划模型，用于考虑多个机场、跑道和航线的到达排序。吴维等[4]考虑相邻跑道干扰影响建立进离场航班排序模型，将有干扰与无干扰跑道的延误时间及跑道空闲时间进行对比，所得结论符合直观分析，验证了模型的有效性。Sandamali 等[5]建立了离场航班动态调度问题的混合整数规划模型，并将分支定界法、元启发式算法等用于解决离场航班调度问题。Lieder 和 Stolletz[6]针对航空器抵离港协同调度问题展开研究，重点探讨了相互依赖于独立运行跑道条件下的航空器排序优化问题，并提出了一种基于动态规划的求解方法。张兆宁和刘珂璇[7]提出改进的动态规划方法，用于求解针对不同跑道运行模式的排序模型。Ma 等[8]提出了一种离港跑道调度的混合整数规划模型，并采用模拟退火算法对模型进行求解。Bikir 等[9]提出了一种基于遗传算法的航班调度方法，帮助空中交通管制员根据标准仪表离港配置组织离场序列。Chandrasekar 和 Hwang[10]采用分支定界算法实现了进离场航班在多跑道系统的排序与调度并用机场实际航班运行数据进行了验证，证明了此算法的优化作用。戴喜妹等[11]探讨了不确定因素引起的交通量猛增下的进离场航

收稿日期：2026-03-10

基金项目：大学生创新训练计划项目（S202510624117）

作者简介：李佳（2002—），女，河南汝州人，硕士研究生，研究方向为航空交通运输；通信作者王旻轲（1986—），男，四川成都人，博士，副教授，研究方向为系统工程与决策科学、物流系统优化；翟果（2000—），女，四川营山人，硕士研究生，研究方向为航空交通运输。

班的排序与调度问题。向征等[12]将强化学习应用于终端区航空器进离场排序中，提出基于强化学习的进离场排序模型。Riahi 等[13]为提高算法对飞机排序问题的搜索效率，提出基于目标函数与约束的知识图谱结构搜索算法。实验结果表明，所提算法明显优于现有的最先进的飞机排序算法。Sáez 等[14-15]考虑到日益紧张的空域资源和交通动态性需求，通过分析航空器性能与运行环境，建立混合整数规划模型，规划适应交通需求的动态航班抵港，安排抵港航班序列，并在一定的时间窗口内高效安排航空器抵港。王建忠等[16]考虑进离场排序中潜在冲突引起的管制员工作负荷增加，以及排序结果的可执行性等问题，提出了基于多目标时间索引模型的进离场航班排序方法。

综上可知，现有研究主要不足在于：较少考虑到复杂跑道构型下的航班抵离港排序问题；并且现有抵离港航班优化调度研究较少考虑到当相邻跑道间距较近时，可能导致两个跑道无法同时进行航空器抵港和离港。因此，通过构建多目标混合整数线性规划(multi-objective mixed integer linear programming，MOMILP)模型，将跑道分配、执行次序和实际起降时间等决策变量纳入统一优化框架，并采用改进的非支配排序遗传算法(non-dominated sorting genetic algorithm II，NSGA-II)求解帕累托前沿，旨在实现总延误时间最小化与跑道空闲时间最小化的双重目标，最终形成考虑安全间隔和实时动态调整的优化调度方案，为多跑道机场的高效运行提供理论支撑与方法创新。

1　问题描述

多跑道机场抵离港航班调度优化研究致力于解决大型枢纽机场在复杂跑道配置条件下的航班调度问题，其核心在于协调不同运行模式跑道(包括专用起飞跑道、专用降落跑道及混合起降跑道)的资源分配与航班时序安排。要求抵港航班必须分配至降落或混合跑道，离港航班则限定于起飞或混合跑道，同时需严格遵循基于飞机类型和起降类别的尾流间隔标准，即同一跑道上相邻航班的执行时间间隔必须满足前序航班尾流消散的安全要求。当相邻跑道间距小于临界距离时，并行运行的航班还需保持额外的安全时间间隔以避免空域冲突，这使得调度过程必须同时考虑时间和空间两个维度的耦合约束。

