

文章编号: 1006-3080(2009)01-0125-07

混合量子衍生进化规划算法及其在并行机拖期调度中的应用

于艾清^{1,2}, 郭平², 顾幸生¹

(1. 华东理工大学自动化研究所, 上海 200237; 2. 上海电力学院电力与自动化工程学院, 上海 200090)

摘要: 针对并行机调度问题的求解, 提出一种新的混合量子衍生进化规划算法(Hybrid Quantum-Inspired Evolutionary Programming, HQEP)。目标函数是最小化所有工件的拖期总和。HQEP 将量子进化算法中的量子位、线性叠加态和量子旋转门的概念引入到进化规划算法中。定义了新的用于调度问题的量子旋转角, 使个体向更好的解靠近。此外, 针对并行机问题本身, 改进了个体的编码方式和新的变异方法。为了验证算法的有效性和收敛性, 将 HQEP 算法应用于同等并行机调度拖期问题的求解并加以不同规模的算例进行仿真实验。结果显示, 即使在小种群情况下, 所得解均优于进化规划求得解。

关键词: 量子计算; 量子衍生进化规划; 同等并行机拖期调度

中图分类号: O22.1

文献标志码: A

A Hybrid Quantum-Inspired Evolutionary Programming and Its Application in Tardiness Identical Parallel Machines Scheduling

YU Ai-qing^{1,2}, GUO Ping², GU Xing-sheng¹

(1. Institute of Automation, East China University of Science and Technology, Shanghai 200237, China; 2. Faculty of Electric and Automation Engineering, Shanghai University of Electric Power, Shanghai 200090, China)

Abstract: In this paper, a hybrid quantum-inspired evolutionary programming (HQEP) is proposed for identical parallel machines scheduling. The objective is to minimize the total tardiness of all jobs. In HQEP, the concept and principles of quantum computing, such as a quantum bit and superposition of states, are combined with evolutionary programming, and the Q-gate is introduced as a variation operator to drive the individuals toward better solutions. Moreover, an improved representation structure of individuals and mutation operator is proposed for scheduling problems in HQEP. Finally, an illustrative experiment is carried out on different scales of randomly generated test problems. Computational results show that HQEP outperforms evolutionary programming, even with a small population.

Key words: quantum computation; quantum-inspired evolutionary programming; tardiness identical parallel machines scheduling

并行机调度是实际生产过程中的一类典型调度问题, 它研究 N 个工件在 M 台机器上的加工过程, 每个工件仅需在某一台机器上加工一次, 要求某调度指标最优。尽管并行机调度问题是混合 Flow

收稿日期: 2007-11-22

基金项目: 国家自然科学基金资助项目(60674075, 60774078); 上海市教委重点学科建设项目资助(J51301)

作者简介: 于艾清(1981-), 女, 博士生, 主要研究方向为生产计划与生产调度。

通讯联系人: 顾幸生, E-mail: xsgu@ecust.edu.cn

Shop 调度问题的一个特例,但因其具有代表性,调度领域还是将其单独归为一类调度问题。目前,对于调度问题的研究方法较多集中于遗传算法^[1],也有基于遗传算法的DNA进化算法^[2]和离散PSO算法^[3]以及混合算法^[4]等进化算法。虽然同传统的优化方法相比,基本的进化算法是鲁棒的、全局的,但是这类方法本身也存在着不足,比如种群的大小、参数的设置依赖于问题,没有规则可循,容易早熟等。针对这些问题,研究者正在积极寻找改进办法。

