

6-8-2020

Hybrid Quantum-behaved Particle Swarm Optimization Algorithm for Solving Mould Job Shop Scheduling Problem

Zhou Kai

Institute of Electrical Automation, Jiangnan University, Wuxi 214122, China;

Wang Yan

Institute of Electrical Automation, Jiangnan University, Wuxi 214122, China;

Zhicheng Ji

Institute of Electrical Automation, Jiangnan University, Wuxi 214122, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Hybrid Quantum-behaved Particle Swarm Optimization Algorithm for Solving Mould Job Shop Scheduling Problem

Abstract

Abstract: Quantum-behaved particle swarm optimization has better performance compared with particle swarm optimization, but it still has the problem of getting trapped into local optimum with premature convergence. *According to the above problem, a hybrid algorithm included quantum-behaved particle swarm optimization and bat algorithm was proposed. On the one hand, the random walk strategy of bat algorithm was used to avoid getting into local optimum, on the other hand, the speed changing of bats' sound was learned to transform the factor of quantum-behaved particle swarm optimization.* The proposed algorithm was tested on five benchmark functions and a mould job shop scheduling example, compared with PSO (Particle Swarm Optimization) and QPSO (Quantum-Behaved Particle Swarm Optimization). The simulated experimental results indicate the validity and superiority of the hybrid algorithm.

Keywords

bat algorithm, quantum-behaved particle swarm optimization, scheduling optimization, mould job shop

Recommended Citation

Zhou Kai, Wang Yan, Ji Zhicheng. Hybrid Quantum-behaved Particle Swarm Optimization Algorithm for Solving Mould Job Shop Scheduling Problem[J]. Journal of System Simulation, 2016, 28(6): 1247-1255.

混合量子粒子群算法求解模具车间调度问题

周恺, 王艳, 纪志成

(江南大学电气自动化研究所, 江苏 无锡 214122)

摘要: 虽然相比较粒子群算法而言, 量子粒子群优化算法有着更好的性能, 但其仍然面临因过早收敛而陷入局部最优的问题。因此尝试将量子粒子群算法与蝙蝠算法相混合, 一方面利用蝙蝠算法中的随机游走策略来避免过早地陷入局部最优, 另一方面学习蝙蝠算法中发声速度的变化方式来改变量子粒子群算法中的因子。将所提算法与粒子群优化算法和量子粒子群优化算法经过 5 个标准测试函数和一个实际模具车间的调度模型的仿真验证, 并与粒子群算法和量子粒子群算法进行对比, 仿真结果表明了该算法在求解连续型问题和离散型问题的有效性和优越性。

关键词: 蝙蝠算法; 量子粒子群算法; 调度优化; 模具车间

中图分类号: TP391.9

文献标识码: A

文章编号: 1004-731X (2016) 06-15-1247-09

Hybrid Quantum-behaved Particle Swarm Optimization Algorithm for Solving Mould Job Shop Scheduling Problem

Zhou Kai, Wang Yan, Ji Zhicheng

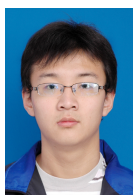
(Institute of Electrical Automation, Jiangnan University, Wuxi 214122, China)

Abstract: Quantum-behaved particle swarm optimization has better performance compared with particle swarm optimization, but it still has the problem of getting trapped into local optimum with premature convergence. According to the above problem, a hybrid algorithm included quantum-behaved particle swarm optimization and bat algorithm was proposed. On the one hand, the random walk strategy of bat algorithm was used to avoid getting into local optimum, on the other hand, the speed changing of bats' sound was learned to transform the factor of quantum-behaved particle swarm optimization. The proposed algorithm was tested on five benchmark functions and a mould job shop scheduling example, compared with PSO (Particle Swarm Optimization) and QPSO (Quantum-Behaved Particle Swarm Optimization). The simulated experimental results indicate the validity and superiority of the hybrid algorithm.

Keywords: bat algorithm; quantum-behaved particle swarm optimization; scheduling optimization; mould job shop

引言

车间调度问题是近些年的热点问题, 得到大家



收稿日期: 2015-12-09 修回日期: 2015-12-07;
基金项目: 国家自然科学基金(61572238), 国家 863 计划(2014AA041505);
作者简介: 周恺(1989-), 男, 安徽铜陵, 硕士, 研究方向为控制理论与控制工程; 王艳(1978-), 女, 江苏盐城, 博士, 教授, 硕导, 研究领域为网络化控制系统、制造系统能效优化控制。

的广泛关注。其中柔性作业车间调度问题(Flexible Job-shop Scheduling Problem, FJSP)更是调度问题中比较复杂的一种类型, 允许工序可以用不同的机器加工, 允许工序在一组机器中的任何一台机器上进行加工^[1]。显然, FJSP 更贴近于实际生产制造过程, 与此同时, 多目标生产调度优化更是企业决策层关心的重点问题。