如图 1 所示，进近管制员需要在抵港航班到达中间进近定位点之前对航班进行排序，并为其分配对应的跑道和降落时间，然后移交给塔台管制员引导其着陆并滑行至机位；对于离港航班，则是由塔台管制员对其进行起飞时间、起飞跑道的分配，引导航空器从机位滑行至跑道前等待位置，等待管制员指令进入跑道并滑跑起飞。

研究的前提假设条件包括：①所有跑道在整个调度周期内保持固定运行模式；②每次飞行必须只分配到一条跑道，并且所有的跑道都不能修改；③模型中所有参与计算的航班中，航空器尾流间隔类型标准、预计抵港时刻和预计离港时刻均已知；④尾流间隔时间仅取决于前后航班的飞机类型组合(重型 H/中型 M/轻型 L)和运动方向；⑤不考虑随机干扰影响航班不能正常起飞或降落。

2　模型构建

表 1 给出了多跑道机场抵离港航班调度模型的相关符号变量含义，据此描述该数学模型的目标函数和约束条件。

2.1　目标函数

综合考虑空中交通管制员、机场、航空公司三方利益，选取飞机总延误时间最小化和跑道空闲时间最小化为目标函数。

飞机总延误时间最小化：

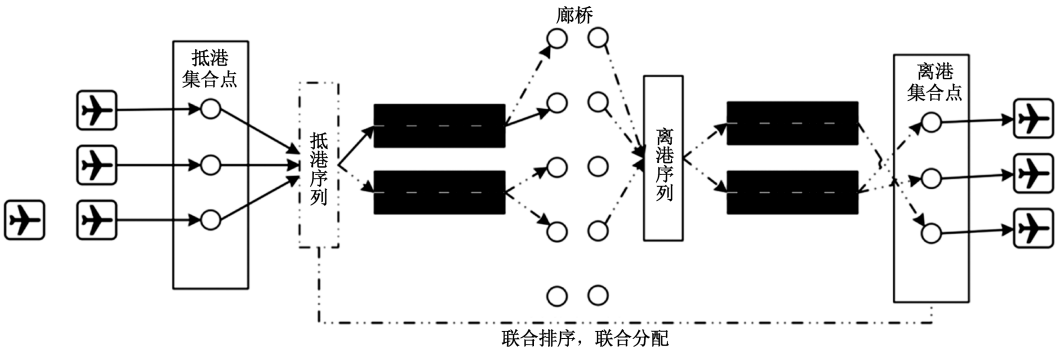


图 1　多跑道机场抵离港航班调度示意图

表 1　变量含义及符号

变量名称	符号	含义
集合	F	所有航班集合，$F=F_1\cup F_2$
	R	所有跑道集合，$R=R_1\cup R_2\cup R_3$
	F_1	抵港航班集合
	F_2	离港航班集合
	R_1	混合起降跑道集合
	R_2	起飞跑道集合
	R_3	降落跑道集合
常规变量	u_i	航班 i 在跑道上的执行次序
	δ_{ij}	航班 i 和 j 的最小尾流间隔
	Δ_{pq}	航班在跑道 p 与 q 的跨跑道干扰的时间段
	T_i^{e}	航班 i 的最早起降时间
	T_i^{l}	航班 i 的最晚起降时间
	M	较大的固定值
	$\bar{t}_i$	航班 i 的计划起降时刻
	Z_{ijr}	航班 i 和 j 在跑道 r 上的执行顺序指示变量
	T_{ij}	航班 i 和 j 之间的尾流间隔时间
	T'_{ij}	航班 i 和 j 的管制安全间隔时间
	T_i	航班 i 占用跑道时间
决策变量	x_{ir}	是否将航班 i 分配至 r 跑道
	y_{ij}	航班 i 是否在 j 之前起飞或降落
	t_i	航班 i 的实际起降时刻
	d_i	航班 i 的延误时长

$$\min Z_1=\sum_{i\in F}\left|t_i-\bar{t}_i\right| \tag{1}$$

采用航班实际起降时刻与计划起降时刻的差值绝对值累加的形式，对应了民航运行准点率的核心指标要求，t_i 作为决策变量动态反映了航班调度计划对时间的影响。航班提前或者晚于计划起降时间进入跑道都作为航班延误处理。

