

文章编号: 1004-4353(2014)04-0335-05

# 基于改进的蚁群算法的教室管理优化问题

怀丽波, 崔荣一, 赵亚慧

( 延边大学工学院 计算机科学与技术系, 吉林 延吉 133002 )

**摘要:** 给出了教室管理问题的一种改进的蚁群优化方法. 考虑教室容量、课间距离和单双周课程等因素, 对抽象出的数据按优化方向排序, 将教室管理问题简化为带权二部图的完备匹配问题; 然后运用基于超立方框架的最大最小蚁群算法进行求解. 为有效减少搜索空间, 该算法按照教室类型对二部图结点进行分块搜索. 实验表明, 与基本蚁群算法相比, 该算法在解决教室管理优化问题上能得到较优解.

**关键词:** 教室管理; 带权二部图; 最大最小蚂蚁系统

**中图分类号:** TP301.6

**文献标识码:** A

## Study of classroom management optimal problem based on improved ant colony algorithm

HUAI Libo, CUI Rongyi, ZHAO Yahui

( Department of Computer Science & Technology, College of Engineering,  
Yanbian University, Yanji 133002, China )

**Abstract:** An improved ant colony algorithm was presented in order to solve the classroom management problem. This paper sorted the abstracted data along the optimized direction and turned the classroom management problem into the complete matching problem of weighted bipartite graph by analyzing the classroom capacity, the distant between classrooms, single and double week courses etc. Then, we gave optimization method using the improved Max-Min ant colony algorithm based on hyper-cube framework, which searched the node partitioned according to classroom type in order to reducing the searching space. Experimental results show that the improved ant system is able to construct significantly better solutions compared with ACA.

**Key words:** classroom management problem; weighted bipartite graph; MMACA

### 0 引言

近年来高校招生规模呈不断扩大趋势,使得许多高校都存在着学生数量多而相应的教室资源有限的情况,因此教室的安排是否合理直接影响到学校教育资源的有效利用. 教室管理问题作为排课问题的子问题,主要完成课程的教室分配任务. 排课问题作为一个多目标、带有模糊约束条件的组合优化问题,1975 年 E. nen 等证明了其属于

NP 完全问题. 目前,解决这个问题的方法有很多,主要有模拟退火算法、图着色算法、遗传算法、粒子群算法,以及几种算法的混合方法等<sup>[1-4]</sup>. 针对教室管理问题的优化,文献[5]给出了从课程表到教室调度表的实现方法,文献[6]针对减少教室流动性问题,提出固定一定教室比例的方法,但目前综合考虑教室管理优化问题的研究还很少. 基于此,本文利用改进的蚁群算法对该问题进行研究.

# 1 教室管理问题的数学描述

教室管理问题主要涉及 5 个要素:时间、教师、教室、班级、课程. 作为排课问题的一个子问题,教室管理优化问题假设时间、教师要素已经排好,故只涉及到 3 个硬性约束条件:①同一个教室在相同时间只能安排一门课;②每个教室的教室容量要大于上课班级的学生人数;③教室总数量在所有的时间里要满足课程总门数.

## 1.1 教室管理问题优化要点

针对教室管理的优化,本文研究主要集中在以下几个软约束条件:

- 1) 减少空闲座位,提高教室利用率;
- 2) 为提高学习效率,连续上课的同一班级尽量安排在同一教室,减少学生的课间流动性;
- 3) 选择教室时,尽量使教室资源集中使用,多余教室可作为自习室或社团活动室灵活使用;
- 4) 为使教室最大化利用,单周和双周同一时间上课的课元使用同一个教室.

## 1.2 教室管理优化问题的二部图模型

### 1.2.1 构造二部图的节点

1) 课元结点  $G_{LCT}$  由数组〈课程,班级,教师〉组成:课程集合包含课程代码、课程名称、上课时间、需要的教室类型等属性;班级有班级编号、名称、人数等属性;教师包括教师编号、姓名、所在部门、职称等属性. 课元是最小单位,每个课元有不同的编号,每周一次以上的课按不同的课元处理.

2) 目标结点  $G_{PR}$  由数组〈上课时间,教室〉组成:上课时间有〈单双周,星期,节次〉3 个属性;教室集合包含教室编号、名称、容量、教室类型等属性;同时需满足  $|G_{LCT}| \ll |G_{PR}|$ .

1.2.2 构造二部图的边和权值 作为教室管理优化问题,课元的上课时间已经确定,所以每个课元结点不是和所有的目标节点连边,二部图的边需要满足以下条件:①课元结点的上课时间和目标结点的上课时间相同;②教室的容量大于上课学生人数;③教室类型要匹配.

边的权值  $w(i, j) = \text{教室容量} - \text{学生人数}$ . 教室管理优化问题转化为为每个课元结点寻找一个对应的目标节点,即转化为带权二部图的完备匹配问题.

