

基于多目标分阶段的跑道调度计划算法*

刘期建, 何元清

(中国民航飞行学院 计算机学院, 四川 广汉 618307)

摘 要: 针对多条跑道环境下离港飞机调度问题,提出了一种基于多目标、两阶段算法。算法第一阶段以飞机重量类型为主要分解参数,生成离港飞机序列。该参数在跑道调度计划问题上比其他参数更具影响力和稳定性。算法第二阶段从离港飞机队列中选取可用序列,将特殊航班指配到目标类型序列中,生成优化的飞行航班时刻表。实验表明,采用两阶段跑道调度计划算法进行多跑道离港飞机调度比采用先来先服务算法调度在跑道总吞吐量上有明显改善,能有效降低机场航班延误,提高跑道运营效率。

关键词: 跑道调度计划; 多目标; 两阶段算法; 重量类型序列; 离港优化

中图分类号: TP301 **文献标志码:** A **文章编号:** 1001-3695(2012)04-1267-03

doi:10.3969/j.issn.1001-3695.2012.04.018

Multi-objective and decomposition-based optimization algorithm for runway operations planning

LIU Qi-jian, HE Yuan-qing

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

Abstract: This paper proposed a multi-objective and two-stage based algorithm to solve the departure optimization problem of multi-runway situation. In the first stage, generated and ranked sequences of departure aircrafts by weight class, which was a primary decomposed parameter. It was more dominant and stabilized as for the runway operations planning than other parameters. In the second stage, populated the target class sequences with specific flights from the pool of available aircrafts to generate optimized aircraft schedules. The experiment shows that the new algorithm is more effective in the total throughput than the algorithm of first come first serve. It can reduce the airplane delays at the airport and improve the efficiency of runway operations.

Key words: runway operations planning; multi-objective; two-stage algorithm; weight class sequences; departure optimization

跑道调度计划(runway operations planning, ROP)是指在给定的时间段内,针对多条跑道上飞机起飞、着陆和穿越等飞行活动,将机场跑道资源进行优化分配和调度,生成关于这些飞行活动的调度计划。目前民航运输高速增长,机场容量^[1]与需求矛盾日显突出,航班计划在时间上相互衔接,常使用集中到港和集中离港方式进行调度,容易引起浪涌式交通流量,使机场容量过载,造成航班延误^[2]。根据美国运输部的统计,20%的航班延误15 min以上,其中四分之一超过1 h,每年由于机场和空域拥塞造成的经济损失高达70亿美元^[3]。机场容量负荷过大,吞吐量趋向饱和,大多枢纽机场已建设多条跑道以缓解调度压力。随着机场调度问题日趋复杂,管制员工作量将大幅增加,跑道调度问题面临严峻考验。在确保安全的前提下,要减少航班延误、提高跑道运营效率是机场管理中亟待解决的课题之一。

近年来,欧美学者广泛运用排队理论来建立ROP模型研究离港调度问题。Shumsky^[4]提出了一个机场容量和滑行迟滞的模型用于预测离港航班时间。Hebert等人^[5]根据观测数据和概率分布理论提出了一个Markov链模型来预测纽约拉瓜迪亚机场(LaGuardia airport)的起飞延误时间。Pujet等人^[6]对上述排队模型进行扩展,建立一个基于“输入/输出”的机场排

队模型,以便提供更多细节来评估起飞调度计划的有效性。Teixeira^[7]采用的启发式算法实际上是一种针对着陆间隔放行的启发式贪心搜索改进算法。Chang^[8]提出了一种启发式搜索算法,用于求解航班排队和停机位分配问题,并按照着陆、机位分配和起飞优先序列进行调度。当有多架飞机准备起飞时,按照每架飞机上的乘客人数降序排列以降低延误时间,从而达到最小经济开销的目标。该方法无法保证跑道利用率的优化。国内研究大多是围绕单跑道环境飞机着陆计划进行^[9,10],对多跑道环境起飞计划调度问题探讨较少。