跑道空闲时间最小化：

$$\min Z_2=\sum_{r\in R}\sum_{i,j\in F}Z_{ijr}\left[t_j-t_i-\max(T_{ij},T'_{ij})-T_i\right] \tag{2}$$

Z_{ijr} 变量确保仅统计同跑道航班序列，$\max(T_{ij},T'_{ij})$ 强制性满足尾流间隔 T_{ij} 与管制间隔 T'_{ij} 中的严格标准，这种限制显著提升了跑道利用效率。

2.2　约束条件

(1)单跑道分配：

$$\sum_{r\in R}x_{ir}=1,\ \forall i\in F \tag{3}$$

必须确保每个航班起降只能分配至一条跑道，这与美国联邦航空管理局(FAA)的航空器运行安全规范完全一致。

(2)跑道功能性约束：

$$\sum_{r\in R_1}x_{ir}+\sum_{r\in R_3}x_{ir}=1,\ \forall i\in F_1 \tag{4}$$

保证每个抵港航班将被分配到混合或降落跑道上。

$$\sum_{r\in R_1}x_{jr}+\sum_{r\in R_2}x_{jr}=1,\ \forall j\in F_2 \tag{5}$$

保证每个离港航班将被分配到混合或起飞跑道上。

(3)时间窗遵守：

$$T_i^{\mathrm{e}}\leqslant t_i\leqslant T_i^{\mathrm{l}},\ \forall i\in F \tag{6}$$

抵离港航班运行过程中，由于终端空域范围有限，运行约束与限制均转化为时间窗约束，为保证时间窗约束的可行性，需确保航班的实际起降时间在规定的时间窗口内，T_i^{e} 为航班 i 的最早起降时间，T_i^{l} 为航班 i 的最晚起降时间。

(4)同跑道安全间隔：

$$t_j\geqslant t_i+\delta_{ij}-M(1-y_{ij})-M(2-x_{ir}-x_{jr}),\ \forall i,j\in F,r\in R \tag{7}$$

当且仅当航班 i、j 同属跑道 r 且 i 优先于 j 时(即 $x_{ir}=x_{jr}=y_{ij}=1$)，约束退化为标准间隔要求，否则被大数 M 松弛化，这种约束方式可将非线性逻辑条件转化为混合整数线性可解形式。

(5)近距跨跑道干扰：

$$|t_i-t_j|\geqslant\Delta_{pq},\ \forall i,j\in F,(r_p,r_q)\in R^2 \tag{8}$$

当实施相关平行进近时，在航空器抵港调度中不仅要考虑尾流间隔，还要把相邻两条跑道上前后进近的航空器之间的距离作为约束条件，该距离为斜距间隔，通常跑道斜距间隔距离为 4 km，速度以 300 km/h 计算，转化为跑道斜距间隔时间为 48 s。

(6)时序变量一致性：

$$y_{ij}+y_{ji}=1,\ \forall i\neq j\in F \tag{9}$$

确保飞机 i 在 j 之前起飞/降落，反之亦然。

(7)延误变量定义：

$$d_i\geqslant t_i-\bar{t}_i,\ \forall i\in F \tag{10}$$

(8)跑道占用连续性约束：

$$t_i\geqslant t_i+\max(T_{ij},T'_{ij})+T_i-M(1-Z_{ijr}),\forall i,j\in F \tag{11}$$

前续航班抵离港运行的结束时间(即实际开始时间、跑道占用时间和尾流间隔时间之和)不早于其后续航班抵离港运行的开始时间，从而形成无间断的航班执行链，确保每架航空器在跑道占用时段内具有明确的连续性，有效避免了资源分配的时间重叠问题。

(9)子巡回消除约束：

$$u_i-u_j+|F|Z_{ijr}\leqslant|F|-1,\ r\in R,\ \forall i,j\in F_r,\ i\neq j \tag{12}$$

