

Job-shop 调度求解的广义蚁群算法

张宏国¹, 宫雪²

(1. 哈尔滨理工大学 软件学院, 黑龙江 哈尔滨 150040;
2. 哈尔滨理工大学 计算机科学与技术学院, 黑龙江 哈尔滨 150080)

摘要:针对蚁群算法求解 Job-shop 调度问题,该算法考虑到利用析取图来描述工件加工关系给算法带来的复杂度,提出了一种广义蚁群算法。在综合考虑设备和工序关系约束的前提下,将信息素更新机制运用到求解 Job-shop 调度问题中,从而提高解的质量。为了提高搜索效率,根据蚁群算法的状态转移规则,对算法中参数的选择进行了详细的研究,设计了信息素更新策略。仿真实验结果表明,广义蚁群算法得到的解更接近最优解,该算法的收敛速度明显高于其他算法的收敛速度。

关键词:Job-shop 调度问题; 广义蚁群算法; 信息素更新机制

DOI: 10.15938/j.jhust.2017.01.016

中图分类号: TP391 文献标志码: A 文章编号: 1007-2683(2017)01-0091-05

A Generalized Ant Colony Algorithm for Job-shop Scheduling Problem

ZHANG Hong-guo¹, GONG Xue²

(1. School of Software, Harbin University of Science and Technology, Harbin 150040, China;
2. School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract: Aiming at the problem of ant colony algorithm for solving Job-shop scheduling problem. Considering the complexity of the algorithm that uses disjunctive graph to describe the relationship between workpiece processing. To solve the problem of optimal solution, a generalized ant colony algorithm is proposed. Under the premise of considering constrained relationship between equipment and process, the pheromone update mechanism is applied to solve Job-shop scheduling problem, so as to improve the quality of the solution. In order to improve the search efficiency, according to the state transition rules of ant colony algorithm, this paper makes a detailed study on the selection and improvement of the parameters in the algorithm, and designs the pheromone update strategy. Experimental results show that a generalized ant colony algorithm is more feasible and more effective. Compared with other algorithms in the literature, the results prove that the algorithm improves in computing the optimal solution and convergence speed.

Keywords: Job-shop scheduling problem; generalized ant colony algorithm; pheromone update mechanism

收稿日期: 2015-09-14

基金项目: 国家自然科学基金(51375128)。

作者简介: 张宏国(1967—),男,博士,教授,硕士研究生导师;

宫雪(1990—),女,硕士研究生。

0 引言

现代制造业的市场竞争日益激烈,生产调度是现代制造业面临的重要问题,它的解决有利于提高企业的生产效率与经济。Job-shop 调度问题^[1-3] (Job-shop scheduling problem, JSP) 是生产调度中最基本的组合优化问题,在满足工艺路线、交货期与资源利用等约束条件下,合理安排工艺加工时间、先后顺序及使用资源,可以减小生产周期,减少库存,获得产品时间或成本的优化。因此 Job-shop 调度问题的优化求解具有重要的理论价值和实际意义,是学术界与工程界共同关注的热点。

目前求解 Job-shop 调度问题的主要方法是建立智能优化算法,以禁忌搜索^[4]、模拟退火^[5]、遗传算法^[6]为代表。蚁群算法^[7]具有正反馈、较强的鲁棒性、全局性、普遍性、优良的分式并行计算机机制,易于与其他方法相结合等诸多优点。近年来,很多学者将蚁群算法逐渐地应用于 Job-shop 调度求解中。文^[8]提出了一种自适应蚁群算法(AACA),自适应地初始化信息素并限定其大小范围以得到最优解。文^[9]提出了最大最小蚁群算法,对信息素进行限制防止算法陷入局部最优。文^[10]提出了混合算法,将蚁群算法和遗传算法相结合的 ACSGA 算法,充分利用它们的特性。