1 算法概述

ROP分解算法是将优化过程划分为两个相对简单的阶段,每阶段建立各自的启发式规则集,算法流程如图1所示。第一阶段的目标为吞吐量最大化,考虑飞机尾流间隔和跑道穿越等因素;第二阶段的目标为延误时间最小化及空中交通管制规定,考虑流量管理、管制员工作负荷限制、交叉跑道环境等因素。算法第一阶段的核心是确定一个最优飞机重量类型序列,需要计算包括跑道穿越在内的每个离港重量类型序列所占用的跑道吞吐量。算法第二阶段的核心是将飞机尾号指配到航班时刻表中,采用整数规划描述时隙序列中的特定航班,生成

收稿日期: 2011-09-03; 修回日期: 2011-10-31 基金项目: 国家自然科学基金资助项目(60879023)

作者简介: 刘期建(1982-),男,四川广汉人,讲师,硕士,主要研究方向为计算机软件工程、民航计算机应用(qijian67@sina.com);何元清(1967-),男,教授,博士,主要研究方向为航空计算机应用、并行算法分析与设计、高性能计算。

优化的起飞计划。

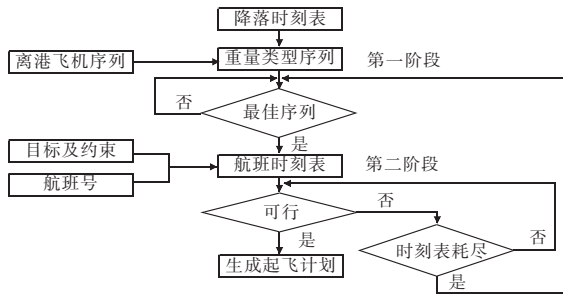


图1 两阶段算法流程

ROP算法的优势在于,当进入第二阶段前,重量类型序列已经确定,第二阶段优化算法的搜索空间将随之减小。假定共有 N 架离港飞机进入调度序列,包括 N_H 架重型飞机、 N_L 架大型飞机和 N_S 架小型飞机。若采用简单搜索算法,设第一阶段搜索空间为 N_1 ,第二阶段搜索空间为 N_2 ,由式(1)可知: $N_1 \geq N_2$,故两阶段ROP算法比一次性算法效率高。

$$\frac{N_1}{N_2} = \frac{N!}{N_H! \times N_L! \times N_S!} = \frac{(N_H + N_L + N_S)!}{N_H! \times N_L! \times N_S!} \quad (1)$$

2 两阶段ROP算法

2.1 分解公式

文献[11]描述了将ROP问题进行功能分解和关系分解后得到的需求集与参数集的耦合关系,通过解耦将飞机的重量类型作为主要分解元,得到第一阶段的目标函数为

$$\max T = \min T_{\text{total}} = \min \sum_{m,n \in N_D} T_{m,n} \quad (2)$$

其中: T 为吞吐量; T_{total} 为完成起飞序列所需的总时间,或任意连续起飞任务 m 和 n 的时间间隔 $T_{m,n}$,而 m,n 均属于起飞计划中的航班任务组 N_D 。每个 $T_{m,n}$ 正好是关于两次连续任务重量类型的函数,可将优化目标函数表示为需求集 $\{FR_1\}$ 和参数集 $\{DP_1\}$ 的函数。

$$\begin{aligned} \{FR_1\} &= \max T = \min T_{\text{total}} = f_1(\{DP_1\}) \\ T_{m,n} &\geq SEP_{m,n} \quad \forall m,n \in N_D \end{aligned} \quad (3)$$

式(3)的规则集中, $SEP_{m,n}$ 是连续起飞任务 m,n 之间的最低时间间隔,是以重量类型作为参数的函数,故规则集可以假定为关于 $\{DP_1\}$ 的函数,第一阶段待优化的问题将重量类型作为设计参数。