式中：n 为跑道 r 的航班数量；$|F|$ 为航班总数。当 $Z_{ijr}=1$ 时强制 $u_i<u_j$，消除调度序列中的循环。

3　改进的 NSGA-II 算法设计

3.1　染色体编码

染色体编码采用双重结构的实数编码方式，对航班序列和航班所使用跑道进行染色体编码，所有基因组的数据类型均为实数数值。对于包含 $|F|$ 个航班的调度场景，染色体总长度为 $2|F|$，前 $|F|$ 个基因位表示同跑道内航班的执行次序，是对同跑道航班按优先级数值进行升序排列；后 $|F|$ 个基因中每个基因位表示航班 i 分配的跑道编号 $r \in R$。假设有 12 个航班 F_1、F_2、…F_{11}、F_{12}，处在排序队列中，生成一条染色体，见表 2。

3.2　启发式规则生成初始可行解

针对多跑道航班调度这一特殊问题，若采用完全随机的方式生成初始解，由于需要同时满足跑道类型匹配、尾流间隔、时间窗口等多重约束条件，获得可行解的概率极低。因此，为提高算法效率，提出了基于启发式规则的初始化方法，其核心实施步骤如下。

步骤 1：对所有航班 $\forall i \in F$，根据航班 i 的计划抵离港时刻 $\bar{t}_i$ 进行排序，令排序结果为 $\{F'_1, F'_2, \cdots, F'_{|F|}\}$。

步骤 2：设定能够完成所有航班 F 的跑道集合，令 $i=1$。

步骤 3：找出能够执飞航班 F_i 的可行跑道集合 R。若 $R \neq \varnothing$，则从 R 中随机选择一条跑道 $\forall r \in R$ 执飞航班 F_i，转至步骤 4；否则，转至步骤 2。

步骤 3.1：令 $R = \varnothing$。

步骤 3.2：找出与航班类型相匹配的任意跑道 $\forall r \in R$，并获取该跑道的最后一个航班 F_r。

步骤 3.3：检查该跑道在完成航班 F_r 后是否能够继续执飞航班 F_i。若满足下列条件之一。①$F_r = \varnothing$，②$t_{F_r} + \max(T_{F_r}, T'_{F_r}) \leqslant t_{F_i}$（前序航班完成时间＋尾流间隔≤当前航班计划起降时间），满足条件则令 $R = R \cup \{r\}$。

步骤 4：令 $i = i + 1$，若 $i \leqslant |F|$，则转至步骤 3，否则，输出结果，算法终止。

3.3　滚动时域控制策略

由于机场航班量源源不断增加，为此，优化求解也必须采用滚动求解。滚动时域控制策略（receding horizon control，RHC）最先由 Hu 和 Chen[17] 引入航班调度研究中。RHC 采用时间分段、逐段优化、时域滚动方法，为了对优化时段 $[T_0, T_{end}]$ 内的所有航班进行优化，需先对预计起降时间在时域 $[T_0, T_0 + NT]$ 的航班进行优化，每次向前滚动，形成新的时域，直至到达 T_{end} 为止。

4　仿真实验

以广州白云国际机场某高峰时段航班信息作为优化调度的研究对象，且该机场跑道的主运行模式为西跑道 R_1 采用混合起降模式、东侧主跑道 R_2 采用起飞模式、东侧辅助跑道 R_3 采用降落模式，研究选取抵离港各占 50％的 50 架次航班作为优化样本，其机型、计划抵/离港时间等信息见表 3。

4.1　参数设置

航班协同优化调度模型的改进 NSGA-II 算法参数配置体系为：种群规模为 150，最大迭代次数为 300；遗传算子设置为 0.8 的交叉概率，变异概率设置为 0.1；在滚动时域控制方面，构建双周期（$N=2$）滚动优化框架，设置时间间隔为 $T=360$ s，实现动态环境下的渐进式寻优。

