

基于约束满足和遗传算法的排课算法

许秀林¹, 胡克瑾²

(1. 南通职业大学电子工程系, 南通 226007; 2. 同济大学经济管理学院, 上海 200092)

摘 要: 针对高校排课过程中存在诸多资源约束因素的问题, 提出一种将遗传算法与约束满足算法相结合的排课算法, 由约束满足算法确定排课任务的优先次序, 遗传算法解决单个排课任务时间片分配的优化问题。算法中单个排课任务的局部最优解具有全局最优性。实验结果表明, 该算法能够改进算法性能, 提高排课效率。

关键词: 约束满足算法; 遗传算法; 排课问题

Course Schedule Algorithm Based on Constraint Satisfaction and Genetic Algorithm

XU Xiu-lin¹, HU Ke-jin²

(1. Department of Electronic Engineering, Nantong Vocation College, Nantong 226007;

2. College of Economic and Management, Tongji University, Shanghai 200092)

【Abstract】 Aiming at the factors of resource constraints that exist in the process of course schedule, this paper proposes an algorithm combining Genetic Algorithm(GA) and constraint satisfaction algorithm to solve course schedule problem. Course schedule tasks are sorted with constraint satisfaction algorithm, and a single course schedule task's timetable is allocated and optimized with GA. In this algorithm, the result of single course schedule task is global optimal. Experimental results show that this method is feasible to improve the performance and the efficiency.

【Key words】 constraint satisfaction algorithm; Genetic Algorithm(GA); course schedule problem

1 概述

排课问题是典型的多类资源组合优化问题。在 1975 年, 排课问题已被 S.Even 等人论证为 NP 完全问题。这类问题的求解只能找到较为合理、较为满意的解, 而并非是最优解。由于排课问题通常包含教学班级、教师、教室、时间等多个约束条件, 有些研究利用约束问题求解方法来求解排课问题^[1-2]。作为随机优化与搜索方法, 遗传算法(Genetic Algorithm, GA)通过对可行解进行选择、交叉、变异等遗传算子的作用使种群不断进化, 最后得到全局最优解或近似最优解。文献[3]对一般的安排问题使用了遗传算法; 文献[4]用具有控制约束的遗传算法解决大学课表安排问题; 文献[5]在课表初始化、课程安排和冲突处理时采用了时间片重叠法; 文献[6]提出了独特的排课问题的基因编码方式, 分别从不同角度应用遗传算法解决排课问题。由于用遗传算法实现大规模的排课时染色体太长, 因此基本解空间规模太大导致该问题的可行解比例较小。有些研究人员在应用遗传算法求解排课时, 往往削弱课程资源约束强度: 假设一个教师只上一门课, 或者假设教室资源足够多, 不考虑教室资源的约束, 或者不考虑奇数周学时课程之间的时间匹配等。

实际上, 在高校排课过程中存在诸多资源约束因素: 合班课, 一个教师任多门课, 一门课有多个任课教师, 课程类型(理论课、实验课、体育课等), 课程性质(公共课、专业课、选修课等), 奇数周课时, 不同起止周等。本文综合考虑了多种排课制约因素, 通过约束搜索算法选择自由度最小的排课任务, 再应用遗传算法对所选排课任务求最优解, 取得了较好的排课效率。

2 排课问题

排课是对课程教学活动的所需资源进行分配的过程。排课分为 2 个阶段: (1)由教学计划生成开课任务书, 为课程分配上课班级和任课教师, 通常由教研室主任手工完成。(2)由开课任务书生成教学课表, 为课程分配教室和时间。本文讨论的排课主要是第(2)阶段的排课, 且假定第(1)阶段的排课结果在相对合理的范围内。