量子衍生进化算法(简称为量子进化算法, Quantum-inspired Evolutionary Algorithm, QEA)^[5-9]是新近发展起来的一种概率进化算法,是量子计算与进化计算理论相结合的新兴交叉产物。量子计算(Quantum Computation, QC)的概念起源于1982年,诺贝尔物理学奖获得者Feynman将其看成是一个量子力学的物理过程^[7],现在它已经成为当今世界各国紧密跟踪的前沿学科之一。目前,量子计算的研究主要集中在两方面:一是量子计算机的研发,另一方面是量子算法的研究。前者着重于量子硬件的开发,但在实际技术上仍有较大挑战。后者利用了量子理论中有关量子态的叠加、纠缠和干涉等特性提出了量子算法,具有较大发展潜力。Shor^[8]和Grover^[9]的研究显示,通过量子计算有可能解决经典计算中的NP问题。量子进化算法基于量子计算的一些概念和理论(诸如量子位和量子叠加态),采用量子比特染色编码,可以在经典计算机上实施。较之于传统的进化算法,能较好地平衡算法的“探索”和“开发”的能力,换句话说,增加了群体的多样性,改善了选择性压力问题。基于量子进化算法的独特计算性能,它已被用于求解一部分组合优化问题,如小规模TSP问题^[5],0-1背包问题^[6],并取得了优于传统遗传算法的效果。然而,相关文献显示,关于量子进化算法在调度问题中的应用研究涉足不多,王凌^[10]提出了一种量子衍生遗传算法并应用于Flow Shop调度问题的研究。

为了深入研究量子进化算法在调度中的应用,本文分析了量子进化算法的基本概念,针对并行机调度问题的求解,提出一种新的混合量子衍生进化规划算法(Hybrid Quantum-Inspired Evolutionary Programming, HQEP)。相应地提出了针对并行机问题的编码方式和变异方法,将量子进化算法中的量子位、线性叠加态和量子旋转门的概念引入进化规划算法中。最后,将HQEP算法应用于同等并行机拖期调度问题的求解并以不同规模的算例进行仿真实验。

1 带顺序相关建立时间的同等并行机拖期调度问题

1.1 假设和符号表示

并行机调度问题一般具有以下形式:给定 M 台机器与 N 个工件($N > M$),每个工件可在任一台机器上完成加工。问题所追求的目标是怎样分配每台机器上所加工的工件及加工顺序,使所有工件的拖期总和最小。

本文研究的同等并行机调度问题作如下假设:

- (1)每个工件只有一个加工工序,可以在任何一台机器上加工并且不能中断;
- (2)每台机器在同一时刻只能加工一个工件;
- (3)所有工件在0时刻可以启动;
- (4)所有的机器都是相同的;
- (5)加工工件之间有顺序相关建立时间。

为了描述该问题的数学模型,设定符号表示如下:

工件 $i: i=1, 2, \dots, N$;

机器 $j: j=1, 2, \dots, M$;

N 个工件的排序: $x=(x_1, x_2, \dots, x_N)$,为整数变量;

每台机器上的加工工件数: $y=(y^1, y^2, \dots, y^M)$, $\sum y_i=N$,为整数变量;

工件 x_i 的加工时间: p_{x_i} ;

工件 x_i 和 $x_{i'}$ 之间的顺序相关建立时间: $s_{x_i x_{i'}}$, $i, i' \in \{1, 2, \dots, N\}$;

工件 x_i 的交货时间: d_{x_i} ;

工件 x_i 的完工时间: C_{x_i} ;

工件 x_i 的拖期时间: T_{x_i} 。

1.2 数学模型

本文提出1个新的并行机调度问题的表示方式,分别用工件加工排序和每台机器上的工件加工数来表示工件的完工时间。

建立该问题的数学模型如下:

$$\min(\sum_{i=1}^N T_{x_i})$$

$$\text{s. t. } C_{x_{i+j}}(x, y) = p_{x_{i+j}}, \quad l = \sum_{k=1}^j y_{k-1}, \\ y_0 = 0, \quad j = 1, 2, \dots, M \quad (1)$$

$$C_{x_{i+j}}(x, y) = C_{x_{i+j-1}}(x, y) + p_{x_{i+j}} + s_{x_{i+j-1} x_{i+j}}, \\ 2 \leq i \leq y_j, \quad j = 1, 2, \dots, M \quad (2)$$

$$T_{x_i} = \max(0, C_{x_i} - d_{x_i}) \quad (3)$$

$$1 \leq x_i \leq N \quad (4)$$

$$\sum_j y_j = N, 1 \leq y_j < N, j = 1, 2, \dots, M \quad (5)$$

式(1)表示第 j 台机器上第 1 个加工工件的完工时间, 式(2)表示第 j 台机器上第 i 个加工工件的完工时间, 式(3)为每个工件拖期时间, 式(4)和式(5)为变量约束。