但上述文献都是采用传统蚁群算法中析取图的方式对工件加工关系进行描述的。虽然析取图是用来描述路径关系比较好的视图形式,但是车间生产中各工件之间的加工关系不同于旅行商问题(TSP)^[11-13]中的路径选择。特别是对于加工规模较大的 Job-shop 调度问题,每增加一个机器,析取图就会增加条边,则形成的析取图^[14]结构会十分复杂,使得算法的复杂度增加,收敛速度变慢,影响解的质量。

本文提出一种广义蚁群算法,将其用于 Job-shop 调度求解中。该算法利用一次调度就是蚂蚁的一次爬行的思想,摆脱传统的析取图方式,把信息素^[15]更新机制应用到 Job-shop 调度问题中,这样能有效简化算法复杂度,提高算法的收敛速度,优化解的质量。将信息素机制合理的应用在每一道工序上,在满足约束条件和解的合法性的前提下,越早完成加工的工序和同一工件优先进行加工的工序上残留的信息素浓度越高。

1 Job-shop 调度问题分析

1.1 问题描述

n 个工件在 m 台机器上加工,每个工件有 m 道工序,加工时间已知,要求总完工时间最小。针对 Job-shop 调度问题的具体约束条件如下:

- 1) 同一时刻同一台机器只能加工一个工序;
- 2) 前一道工序加工完毕后才能加工下一道工序;
- 3) 每个工序在可用机器上的加工时间不同且已经确定;
- 4) 允许工件在工序间等待,也允许机器在工件未到达前闲置;
- 5) 工序一旦开始,不允许中断。

1.2 问题建模

关于 Job-shop 调度问题,作如下定义: 1) n : 工件数目;

2) m : 机器数目;

3) 工件集 $J = \{J_1, J_2, \dots, J_n\}$, $1 \leq i \leq n$;

4) 机器集 $M = \{M_1, M_2, \dots, M_m\}$, $1 \leq k \leq m$;

5) 工序集 $O_i = \{O_{i,1}, O_{i,2}, \dots, O_{i,m}\}$, 其中 $O_{i,j}$ 表示工件 i 的第 j 道工序;

6) $M_{i,j}$ 表示 $O_{i,j}$ 占用的机器, $M_{i,j} \in M$;

7) $T_{i,j}$ 表示第 i 个工件第 j 道工序的加工时间;

8) $S_{i,j}$ 表示第 i 个工件第 j 道工序的开始时间;

9) $C_{i,j}$ 表示第 i 个工件第 j 道工序的完工时间, $C_{i,j} = S_{i,j} + T_{i,j}$ 。

因为工件完工要求所有工序加工完毕,并且每道工序开始加工时,它的紧前工序必须加工完毕,所以工件加工完毕的时间为各设备完工的最大时间值。因此在给定约束条件的前提下,调度的最短时间值是所求的结果,其数学描述为:

$$F = \min_{1 \leq i \leq n} \max_{1 \leq j \leq m} C_{i,j} \quad (1)$$

$$\text{s. t. } S_{i,j+1} \geq S_{i,j} + T_{i,j} \quad (2)$$

$$S_{i,j} \geq S_{p,q} + T_{p,q} \text{ 或 } S_{p,q} \geq S_{i,j} + T_{i,j}, M_{i,j} = M_{p,q} \quad (3)$$

在上面式子中,式(1)表示目标函数,调度的整体完工时间最短。式(2)表示对于同一工件 i 来说,工序 $j+1$ 的开始加工时间,一定会大于其紧前工序 j 的开始加工时间与连续加工时间之和。式(3)表示对于同一设备来说,同一设备在同一时刻只能加工一道工序。