排课问题实际上是一个由 Course、Class、Teacher、Classroom、Timetable 组成的五元组, 用 $(A_c, B_c, C_c, D_c, E_c)$ 表示。在五元组中, 除 A_c 以外, 其他元素都是集合变量, 分别为班级集合、教师集合、教室集合和时间片集合的子集。在给定 A_c 值后, 在排课第 1 阶段分配 B_c, C_c 的值, 排课第 2 阶段是在给定 A_c, B_c, C_c 3 个元素的值后求解 D_c, E_c 的值。

排课活动是将教师与学生在时间和空间上根据不同的约束条件进行排列组合, 由于课程资源的有限性, 因此不同课程之间在资源利用上必然存在相互制约。例如, 在同一时间同一班级不能上 2 门不同的课程, 在同一时间同一教师不能上 2 门不同的课程, 在同一时间同一教室不能上 2 门不同的课程。上述排课必须满足的条件称为硬约束, 数学描述如下:

设任意 2 门课程 $a_i, a_j, i \neq j$, 它们的解分别为 (b_i, c_i, d_i, e_i) 和 (b_j, c_j, d_j, e_j) , 则有:

若 $b_i \cap b_j \neq \emptyset$, 则 $e_i \cap e_j = \emptyset$;

若 $c_i \cap c_j \neq \emptyset$, 则 $e_i \cap e_j = \emptyset$;

作者简介: 许秀林(1965 -), 男, 副教授, 主研方向: 组合优化, 信息安全与审计; 胡克瑾, 教授、博士生导师

收稿日期: 2009-12-29 **E-mail:** xuxiulin@sina.com

若 $d_i \cap d_j \neq \emptyset$, 则 $e_i \cap e_j = \emptyset$;

在实际排课中, 还需根据教学规律要求安排教学活动, 比如, 一个班级的上课时间安排每天应尽量分布均匀, 一个教师的教学时间尽量分布均匀, 教室的利用率尽量分布均匀, 一门课程的教学时间尽量分布均匀。此外, 还有特殊课程对教师、教室、上课节次的特殊要求, 上述要求尽可能满足的条件称为软约束。

排课过程是对课程五元组进行求解的过程, 满足硬约束的排课方案是可行解, 满足软约束的排课方案则是优化解。由于优化解是多个资源交叉约束条件下相互妥协的结果, 因此本文取各种软约束条件满足系数的加权平均值作为优化解的判别标准, 加权平均值最高的解视为最优解。排课的目标就是尽可能找到这样的优化解以满足实际排课的需要。

3 排课算法

3.1 约束满足算法

约束满足问题(Constraint Satisfaction Problem, CSP)的研究是人工智能领域一个研究分支。通常, 一个 CSP 问题可定义为三元组 $P=(V, D, C)$, 其中, V 是 n 个变量($V_1, V_2, \dots, V_i, \dots, V_n$)的集合; D 是 n 个域 ($D_1, D_2, \dots, D_i, \dots, D_n$)的集合, 其中, D_i 是 V_i 可能的取值的集合; C 是变量 V 之间的约束关系集。

若把排课任务对应的五元组(A_c, B_c, C_c, D_c, E_c)定义为变量 V_i , 五元组中各元素的取值范围定义为域 D_i , 课程资源分配的硬约束定义为约束关系集 C , 则排课问题的求解可转化为一个约束满足问题的求解。

根据回溯搜索算法, 排课则是对排课任务变量逐一分配空间资源和时间资源, 当排课任务空间资源或时间资源分配失败时, 则退回到前一排课任务重新分配资源。为了减少回溯, 必须优先安排受约束最多的排课任务或者解的值域最小的排课任务。因此, 在排课之前, 必须先计算每个排课任务的资源约束度, 并由高到低排序。

算法主要步骤如下:

(1)录入班级、教师、教室等资源的基本信息, 并以教室为单位, 按校区、教室类型、教室容量排序建立教室队列。

(2)建立空的班级课程表、教师课程表、教室课程表。

(3)录入课程的基本信息, 并以课程为单位建立排课任务队列。

(4)计算队列中每个排课任务的资源约束度, 并将排课任务按资源约束度由高到低排序。

(5)取队列中资源约束度最高的排课任务。

(6)从教室队列中根据排课任务中课程教室资源要求符合条件且容量最小的教室。如果未能选择到符合条件的教室, 则回退到前一排课任务, 将前一排课任务重新放回队列, 转(5)。