2 求解并行机调度的混合量子衍生进化规划

2.1 量子位和量子个体的表示

在本文提出的 HQEP 中, 采用量子位^[6] (Q-bit or qubit)表示量子个体。这种方式的特点是可以表示解的任何线性叠加态。Q-bit 是二状态量子计算机中存储信息的最小单位。1 个量子位可能处于 $|0\rangle$ 或 $|1\rangle$ 或两种状态的线性叠加。因此, 1 个量子位可表示为:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (6)$$

式中 α, β 为代表相应状态出现概率幅值的 2 个复数。 $|\alpha|^2$ 和 $|\beta|^2$ 分别表示 Q-bit 处于状态“0”和状态“1”的概率, 满足 $|\alpha|^2 + |\beta|^2 = 1$ 。

1 个量子个体是由 m 个量子位的概率幅值组成的串, 定义如下:

$$\left[\begin{array}{c} \alpha_1 \\ \beta_1 \end{array} \middle| \begin{array}{c} \alpha_2 \\ \beta_2 \end{array} \middle| \dots \middle| \begin{array}{c} \alpha_m \\ \beta_m \end{array} \right],$$
$$|\alpha_i|^2 + |\beta_i|^2 = 1, i = 1, 2, \dots, m \quad (7)$$

可以看出, 1 个具有 m 个量子位的量子个体, 可以表示 2^m 个状态。例如, 假设 1 个量子个体由 2 个量子位组成, 它的 2 对幅值为 $\left[\begin{array}{c} 1/2 \\ \sqrt{3}/2 \end{array} \middle| \begin{array}{c} -1/\sqrt{2} \\ 1/\sqrt{2} \end{array} \right]$, 那么它的状态则可表示为 $-\frac{1}{\sqrt{8}}|00\rangle + \frac{1}{\sqrt{8}}|01\rangle - \frac{\sqrt{3}}{\sqrt{8}}|10\rangle + \frac{\sqrt{3}}{\sqrt{8}}|11\rangle$ 。换句话说, 它表示状态 $|00\rangle, |01\rangle, |10\rangle$ 和 $|11\rangle$ 的概率分别为 $1/8, 1/8, 3/8$ 和 $3/8$ 。因此, 1 个具有 2 个量子位的量子个体可以表示 4 种状态, 这样, 种群的多样性得到丰富, 更利于算法在搜索空间探索。

2.2 量子旋转门

量子变异在量子理论中是通过量子门变换矩阵实现的。量子门的选择有多种, 如非门 (NOT gate)、受控非门 (Controlled NOT gate)、哈玛达门 (Hadamard gate)、旋转门 (Rotation gate) 等。在本文提出的 HQEP 中, 选择量子旋转门进行量子变异, 量子旋转门的极坐标图如图 1 所示。

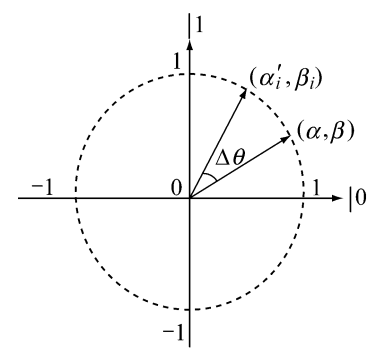


图 1 量子位个体的旋转门的极坐标图
Fig. 1 Polar plot of the rotation gate for Q-bit individuals

量子个体的第 i 个量子位 (α_i, β_i) 用旋转门 $U(\theta)$ 进行量子变异, 如下所示:

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = U(\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (8)$$