第二阶段优化问题的参数集设计如式(4)所示。其中: SA 为飞行任务间隔; MPS 为最大位移; TOD 为滑行延迟。

$$\{DP\} - \{DP_1\} = \{DP_2\} = \{SA, MPS, TOD\} \quad (4)$$

式(5)表示优化函数,规则集中 DFC 为降落流量限制, PSC 为位置限制:

$$\begin{aligned} \{FR_{2\text{obj}}\} &= \min T_d = f_2(\{DP_2\}) \\ \{FR_{2\text{const}}\} &= \{DFC, PSC\} = g_2(\{DP_2\}) \end{aligned} \quad (5)$$

2.2 算法第一阶段

第一阶段的目标是起飞吞吐量最大化,这表明作为线性计划和决策变量的重量序列 WC_i 将联系离港序列。每个决策变量表示重型(H)、大型(L)、小型(S)三种机型。

第一阶段规则集建立主要考虑飞机间隔必须满足空中交通管制(air traffic control, ATC)对跑道调度的最小间隔要求,即在同一跑道运行时间间隔或距离间隔的集合。对起飞而言,前后两架离港飞机间隔是调度系统矩阵的一个输入参数,矩阵

中的行对应领先飞机重量序列,列对应跟随飞机重量序列。可根据实时调度情况,适当修改矩阵值,如另一条跑道有降落飞机要滑行到停机位且必须穿越正在使用的跑道,那么需考虑另外两个规则:a)限制连续降落,即连续到港的飞机数量必须小于等于滑行道区域起飞和降落的吞吐量容限;b)控制穿越等待飞机最大延迟时间。

采用最小化运行时间以达到最大化吞吐量的目标,得出目标函数。设 N_A 为到港飞机总数, N_D 为离港飞机总数,那么, $N = N_A + N_D$ 表示当前调度窗口的混合运行总数,则最大化吞吐量 T 可以通过最小化跑道调度时间 T_i 实现,即

$$\min T = \max T_i \quad 1 \leq i \leq N_A + N_D \quad (6)$$

另一种方式通过累加各起飞时间间隔 $T_{m,n}$ 的总时间 T_{total} 制定目标函数:

$$T_{\text{total}} = \sum_{m,n \in N_D} T_{m,n} \quad (7)$$

最大化跑道吞吐量(T)可通过最小化 T_{total} ,或者说最小化各 $T_{m,n}$ 的值来实现,如式(2)所示。每个时间间隔的长度都取决于决策变量 WC_i 以及离港飞机尾号,目标函数的形式取决于前后飞机间隔标准。

$$T_{m,n} = f(WC_i, WC_{i-1}) \quad \forall \text{departure } i \in [2, \dots, N_D] \quad (8)$$

第一阶段的核心元素为离港飞机重量类型序列,是一个由飞机重量决定的起飞序列,其功能函数主要实现最大化离港吞吐量。该阶段算法模块的基本思想就是如何改善过去几年间对飞机序列^[12,13]以及到港时间准确预测^[14]的能力。由于跑道形态的影响,部分或是全部计划到达航班在着陆或减速后,均可能穿越使用跑道。此时可根据降落飞机的重量类型以及滑行道间隔要求计算出穿越所需时间。多数情况下,到达航班所穿越的跑道与离港航班占用的跑道相同,故最大化跑道吞吐量需要确定恰当的起飞和穿越队列。

第一阶段输出是一个关于起飞重量类型序列 CS 的矩阵,如式(9)所示, m 为产生的重量类型序列数量,每个序列中含有若干飞机类型序列。按照容量排序,每一行为类型序列,序号表示相对其他序列的特殊类型队列,最优序列置于首行。

$$CS = [CS_1, CS_2, \dots, CS_m] \quad (9)$$

2.3 算法第二阶段

将第一阶段的输出 CS 矩阵中得到的最优序列作为目标类型序列(target class sequence, TCS),第二阶段目标是将特殊飞机重量类型间隔指配到TCS中。若已选择的TCS不能产生可行的解决方案,将读取 CS 矩阵中下一行形成新的TCS。