(7)将排课任务中相关的班级、教师及教室在起止周范围内的课程表进行“或”运算, 生成排课任务可排时间的值域。

(8)如果排课任务的可排时间域为空, 则转(6)(从教室队列中另选符合条件的教室); 否则, 在可排时间域中分配上课时间, 并修改班级、教师、教室的课程表。

(9)从排课任务队列中删除本排课任务。

(10)如果排课任务队列为空, 则排课结束, 否则转(5)(取下一排课任务)。

算法步骤(8)时, 在班级、教师、教室课程表组合所生成空间时间域中, 分配排课任务的上课时间只是满足了硬约束,

确保所分配的时间不与课程资源已分配的时间发生冲突, 因此, 所求出的解只是可行解。

3.2 遗传算法

遗传算法是基于进化过程中的信息遗传机制和优胜劣汰的自然选择原则的搜索算法。它从一个种群开始, 利用选择、交叉、变异等遗传算子对种群进行不断进化, 最后得到全局最优解或近似最优解。

3.2.1 染色体编码

一条染色体是由若干个基因组成的基因组。本文染色体对应排课任务的时间分配表, 基因对应上课时间片。具体描述如下:

染色体结构

基因数	基因 1	基因 2	基因 3	...	基因 n
-----	------	------	------	-----	--------

基因结构

星期	单双周	时段	起始节次	终止节次
----	-----	----	------	------

一条染色体的二进制编码如下:

010 010|01|001|010 100|11|10|011|100

该染色体包含 2 个基因, 对应的时间片为星期二(单周)上午第 1 节、第 2 节和星期四下午第 3 节、第 4 节。其中, 单双周编码为: 单周——01, 双周——10, 不分单双周——11; 时段的编码: 上午——01, 下午——10, 晚上——11。

3.2.2 初始种群生成与适应度计算

本文应用遗传算法求单个排课任务的最优解, 初始种群来自于第 3 节排课算法的可行解, 初始种群产生后, 需对每条染色体进行适应度评价。排课的软约束根据不同学校的要求可以有多种, 与适应度函数定义相关的 7 个指标如下:

(1)上课时间均衡度: 一门课程不同时间片间隔长度的评价价值。(2)上课时段适合度: 课程性质与分配时段是否适合的评价价值。(3)上课节次均衡度: 一门课程相同时段相同节次的重复次数评价价值。(4)教室利用均衡度: 一门课程占用教室数的评价价值。(5)教师上课意愿满足度: 是否满足教师上课意愿的评价价值。(6)班级负担均衡度: 学生(班级)一天上课节次数的评价价值。(7)教师负担均衡度: 教师一天上课节次数的评价价值。

排课问题一般是多个目标的优化, 应用遗传算法求解多目标优化问题时, 适应度的计算必须按照多目标优化理论确定。当前比较典型的多目标优化算法有聚集函数法、向量评估进化算法、非支配排序法、ELECTRE 法等, 本文采用计算较为简单的目标聚集法。

3.2.3 遗传算子

遗传过程中的 3 种操作算子分别是选择、交叉和变异。

(1)选择操作。选择操作采用轮盘赌选择的方法。轮盘中不同比例的区域是根据个体适应度大小分配的, 适应度高的个体由于在轮盘中占据较大比例的区域, 因此被选中的概率较大, 得以生存的概率大。相反, 适应度较低的个体则被淘汰的概率较大。为了保持种群一定规模, 被淘汰的染色体用适应度高的染色体自我复制来补充。(2)交叉操作。交叉是遗传算法中最主要的一种操作。在交叉操作中, 首先产生一个随机数, 如果小于交叉率(默认为 0.7), 再产生一个随机数, 确定基因交换位置, 然后从种群中随机选择 2 条染色体, 交换相同位置的一对基因。(3)变异操作。在变异操作中, 对每条染色体的基因中表示星期、单双周和时段的二进制位在变异率(默认为 0.01)范围内进行 0 和 1 的变异, 即产生一个随机数, 如果小于变异率, 则进行变异操作。变异操作有可能使可行解变成不可行解, 所以, 对发生变异的染色体需进行