# 2 改进的蚁群算法求解教室管理优化问题

蚁群算法是对蚂蚁群落食物采集过程的模拟,其主要特点是正反馈、分布式计算. 蚁群算法先后被应用至 TSP 问题、车间作业调度问题、车辆路径问题等多个经典组合优化问题<sup>[7-8]</sup>;近年来,一些学者针对蚁群算法进行了改进,比如应用比较广泛的最大最小蚂蚁系统<sup>[9]</sup>等.

## 2.1 基本蚁群算法的数学模型

算法描述如下:初始时刻,各条路径上的信息素量相等,设  $\tau_{ij}(0) = C$  ( $C$  为常数),蚂蚁  $k$  ( $k = 1, 2, 3, \dots, m$ ) 在运动过程中根据各条路径上的信息量决定转移方向. 用  $k$  个禁忌表  $tabu_k$  ( $k = 1, 2, \dots, m$ ) 记录每只蚂蚁走过的路径,并随着蚂蚁位置的变化做出相应的动态调整. 在整个寻优过程中,蚂蚁根据每条边上的信息量和启发信息来确定转移概率以决定转移到哪个结点. 若  $p_{ij}^k(t)$  表示蚂蚁  $k$  在  $t$  时刻由  $i$  节点转移到  $j$  节点的概率,那么基本蚁群算法的转移概率为

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{s \in allowed} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}]^\beta}, & j \in allowed_k; \\ 0, & \text{否则.} \end{cases} \quad (1)$$

其中:  $allowed_k = \{0, 1, 2, \dots, n-1\} - tabu_k$  ( $k = 1, 2, 3, \dots, m$ ) 表示蚂蚁  $k$  下一个可以选择的节点;  $\eta_{ij}$  代表启发函数,一般取  $\eta_{ij} = 1/d_{ij}$ ,  $d_{ij}$  代表结点  $i, j$  之间的长度;  $\alpha$  代表信息量残留的重要程度;  $\beta$  代表启发函数的重要程度. 每条路径的信息量要在每只蚂蚁走完一步或完成遍历后,根据公式(2) 做出调整:

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij},$$
$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k, \quad (2)$$

其中  $\rho$  表示信息量随着时间挥发的系数,  $1-\rho$  表示信息量残留因子,  $\Delta\tau_{ij}$  表示每次循环中  $(i, j)$  上的信息量增量,  $\Delta\tau_{ij}^k$  表示第  $k$  只蚂蚁在本次循环中给路径  $(i, j)$  增加的信息量.

## 2.2 针对教室管理问题改进的蚁群算法

教室管理优化问题是为带权二部图寻找最小的完备匹配,即寻找一个单射  $f: G_{LCT} \rightarrow G_{PR}$ , 在尽量满足软约束条件下,总权值最小.

**2.2.1 数据准备** 1) 排序. 为提高蚁群算法的局部搜索能力,将课元按照班级学生人数降序排列(人数多的优先级高),同一个班级按照上课时间的先后排序;目标集按照教室容量升序排列,教室容量相同的按照上课时间先后排序;并将同一个班级单双周的课程合并成一个课元. 2) 分组. 为缩小蚂蚁的搜索空间,将两个节点集按匹配的教室类型进行分组,将二部图划分为几个二部子图.

**2.2.2 算法的主要内容** 1) 转移概率. 使用蚁群系统的伪随机比例规则:

$$j = \begin{cases} \arg \max \{ \tau_{ij}^\alpha \cdot \eta_{ij}^\beta \}, & \text{if } q \leq q_0; \\ J, & \text{否则.} \end{cases} \quad (3)$$

其中:  $\eta_{ij} = 1/\omega_{ij}$ ;  $q_0 (0 \leq q_0 \leq 1)$  是初始设定的参数;  $q$  是一个随机数,  $q \in [0, 1]$ ; 参数  $q_0$  的大小决定了利用先验知识和探索新路径之间的相对重要性. 每当蚂蚁要选择下一个目标时,它就选取一个随机数  $0 \leq q \leq 1$ , 如果  $q \leq q_0$ , 则根据先验知识公式来选择最好的边, 否则  $J$  按照公式概率(1)选择. 该转移策略将确定性选择和随机选择相结合, 既保证搜索收敛性好, 又避免过早陷于搜索停滞.

2) 信息素更新策略. 更新策略使用基于超立方框架的最大最小蚁群算法<sup>[13]</sup>, 通过设置信息素的上下限, 将各路径初始化信息素浓度设为  $\tau_{\max}$ , 这样可解决应用过程中出现的停滞和扩散问题, 避免信息素的无限制累加和可能出现的信息素为零的现象, 提高算法的鲁棒性.