<http://www.china-simulation.com>

• 1247 •

在研究的早期,往往是通过数学规划方法来解决一些小型的 FJSP 问题,但随着生产规模的扩大,许多群智能算法如粒子群算法(Particle Swarm Optimization, PSO)^[2]、蚁群算法^[3]、蜂群算法^[4-5]、萤火虫算法^[6]等被用来求解大规模的 FJSP 问题。其中 PSO 算法以其收敛速度快,收敛精度高等优势得到了众多学者广泛使用,但其在求解 FJSP 问题时,常常出现早熟状况。

2004 年, Sun 等人^[7]基于量子力学提出了量子粒子群算法(Quantum-behaved particle swarm optimization, QPSO),不仅可以有效跳出局部最优,还提高了算法收敛的精度和速度。因此在利用 QPSO 求解 FJSP 问题时,相对 PSO 算法可以得到较好的调度解。

然而 QPSO 算法在陷入局部最优的问题上依然有改进的空间,田娜等^[8]提出了在 QPSO 算法中对每个粒子施加扰动来增加种群的多样性,有效避免算法早熟;方伟等^[9]提出在一定概率对粒子的每一维微分进化来增加随机性,增强了算法的搜索性能;朱红求^[10]提出将量子粒子群算法的全局搜索与 Powell 的局部搜索相混合,有效提高算法的求解速度与精度。

本文根据蝙蝠算法(Bat Algorithm, BA)^[11]中发射声波速度的变化方式以及其为了增强局部搜索而进行的随机游走策略,与 QPSO 算法相混合,提出蝙蝠量子粒子群算法(Bat-Quantum-behaved particle swarm optimization, BAT-QPSO),该算法能够有效跳出局部最优,得到更好的收敛精度与收敛速度。

在利用 BAT-QPSO 算法求解标准测试函数来验证其有效性后,再用该算法与 PSO 算法及 QPSO 算法对比求解实际模具车间的多目标 FJSP 问题,并采取常用的 Pareto 优化策略^[12],能够针对多目标的调度类型得到一组互不支配的折衷解,经过最后求解结果的对比发现, BAT-QPSO 算法可以得到更优的调度方案。

1 模具车间柔性作业调度问题研究

1.1 模具车间生产流程简介

某模具车间的平面图如图 1 所示。

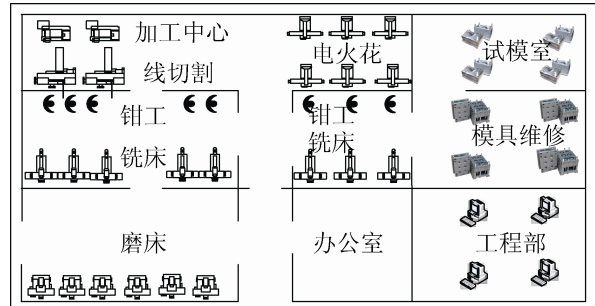


图 1 某模具车间平面图

生产流程涉及如下几种机器:计算机数字控制机床(Computerized Numerical Control Machine, CNC),磨床,电火花镜面机,电火花普通机,线切割机器,不同的机器分别有不同的工序在其上面加工。例如 CNC 上有 CNC 开框、CNC 粗加工、CNC 精加工、CNC 铜公等多道工序加工。同时,该车间所有模具的加工工艺基本由以下路线设置,如图 2 所示。

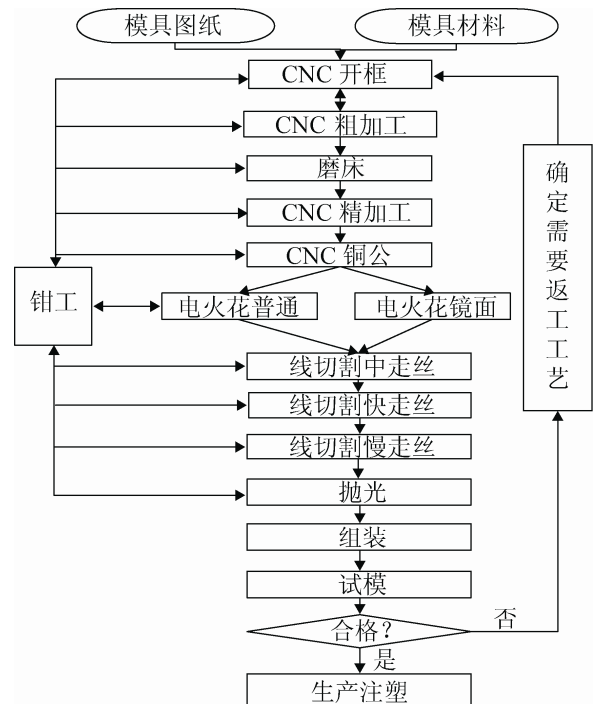


图 2 加工工艺路线图

从根据图纸的开始到最后生产注塑, 其中 CNC、线切割、电火花为机加工部分, 而钳工、磨床、抛光为非机加工部分, 外协(热处理)则是将热处理这道工序进行委外加工, 试模、返工则是在实际车间加工制造中常常出现的问题。