式中: θ_i 是旋转角, $\theta_i = s(\alpha_i, \beta_i) \Delta\theta_i$, $s(\alpha_i, \beta_i)$ 是 θ_i 的符号位, 决定旋转方向; $\Delta\theta_i$ 是旋转角旋转幅度, 可查询旋转角表格, 不同的优化问题, 可以有不同的旋转表。根据调度问题的特性, 本文定义了新的量子旋转角查询表格。

2.3 混合量子衍生进化规划

本文提出 HQEP 将量子个体和量子旋转门引入到进化规划算法中, 用量子个体来丰富种群的多样性, 用进化规划的选择操作来注重种群行为上的进化。

进化规划 (Evolutionary Programming, EP)^[11-13] 也是模拟自然界中生物的进化规律, 在优胜劣汰的竞争环境中不断进化, 逐渐逼近最优解。同遗传算法着重于对染色体的算子研究不同, 进化规划更注意个体和种群行为上的进化, 即强调父代和子代行为上的联系, 而不是基因方面的联系。EP 的显著特征是没有杂交过程, 其适应值用于指导子代个体的选择。

针对并行机调度问题, HQEP 主要包括初始种群产生、适应度评价、选择、变异、停止条件, 具体算法如下:

(1) 初始种群的产生。对于 HQEP 中的初始种群个体的编码, 采用 2 部分编码 $X|Y$ 。 $X = (x_1, x_2, \dots, x_n)$ 为 n 个工件的排序, 为字符编码, 字符串长度为 n ; $Y = (y_1, y_2, \dots, y_m)$ 为每台机器上的加工工件数, 为字符编码, 满足约束条件 $\sum_{j=1}^m y_j = n$, 字符串长度为 m 。

对于 X 部分编码, 先将量子个体转换成二进制

串个体,再换算成十进制编码,最后转换成工件排序。对于 Y 部分编码,考虑到每台机器上至少要加工 1 个工件的约束,每个 y_j 是在一定区间范围内随机产生的,区间为 $[1, n - (m - j) - \sum_{k=1}^j y_{k-1}]$,最后 1 台机器上加工的工件数为 $n - \sum_{k=1}^j y_{k-1}$ 。

如图 2 所示,考虑 1 个 3 工件 2 机器的并行机调度问题,采用 2 量子位来表示量子个体,1 个量子个体对应 1 个工件,随机生成 3 工件的 X 部分的量子编码为:

$$\begin{bmatrix} 1/2 & -1/\sqrt{2} \\ \sqrt{3}/2 & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -\sqrt{3}/2 \\ 1/\sqrt{2} & 1/2 \end{bmatrix} \begin{bmatrix} \sqrt{2}/\sqrt{3} & 1/\sqrt{2} \\ 1/\sqrt{3} & -1/\sqrt{2} \end{bmatrix}$$

对于每个量子位随机生成 1 个数 $r \in [0, 1]$,判断是否满足 $r < |\beta_i|^2$,满足则为“1”,否则为“0”。由此,量子个体转化为二进制个体 $[1 \ 0] [0 \ 0] [1 \ 0]$ 。换算成十进制为 $[2 \ 0 \ 2]$,最小的数对应工件 1,以此类推,如遇到相同数,则按出现次序排工件次序,进而得到工件排序为 $[2 \ 1 \ 3]$ 。 Y 部分编码则随机生成。1 个完整的个体则为 $[2 \ 1 \ 3 | 2 \ 1]$ 。

(2) 适应度评价。将本文所研究的目标函数 $\sum_{i=1}^N T_{x_i}$ 定为适应度。作为评判标准,根据适应度来选择子代个体。

(3) 选择操作。HQEP 采用随机 q -竞争选择方法使优良个体尽可能入选,但也允许极少数性能较差的个体入选,从而使少数性能较差的个体也可以保留在下一代群体中,这种选择方式也是动态的、不确定的。在混合群体中,通过 q -竞争选择法产生新一代群体。