2 Job-shop 调度求解的广义蚁群算法

2.1 广义蚁群算法的总体思想

蚁群算法是一种基于反馈机制和并行机制的模拟进化算法,在大规模的调度问题中有其固有的优势,但也容易出现搜索时间过长导致局部最优解^[14]。传统的蚁群算法用析取图来求解 JSP 问题,析取图被描述为一个有向图,更直观的展现便于分析。但对于规模大的 JSP 问题会加剧蚁群算法的缺陷,因需要优化的问题规模变大,那么得到相应的解就会增加迭代次数,增加了相应的计算时间,进而影响解的质量和算法的收敛速度^[17-19]。

广义蚁群算法利用每一次的调度需要蚂蚁爬行一次的思想,寻找的最优路径就是蚂蚁爬行所走的最短路径。运用广义蚁群算法求解时,一只蚂蚁从一个工件的第一道工序开始爬行,根据转移状态规则^[20]对可调度的工序进行选择。将选择的工序标记为已操作工序,计算概率继续选择接下来要调度的工序,直到每台机器未调度的工序集为空为止,即完成一次调度。

每次调度之后更新信息素,信息素的改变直接影响工序转移概率的计算结果,因此信息素的更新要与工序的优先级紧密联系。针对蚁群算法的特点,对信息素更新机制进行改进,以便更快找到工序在各机器上加工的一组优先序列。

2.2 信息素更新机制

1) 状态转移规则

蚂蚁根据信息素的浓度和工序间的启发式信息,在其可选尚未调度的集合中选择接下来的目标工序。选择工序的时候需要判断各工序在机器上的加工概率,状态转移概率公式如下:

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{j \in AS} [\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta} \quad (4)$$

从式(4)可看出,状态转移概率主要根据 t 时刻工序 i 转移到工序 j 上的信息素浓度 $\tau_{ij}(t)$ 以及工序 i 到工序 j 的启发式信息 $\eta_{ij}(t)$ 。其中, α 和 β 是两个参数,分别表示信息素的影响力和启发式信息的相对重要性, AS 表示可选尚未调度的集合。

2) 信息素计算公式

在求解 Job-shop 调度问题时,蚂蚁在遍历过程中留下信息素,路径上的信息素浓度随着蚂蚁每次遍历进行更新。信息素浓度的强弱引导着后面的蚂蚁对工序的选择,某个路径上蚂蚁遍历的次数越多,

信息素浓度越大,后来者选择该路径遍历的概率就越大。当蚂蚁所走的路径越短即调度结果越优时,该路径的信息素浓度越大。

信息素更新机制是工件加工排序中的重中之重,因此本文对信息素的更新公式进行优化,会着重突出信息素的作用和影响,削弱启发式信息的重要性。信息素浓度的变化受工序节点开始时间以及局部目标函数完成时间的影响,根据这一原则区分工序的优先级,优先安排信息素浓度大的工序。

在一次遍历工序中,蚂蚁遍历工序的完成时间越早,信息素浓度越大。其次,开始时间越早的工序,信息素浓度越大,越优先安排。在传统的信息素公式中,由于开始时间和完成时间都与信息素的大小有直接影响,所以在信息素浓度相同的情况下,确定不了工序的优先顺序。因此对计算信息素浓度的公式加以完善,这样会更快得到较优的加工序列。计算信息素的公式如下:

$$\tau_{ij} = \frac{\gamma}{S_{i,j} + z \max_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} C_{i,j}} \quad (5)$$

通过式(5)对信息素浓度的赋值,保证了越早完成加工的工件对应的工序上信息素浓度越大,并且同一工件的前道工序获得的信息素浓度比后道工序获得的信息素浓度大。

3) 公式参数的分析

为了避免信息素过小而影响解的计算,故在式(5)的分母上加上系数 γ 以保证解的合理性公式, γ 为信息素放大系数,取经验值 $\gamma = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} T_{ij}$ 。同时为

了保证工序的选择不会因为开始时间而淹没完工时间的影响,故在式(5)的分子中的完成时间上乘上一个常数 z ,适当扩大完工时间对信息素的影响, z 为调整信息素浓度系数, z 取经验值。在算法的具体实行时通常取 $z = n$,这样通过对公式参数的设置更有效的保证了对调度周期的优化。