由于在实际模具车间中, 存在某些工件的某种工序只能在一种机器上加工, 因此该模具车间的调度是一个部分柔性作业调度问题。

1.2 模具车间生产调度模型

上述精简的模具车间调度问题可以描述如下^[13]:

假设有 n 个不同的工件在 m 台机器上进行加工的情况:

- (1) 每个工件有一道工序或者是多道工序;
- (2) 同一个工件按一定的顺序做不同的工序;
- (3) 每道工序能够选择在一台或者多台机器上进行加工, 每个工序所选的加工机器不一样其对应的加工时间也不尽相同。

约束条件如下:

- (1) 同一机器在同一时刻只能加工一道工序;
- (2) 一道工序在同一时刻内只能选择在众多机器中的一台机器上加工;
- (3) 同一工件其工序有一定的先后顺序, 而不同工件的加工工序之间并没有先后顺序的要求;
- (4) 不同的工件之间具有相同的优先级别;
- (5) 不考虑机器故障, 工件加工过程中途中断等随机性因素的发生。

调度优化目标如下:

- (1) 最小化最大完工时间

每个工件完成最后一道工序的时间便是该工件的完工时间, 最后一个工件的最后一道工序的完成时间为最大完工时间。最大完工时间可以看成是评价一种调度方案的最根本指标, 表示着车间生产的效率高低, 公式如下:

$$f_1 = \text{minimize}(C_{\max}) \quad (1)$$

式中: C_{\max} 为最大完工时间。

- (2) 最小化机器最大负载

在车间调度问题当中, 每台机器的负载各不相同, 而随着调度方案的改变每个机器的负载也相应的做出变化, 针对那些负载最大的机器就相当于生产过程中的瓶颈设备, 为了提高机器的利用率, 应用的调度方案要能够使每台机器的负载平衡且尽量小, 公式如下:

$$f_2 = \min \left\{ \max_{1 \leq p \leq n} \sum_{q=1}^{n_p} T_{pqk} x_{pqk} \right\} \quad (2)$$

式中: n 为总工件数; p 为工件号; q 为工序号; k 为机器号; n_p 表示第 p 个工件的工序总数; T_{pqk} 表示工件 p 的第 q 道工序在机器 k 上的处理时间。 x_{pqk} 定义为当工件 p 的第 q 道工序在机器 k 上加工时, $x_{pqk} = 1$, 否则 $x_{pqk} = 0$ 。

- (3) 最小化机器总负载

所有机器的总负载随着调度方案的变化而便变化, 一个好的调度方案需要尽量减小所有机器的总负载, 公式如下:

$$f_3 = \min \left\{ \sum_{k=1}^m \sum_{p=1}^n \sum_{q=1}^{n_p} T_{pqk} x_{pqk} \right\} \quad (3)$$

式中: m 为总机器数。

2 混合蝙蝠算法的量子粒子群算法

2.1 粒子群优化算法

PSO 算法源于对鸟群觅食行为的研究, 是一种可以有效求解复杂优化问题的群智能优化算法^[14]。其种群中的每一个粒子以某一定的速度在 D 维的空间中飞行, 它们依据自身的个体所达到的最优位置以及全局的最优位置搜索解空间, 直至找到最优解。在搜索过程中, 粒子 i 通过如下方式在每一迭代中更新自身的速度变量 $v_{id}(t)$ 及位置变量 $x_{id}(t)$:

$$v_{id}(t+1) = wv_{id}(t) + c_1r_1(pbest_{id}(t) - x_{id}(t)) + c_2r_2(Gbest_d(t) - x_{id}(t)) \quad (4)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (5)$$

式中: $1 \leq d \leq D$, t 是迭代的次数; r_1 , r_2 是均匀分布在 $[0, 1]$ 间的随机数; w 为惯性权重; c_1 , c_2

是学习因子; $pbest_{id}(t)$ 为粒子 i 在 t 时刻的个体最优位置; $Gbest_j(t)$ 为当前种群中粒子最优的位置。

2.2 量子粒子群优化算法

在 PSO 算法的基础上, Sun 等人提出了 QPSO 算法, 该算法中具有量子行为的粒子可以搜索整个解空间, 有着更好的全局搜索能力^[7], 同时, QPSO 算法有着参数少、收敛速度快、收敛精度高等优势。