可行解的校验。

4 算法分析

本文基于约束满足算法和遗传算法按照排课任务约束度的高低,依次为每个排课任务分配合适的时间片,严格意义上说,仍属于局部最优解。由于在每个待排课程求解过程的适应度计算中优先考虑每个排课任务占用资源的均衡度,因此先排课程所占资源不一定优于后排课程。

定理 1 一门课程时间片间隔分配后,同一班级后一门课程的时间片间隔分布的概率大于连续分布的概率。

证明:设 n 门待排课程 A_1, A_2, \dots, A_n , 周课时为 1~6 课时,每个时间片均为 2 课时。课程划分为 3 个等价类: $S_1=\{A_i|A_i$ 的周课时为 1 或 2 $\}$, $S_2=\{A_i|A_i$ 的周课时为 3 或 4 $\}$, $S_3=\{A_i|A_i$ 的周课时为 5 或 6 $\}$ 。

(1) $A_k S_3, A_{k+1} S_3$, 因为 A_k 是间隔分配的,所以最佳方案是周一、周三、周五,简记为(1, 3, 5)(下同),根据耦合原则 A_{k+1} 的时间片组(t_1, t_2, t_3)的可选方案是(1, 2, 4), (2, 3, 4), (2, 4, 5), 令 $d=(t_3-t_2)+(t_2-t_1)=t_3-t_1$ 。显然,当 $d=2$ 时,时间片呈连续分布,当 $d>2$ 时,时间片呈间隔分布。在 A_{k+1} 的时间片分配方案中, $p(d>2)=66.7\%$, $p(d=2)=33.3\%$, 得 $p(d>2)>p(d=2)$ 。

(2) $A_k S_3, A_{k+1} S_2$, 因为 A_k 是间隔分配的,所以最佳方案是(1, 3, 5), 根据耦合原则, A_{k+1} 的时间片组(t_1, t_2)的可选方案是(2, 4)。令 $d=t_2-t_1$ 。显然,当 $d=1$ 时,时间片呈连续分布,当 $d=2$ 时,时间片呈间隔分布,在 A_{k+1} 的时间片分配可选方案中, $p(d=2)=100\%$, $p(d=1)=0\%$, 得 $p(d=2)>p(d=1)$ 。

(3) $A_k S_2, A_{k+1} S_2$, 因为 A_k 是间隔分配的,所以最佳方案是(1, 4)或(2, 5), 根据耦合原则, A_{k+1} 的时间片组(t_1, t_2)的可选方案是{(2, 3), (2, 5), (3, 5)}或{(1, 3), (1, 4), (3, 4)}。同样,令 $d=t_2-t_1$, 当 $d=1$ 时,时间片呈连续分布,当 $d=2$ 时,时间片呈间隔分布,在 A_{k+1} 的时间片分配可选方案中, $p(d=2)=66.7\%$, $p(d=1)=33.3\%$, 得 $p(d=2)>p(d=1)$ 。

(4) $A_{k+1} S_1$ 时由于只有一个时间片,因此不存在间隔分布问题。

综合(1)~(4), 定理得证。

定理 2 一门课程不同时间片、相同时间段节次交叉,同一班级后一门课程相应时间片、时间段的节次交叉的概率大于节次一致的概率。

证明:设 n 门待排课程 A_1, A_2, \dots, A_n , 周课时为 1~6 课时,每个时间片均为 2 课时。课程划分为 3 个等价类: $S_1=\{A_i|A_i$ 的周课时为 1 或 2 $\}$, $S_2=\{A_i|A_i$ 的周课时为 3 或 4 $\}$, $S_3=\{A_i|A_i$ 的周课时为 5 或 6 $\}$ 。