4) 信息素更新方式

蚂蚁完成一次调度后,更新当前路径上的各工序信息素浓度,更新方式如下:

$$\Delta \tau_{ij}^k(t, t+1) = \sum_{k=1}^m \Delta \tau_{ij}^k(t, t+1) \quad (6)$$

$$\tau_{ij}(t+1) = (1 - \rho) \times \tau_{ij}(t) + \Delta \tau_{ij}^k(t, t+1) \quad (7)$$

式(6)中, $\Delta \tau_{ij}^k(t, t+1)$ 表示第 k 只蚂蚁在 $(t, t+1)$ 时刻从工序 i 移动到工序 j 这一过程的信息素增量。式(7)表示信息素的更新方式,其中 ρ 表示信息素挥发系数,信息素会随着时间不断挥发, $\rho \in (0, 1)$ 。

2.3 算法步骤

符号表示: AS 为可调度的工序集合, CS 为完成调度的工序集合, GS 为未调度的工序集合, IS 为正在调度的工序集合。

步骤 1: 初始化 $AS = \{O_{ij} | i = 1 \cdots n\}$, $CS = \phi$, $GS = \{O_{ij} | i = 1 \cdots n, j = 1 \cdots m\}$, $IS = \phi$ 。

步骤 2: 初始化蚂蚁数目 n , 参数 α, β, ρ , 机器 M_{ij} 的可用时刻为 MT_{mij} 以及当前时刻 ST_{ij} 。

步骤 3: 当调度解多次不再优化时, 算法停止, 输出结果。否则重置并转到步骤 1。

步骤 4: 一共有 $k = 1 \sim n$ 只蚂蚁, 当初始时刻 $t = 0$ 时, 分别放在各工件的第一道工序 $O_{,1}$ 上, 更新当前时刻 $ST_{ij} = ST_{k,1}$, 各机器可用时刻 $MT_{mij} = T_{k,1}$, 更新集合 $AS = AS - \{O_{k,1}\}$, $GS = GS - \{O_{k,1}\}$, $IS = IS + \{O_{k,1}\}$ 。

步骤 5: 对于 m 个机器, 如果机器可用时刻 $MT_{mij} \leq T_{k,1}$, 则机器空闲, 进行下一个工序的选择。如果 AS 中只有一个可调度工序, 则直接将此工序放入 IS ; 如果存在多个可调度工序, 根据式 (7) 计算转移概率, 按轮盘赌的方式选择接下来要加工的工序 O_{ij} ; 如果不存在直接跳过, 转步骤 7。

步骤 6: 当蚂蚁移动到所选工序 O_{ij} 时, $AS = AS - \{O_{ij}\}$, $GS = GS - \{O_{ij}\}$, $IS = IS + \{O_{ij}\}$ 。 O_{ij} 完成调度之后, $CS = CS + \{O_{ij}\}$, 当前时刻 $ST_{ij} = ST_{k,1} + T_{ij}$, 将接下来要调度的工序 $O_{i,j+1}$ 放入 AS 中。

步骤 7: 判断 GS 是否为空。如果 $GS = \phi$, 说明蚂蚁遍历所有工序, 完成了一次调度, 执行步骤 8; 否则, 转步骤 4。

步骤 8: 蚂蚁完成一次调度后, 记下调度序列, 对该蚂蚁当前路径各工序信息素浓度进行更新。

步骤 9: 比较前后两只蚂蚁爬行所用的完成时间, 选择完成时间短的。每次调度后都选出较优的调度序列, 对目标函数值进行更新并记录调度序列。

步骤 10: 直到所有蚂蚁都完成寻优, 得到最优的完工时间及调度序列。

3 仿真实验

3.1 实例验证

根据上述理论, 针对所描述的广义蚁群算法进行模拟仿真实验, 并做出分析与评价, 验证算法的可行性。应用实例是 6 个待加工工件 ($J_1, J_2, \dots, J_6$) 在 6 台机器 ($M_1, M_2, \dots, M_6$) 上加的典型 Job-shop 调度问题。