QPSO 算法的位置方程为:

$$X(t) = P \pm \frac{L}{2} \ln\left(\frac{1}{u}\right) \quad (6)$$

式中: u 是在 $[0, 1]$ 上均匀分布的随机数; P 是 δ 势阱中粒子收敛到的点; P 点在算法中是一个变化的随机变量; L 是随时间变化的特征长度。

同时, QPSO 算法中提出种群粒子的平均最优位置 $mbest$, 其通过如下公式计算:

$$mbest = \frac{1}{M} \sum_{i=1}^M Pbest(i) \quad (7)$$

式中: M 为种群中粒子的数目; $Pbest(i)$ 为粒子 i 的个体最好位置。

因此得到以下 2 个公式:

$$L(t+1) = 2\beta |mbest - X(t)| \quad (8)$$

$$P = \alpha \times Pbest(i) + (1 - \alpha) \times Gbest \quad (9)$$

式中: α 是在 $[0, 1]$ 上均匀分布的随机数; $Gbest$ 为全局最优位置; β 是收缩扩张因子, 一般将 β 的值设置为 1.0~0.5 的线性递减效果较好。

将式(8)、式(9)代入式(6)得到 QPSO 算法中粒子位置变量 $X(t)$ 进化公式:

$$X(t+1) = P \pm \beta \times |mbest - X(t)| \times \ln(1/u) \quad (10)$$

2.3 蝙蝠算法

在自然界中, 蝙蝠这种生物根据自身的声呐系统来生存, 它们通过发出不同频率的声波来判断周围的障碍物, 计算猎物的位置、速度等, 从而捕获食物。2010 年, Yang 等人根据蝙蝠这种利用回声定位原理来捕获食物的行为, 提出了蝙蝠算法^[11]。

2.3.1 蝙蝠算法的速度更新与位置更新

算法基于 n 只蝙蝠组成的蝙蝠群体之上, 在 D 维空间中, 算法中的速度更新公式与位置更新公式如下所示:

$$f_i = f_{\min} + (f_{\max} - f_{\min})\eta \quad (11)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \quad (12)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (13)$$

式中: f_i 为蝙蝠 i 所发声波的频率, f_{\min} 为声波频率的最小值, f_{\max} 为声波频率的最大值, v_i^{t-1} 和 v_i^t 分别表示蝙蝠 i 在 $t-1$ 和 t 时刻的速度, x_* 为蝙蝠位置的全局最优解, x_i^{t-1} 和 x_i^t 分别为蝙蝠 i 在 $t-1$ 和 t 时刻的位置, η 为 $[0, 1]$ 上均匀分布的随机数。

考虑到蝙蝠在最初寻找猎物的时候, 面对广阔的搜索范围, 初始阶段需要强度大、发声速度慢的声波来提高自身的搜索速率; 待找到目标后, 减小声波强度, 同时增加发声速度, 从而做到对目标的精确定位^[15-16]。这种声波强度与发声速度的变化情况以如下公式表达:

$$A_i^{t+1} = \alpha A_i^t \quad (14)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma \times t)] \quad (15)$$

式中: A_i^t , A_i^{t+1} 分别表示蝙蝠 i 在 t 和 $t+1$ 时刻的声波强度; α 为声波强度的衰减系数; r_i^0 为蝙蝠 i 最大发射声波的速度; r_i^{t+1} 为蝙蝠 i 在 $t+1$ 时刻的发射声波的速度; γ 为发射声波速度增加系数。

2.3.2 局部搜索部分

在每一次蝙蝠速度与位置更新的同时, 采用一种随机游走的策略来增强局部搜索的能力, 这种随机游走的方式是从当前最优解当中随机地选择一个最优解, 并以式(16)的方式进行位置的更新, 从而有效的避免陷入局部最优的问题。

$$x_n = x_o + \varepsilon A^t \quad (16)$$

式中: ε 为 $[-1, 1]$ 中的一个随机数, A^t 为蝙蝠群体在某一时刻的平均声波强度。

2.4 混合蝙蝠算法的量子粒子群算法

2.4.1 混合蝙蝠算法中的随机游走策略

与 PSO 算法相比, 加入量子行为的 QPSO 算法无论是在收敛精度还是收敛速度上都优于 PSO 算法, 然而 QPSO 算法在迭代过程中, 容易陷入局部最优解的问题依然存在。因此本文利用蝙蝠算法中针对局部搜索的随机游走策略以及声波强度与发声速度的变化方式, 与 QPSO 算法进行混合。