(1) $A_k S_3, A_{k+1} S_3$, 因为 A_k 的相同时段上课节次是交叉的,所以分配方案是(1-2, 3-4, 1-2), (3-4, 1-2, 3-4), (1-2, 1-2, 3-4), (3-4, 3-4, 1-2)。根据耦合原则, A_{k+1} 相同时间片、相同时段的可选方案依次是(3-4, 1-2, 3-4), (1-2, 3-4, 1-2), (3-4, 3-4, 1-2), (1-2, 1-2, 3-4)。显然,所有可选方案的节次均是交叉的,即节次完全一致的概率为 0。

(2) $A_k S_3, A_{k+1} S_2$, 因为 A_k 的相同时段上课节次是交叉的,所以分配方案是(1-2, 3-4, 1-2), (3-4, 1-2, 3-4), (1-2, 1-2, 3-4), (3-4, 3-4, 1-2)。根据耦合原则, A_{k+1} 相同时间片、相同时段的可选方案依次是(3-4, 1-2), (1-2, 3-4), (3-4, 3-4), (1-2, 1-2)。节次交叉和节次一致的概率相同,均为 50%。

(3) $A_k S_2, A_{k+1} S_2$, 因为 A_k 的相同时段上课节次是交叉的,所以分配方案是(1-2, 3-4), (3-4, 1-2)。根据耦合原则, A_{k+1} 相同时间片、相同时段的可选方案依次是(3-4, 1-2), (1-2,

3-4)。所有可选方案的节次均是交叉的,即节次一致的概率为 0。

(4) $A_{k+1} S_1$ 时由于只有一个时间片,因此不存在节次交叉的问题。

综合(1)~(4), 定理得证。

定理 3 一门课程时段分配后,与同一班级后一门课程时段相异概率大于时段相同概率。

证明:设 n 门待排课程 A_1, A_2, \dots, A_n , 根据课程性质划分为 4 个等价类: $S_1=\{A_i|A_i$ 是公共基础课或专业基础课 $\}$, $S_2=\{A_i|A_i$ 是专业课 $\}$, $S_3=\{A_i|A_i$ 是实验实训课 $\}$, $S_4=\{A_i|A_i$ 是专业选修课或公共选修课 $\}$ 。假设 4 类课程数量相同或相近。

根据教学规律,对 4 类课程的上课时段进行如下软约束:

S_1 类课程,时段为上午;

S_2 类课程,时段为上午或下午;

S_3 类课程,时段为下午;

S_4 类课程,时段为晚上。

同一班级不同课程时段交叉或一致的概率计算见表 1, 计算结果表明, $p(tp_k \neq tp_{k+1}) > p(tp_k = tp_{k+1})$ 。定理得证。

表 1 不同课程时段相异或相同概率

A_k	A_{k+1}	$p(tp_k=tp_{k+1})$	$p(tp_k \neq tp_{k+1})$
S_1	S_1	1.0	0.0
S_1	S_2	0.5	0.5
S_1	S_3	0.0	1.0
S_1	S_4	0.0	1.0
S_2	S_1	0.5	0.5
S_2	S_2	0.5	0.5
S_2	S_3	0.5	0.5
S_2	S_4	0.0	1.0
S_3	S_1	0.0	1.0
S_3	S_2	0.5	0.5
S_3	S_3	1.0	0.0
S_3	S_4	0.0	1.0
S_4	S_1	0.0	1.0
S_4	S_2	0.0	1.0
S_4	S_3	0.0	1.0
S_4	S_4	1.0	0.0
合计		5.5/16=34.48%	10.5/16=65.52%

本文中遗传算法所求的课程调度最优解,其最佳适应度是占用资源最佳均衡度综合加权计算结果。由定理 1~定理 3 可知,一门课程均衡占用资源会导致另一门课程均衡占用资源,即先排课程的最优解并不排斥后排课程获得最优解,最终呈现为全局最优解。