同时为进一步增大最优路径边与劣质路径边之间的信息素量差异, 引入负反馈机制, 使蚂蚁的搜索行为更集中于最优解的附近. 每条路径的信息量在迭代后根据公式(4) 做出调整:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) - K \frac{L_{\text{worst}}}{L_{\text{best}}},$$
$$\tau \in [\tau_{\min}, \tau_{\max}], \quad (4)$$

其中  $K$  为引入的一个参数,  $L_{\text{worst}}$  表示当前循环中最差蚂蚁的路径长度,  $L_{\text{best}}$  表示当前最好的路径长度,  $\tau_{\min}$  和  $\tau_{\max}$  表示信息素的上下限.

**2.2.3 算法的主要步骤** 本文提出的教室管理优化问题的蚁群算法的实现步骤为:

step 1 参数初始化: 设置蚂蚁个数  $m$ , 迭代次数  $N=0$ ; 最大迭代循环次数  $N_{\max}$ , 初始化新信息素  $\tau_{ij}(0)=\tau_{\max}$ , 将  $m$  只蚂蚁随机分布在  $G_{\text{LCT}}$  中的

起点上;

step 2  $N := N + 1$ ;

step 3 当  $N < N_{\max}$  时, 蚂蚁  $k$  按照(1) 式和(3) 式访问  $G_{\text{LCT}}$  中的元素  $j (j \in (G_{\text{LCT}} \cup G_{\text{PR}}) - \text{tabu}_k)$ . 在访问过程中: 当两条边的权值相同时, 选择占用过的教室; 判断一个课元的上课时间是否为单双周, 如果是单双周上课的, 教室可重复占用;

step 4 选择具有最大状态转移概率的元素  $j$ , 修改蚂蚁  $k$  的禁忌表, 将  $j$  加入禁忌表中;

step 5 若没有遍历完  $G_{\text{LCT}}$  中的所有点, 跳转到第 3 步, 否则转到第 6 步;

step 6 计算蚂蚁  $k$  的路径长度  $L_k$ , 通过比较其大小求得最短长度  $L_{\text{best}}$  和最长度  $L_{\text{worst}}$ ;

step 7 按照(4) 式进行信息素更新;

step 8 如果满足结束条件, 即  $N = N_{\max}$ , 迭代结束, 输出结果; 否则, 清空禁忌表, 跳转到第 2 步.

### 3 实验结果与分析

为验证算法的有效性, 以延边大学工学院 69 门课程、6 个班级、16 个教室为数据进行仿真实验, 实验参数设置参照文献[14]. 对基本蚁群算法和改进蚁群算法进行对比试验, 对每次试验记录下最优解  $\text{Length\_best}$ , 以实现算法性能的比较.

图 1 和图 2 分别给出了基本蚁群算法和改进蚁群算法得出的类型相同、容量相同的教室排课表: 教室编号 6 和 7 的教室容量和类型相同, 矩阵表示的是周一到周五每天 5 个时间片安排的课程. 比较图 1 和图 2 可以看出: 在没有冲突的情况下, 基本蚁群算法在课程分布上比较均匀; 而改进的蚁群算法尽量把课程集中在一个教室, 这样可以空出其他教室作为自习室或社团活动室, 满足教室资源的合理利用.

图 3 是以班级为单位生成的课表, 矩阵表示的是周一到周五 5 个时间片所使用的教室情况. 从图 3 可以看出: 基本蚁群算法中, 班级 1 在相邻时间片占用 15231、15221 两个多媒体教室和两个机房 15252 和 23062; 改进的蚁群算法中, 班级 1 在相邻的时间片占用一个多媒体教室 15221 和一个 32062 机房; 班级 2 情况类似. 由以上知改进的蚁群算法减少了学生的课间流动性, 有利于改善教学秩序.

val (:, :, 6) =					
0	129	0	0	72	
0	0	124	0	92	
0	0	0	0	0	
0	122	121	102	100	
0	0	0	0	0	

val (:, :, 7) =					
0	113	0	0	0	
0	71	0	70	0	
0	0	62	0	127	
73	87	0	122	0	
0	0	0	0	0	

图 1 ACS 算法的教室排课表(教室编号 6 和 7)

val (:, :, 6) =					
0	77	65	0	72	
0	71	124	70	92	
82	0	62	84	86	
73	87	121	102	100	
0	89	0	0	0	

val (:, :, 7) =					
0	113	0	0	0	
0	0	0	0	114	
0	0	0	0	127	
0	122	0	87	117	
0	0	0	0	0	