式(16)中的 A^t 为蝙蝠群体在某一时刻的平均声波强度, 我们引出 QPSO 算法中的平均最优位置 $mbest$ 来替换 A^t , 因此在 QPSO 算法进行粒子位置更新之后, 加入了新的局部搜索:

$$X_n(t+1) = X(t+1) + \varepsilon \times mbest \quad (17)$$

式中: $X(t+1)$ 为利用 QPSO 算法进行粒子位置更新的值; $X_n(t+1)$ 为与蝙蝠算法混合后所得到的粒子最新位置。

2.4.2 收缩扩张系数变化方式

在 BA 算法中, 根据蝙蝠前期要寻找猎物, 后期要对猎物进行位置、速度精确判断等问题, 要求发声速度是先慢后快, 以式(15)的方式来变化的。

而在 QPSO 算法中, β 是收缩扩张系数, 其是算法中仅有的控制参数, 综合影响着全局搜索能力和局部搜索能力。当 β 值比较大时, 全局搜索能力比较强; 而当 β 值比较小时, 对局部搜索有着明显的效果。

因此, 可以理解为 β 要先大后小的进行变化能得到更好的效果。根据 BA 算法中的变化方式, 并综合考虑到 β 本身是在 $[0, 1]$ 区间内变化的, 本文对式(15)进行变换, 得到式(18):

$$\beta = 1 \times [\exp(-\gamma' \times gen)] \quad (18)$$

式中: gen 为算法当前的迭代步数; $Ngen$ 为算法的总迭代步数; γ' 相当于 β 减少系数。

2.4.3 BAT-QPSO 算法步骤

本文算法的流程如下:

Step 1: 确定算法当中相关参数并初始化种群;

Step 2: 计算当前种群每一个粒子的适应度函数值;

Step 3: 初始化 $Pbest$ 和 $Gbest$;

Step 4: 根据粒子更新方程生成种群中新的粒子;

Step 5: 根据随机游走策略计算出粒子的最新位置;

Step 6: 判断是否需要更新种群的个体极值 $Pbest$ 和全局极值 $Gbest$;

Step 7: 判断是否满足算法的终止条件, 若满足, 执行 Step 8; 否则, 执行 Step 4;

Step 8: 算法结束, 输出最优值。

2.5 测试函数仿真分析

表 1 中 5 个标准测试函数用来验证 BAT-QPSO 算法的有效性, 其中 Sphere 和 Rosenbrock 为单峰函数, Rastrigrim、Griewank 和 Schaffer f6 三个函数为多峰函数。由于文献^[17]验证了 PSO 和 QPSO 算法在求解连续型问题上相较于遗传算法的优越性, 因此考虑使用该三算法进行标准函数测试的对比验证。

表 1 标准测试函数

函数	表达式	搜索空间
Sphere	$f_1 = \sum_{i=1}^d x_i^2$	$(-100, 100)$
Rosenbrock	$f_2 = \sum_{i=1}^{d-1} (100(x_{i+1}^2 - x_i)^2 + (1 - x_i)^2)$	$(-30, 30)$
Rastrigrim	$f_3 = \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$(-5.12, 5.12)$
Griewank	$f_4 = \frac{1}{4000} \sum_{i=1}^d x_i^2 - \prod_{i=1}^d \cos[\frac{x_i}{\sqrt{i}}] + 1$	$(-600, 600)$
Schaffer f6	$f_5 = 0.5 + \frac{\sin^2(\sqrt{x^2 + y^2}) - 0.5}{(1 + 0.001(x^2 + y^2))^2}$	$(-100, 100)$

为了体现比较的公平性, 3 种算法的参数设置如下: 函数维数为 10, 种群粒子个数为 20, 迭代 1 000 步, 3 种算法各独立运行 50 次。另外 PSO 算法加速常数为 $c_1 = c_2 = 2.0$, 惯性权重 w 为 0.9 到 0.4 的线性递减; QPSO 算法收缩扩张系数 β 从 1.0 到 0.5 线性递减; BAT-QPSO 算法设置收缩扩张系

数 β 以式(17)的方式递减。

5 个函数的收敛曲线如图 3 所示。

3 种算法运行 50 次的最优值, 平均值和标准差如表 2 所示, 其中的粗体值表示 3 种算法求解当前标准函数获得的最优值。

从图 3 和表 2 可以看出无论是单峰函数还是多

峰函数, BAT-QPSO 算法比 PSO 算法和 QPSO 算法在收敛速度和收敛精度上有很大的提高。只是在 Rosenbrock 这个单峰函数上, BAT-QPSO 求解出的最好值与另外 2 种算法相比较差一些, 但是在平均值和标准差上都有明显的优势。