4.2　算例分析

4.2.1　不同的 NSGA-II 算法对比

传统 NSGA-II 算法以及改进的 NSGA-II 算法求解结果分别如图 2(a)和图 2(b)所示。

图 2(a)、图 2(b)展示了传统 NSGA-II 算法与改进的 NSGA-II 算法在迭代过程中目标函数 1 和目标函数 2 值的变化趋势。在图 2(a)中，以迭代过程为横轴、以优化目标——总延误时间为纵轴。从迭代趋势来看，改进 NSGA-II 算法因引入启发式规则生成初始可行解，其迭代起点的总延误时间显著低于传统 NSGA-II 算法，且在迭代过程中呈现出更平稳的下降态势，收敛速度明显优于传统算法，最终改进算法将航班总延误时间控制在 2 047 s。传统 NSGA-II 算法虽然在初期也呈现下降趋势，但整体下降速度明显较慢，最终保持在 2 924 s，与改进后的 NSGA-II 相比存在显著差距。从收敛精度上看，改进算法相较于传统算法实现了约 30％的优化幅度。因此，相较于传统方法，具有收敛更快、解质量更高、鲁棒性更强的综合优势。

表 2　NSGA-II 染色体编码示例

航班序号	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}
航班次序	1	2	3	4	5	6	7	8	9	10	11	12
跑道分配	1	3	2	2	1	3	1	2	1	3	1	3

表 3　航班基本信息

序号	航班号	机型	计划抵/离港时间/s	抵/离港	序号	航班号	机型	计划抵/离港时间/s	抵/离港
1	CZ7871	H	0	抵港	26	MU2077	M	38	离港
2	AK119	M	120	抵港	27	ET607	H	172	离港
3	8B168	M	187	抵港	28	CZ305	H	325	离港
4	VJ3909	M	320	抵港	29	CZ7871	H	459	离港
5	8B861	M	480	抵港	30	CZ347	H	600	离港
6	AK1395	M	615	抵港	31	CZ5055	M	743	离港
7	OD613	H	762	抵港	32	ZH3291	H	880	离港
8	VJ7527	M	908	抵港	33	SQ853	M	1 021	离港
9	KR938	M	1 050	抵港	34	AK1533	M	1 165	离港
10	C6512	M	1 190	抵港	35	BS326	M	1 308	离港
11	K6611	M	1 340	抵港	36	RA418	M	1 452	离港
12	ZA368	M	1 486	抵港	37	CX953	H	1 600	离港
13	6E1704	M	1 630	抵港	38	6E1702	M	1 883	离港
14	9C7007	M	1 780	抵港	39	9C8893	M	1 982	离港
15	AQ1139	M	1 921	抵港	40	AH3181	H	2 062	离港
16	GJ8059	M	2 065	抵港	41	9C8893	M	2 170	离港
17	AQ1157	H	2 210	抵港	42	TK299	H	2 314	离港
18	CZ8846	M	2 500	抵港	43	CZ301	M	2 458	离港
19	AQ1193	M	2 645	抵港	44	CZ3055	M	2 600	离港
20	3U6710	M	2 930	抵港	45	JL5032	H	2 745	离港
21	AQ1045	M	3 080	抵港	46	QF5004	H	2 889	离港
22	CZ9954	M	3 369	抵港	47	9C7419	M	3 032	离港
23	AQ1723	M	3 513	抵港	48	CX983	H	3 176	离港
24	9C6884	M	3 650	抵港	49	SQ8431	H	3 320	离港
25	CZ8481	M	3 800	抵港	50	CZ377	H	3 600	离港