q -竞争选择法是一种随机选择,总体上讲,优良

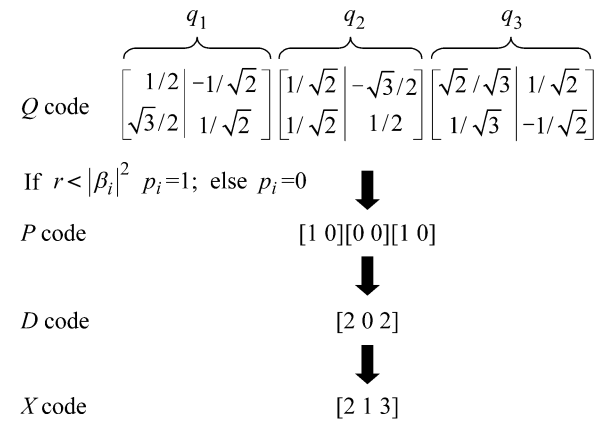


图 2 1 个 3 工件量子个体的 X 编码转换过程

Fig. 2 Transformation of X code for a Q-bit

individual of 3 jobs

个体入选的可能性较大。但是由于测试群体 q 每次都是随机选择的,当 q 个个体都不甚好时,有可能使较差的个体因得分高而入选。这也正是随机选择的本意。

(4) 量子进化规划变异。针对并行机调度问题的 2 部分编码形式, HQEP 的变异方式也有所改进。不同的编码部分采用不同的变异方法。

对于 X 部分编码,在保有原来的进化规划变异的基础上,同时采用量子旋转门来进行量子变异,保留 2 种变异后适应值较好的个体。针对并行机调度问题的编码,借鉴文献[11]中的变异方法,随机选取 X 部分编码中的 2 点并反转两点间的编码产生新的个体。同时定义了并行机量子旋转角查询表,如表 1。由于 X 部分编码的特殊性,为了使个体具有更好的多样性,设置 0-1 状态间的转换,旋转角度为 0.5π ;而 0-0 和 1-1 状态的逼近,选择较小的角度 0.2π 。 p_i 为量子个体的二进制码, b_i 为最优个体的二进制码, $f(\cdot)$ 为个体的适应值, $\Delta\theta_i$ 是旋转角旋转幅度, $\alpha_i\beta_i > 0$ 表示此时位于一、三象限, $\alpha_i\beta_i < 0$ 表示此时位于二、四象限,对于不能判断象限的保持原有状态。

表 1 并行机调度问题的旋转角查询表

Table 1 Lookup table of rotation angle for parallel machine scheduling problems

p_i	b_i	$f(p) < f(b)$	$\Delta\theta_i$	$s(\alpha_i, \beta_i)$	
				$\alpha_i, \beta_i > 0$	$\alpha_i, \beta_i < 0$
0	0	False	0.2π	-1	+1
0	0	True	0	0	0
0	1	False	0.5π	+1	-1
0	1	True	0	0	0
1	0	False	0.5π	-1	+1
1	0	True	0	0	0
1	1	False	0.2π	+1	-1
1	1	True	0	0	0

对于 Y 部分编码,根据所考虑问题编码形式的特殊性,在机器编码部分提出 1 种新的变异方法。机器编码部分必须满足两点约束: y_j 为正整数;加工工件数必须为 n 。这促使了新的变异方法的产生。

随机选取 Y 部分编码的两点,将其中 1 点进行高斯变异,为了满足约束,另 1 点则补偿其变异前后的变化量。这 1 点的高斯变异原则如下:

$$y'_j = y_j + [\sigma \cdot N(0, 1)] \tag{9}$$

$$\sigma = \beta \times \frac{FIT_{old}}{FIT_{opt}} \tag{10}$$

式中: y'_{ij} 和 y_j 分别代表子代和父代在 Y 部分中的变异点 j ; $N(0, 1)$ 表示均值为 0 且标准差为 1 的高斯随机变量; σ 表示变动范围; FIT_{old} 父代的适应度; FIT_{opt} 为种群中的最优适应度; β 是尺度因子; $[]$ 表示取整。

