

单机调度问题在终端区排序中的应用研究

刘传涛 王仲生

(西北工业大学航空学院,西安 710072)

摘要 当终端区比较拥挤的时候,就需要对到达流航班着陆顺序进行调整,以有效缓解终端区的压力、缩短队列完成时间、提高飞行的安全性。飞机排序问题(ASP)属于 NP-hard 问题,通过将飞机排序问题表示成单机调度问题,设计了求解 ASP 的蚁群算法,以减少飞机队列总的完成时间最小为优化目标。通过与 FCFS 调度方法进行对比来验证基于蚁群算法的单机调度问题在单跑道机场终端区到达流排序中具有更好的适用性。

关键字 终端区排序 蚁群算法 单机调度

中国法分类号 V351.11; **文献标志码** A

近 20 年来,随着航空运输业的迅猛发展,仅靠机场基础设施的更新已经不能满足航空运输业的发展需求,终端区飞机排序问题开始得到各国研究机构和空中交通管理部门的广泛关注。以提高系统容量、减少航班延误和增加飞行的安全性为目的,研究表明,对到达流排序算法进行优化可以使系统容量在极端条件下提高 10% 以上。传统的飞机排序是采用先到先服务(FCFS)调度方法,即以先到先服务原则,在满足最小安全间隔等各项约束的前提下,确定最终的着陆顺序,其在不改变各航班间位置关系^[1]。

终端区到达流排序问题就是在不违反飞机安全间隔的前提下,为保证空中交通流快速、有序地流动,对飞机队列降落次序按照某种规则进行重新排列,合理安排飞机着陆次序,使得由于交通拥挤造成的航班延误所带来的影响最小。当航班为 n 架次时,其下降顺序的全排列有 $n!$ 种。因此,当航班架次较多时计算量很大,有可能造成计算机的计算溢出;另外,在实际情况中对航班的位置不能进行

大的调整,要限制其调整范围。当我们进行算法设计时,会加入一定的限制条件,所以全排列中的很多情况在实际计算时是不予考虑的^[2]。

蚁群算法最初由意大利学者 Dorigo M 于 1991 年首次提出,本质上是一个复杂的智能系统,它具有较强的鲁棒性、优良的分布式计算机制和正反馈机制、易于与其他方法结合等优点,具有较强的搜索能力,较好的全局收敛性,能较快得到全局最优解^[3]。

1 基于蚁群算法的单机调度问题模型

车间作业调度问题是蚁群算法的重要研究领域,蚁群算法的特点使得它在理论上比其他算法在求解单机调度问题时具有更大的优越性^[4]。一般意义的单机调度问题(Single Machine Scheduling, SMS)是车间作业调度问题的特例,特征模型可描述为: n 项相互独立的任务需要在一台机器上序贯处理,每项任务都有加工时间、交货期等参数,此外还要满足一些调度环境和约束条件的要求,调度目标就是要找到一个最优的任务序列使得系统加权总完成时间最小^[5]。

基于组合优化问题的定义,我们给出单机调度问题的理论模型:

2010 年 2 月 26 日收到

第一作者简介:刘传涛(1981—),男,汉族,河南商丘人,西北工业大学航空学院硕士研究生;研究方向:交通监控及系统规划。E-mail:tel_664@yahoo.cn。

SMS 问题 (S, f, Ω) 中,离散决策变量 $x_i (x_i \in [1, n])$ 表示待处理的 n 项任务,与之对应的值 $y_i (y_i \in [1, n])$ 表示该任务在机器上加工的次序即工序,状态节点 (x_i, y_i) 表明任务 x_i 在工序 y_i 进行加工,满足 $x_i \neq x_j, y_i \neq y_j, \forall i, j \in [1, n], i \neq j$ 。最小化目标函数 $f: S \rightarrow R^+$, 其中 S 为解空间, $S = \{s = \{(x_i, y_i), \dots, (x_n, y_n)\}\}$, 由所有满足问题约束条件 Ω 的可行解 S 构成,而任何一个可行解 $s \in S$ 都有相应的目标函数值 $f(s)$, SMS 问题的目标函数就是找到全局最优解 $s^* \in S$ 使 $f(s^*) \leq f(s), \forall s \in S$ [6]。