在对机场时刻进行指配时,基本规则是将所需起飞时隙指配给每架飞机,即所指配起飞时隙要早于飞机能够滑至跑道的最早时隙,TCS中的机场时隙序列须满足此规则。

$$\sum_{j \in L} X_{ij} = 1 \quad \forall L_i \quad (10)$$

其中: L 为TCS中大型飞机的时隙序列。如果机场时隙包括2、3、4、5、6,且均为大型飞机时隙,而飞机1是剩余飞机之一,那么最终解决方案应满足 $X_{1j} = 0, \forall j \in [1, 7, 8, 9, 10]$,以保证规则集 $\sum_j X_{1j} = 1, j \in [2, 3, 4, 5, 6]$ 成立。

此外,令机场时隙中 N_s 等于起飞计划时隙 N_D ,每个时刻均有确定的飞机占用:

$$\sum_{i=1}^{N_D} X_{ij} = 1 \quad \forall j \quad (11)$$

显然,每架飞机只能占用一个时隙,如式(12)所示。

$$\sum_{j=1}^{N_s} X_{ij} = 1 \quad \forall \text{ aircraft } i \tag{12}$$

在 ATC 运行中,存在多种位置序列和时间序列约束,以保证飞机正常放行和安全起飞。第二阶段是通过最小化时间延迟以达到提高机场运营效率,同时要遵循每个特殊起飞计划的规则集。机场运营延迟涉及降落时间 T_{on_i} 、跑道穿越时间 T_{xi} 和起飞时间 T_{off_i} ,以及对应的每架飞机初始时刻 E_{on_i} 、 E_{xi} 和 E_{off_i} 。然后利用起降时隙和平均滑行时间计算出起飞时间或预测降落占用跑道的的时间。因此,机场运营中的时延体现了一次调度中航班延误的时长。最小化调度累积时延为

$$\min \left(\sum_{i=1}^{N_D} |T_{off_i}(p_{TO_i}) - E_{off_i}|^k + \sum_{j=1}^{N_A} |T_{on_j} - E_{on_j}|^k + \sum_{m=1}^{N_A} |T_{x_m} - E_{x_m}|^k \right) \tag{13}$$

其中: p_{TO_i} 是离港时刻飞机 i 的位置。 k_A 、 k_D 和 k_X 为正值,且 $k_A \geq 1, k_D \geq 1, k_X \geq 1$ 。若 $k = 1$,目标函数就是一个线性量。

第二阶段的核心是对离港飞机的调度。每架飞机调度应符合 TCS 中的特定起降时隙。为了生成第一阶段未确定的离港飞机调度算法,在每个 TCS 中的起降时刻,采用分枝界限法^[15]对相同重量类型的离港飞机进行抽样。生成的调度方案不一定是适和系统的优化算法,也有部分可能与规则集冲突。遇到这种情况,将丢弃第一个不可行的调度方案,选取下一个飞机调度方案并检查可行性。若所有调度算法均被认为不可行,需从 CS 矩阵中选取下一个可用的类型序列替换原 TCS。接下来按照算法第二阶段将特殊飞机类型指配到第一阶段重量类型中,并将飞机平均起飞时刻分配到 TCS 表中,作为目标类型序列,最终生成一个可行的最佳跑道调度计划时刻表。

第二阶段的输出是由飞机时刻表,如式(14)所组成的 AS 矩阵,矩阵中的每行为飞机时刻序列,行号为特定调度序列。AS 矩阵形式如下,其中第一行是最优调度序列。

$$AS = [AS_1, AS_2, \dots, AS_m] \tag{14}$$

3 实验结果

ROP 算法第一阶段生成的所有重量类型序列按照随机吞吐量排序之后,在第二阶段都会作为一个目标类型序列指配给飞机,即将飞机编号分配到重量类型序列中,产生最优跑道调度计划。实验以 15 架起飞航班构成的目标类型序列,其中 3 架小型机、10 架大型机、2 架重型机,与 16 架降落飞机进行算法模拟,保证最大穿越延迟和滑行道容量。实验参数设置如下:最大穿越延迟为 200 s;滑行道容量设置为 4 架小型飞机,或者 2 架大型飞机,或者 1 架重型飞机与 1 架小型飞机;假设所有小型机只使用同一个穿越点,并且当其他穿越点有大型机或重型机时优先离港。