5 实验分析

5.1 实验过程

(1)实验对象:1 000 门课程,课程性质分为公共基础课、专业基础课、专业课、选修课,课程类型分为理论课、实验课和体育课,周课时为 2~6 课时,每次课时间为 2~4 节课。每门课程合班班级不超过 3 个,任课教师不超过 2 位。

(2)课程资源:250 个班级,400 位教师,150 个教室。教室分为普通教室、多媒体教室、实验室、体育馆或操场。教室容量为 50 人、80 人、120 人和 160 人 4 个等级。课程资源数据随机产生。

(3)实验任务:在软、硬约束条件下为课程分配合适的上课时间。

(4)强约束条件:在上文提到的约束条件中,将部分软约束转化为硬约束。如不允许一门课在同一天安排 2 次及以上,不允许一个班级同一天超过 8 节课,不允许一个教师同一天超过 6 节课。

(5)奇数课时的处理:除少数奇数课时课程要求 3 节课连上以外,一般奇数课时课程的单课时安排在单周或双周上课。为了合理排课、节约课程资源,本实验中首先分配课程的单

课时,优先与其他课程的单课时合用一个时间片(如课程 A 分配单周的星期一第 3、第 4 节,课程 B 尽量分配在双周的星期一的第 3、第 4 节)。剩下的按偶数课时课程分配。

5.2 实验环境

本实验在 CPU(Intel Celerol 1.5 GHz)、Windows XP、VC++ 6.0 环境下进行。

5.3 实验结果

本实验选择了是否约束满足(即是否对排课任务按约束度的大小进行排序)、遗传代数、种群规模等因素作为自变量,选择了排课任务失败数、适应度和排课占用时间等指标作为因变量,对相同的实验数据共实验 132 次,实验统计结果见表 2。为了分析算法的实验效果,本实验没有使用回溯,即排课任务的失败是未回溯情形下的失败。

表 2 实验数据统计结果

自变量		平均失败数	平均适应度	平均占用时间/ms	实验次数
是否约束满足	否	61.48	0.867 7	20 030	66
	是	36.09	0.859 8	21 011	66
遗传代数	0	115.17	0.832 6	22 17	12
	10	69.92	0.865 1	6 249	12
	20	56.92	0.866 8	9 790	12
	30	48.08	0.866 1	13 546	12
	40	43.67	0.867 6	17 523	12
	50	38.50	0.866 2	20 380	12
	60	35.42	0.866 9	24 064	12
	70	33.25	0.868 0	27 425	12
	80	34.00	0.868 0	31 241	12
	90	32.58	0.867 6	35 095	12
种群数	100	29.17	0.866 7	38 194	12
	10	69.82	0.852 5	6 906	22
	20	55.32	0.863 3	12 330	22
	30	46.82	0.865 0	18 362	22
	40	42.59	0.866 9	23 474	22
	50	39.36	0.867 3	28 123	22
	60	38.82	0.867 6	33 928	22

实验结果显示:如果对待排课程即排课任务同时采用约束满足算法和遗传算法,与仅采用遗传算法相比,平均适应度和平均占用时间并无明显差异,但平均失败数由 61.48 降低到 36.09,降幅为 41.3%。随着遗传代数、种群数的增加,采用遗传算法后平均失败数不断降低,但降幅趋于平稳。

132 次实验中,失败数最少的 5%(7 次)的实验结果见表 3,实验均同时采用了约束满足算法和遗传算法,遗传代数在 60~100 之间,种群数为 40~60,平均占用时间为 32 906 ms~

63 406 ms。

表 3 失败数最少的 5%(7 次)的实验结果

是否约束满足	遗传代数	种群数	失败数	适应度	占用时间/ms
是	70	50	14	0.866 085	37 609
是	60	50	12	0.865 641	32 906
是	100	50	11	0.866 350	55 156
是	90	60	11	0.870 630	57 531
是	70	60	9	0.866 493	45 125
是	100	60	9	0.862 631	63 406
是	100	40	8	0.863 495	44 641