2 终端区到达流排序数学模型

假设某枢纽机场在 ΔT 时间段内有 n 架航班等待降落,具有各自不同的预期降落时间(ETS),要求在满足飞行最小安全时间间隔的基础上对它们排序,使得队列总的完成时间最小。连续两架航班的最小安全间隔应满足 ICAO 规定的时间间隔 s_{ij} (如表 1)。

表 1 ICAO 规定的飞机尾流间隔标准/s

尾流间隔		后机		
		重型	大型	小型
前	重型(H)	99	133	196
	大型(L)	74	107	131
	小型(S)	74	80	98

即在给定的时间窗 ΔT 内以合适的排序完成所有航班的降落任务,使得延误时间最小。目标函数为 $\min C_{\max}$ ($C_{\max}$ 为最后一架飞机的实际降落完成时间)。

基于单机调度问题的终端区到达流排序问题的数学模型可以表述为:

ASP 问题 (S, f, Ω) 中,离散决策变量 $x_i (x_i \in [1, n])$ 表示待降落的 n 架航班,与之对应的值 $y_i (y_i \in [1, n])$ 表示该航班在跑道上降落的次序,状态节点 (x_i, y_i) 表示航班 x_i 在次序 y_i 降落,满足 $x_i \neq x_j, y_i \neq y_j, \forall i, j \in [1, n], i \neq j$ 。最小化目标函数 $f: S \rightarrow R^+$, 其中 S 为解空间, $S = \{s = \{(x_i, y_i), \dots, (x_n, y_n)\}\}$, 由所有满足问题约束条件 Ω 的可行解 S 构成,而任何一个可行解 $s \in S$ 都有相应的

目标函数值 $f(s)$, ASP 问题的目标函数就是找到全局最优解 $s^* \in S$ 使得 $f(s^*) \leq f(s), \forall s \in S$ 。形成单机调度问题中的“工件最大推迟完成时间 $C_{\max}$ 最小”的模型 $n/(1/C_{\max})$ 。

约束条件有:(1)任意两架飞机之间必须满足最小安全间隔要求,即 $x_j \geq x_i + s_{ij}$;

(2) $d_i - i < c$, 表示航班 i 排序前后的位置差应小于 c , 是对航班调整的位置约束。这个约束条件主要考虑到飞机性能、管制员的负荷以及先到尽量先服务的公平原则等等,根据不同的要求选择不同的 c 值。

(3)任意一架飞机只能有一架前机或者后机,且所有飞机均在同一跑道降落。

2.1 终端区到达流排序的蚁群算法描述

基于单机调度问题的终端区到达流排序算法可描述为:在算法开始的时候,每只蚂蚁任意选取一个待着陆飞机作为起始出发点,根据信息素和状态转移概率,选择下一个着陆飞机,直到该蚂蚁遍历所有待着陆飞机,从而得到每只蚂蚁的排序结果,找出队列中完成时间最少的一个排序组合,增强该组合中所有状态点的信息素,挥发其他组合中所有点的信息素;如果未满足算法结束条件,算法转入下一次迭代,一旦超出最大迭代次数,算法结束。

2.2 信息素更新策略和状态转移概率

蚂蚁 $k (k = 1, 2, \dots, m)$ 在运动过程中,根据各状态点上的信息量决定其转移方向,这里用禁忌表 $tabu_k (k = 1, 2, \dots, m)$ 来记录蚂蚁 k 当前所走过的元素,集合随着 $tabu_k$ 进化过程作动态调整。在搜索过程中,蚂蚁根据各状态点上的信息量及其启发信息来计算状态转移概率,选择下一个元素。 $p_{ij}^k(t)$ 表示在 t 时刻蚂蚁 k 由元素 i 转移到元素 j 的状态转移概率:

$$p_{ij}^k(t) = \begin{cases} \frac{[t_{ij}(t)]^\alpha \cdot [\eta_{ik}(t)]^\beta}{\sum_{s \in allowed_k} [t_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & j \in allowed_k \\ 0, & else \end{cases} \quad (1)$$

式(1)中, $allowed_k = n - tabu_k$ 表示蚂蚁 k 下一步允许选择的飞机; α 为信息启发因子,表示轨迹的相对重要性,反映了蚂蚁在运动过程中所积累的信息在蚂

蚁运动时所起的作用; β 为期望启发式因子,反映了蚂蚁在运动中启发信息在蚂蚁选择路径时的受重视程度; $\eta_{ij}^l(t)$ 为启发函数,其表达式为 $\eta_{ij}^l(t) = 1/s_{ij}$ 。