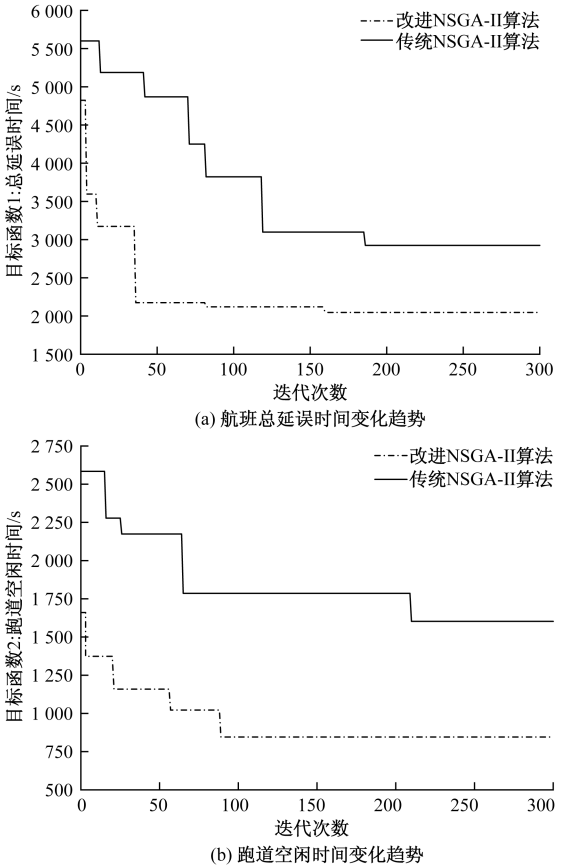


图 2　不同算法下目标函数 1、目标函数 2 的变化趋势

图 2(b)中,以迭代过程为横轴、以优化目标跑道空闲时间为纵轴。从图中得知,改进的 NSGA-II 算法在前 100 次迭代中快速下降,收敛速度更快且解的质量更优,而传统 NSGA-II 算法在整个 300 次迭代中下降幅度较小,最终目标函数值为 1 602 s。且改进的 NSGA-II 算法相较于传统 NSGA-II 算法在跑道空闲时间的最优值上提升幅度超过 47%,进一步验证了改进策略在航班调度问题中的有效性与优越性。

4.2.2　帕累托前沿解

图 3 展示了采用不同优化算法(先到先服务策略 FCFS、传统 NSGA-II 算法及改进 NSGA-II 算法)求解多跑道航班动态协同调度优化模型后,所得非劣解集在“总延误时间最小化”与“跑道空闲时间最小化”这两个目标函数构成的解空间中的分布情况,即帕累托前沿。

从图 3 中得知,先到先服务(FCFS)策略作为基准对比方案,其产生的非劣解集规模有限,主要聚集在解空间的右上方区域,且由于其调度决策仅依据航班计划抵港/离港的先后顺序,缺乏对系统全局资源的主动优化配置,导致调度结果的总延误时间与跑道空闲时间均处于较高水平,证明在高峰时

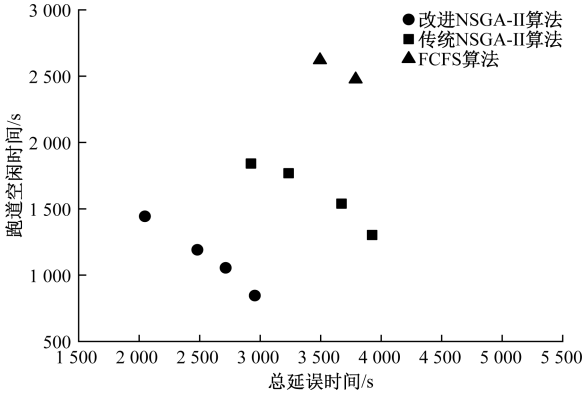


图 3　迭代过程中的 Pareto 前沿

段的复杂多跑道调度场景下，资源调配效率存在明显不足。传统 NSGA-II 算法的解集在目标空间中表现出较广的探索范围，表明该算法具备一定的全局搜索能力。然而，对应地，解集在跑道空闲时间指标上往往处于相对较高的水平，其收敛精度与解的质量存在一定局限。

相比之下，改进的 NSGA-II 算法所产生的帕累托前沿展现出更优的综合性能。在总延误时间和跑道空闲时间两个目标维度上同时获得更优的值。改进的 NSGA-II 算法通过引入启发式规则生成初始可行解、设计双层染色体编码以及滚动时域控制（RHC）策略，显著增强了对解空间的精细搜索，能够在种群迭代进化过程中，更有效地逼近真实的帕累托最优边界。