6 结束语

本文提出了约束满足算法和遗传算法的组合应用算法,在未采用回溯的前提下,排课成功率最高达到 99.2%,能较好地解决排课问题。在本算法中,遗传算法只是用于单个排课任务求解问题的优化,经算法分析,并不影响解的全局优化。但与在所有排课任务整体域上应用遗传算法相比,染色体长度短,种群规模较小,而且在交叉运算后无须进行解的修正,有较高的排课效率。实验表明:随着遗传代数和种群数的增加,排课失败率趋于稳定,但平均占用时间却呈线性增加。本实验中当遗传代数为 60~70、种群数为 40~50 时,遗传算法的效益最佳。

参考文献

[1] 徐成刚, 易军凯, 肖 洋. 基于约束逻辑程序设计的排课算法研究[J]. 计算机工程与应用, 2006, 42(31): 197-199.

[2] 任克强, 赵光甫. 基于约束满足的高校排课问题研究[J]. 江西理工大学学报, 2006, 27(6): 70-72.

[3] Chu P C, Beasley J E. A Genetic Algorithm for the Generalized Assignment Problem[J]. European Journal of Operational Research, 1997, 24(1): 17-23.

[4] Safaai D, Sigeru O. Incorporating Constraint Propagation in Genetic Algorithm for University Timetable Planning[J]. Engineering Applications of Artificial Intelligence, 1999, 12(3): 241-253.

[5] 滕 姿, 邓辉文, 杨久俊. 基于遗传算法的排课系统的设计与实现[J]. 计算机应用, 2007, 27(12): 199-201.

[6] 祝勇仁, 曹焕亚. 应用遗传算法求解排课问题[J]. 计算机应用与软件, 2007, 24(12): 130-132.

编辑 张正兴

(上接第 280 页)

触发源	流量检测系统	平均处理总时间		33 min	
处理步骤					
序号	响应内容	工作方式	响应操作系统	时间/min	效果
1	分析拥堵状况	人工分析	流量分析系统	3	根据人经验而定
2	标注位置	人工查找	交通 GIS 系统	5	处理灵活
3	发布诱导	人工发布	诱导发布系统	10	较灵活, 易出错
4	信号灯调整	人工	信号控制系统	10	易出错
5	通知巡逻车	人工	GPS 巡逻定位系统	5	处理灵活

图 4 交通拥堵处理流程

4 结束语

本文对信息集成中的信息集合和信息变换关系进行了分析和定义,从而构建一种交通信息的集成服务单元——频道,实现消息类型的分类和消息集成代理的部署机制,提高了消息总线对集成的业务系统的适应性和融合功能。在交通信息

集成当中,信息的实时性要求较高,需要一种比较复杂的消息多播服务,如何根据不同交通信息的特点和传输要求而提供相应的传输 QoS 服务^[5],以满足交通信息实时性的要求,将是下一个研究重点。

参考文献

[1] 范玉顺. 企业集成系统技术[J]. 新技术新工艺, 2005, (7): 4-7.

[2] 徐 罡, 黄 涛, 刘绍华, 等. 分布应用集成核心技术研究综述[J]. 计算机学报, 2005, 28(4): 433-444.

[3] 闫晓芬, 郭银章. 基于 P/S 模式的分布对象中间件异步通信接口[J]. 计算机工程, 2009, 35(6): 125-126, 129.

[4] The Object Management Group. Notification Service Specification[EB/OL]. (2004-10-11). <http://www.omg.org/cgi-bin/doc?formal/2004-10-11.pdf>.

[5] 阳王东, 何焕民, 祝 青. 基于统计分析对交通信息集成的优化[J]. 交通运输系统工程与信息, 2007, 7(6): 39-44.

编辑 任吉慧