为了避免残留信息素过多引起残留信息淹没启发信息,在每只蚂蚁走完一步或者完成对所有飞机的遍历后,要对残留信息进行更新处理。由此, $t + n$ 时刻在路径 (i,j) 上的信息量可按如下规则进行调整:

$$t_{ij}(t + n) = (1 - \rho)t_{ij}(t) + \Delta t_{ij} \tag{2}$$

$$\Delta t_{ij} = \sum_{k=1}^{n-ant} \Delta t_{ij}^k(t) \tag{3}$$

式(2)和式(3)中, ρ 表示信息素挥发系数,则 $(1 - \rho)$ 表示信息素残留因子,为了防止信息的无限积累, $\rho \in [0,1)$; Δt_{ij} 表示本次循环中在点 (i,j) 上的信息素增量,初始时刻 $\Delta t_{ij}(0) = 0$ 。Dorigo M提出的 Ant-Cycle 模型中规定:

$$\Delta t_{ij}(t) = \begin{cases} \frac{Q}{L_k}, & \text{若第 } k \text{ 只蚂蚁在本次循环中经过}(i,j) \\ 0, & \text{否则} \end{cases} \tag{4}$$

式(4)中, Q 表示信息素浓度,它在一定程度上影响算法的收敛速度; L_k 表示第 k 只蚂蚁在本次循环中所走路径的总长度。

2.3 基于 SMS 的 ASP 蚁群算法描述

ASP 蚁群算法的实现步骤如下:

STEP0:初始化, $P = \{1,2,\cdots,n\}$ 为待着陆飞机集合, $S = (s_{ij})_{n \times n}$ 为飞机最小安全间隔矩阵, $NC_max = 200$, 蚂蚁总数 $M = 200$, $\alpha = 0.5$, $\beta = 1$, $\eta_{ij} = 1/s_{ij}(i \neq j)$, $\rho = 0.7$, $Q = 120$ 。

STEP1:让蚂蚁随机选择第一架着陆飞机;

STEP2:For $n = 2$ to N

For $m = 1$ to M

If $allowed_m = \phi$,则完成第 m 只蚂蚁的计算;

Else 蚂蚁 m 从飞机 i 以概率 p_{ij}^m 寻找后机 $j(i \neq j)$, 更新 $allowed_m$;

End If

End For

End For

STEP3:检查每只蚂蚁的 $tabu(m)$, 得到飞机排序组合并计算队列完成时间,

For $m = 1$ to m

$$x_1 = E_0;$$

For $i = 2$ to N

$$x_i = x_{i-1} + s_{i,i-1};$$

End For

End For

队列中最后一架飞机完成时间 $C(m) = x_n + p(n)$ (跑道占用时间),找出所有循环中的 $\min C(m)$, 对应的飞机排序组合为 $tabu(m)$;

STEP4:更新信息素, 对 $tabu(m)$ 中的飞机邻接组合 (i,j) 的信息素进行加强, 否则减少,

If $(i,j) \in tabu(m)$

$$\Delta t_{ij} = \Delta t_{ij} + Q/\min C(m);$$

$$t_{ij} = (1 - \rho)t_{ij} + \Delta t_{ij};$$

$$\text{Else } t_{ij} = (1 - \rho)t_{ij}$$

End

STEP5: 记录迭代次数, $NC = NC + 1$, 若 $NC < NC_max$, 清空 $tabu(m)$, 转 STEP1; 否则循环结束, 输出结果。

3 实例仿真验证

本文选择某一单跑道机场的繁忙时间段内的 15 架待降航班进行验证, 表 2 中 A 代表航班号, ETA 表示飞机的预计降落时间, ACA 为蚁群算法排序结果。本例中参数选取为^[7]: $\alpha = 0.5$; $\beta = 1$; $\rho = 0.7$; $Q = 120$; 权重 (q_i) 分别取 2、1.5 和 1; 取 $c = 2$ 。在 matlab7.0 环境下对基于蚁群算法的单机调度模型进行仿真, 结果如下(如图 1):

表 2 待降航班各参数信息

航班 A	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
机型	S	L	H	L	H	L	L	L	H	S	L	H	L	S	L
ETS	45	70	131	183	395	460	575	740	875	885	906	1 193	1 289	1 350	1 493
权重	1	1.5	2	3	1	2	1	1.5	2	1.5	2	1.5	2	1	2