1) 参数确定: $\alpha = 2, \beta = 3, \rho = 0.5$;

2) 实例中加工机器对应的加工工序如表 1 所示, 加工时间如表 2 所示。

表 1 加工机器对应的加工工序

机器	工件					
	J_1	J_2	J_3	J_4	J_5	J_6
M_1	2	5	4	2	5	4
M_2	3	1	5	1	2	1
M_3	1	2	1	3	1	6
M_4	4	6	2	4	6	2
M_5	6	3	6	5	3	5
M_6	5	4	3	6	4	3

表 2 工序在各台机器上的加工时间

机器	工件					
	J_1	J_2	J_3	J_4	J_5	J_6
M_1	3	10	9	5	3	10
M_2	6	8	1	5	3	3
M_3	1	5	5	5	9	1
M_4	7	4	4	3	1	3
M_5	6	10	7	8	5	4
M_6	3	10	8	9	4	9

表 2 的加工时间与表 1 中的加工工序对应着, 根据广义蚁群算法得到最优调度序列, 用 6 行 6 列的 S 矩阵表示。矩阵的每一行代表一个加工机器, 矩阵的每一列表示工序的由开始到最后的加工顺序。例如一行一列的 (1, 2) 代表在 M_1 上首先加工的是 O_{12} , 第二行第二列的 (4, 1) 代表在 M_2 上第二个加工的是 O_{41} , 依此类推。

$$S = \begin{bmatrix} (1,2) & (4,2) & (3,4) & (6,4) & (2,5) & (5,5) \\ (2,1) & (4,1) & (6,1) & (1,3) & (5,2) & (3,5) \\ (3,1) & (1,1) & (2,2) & (5,1) & (4,3) & (6,6) \\ (3,2) & (6,2) & (4,4) & (1,4) & (2,6) & (5,6) \\ (2,3) & (5,3) & (4,5) & (6,5) & (3,6) & (1,6) \\ (3,3) & (5,3) & (2,4) & (1,5) & (5,4) & (4,6) \end{bmatrix}$$

为了更直观的看出机器的加工状态, 根据 S 矩阵绘制甘特图如图 1 所示。图中每个实线方块代表一个加工工序 O_{ij} , 方块的长度代表此工序在此台机器上的加工时间, 方块外的空白之处代表机器的空闲时间。从甘特图可看出完成时间是 55, 也是已知的最优解, 该算例证明了广义蚁群算法的可行性和有效性。

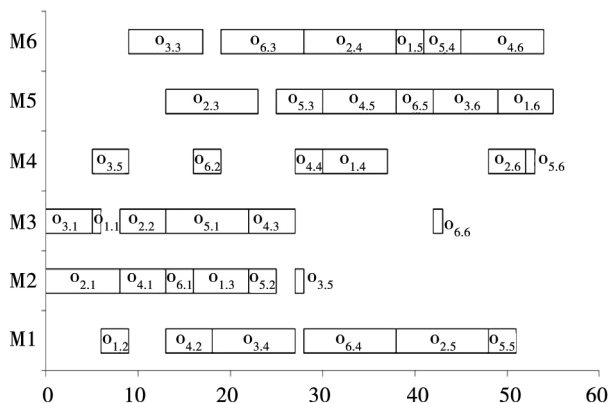


图 1 广义蚁群算法的甘特图

3.2 算法评价

用广义蚁群算法求解 Job-shop 调度问题的 5 个经典算例,采用 Matlab 软件进行编程仿真。仿真结果与其他算法得到的结果进行对比,其他算法的结果由文献给出,如表 3 所示。由表可知,与遗传算法、传统蚁群算法相比,广义蚁群算法的结果更接近最优解,可见达到了优化的目的。