表 1 为第二阶段优化后输出的飞行时刻表,含“*”的时刻表示存在跑道穿越活动。该时刻表采用第一阶段的飞机重量类型序列为 $\{L, S, L, s, s, s, L, L, l, s, s, L, L, s, s, s, L, S, H, s, s, s, s, L, H, s, s, s, L, S, L\}$,其中小写字母表示机场穿越。

若采用照先来先服务 (first come first serve, FCFS) 算法进行调度,结果如表 2 所示。由于进场飞机的优先级较高,若飞机连续降落时,进场飞机将长时间持续占用离场跑道。假设每架飞机穿越跑道的的时间为 40 s,由于没有采用分组穿越,故跑道穿越时间开销为 640 s。然而采用两阶段 ROP 算法的开销

为 310 s,比 FCFS 算法减少 330 s。由于进场飞机的优先级较高,即使离场飞机已经到达跑道,只要有飞机降落,离场飞机就必须等待。FCFS 算法中的穿越时间及跑道空闲时间将全部累加到离场飞机的起飞延迟上,从而导致离场跑道使用效率和机场吞吐量的下降。将表 3 中两阶段 ROP 优化算法与 FCFS 算法仿真结果比较发现,两阶段 ROP 优化算法产生的跑道调度计划通过对飞机重量类型序列的优化设计及飞机编号在重量序列上的分配优化,有效地提高了跑道的吞吐量,其中起飞吞吐量提高了 8.55 架次/h,跑道总吞吐量提高了 17.59 架次/h,有效减少了离场延误。同时,在保证进场飞机一定优先级及穿越点容量允许的前提下,采用成组交叉的原则避免了离场飞机长时间等待的状态。

表 1 二阶段 ROP 算法输出的飞行时刻表

时刻	航班号	重量类型	推出顺序	到达顺序	起飞顺序	耗时/s	穿越延迟/s
8:37	DEP003	L	4		1	600	
8:38	DEP007	S	1		2	924	
8:39	DEP006	L	5		3	690	
8:40 *	ARR001	S		13		0	174
	ARR002	S		14		0	114
	ARR003	S		15		0	54
8:41	DEP002	L	3		4	870	
8:42	DEP001	L	2		5	990	
8:43 *	ARR004	L		16		0	159
	ARR005	S		17		0	104
	ARR006	S		18		0	39
8:44	DEP004	L	6		6	942	
8:45	DEP008	L	8		7	852	
8:46 *	ARR007	S		19		0	154
	ARR008	S		20		0	94
	ARR009	S		21		0	24
8:47	DEP005	L	13		8	678	
8:48	DEP010	S	10		9	942	
8:49	DEP011	H	7		10	1 212	
8:50 *	ARR010	S		22		0	194
	ARR011	S		23		0	139
	ARR012	S		24		0	74
	ARR013	S		25		0	4
8:51	DEP009	L	11		11	1 089	
8:52	DEP015	H	9		12	1 260	
8:53 *	ARR014	S		26		0	129
	ARR015	S		27		0	64
	ARR016	S		28		0	4
8:54	DEP012	L	14		13	1 044	
8:55	DEP013	S	12		14	1 194	
8:56	DEP014	L	15		15	1 188	