(5) 停止条件。终止的准则选择最大迭代次数和最优个体与平均值的偏差作为判据。

2.4 算法步骤

- (1) 种群初始化, 将量子个体转化为工件排序;
- (2) 评价个体, 并存储当代最好个体;
- (3) 将个体根据不同编码实施进化规划变异、量子变异和高斯变异;
- (4) 变异后的子代加上原来的父代共同形成混合群体, 并计算其适应值;
- (5) 进行 q 竞争选择, 产生下一代个体;
- (6) 评价个体, 并存储当代最好个体;
- (7) 判断是否满足停止条件, 如满足, 则停止, 输出最优解; 如不满足, 回到步骤(3)。

3 仿真实验

为了验证 HQEP 算法的有效性, 考虑不同规模下随机生成的同等并行机调度算例。仿真研究在 Intel Pentium IV 3.07 GHz 处理器、512 M 内存的计算机上运行程序, 编程环境为 Matlab 6.5。工件处理时间在均匀分布的区间 $[1, 20]$ 内随机产生, 顺序相关建立时间 $s_{ij} = a \times \min(p_i, p_j)$, 其中 a 为区间 $[A, B]$ 内的任意数, 本文设定 $[A, B] = [0.2, 0.5]$ 。根据文献[13]中的方法, 工件的交货期在均匀分布的区间 $[(1 - T - R/2)\bar{p}, (1 - T + R/2)\bar{p}]$ 中随机产生。设置 $T = 0.2$ 控制交货期的松紧度, $R = 0.8$ 调节交货期的相对范围。工件数分别为 10, 20, 50 和 100, 机器数为 2。

考虑种群数和 q 值对于 HQEP 性能的影响, 分别给 HQEP 算法定义不同的参数组。HQEP1 中, $popsiz = 5, q = 2 \times popsiz$ 。HQEP2 中 $popsiz = 10, q = 1.5 \times popsiz$ 。HQEP3 中 $popsiz = 20, q = popsiz$ 。每个算例运行 10 次, 最大代数为 500, 取其最优值(Best)、平均值(Avg)和最差值(Worst)分别列于表 2、表 3 和表 4 中, 并与相同参数设置下的基本进化规划(EP for Scheduling, SEP)进行比较。

表 2~4 中列出了各个算例运行 10 次的最优结果, 其中小规模问题 10 工件的算例, 在 HQEP 算法

表 2 HQEP1 与 SEP 求解的最优值, 平均值, 最差值的结果比较

Table 2 Results comparison between HQEP1 and SEP in terms of Best, Avg and Worst

N	HQEP1			SEP		
	Best	Avg	Worst	Best	Avg	Worst
10	1.44	3.139	5.46	2.00	3.834	6.95
20	34.50	43.235	56.55	36.56	48.711	68.08
50	173.96	243.716	341.09	174.21	277.829	356.39
100	738.31	1 209.70	1 791.60	785.14	1 268.30	1 585.50

表 3 HQEP2 与 SEP 求解的最优值, 平均值, 最差值的结果比较

Table 3 Results comparison between HQEP2 and SEP in terms of Best, Avg and Worst

N	HQEP2			SEP		
	Best	Avg	Worst	Best	Avg	Worst
10	1.44	2.896	5.92	1.44	3.214	6.50
20	32.62	39.286	46.02	33.16	42.898	59.26
50	141.43	195.828	299.19	154.40	219.755	317.60
100	552.06	767.393	991.58	577.19	938.633	1 423.60

表 4 HQEP3 与 SEP 求解的最优值, 平均值, 最差值的结果比较

Table 4 Results comparison between HQEP3 and SEP in terms of Best, Avg and Worst

N	HQEP1			SEP		
	Best	Avg	Worst	Best	Avg	Worst
10	1.44	2.146	3.69	1.44	2.829	4.29
20	30.73	36.368	41.93	31.05	38.985	60.17
50	114.91	145.146	182.13	125.29	162.238	254.44
100	506.15	628.621	863.55	608.30	846.409	1 176.80