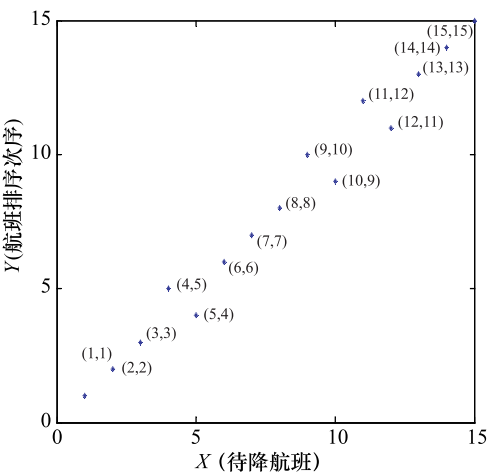


图 1 ASP 仿真结果

Shortest_Route = 1 2 3 5 4 6 7 8 10
9 12 11 13 14 15
Shortest_Time = 1 349.00 s

表 3 ACA 与 FCFS 仿真结果比较

待排序	FCFS	ACA	ACA
航班数量	调度结果	调度结果	改善水平
15	1 537 s	1 349 s	12.3%

从仿真结果(如图 1 和表 3)可以看出,ASP 蚁群算法优于 FCFS 调度方法,可有效减少着陆飞机队列完成时间,在本例中终端区待降航班数量为 15 架时 ASP 蚁群算法优化排序可以使队列总完成时间减少 12.3%。最大调整量和权值的引入使得优化结果兼顾了公平原则,使时间效益和经济效益同时得以体现,有利于其得到更进一步的推广适用。

4 结语

ASP 问题属于 NP 难问题,基于蚁群算法的单机调度问题模型在终端区到达流排序问题的求解中具有很好的适用性。单机调度问题是车间作业调度问题的特例,适用单机调度问题可以将飞机队列的延误损失也加以一定的考虑,体现了兼顾公平的原则。之所以采用蚁群算法进行求解,是因为它

采用正反馈机制,能够很快地发现较优解;采用分布式计算方式,避免过早地陷入局部最优解;采用贪婪式搜索方式,有助于搜索过程中早期找出可接受的解决方案,缩短了搜索时间。研究结果表明,此算法较 FCFS 调度方法在寻找最优解和有效解的实时性方面有很大优势。仿真结果表明当飞机较多时,该方法能使跑道利用率得到有效提高,能在缓解终端区压力方面起到很大程度的作用。

参 考 文 献

1 Hu X B,Chen W H. Genetic algorithm based on recedinghorizon control for arrival sequencing and scheduling. Egi-neering Applications of Artificial Intelligence,2005;18(6):633—642
2 李志荣,张兆宁. 基于蚁群算法的航班着陆排序. 交通运输工程与信息学报,西安. 2006;(02):1672—4747
3 段海滨. 蚁群算法原理及其应用. 北京:科学出版社,2005
4 赵伟,韩文秀,罗永泰. 准时生产方式下混流装配线的调度问题. 管理科学学报,2000;3(4):23—28
5 郑大钟,赵千川. 离散事件动态系统. 北京:清华大学出版社,2001
6 叶强. 基于改进蚁群算法的一类单机调度问题研究. 合肥:合肥工业大学博士论文,2008
7 陈欣,杨文东,陆迅,等. 一种机场终端区飞机排序问题的蚁群算法研究. 山东大学学报(工学版),2007;(06):1672—3961
8 唐恒永,赵传立. 排序引论. 北京:科学出版社,2002

Researchs on Aircrafts Sequencing in the Airport Terminal Area Based on Single Machine Scheduling Problem

LIU Chuan-tao,WANG Zhong-sheng

(College of Aeronautics , Northwestern Polytechnical University , Xi'an 710072, P. R. China)

[Abstract] When the airport terminal area busy,the sequence of the arriveing airplanes is need to adjust in order to realease the presure of airport terminal area,minimize the makespan of the landing airplanes,heighten the securi-ty of flighting. An aircraft sequenceing problem(ASA) is a non-deterministic polynomial hard(NP-hard) problem. ASP was treated as a particular Single Machine Scheduling(SMS), and an Ant Colony Algorithm (ACA) for the ASP is designed with the objective of minimizing the makespan of landing airplanes. The SMS based on ACA was in-vestigated by comparing with the scheduling results of FCFS,proves to be better on an airport with one runway.

[Key words] aircrafts sequencing in the airport terminal aera ant colony algorithm single machine scheduling