表 3 算法结果对比

序号	JSP 实例	工件数 × 机器数	已知最优解	遗传算法最优解	传统蚁群算法最优解	本文算法最优解
1	FT03	3 × 3	12	12	12	12
2	FT06	6 × 6	55	55	55	55
3	FT20	20 × 5	1165	1178	1175	1169
4	LA16	10 × 10	945	979	950	945
5	LA21	15 × 10	1046	1097	1222	1054

收敛曲线对比: 图 2 中纵坐标表示目标函数值,横坐标表示循环次数,黑色曲线是本文提出的广义蚁群算法的收敛曲线,红色曲线是基本蚁群算法的收敛曲线,蓝色曲线是遗传算法的收敛曲线。从图中可以看出本文的算法更优,不仅收敛的速度快而且得到的目标函数值更贴近最优解。

4 结 论

根据蚁群算法求解 Job-shop 调度问题,考虑到用析取图求解规模较大的调度问题时会非常复杂,而且缺少对蚁群算法并行特性的体现。本文从算法的调度方式、信息素的更新机制方面进行改进,提出了广义蚁群算法。通过应用实例进行试验,检验了该算法对实际问题的有效性,与传统蚁群算法,遗传

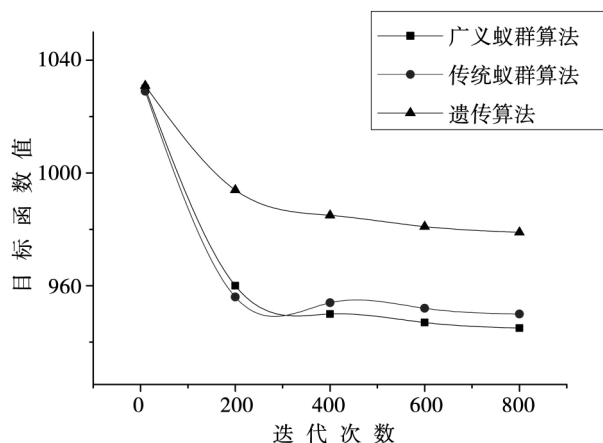


图 2 LA16 问题的收敛曲线对比图

算法进行对比,结果表明本文提出的广义蚁群算法获得的解更接近最优解,而且算法的收敛速度更快,这说明本文求解 Job-shop 调度问题的方法更有效。

参 考 文 献:

- [1] 彭建刚,刘明周,张铭鑫,等. 多目标柔性作业车间调度算法研究综述 [J]. 中国机械工程, 2014, 25(23): 3244-3254.
- [2] CHAUDHRY I A. Job Shop Scheduling Problem With Alternative Machines Using Genetic Algorithms [J]. Journal of Central South University, 2012, 5(19): 1322-1333.
- [3] 范营营,刘海彬,赵博选,等. 基于多任务的车间生产调度算法的研究 [J]. 机械设计与制造工程, 2014(3): 15-20.
- [4] 戚峰,俞晶菁,黄召杰. 基于禁忌搜索算法求解车间作业调度问题 [J]. 兰州交通大学学报, 2011, 30(3): 79-84.
- [5] 李俊,刘志雄,张煜,等. 柔性作业车间调度优化的改进模拟退火算法 [J]. 武汉科技大学学报, 2015, 38(2): 111-115.
- [6] 柳青红,袁逸萍,李晓娟,等. 基于改进遗传算法交叉算子的作业车间调度 [J]. 组合机床与自动化加工技术, 2014(12): 150-153.
- [7] 王硕,顾幸生. 基于改进蚁群算法的作业车间调度 [J]. 青岛科技大学学报, 2012, 33(5): 489-494.
- [8] 梁德赛. 基于自适应蚁群算法的 JSP 问题的仿真研究 [J]. 计算机仿真, 2012, 29(6): 228-239.
- [9] 张宏国,王彦俊. Pareto 多目标模糊调度算法研究 [J]. 哈尔滨理工大学学报, 2013(5): 47-51.
- [10] CANSIN Turguner, OZGUR Koray Sahingoz. Solving Job Shop Scheduling Problem With Ant Colony Optimization [J]. Computational Intelligence and Informatics (CINTI), 2014 IEEE 15th International Symposium on, 2014: 385-389.
- [11] YI Minjie. The Research of path-oriented test data generation based on a mixed ant colony system algorithm and genetic algorithm [J]. Jiangsu Automation Research Institute, 2012: 2161-2164.
- [12] LI Xiaojiang, LIAO Jiabin, MIN Cai. Ant Colony Algorithm for Large Scale TSP [J]. Electrical and Control Engineering (ICECE), 2011: 573-576.