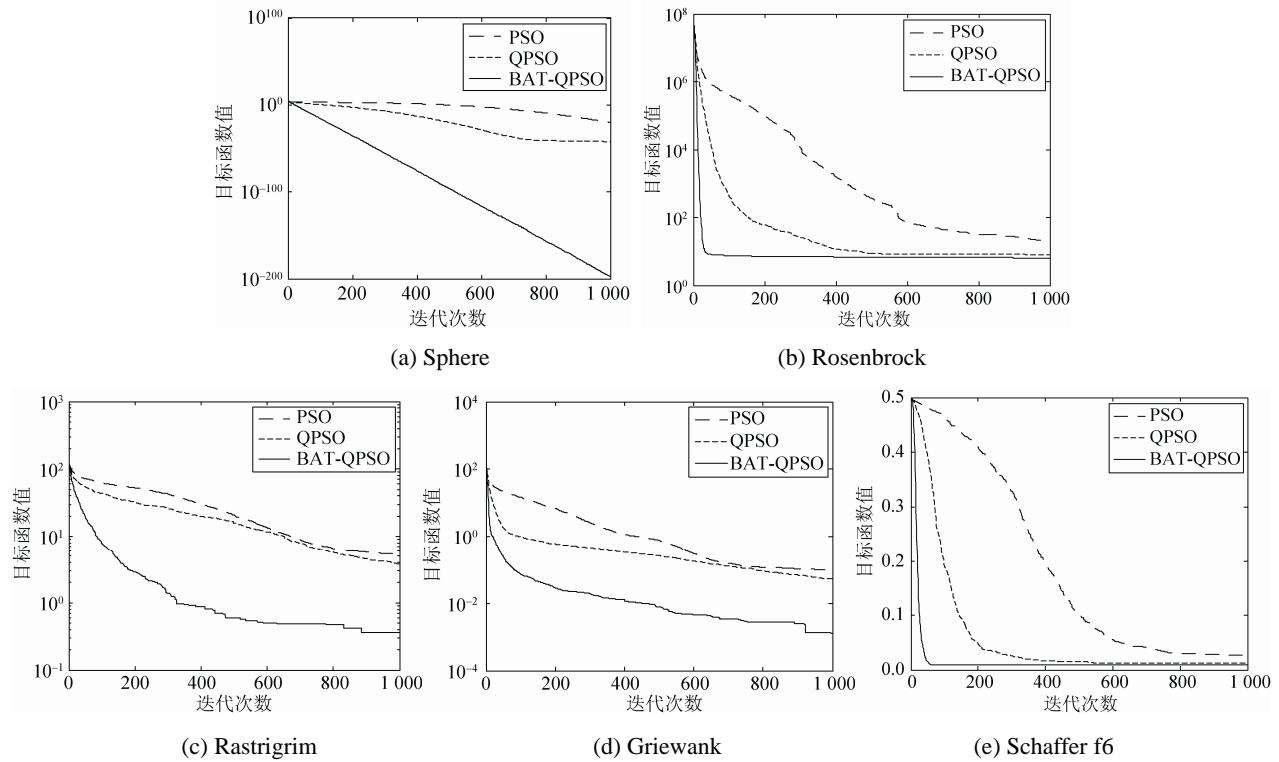


图 3 5 种标准测试函数优化曲线

表 2 3 种算法求解标准测试函数

函数	PSO			QPSO			BAT-QPSO		
	最好值	平均值	标准差	最好值	平均值	标准差	最好值	平均值	标准差
Sphere	7.993 7e-024	1.264 3e-020	5.737 2e-020	3.535 9e-058	9.886e-043	6.079 6e-042	3.792 6e-207	4.783 1e-198	0
Rosenbrock	0.165 2	20.865 3	48.466 4	0.108 4	8.129 8	13.900 9	5.514 2	6.464 7	0.411 0
Rastrigrim	1.989 9	5.363 5	2.279 3	0.994 9	3.830 8	1.976 6	0	0.363 9	1.796 2
Griewank	0.031 9	0.098 6	0.052 5	8.754 2e-006	0.054 3	0.029 7	0	0.001 3	0.005 6
Schaffer f6	0.009 7	0.026 4	0.015 5	0.009 7	0.012 4	0.008 3	0.009 7	0.009 7	3.218 1e-008

3 利用 BAT-QPSO 算法求解模具车间调度问题

3.1 粒子编码及解码

模具车间优化调度属于离散优化的问题, 利用

BAT-QPSO 算法在求此类问题时, 需要进行粒子的编码与解码, 本文采取常用的基于工序和机器的双层编码方式^[18]。

粒子位置由 2 个 L 维向量 $X_p[L]$ 与 $X_m[L]$ 表示, L 代表总工序数, $X_p[L]$ 表示了每个工件的操

作工序及其优先顺序, $X_m[L]$ 则与 $X_p[L]$ 对应显示了每个工件在加工每道工序时用的是哪一台机器。粒子的编码如表 3 所示。

在表 3 当中, $X_p[L]$ 一行中的 2, 1, 3, 1 分别表示工件 2 的第一道工序, 工件 1 的第一道工序, 工件 3 的第一道工序以及工件 1 的第 2 道工序; 与之对应的 $X_m[L]$ 分别为 3, 1, 2, 3, 则表示上述 4 个工序分别在在 3, 1, 2, 3 这 4 台机器上加工。