表 2 FCFS 算法输出的飞行时刻表

时刻	航班号	重量类型	时刻	航班号	重量类型
8:37:00	DEP001	L	8:49:20 *	ARR012	S
8:38:00 *	ARR001	S	8:50:00 *	ARR013	S
8:38:40 *	ARR002	S	8:50:40	DEP006	L
8:39:20 *	ARR003	S	8:51:40 *	ARR014	S
8:40:00	DEP002	L	8:52:20 *	ARR015	S
8:41:00 *	ARR004	L	8:53:00 *	ARR016	S
8:41:40 *	ARR005	S	8:53:40	DEP007	S
8:42:20	DEP003	L	8:54:40	DEP008	L
8:43:20 *	ARR006	S	8:55:40	DEP009	L
8:44:00 *	ARR007	S	8:56:40	DEP010	S
8:44:40 *	ARR008	S	8:57:40	DEP011	H
8:45:20	DEP004	L	8:58:40	DEP012	L
8:46:20 *	ARR009	S	8:59:40	DEP013	S
8:47:00 *	ARR010	S	9:00:40	DEP014	L
8:47:40	DEP005	L	9:01:40	DEP015	H
8:48:40 *	ARR011	S			

始的不一致率 $\xi_{\text{Page-blocks}} = 0.008$ 。由于空间有限,将在以后的工作中调整 β 和 ξ 值,以获得更好的学习精度。

从表3中可以明显看出,VDP方法的平均分类学习精度最高。最后一行是指12个数据集的每个离散化算法的平均级别,如果算法的性能最佳,则平均级别为1,……。接下来对实验结果进行统计性分析。

2.2 统计性分析

为了衡量VDP的有效性,采用Friedman统计^[10]来测试所有离散化方法是否有显著的差异,如果存在显著的差异,Holmc统计测试将被采用,目的是进一步衡量VDP方法与其他方法相比是否存在统计意义上的显著差异,统计中用到的显著水平 $\alpha = 0.05$ 。Friedman统计表示如下:

$$\chi^2 = \frac{12J}{T(T+1)} \left[\sum_j Q_j^2 - \frac{T(T+1)^2}{4} \right] \quad (5)$$

其中: J 是数据集的个数; T 是离散化算法的个数; $Q_j = \frac{1}{J} \sum_i v_i^j$, v_i^j 指第*i*个数据集、第*j*个算法的平均级别。

通过计算,表3的Friedman统计值为45.5383,大于阈值11.1,说明六种离散化方法存在差异。这样,Holm's post-hoc统计测试^[10]将被采用,统计表明VDP、CHI²和CAIM性能相当。然而,如果去掉entropy-based和EFD方法,通过测试,VDP则显著地优于其他三种方法。

3 结束语

离散化方法对计算机多方面的研究与应用有着重要作用。近年来,尽管有多种离散化方法被提出,但具有一定影响和一定理论意义的并不多。本文提出的连续属性值域划分的离散

化新方法,很好地提高了C5.0决策树的分类学习精度。

参考文献:

- [1] SU C T, HSU J H. An extended CHI2 algorithm for discretization of real value attributes [J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17(3): 437-441.
- [2] 谢宏,程浩忠,牛东晓. 基于信息熵的粗糙集连续属性离散化算法[J]. 计算机学报, 2005, 28(9): 1570-1574.
- [3] FAYYAD U M, IRANI K B. Multi-interval discretization of continuous-valued attributes for classification learning[C]//Proc of the 13th International Joint Conference on Artificial Intelligence. San Mateo, CA: Morgan Kaufmann, 1993: 1022-1027.
- [4] TSAI C J, LEE C I, YANG Wei-pang. A discretization algorithm based on class-attribute contingency coefficient [J]. Information Sciences, 2008, 178(3): 714-731.
- [5] KURGAN L A, CIO S K J. CAIM discretization algorithm [J]. IEEE Trans on Knowledge and Data Engineering, 2004, 16(2): 145-153.
- [6] RUIZ F J, ANGULO C, AGELL N. IDD: a supervised interval distance-based method for discretization [J]. IEEE Trans on Knowledge and Data Engineering, 2008, 20(9): 1230-1238.
- [7] JIN R, BREITBART Y, MUOH C. Data discretization unification [J]. Knowledge and Information System, 2008, 19(1): 1-29.
- [8] ZIARKO W. Variable precision rough set model [J]. Computer and System Science, 1993, 46(1): 39-59.
- [9] DEMŠAR J. Statistical comparisons of classifiers over multiple data sets [J]. Journal of Machine Learning Research, 2006, 7: 1-30.
- [10] HETTICH S, BAY S D. The UCI KDD archive [EB/OL]. (1999). <http://kdd.ics.uci.edu/>.
- [5] HEBERT J E, DIETZ D C. Modeling and analysis of an airport departure process [J]. Journal of Aircraft, 1997, 34(1): 43-47.
- [6] PUJET N, FERON E. Input-output modeling and control of the departure process of congested airports [J]. Air Traffic Control Quarterly Journal, 2000, 8(1): 1-32.
- [7] TEIXEIRA R. A heuristic for the improvement of aircraft departure scheduling at airports [D]. Leicestershire: Loughborough Technical University, 1992.
- [8] CHANG C. Flight sequencing and gate assignment at airport hubs [D]. College Park: University of Maryland, 1994.
- [9] 张洪海,胡明华. 多跑道着陆飞机协同调度多目标优化[J]. 西南交通大学学报, 2009, 44(3): 402-409.
- [10] JIANG Yu, YANG Ying-bao, ZHOU Hang. Innovative predatory search algorithm for aircraft arrival sequencing and scheduling problems [J]. Transactions of Nanjing University of Aeronautics & Astronautics, 2010, 27(4): 361-366.
- [11] 刘期建. 多目标分阶段的跑道调度计划算法研究 [D]. 成都: 电子科技大学, 2009.
- [12] 牟奇峰,王慈光. 飞机进近序列快速优化 [J]. 西南交通大学学报, 2008, 43(3): 404-408.
- [13] 马正平,崔德光,陈晨. 空中交通进近排序及优化调度 [J]. 清华大学学报: 自然科学版, 2004, 44(1): 122-125.
- [14] 杨军利,向小军. 基于最小成本的飞机着陆规划算法 [J]. 中国民航学院学报, 2006, 24(2): 34-36.
- [15] 王敏. TSP问题及几种常见算法的比较研究 [J]. 长春理工大学学报, 2010, 5(5): 184-185.

(上接第1269页)

表3 二阶段ROP优化算法调度与FCFS调度结果比较

参数	两阶段ROP算法	FCFS算法
调度时间	20.00 min	24.67 min
起飞吞吐量	45 架次/h	36.45 架次/h
跑道总吞吐量	93 架次/h	75.41 架次/h

4 结束语

基于多目标分阶段的启发式ROP优化算法以飞机的重量类型作为主要分解元,将起飞调度计划分解为两个阶段,第一阶段以吞吐量最大化为目标,生成一个最优重量类型序列;第二阶段通过最小化延迟时间以达到提高运营效率的目的,输出航班时刻。通过算法仿真及数据分析发现两阶段的ROP优化算法在跑道调度效率上比FCFS算法更高,采用ROP优化算法进行跑道调度有助于降低起飞延迟,提高机场运营效率。

参考文献:

- [1] 彭莉娟,吴鹏,余静. 机场跑道最大容量评估模型的研究 [J]. 四川大学学报: 自然科学版, 2006, 43(5): 1018-1022.
- [2] 余江. 机场扩展终端区的运行优化策略研究 [D]. 成都: 西南交通大学, 2005.
- [3] BUSHNELL D M. Civil aeronautics: problems, solutions and revolutions [J]. Aerospace America, 2003, 41(4): 28-32.
- [4] SHUMSKY R A. Real-time forecasts of aircraft departure queues [J]. Air Traffic Control Quarterly Journal, 1997, 5(4): 281-308.