(下转第 102 页)

- [3] 崔永强,高晓丁,贺素馨. 基于 FPGA 分布式算法的滤波器设计[J]. 现代电子技术,2010,33(16):117-119.
- [4] 刘砚涛,刘玉蓓,尹伟. LC 滤波器设计方法介绍及其仿真特性比较[J]. 电子测量技术,2010,33(5):17-18.
- [5] HERMANOWICZ E,JOHANSSON H. A Complex Variable Fractional-Delay FIR filter Structure[J]. IEEE Transactions on Circuits and Systems II,2011,54(9):785-789.
- [6] 王竹萍,仲顺安,王晓清. 低失调开关电容滤波器的设计[J]. 北京理工大学学报,2010,30(10):1209-1201.
- [7] 杨军,朱江,翟海涛. 一种基于 FPGA 的高速 FIR 滤波器的设计[J]. 信息化研究,2009,35(4):106-111.
- [8] 聂伟,杨胜姚,王岩. 基于 FPGA 的通用 FIR 滤波器设计[J]. 实验技术与管理,2010,27(10):69-72.
- [9] 王建元,于洪洋,孙澄宇,等. 基于特定消谐技术的单相有源滤波器设计[J]. 继电器,2008,36(8):46-48.
- [10] 刘建成,邹应全,徐伟. 基于 FPGA 的 FIR 滤波器设计与仿真[J]. 南京信息工程大学学报:自然科学版,2010,2(5):400-404.
- [11] TAKAHASHI N., K. Suyama. Design of CSD Coefficient FIR Filters Based on Branch and Bound Method[J]. Communication and Information Technologies,2010,27(3):575-578.

(编辑:关毅)

(上接第 95 页)

- [3] LING Lin, DONG Huailin, WU Qingfeng. Research and Improvement of Ant Colony Algorithm Based on TSP[J]. //Information Technology and Artificial Intelligence Conference, 2011: 453-456.
- [4] 夏欣. 批处理机调度问题的蚁群算法求解[J]. 计算机科学,2013,40(5):247-250.
- [5] 张洁,张朋,刘国宝. 基于两阶段蚁群算法的带非等效并行机的作业车间调度[J]. 机械工程学报,2013,49(6):136-144.
- [6] 景波,刘莹,黄兵. 基于遗传算法的 Job Shop 调度问题研究[J]. 计算机应用研究,2013,30(3):688-691.
- [7] 曾强,沈玲,任华,等. 双资源批量生产柔性作业车间多目标调度方法[J]. 计算机工程与应用,2015,50(1):250-256.
- [8] 刘刚,王瑛,张发. 基于关键路径求解作业车间调度问题的收敛性分析[J]. 计算机集成制造,2014,20(5):1078-1087.
- [9] 唐毓毅,朱怡安,黄姝娟,等. 一种有约束关系的实时周期任务调度算法研究[J]. 计算机技术与发展,2013,23(7):1-5.
- [10] 杨小龙,张洁. 基于三阶段改进蚁群算法的多工艺路线柔性作业车间调度[J]. 制造业自动化,2014,37(4):51-55.

(编辑:关毅)