表 3 粒子编码示意图

粒子位置	L							
	1	2	3	4	5	6	7	8
$X_p[L]$	2	1	3	1	2	2	1	3
$X_m[L]$	3	1	2	3	2	1	2	3

3.2 位置更新及修正

算法中的位置公式在进行位置更新后, 往往是小数, 在处理调度问题时需要修正。在根据位置更新公式即式(17)计算更新出每个粒子对应的 $X'_p[L]$ 与 $X'_m[L]$ 。若 $X'_m[L]$ 中的元素超出区间 $[1, m]$, 则取边界值; 计算更新得到的 $X'_p[L]$ 中的元素, 按照数值递增的方式重新排列, 根据重新排列的结果再对初始值进行排序^[19]。

例如在表 3 中, $X_m[L]$ 为 [3 1 2 3 2 1 2 3], 若更新得到 $X'_m[L]$ 为 [1.4 2.7 0.6 1.2 0.8 2.1 1.7 2.4], 通过向上取整的方式对其进行整数修正, 则得到的 $X'_m[L]$ 为 [2 3 1 2 1 3 2 3]。

表 3 中的 $X_p[L]$ 为 [2 1 3 1 2 2 1 3], 若更新后得到 $X'_p[L]$ 为 [1.9 3.2 1.3 0.8 3.6 2.7 1.6 2.4], 将其中的值按从小到大的方式进行排序, 得到 $X'_p[L]$ 为 [0.8 1.3 1.6 1.9 2.4 2.7 3.2 3.6], 此时重新排序的 $X'_p[L]$ 对应到原来的 $X_p[L]$ 为 [1 3 1 2 3 2 1 2]。

3.3 模具车间实例仿真分析

根据图 2 的描述知道在实际模具车间中所涵盖的工序太多, 同时还包含了返工、插单、机器故障等特殊问题。因此在整个模具车间调度问题当中提取一个 14 个工件, 12 台机器, 共 84 道工序的实际调度模型并精简成如图 4 所示的流程。

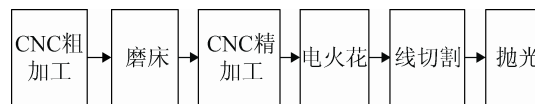


图 4 模具车间精简调度模型流程

每个工件包含 CNC 粗加工、磨床、CNC 精加工、电火花、线切割、抛光等 6 道工序。在 12 台机器中有 2 台机器可以处理 CNC 粗加工, 1 台机器对应磨床工艺, 2 台机器处理 CNC 精加工, 3 台机器加工电火花工艺, 3 台机器加工线切割工艺, 1 台机器进行抛光处理。

在文献[20]中, 已经验证 PSO 和 QPSO 算法在求解离散调度问题时优于遗传算法, 因此 BAT-QPSO 在此静态调度模型上依旧与二者对比。求解目标包含最小化最大完工时间 f_1 、最小化机器最大负载 f_2 、最小化机器总负载 f_3 。其中 PSO 算法设置粒子数为 100, 加速常数 $c_1 = c_2 = 2.0$, 惯性权重 w 为 0.9 到 0.4 的线性递减; QPSO 算法设置粒子数为 100, 收缩扩张系数 β 从 1.0~0.5 线性递减; BAT-QPSO 算法设置参数为 100 个粒子, 收缩扩张系数 β 以式(17)的方式递减。3 种算法分别独立运行 20 次, 迭代次数为 1 000。

利用 3 种算法求解上述模具车间调度模型, 所得的多目标值, 如表 4 所示。对比可见, BAT-QPSO 算法所得的调度结果在 3 个目标函数上都有明显的优势。3 个求解目标所得最优解已用粗体标识。利用 BAT-QPSO 算法所求甘特图如图 5 示。甘特图所表示的是算法 BAT-QPSO 所仿真求得 3 个目标的最优解, 与表 4 中所得最优解相对应, 是更好的调度方案。