4.2.3　不同算法下航班占用跑道情况对比

图 4(a)、图(b)呈现了传统 NSGA-II 算法与改进 NSGA-II 算法在广州白云国际机场三跑道资源分配上的核心差异，其中传统 NSGA-II 算法中混合跑道承担了超过半数航班调度任务，虽满足跑道功能匹配的基础约束与机场既定运行模式，但混合跑道负载过重且专用降落跑道（R_3）资源未充分利用；而改进的 NSGA-II 算法显著降低了混合跑道的航班占比，减少了混合跑道的调度压力和不同类型航班的尾流间隔冗余，同时显著提升了两条专用跑道的利用率，使 3 条跑道的负载更趋均衡。

5　结论

由于航空运输业高速发展，多跑道枢纽机场面临资源分配不均、航班延误累积、跨跑道运行冲突等问题，传统先到先服务（FCFS）策略难以应对高峰时段动态调度需求。因此，通过构建多目标混合整数线性规划（MOMILP）模型，以总延误时间最小化与跑道空闲时间最小化为核心目标，纳入跑道功能

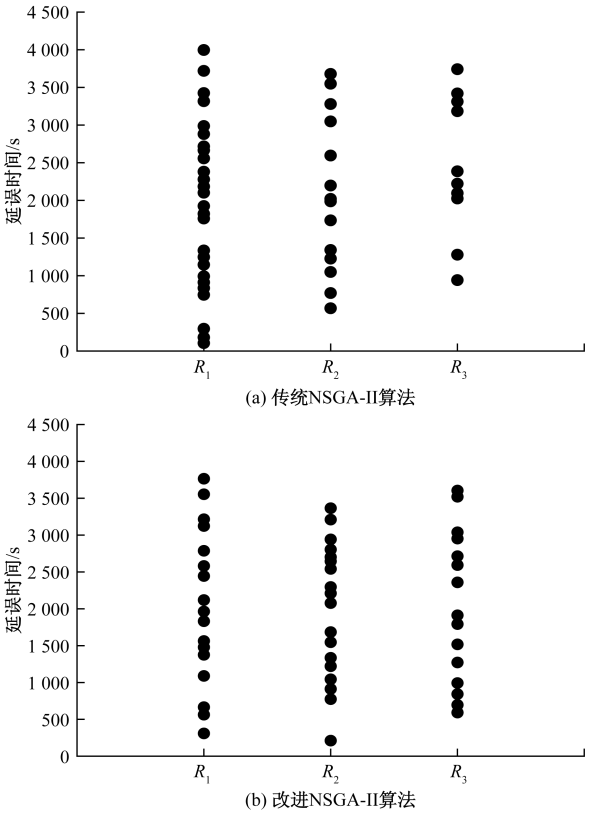


图 4　不同算法跑道使用情况对比

匹配、尾流间隔、近距跨跑道干扰等多重约束。在算法层面，提出改进的 NSGA-II 算法，通过启发式规则生成初始可行解、设计双层编码结构与滚动时域控制（RHC）策略，提升算法收敛速度与解的质量。对广州白云国际机场具体航班数据进行仿真实验，得到如下结论。

（1）相较于传统 NSGA-II 算法，改进后的 NSGA-II算法在优化效果与收敛性能上均表现出显著优势：在总延误时间优化方面，改进算法最终将总延误时间控制在 2 047 s，较传统算法的 2 924 s 降低约 30%。

（2）在跑道空闲时间优化方面，改进算法的优化幅度超过 47%，且实现了跑道资源的均衡分配，有效降低了混合跑道的负载压力，减少了不同类型航班的尾流间隔冗余。

（3）通过对比 FCFS 策略与两种不同 NSGA-II 算法的帕累托前沿解分布可知，改进 NSGA-II 算法的解集在总延误时间与跑道空闲时间两个目标维度上均占据优势，综合性能最优。

然而，模型研究主要是针对正常航班的优化调度，未考虑航班运行过程中的随机干扰因素，后续可引入航班延误、跑道临时占用等随机场景。未来在航班优化调度中需进一步研究并给出航班调度

策略(具体的航向、速度和高度建议)的方法,从而为管制员提供更为精细化的决策支持。

参考文献

[1] 马晓萱,张军峰,朱玲. 复杂多跑道机场离场运行的动态协同调度[J]. 交通运输工程与信息学报,2024,22(4):154-165.