图 2 改进的蚁群算法的教室排课表(教室编号 6 和 7)

val (:, :, 1) =					
15231	15252	14262	15231	32062	
15231	15252	15221	15252	15221	
15231	0	32062	0	15221	
15252	15231	15221	0	15231	
0	0	0	0	0	

val (:, :, 2) =					
0	15221	15221	15221	15231	
0	0	15231	0	32062	
15252	0	15221	15252	32062	
0	14262	0	32062	15221	
0	14262	0	0	0	

(a)基本蚁群算法的班级编号为 1 和 2 的排课表

(b)改进蚁群算法的班级编号为 1 和 2 的排课表

图 3 两种算法的班级课表对比

图 4 为两种算法随迭代次数的增加而生成的最优解的变化曲线图. 由图 4 也可以看出,随着迭代次数的增加,两种算法都收敛,其中改进的蚁群算法在迭代次数 30~40 之间的效果最好,这与所选数据的规模有关.

表 1 给出了两种算法最优值的对比数据(单位:座位数). 从图 4 可以看出,迭代次数在 35 左右时的效果最好,所以对比数据的迭代次数选为 35 次. 结果表明,改进的蚁群算法权值更低,得到了更优解.

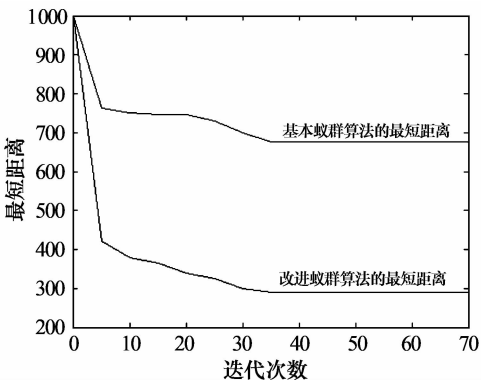


图 4 两种算法的迭代次数和最短距离的对比曲线

表 1 两种算法的最优值的对比

算法	不同迭代次数下的最优值					
	5	10	15	20	30	35
ACA	762	762	750	746	746	746
改进的 ACA	422	378	366	340	326	326

4 结论

本文实验结果表明,改进的蚁群算法能够满足硬性约束,未出现教室冲突或教室类型不匹配现象,达到了优化的目的.通过在边的权值优化方面加以改进,可保证学生人数少的课程尽量占用容量小的教室,也可解决大部分连续两节课之间的教室距离问题、单双周课程安排问题;本文的研究对教室资源有限的高校教室管理有一定的应用价值.本文的算法只针对教室管理进行了优化,课元的上课时间预先给定,没有考虑时间冲突,下一步的工作将主要解决排课问题的时间冲突,对蚁群算法信息素做进一步改进,以提高算法效率.

参考文献:

[1] 侯文静,马永杰,张燕,等.求解 TSP 的改进蚁群算法[J].计算机应用研究,2010,27(6):2087-2089.

[2] 崔旭,崔荣一,金小峰,等.基于时间资源的大学排课问题研究[J].延边大学学报:自然科学版,2006,32(4):256-258.

[3] 詹亚坤.基于模拟退火算法的高校排课系统研究

[D].东北师范大学,2012.

[4] 王凤,林杰.高校排课问题的图论模型及算法[J].计算机工程与应用,2009,45(27):240-242.

[5] 景雪琴,朱玉芳,杜栋,等.从排课表到教室调度表的设计与实现[J].计算机应用与软件,2004,21(2):123-125.

[6] 余斌,谢昕.基于减小教室流动性的排课算法研究[J].计算机时代,2004(2):22-24.

[7] 倪庆剑,邢汉承,张志政.蚁群算法及其应用研究进展[J].计算机应用与软件,2008,25(8):12-15.

[8] 池元成,蔡国飙.基于蚁群算法的多目标优化[J].计算机工程,2009,35(15):168-169.

[9] Thomas Stützle, Holger H Hoos. MAX-MIN ant system[J]. Future Generation Computer Systems, 2000,16(8):889-914.

[10] 何小虎.优化蚁群算法在排课中的应用策略[J].计算机与数字工程,2012,40(7):33-35.

[11] Broderick Crawford, Ricardo Soto. A Max-Min ant system algorithm to solve the software project scheduling problem[J]. Expert Systems with Applications, 2014(41):6634-6645.

[12] 吴小娟,吕强.新蚁群算法模型在大学课程时间表问题中的应用[J].计算机应用与软件,2009,26(6):80-83.

[13] Christian Blum. The Hyper-Cube framework for ant colony optimization[J]. IEEE Transactions on Systems Man, and Cybernetics Cybernetics-part B: Cybernetics, 2004,34(2):1161-1171.

[14] Li Zhiyong, Wang Yong, Dai Yun, et al. The cloud-based framework for ant colony optimization [C]//Proceedings of the 1st ACM/SIGEVO Summit on Genetic and Evolutionary Computation. Shanghai, 2009:279-286.