表 4 3 种算法的仿真解

算法	f_1	f_2	f_3
PSO	139	91	596
QPSO	117	91	599
BAT-QPSO	98	91	594

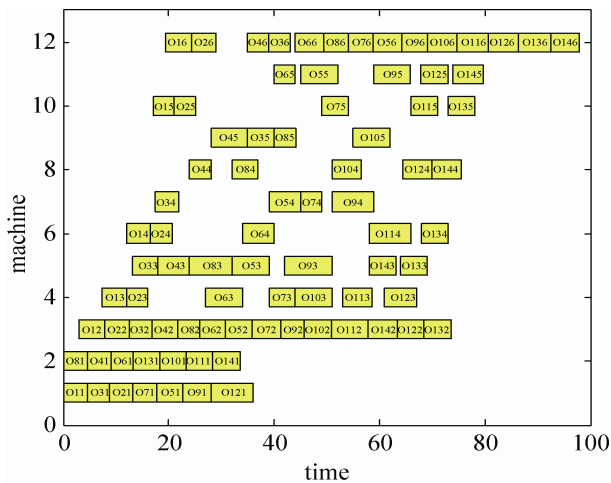


图 5 利用 BAT-QPSO 算法所得调度解

4 结论

本文一方面从模具车间生产出发整理出实际的加工流程,另一方面针对 PSO 算法以及 QPSO 算法出现早熟等问题,将 BA 算法与 QPSO 算法结合,取长补短,避免算法过早的陷入局部最优解,并将其用于解决文中提出的实际模具车间中的理想模型。由标准测试函数和实际调度问题两者的仿真分析可知,本文所混合的 BAT-QPSO 算法具有良好的收敛速度和收敛精度,并且能够有效的跳出局部最优。在求解实际调度问题时,能得到更优的调度解。

本文所提算法在求解该模具车间调度模型时,依然是一个静态的,较为理想化的调度问题,关于如何处理动态调度问题,或者是实际的考虑机器故障,临时插单等特殊的调度问题将是后续要研究的重点。

参考文献:

- [1] Shao X, Liu W, Liu Q, et al. Hybrid discrete particle swarm optimization for multi-objective flexible job-shop scheduling problem [J]. The International Journal of Advanced Manufacturing Technology (S0268-3768), 2013, 67(9/12): 2885-2901.
- [2] Zhang J, Wang W, Xu X, et al. Hybrid particle-swarm optimization for multi-objective flexible job-shop scheduling problem [J]. Control Theory & Applications (S1751-8644), 2012, 29(6): 715-722.
- [3] Yao B, Yang C, Hu J, et al. An improved ant colony optimization for flexible job shop scheduling problems [J]. Advanced Science Letters (S1936-6612), 2011, 4(6/7): 2127-2131.
- [4] Li J Q, Pan Q K, Tasgetiren M F. A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities [J]. Applied Mathematical Modelling (S0307-904X), 2014, 38(3): 1111-1132.
- [5] 张明, 田娜, 纪志成. 基于适应值欧式距离比的均衡蜂群算法 [J]. 系统仿真学报, 2015, 27(5): 980-989.
- [6] Marichelvam M K, Prabaharan T, Yang X S. A discrete firefly algorithm for the multi-objective hybrid flowshop scheduling problems [J]. Evolutionary Computation, IEEE Transactions on (S1063-6560), 2014, 18(2): 301-305.
- [7] Sun J, Feng B, Xu W. Particle swarm optimization with particles having quantum behavior [C]// Congress on Evolutionary Computation, Hawaii, USA: IEEE, 2004: 325-331.
- [8] 田娜, 纪志成. 带扰动算子的量子粒子群在水污染源识别中的应用 [J]. 系统仿真学报, 2015, 27(7): 1628-1637.
- [9] 方伟, 孙俊, 须文波. 基于微分进化算子的量子粒子群优化算法及应用 [J]. 系统仿真学报, 2008, 20(24): 6740-6744.
- [10] 朱红求, 阳春华, 王觉, 等. 基于混合 QPSO 的 LS-SVM 参数优化及其应用 [J]. 中南大学学报(自然科学版), 2011, 42(4): 1000-1004.
- [11] Yang X S. A new metaheuristic bat-inspired algorithm [M]// Nature inspired cooperative strategies for optimization (NICSO 2010). Germany: Springer Berlin Heidelberg, 2010: 65-74.
- [12] Moslehi G, Mahnam M. A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search [J]. International Journal of Production Economics (S0925-5273), 2011, 129(1): 14-22.
- [13] 王进峰, 范孝良, 万书亭. 一种基于图的柔性作业车间调度方法 [J]. 系统仿真学报, 2013, 25(10): 2499-2502, 2508.
- [14] Eberhart R, Kennedy J. Particle Swarm Optimization [C]// Proceeding IEEE Inter Conference on Neural Networks, Perth, Australia: IEEE, 1995: 1942-1948.
- [15] 盛晓华, 叶春明. 蝙蝠算法在 PFSP 调度问题中的应用研究 [J]. 工业工程, 2013, 16(1): 119-124.

(下转第 1280 页)