[2] 张军峰,马婴,杜卓铭,等. 多跑道混合运行模式进场航班动态鲁棒调度[J]. 航空学报,2025,46(7):284-297.

[3] JIANG F, ZHANG Z. Optimal sequencing of arrival flights at metroplex airports: a study on shared waypoints based on path selection and rolling horizon control[J]. Aerospace, 2023, 10: 881.

[4] 吴维,魏明,叶志坚,等. 考虑相邻跑道干扰的多目标进离场航班排序优化模型[J]. 科学技术与工程,2021,21(6):2512-2517.

[5] SANDAMALI G, 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] LIEDER A, STOLLETZ R. Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways[J]. Social Science Electronic Publishing, 2016, 88: 167-188.

[7] 张兆宁,刘珂璇. 基于跑道运行类别的航班优化排序方法[J]. 重庆交通大学学报(自然科学版),2020,39(5):32-37.

[8] MA J, DELAHAYE D, LIANG M. Arrival and departure sequencing, considering runway assignment preferences and crossings[J]. Aerospace, 2024, 11: 604.

[9] BIKIR A, IDRISSI O, MANSOURI K, et al. An optimized air traffic departure sequence according to the standard instrument departures[J]. Advances in Computer Science and Applications, 2024, 15: 1364-1371.

[10] CHANDRASEKAR S, HWANG I. Algorithm for optimal arrival and departure sequencing and runway assignment[J]. Journal of Guidance, Control and Dynamics, 2015, 38(4): 601-613.

[11] 戴喜妹,张军峰,赵鹏力. 离场航班多目标优化排序研究[J]. 哈尔滨商业大学学报(自然科学版),2019,35(2):241-245.

[12] 向征,周鼎凯,孙赫阳,等. 基于强化学习的终端区离场排序研究[J]. 航空计算技术,2023,53(3):30-34.

[13] RIAHI V, NEWTON M A H, POLASH M M A, et al. Constraint guided search for aircraft sequencing[J]. Expert Systems with Applications, 2019, 118: 440-458.

[14] SÁEZ R, POLISHCHUK T, CHRISTIANE S, et al. Automated sequencing and merging with dynamic aircraft arrival routes and speed management for continuous descent operations[J]. Transportation Research Part C: Emerging Technologies, 2021, 132: 103402.

[15] SÁEZ R, PRATS X, POLISHCHUK T, et al. Automation for separation with continuous descent operations: dynamic aircraft arrival routes[J]. Journal of Air Transportation, 2020, 28(4): 144-154.

[16] 王建忠,丁小芹,王树伟. 考虑潜在冲突的进离场航班协同排序[J]. 交通运输系统工程与信息,2023,23(5):312-320.

[17] HU X B, CHEN W H. Receding horizon control for aircraft arrival sequencing and scheduling[J]. IEEE Transactions on Intelligent Transportation Systems, 2005, 6(2): 189-197.

Optimisation of Coordinated Scheduling for Arriving and Departing Flights in Multi-runway Airport Operations

LI Jia, WANG Minke, ZHAI Guo

(Airport College, Civil Aviation Flight University of China, Guanghan 618307, China)

Abstract: Uneven resource allocation and cumulative flight delays are prominent issues at multi-runway hub airports. To address these problems, a dual-objective mixed-integer linear programming model was developed. The model aimed to minimize both total flight delay and runway idle time. Constraints including runway functional matching and wake turbulence safety separation were incorporated. An improved non-dominated sorting genetic algorithm (NSGA-II) was employed to generate solutions. Pareto optimality relationships were analyzed to determine the optimal runway assignments and execution sequences for flights. Simulation was conducted using peak-hour flight data from Guangzhou Baiyun Airport. It is found that the improved algorithm reduces total delays by 30% and optimizes runway idle time by more than 47%, which achieves balanced runway utilisation. Its performance surpasses that of traditional algorithms and first-come-first-served strategies. Theoretical and methodological support is thus provided for relevant scheduling operations at multi-runway hub airports.

Keywords: mixed-integer linear programming model; flight execution sequence; wake turbulence safety separation