种群数为 5 的情况下, 都找到了最优解。与相同参数下进化规划算法相比, 即使在种群数较小的情况下, HQEP 依然能有较好的搜索和开发能力。随着种群数的增大, HQEP 算法搜索到最优解的次数也逐渐增多, 20、50 和 100 工件规模的问题所找到的解都优于 SEP。在 q 竞争选择法中, q 参数取 $2 \times popsiz$ 表示是确定性选择, 即所有个体包括父代和子代都参与竞争, 这样保证了最优解的趋势, 但也容易早熟。当 $q = 1.5 \times popsiz$ 或者 $q = popsiz$ 时, 表示是随机性选择, 在保证最优解的趋势的同时, 也允许较差个体的存在, 可以帮助跳出局部最优。所以, 纵向对比上述表格, 用 HQEP2 和 HQEP3 找到的解都优于 HQEP1。

图 3 为不同规模算例下 SEP 和 HQEP 算法所

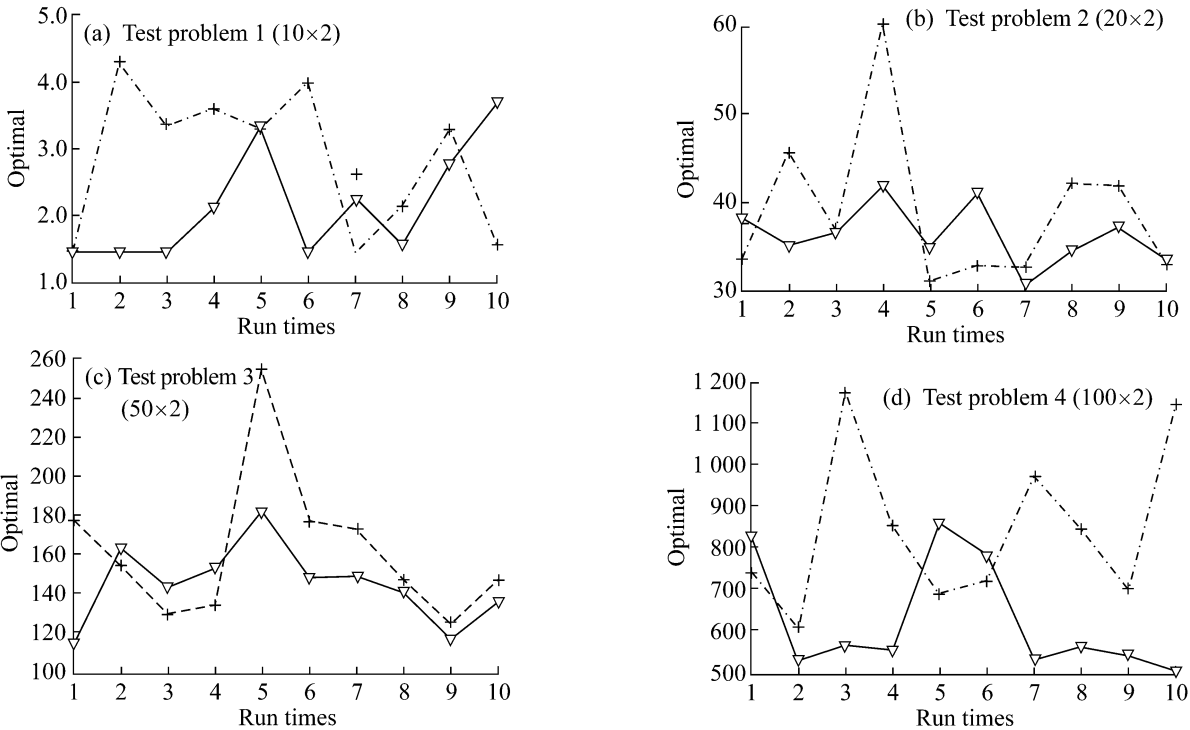


图 3 不同规模算例下用 SEP 算法和 HQEP 算法求得的解分布图

Fig. 3 Distribution of solutions generated by SEP and HQEP for different test problems

--+-- SEP; —▽— HQEP3

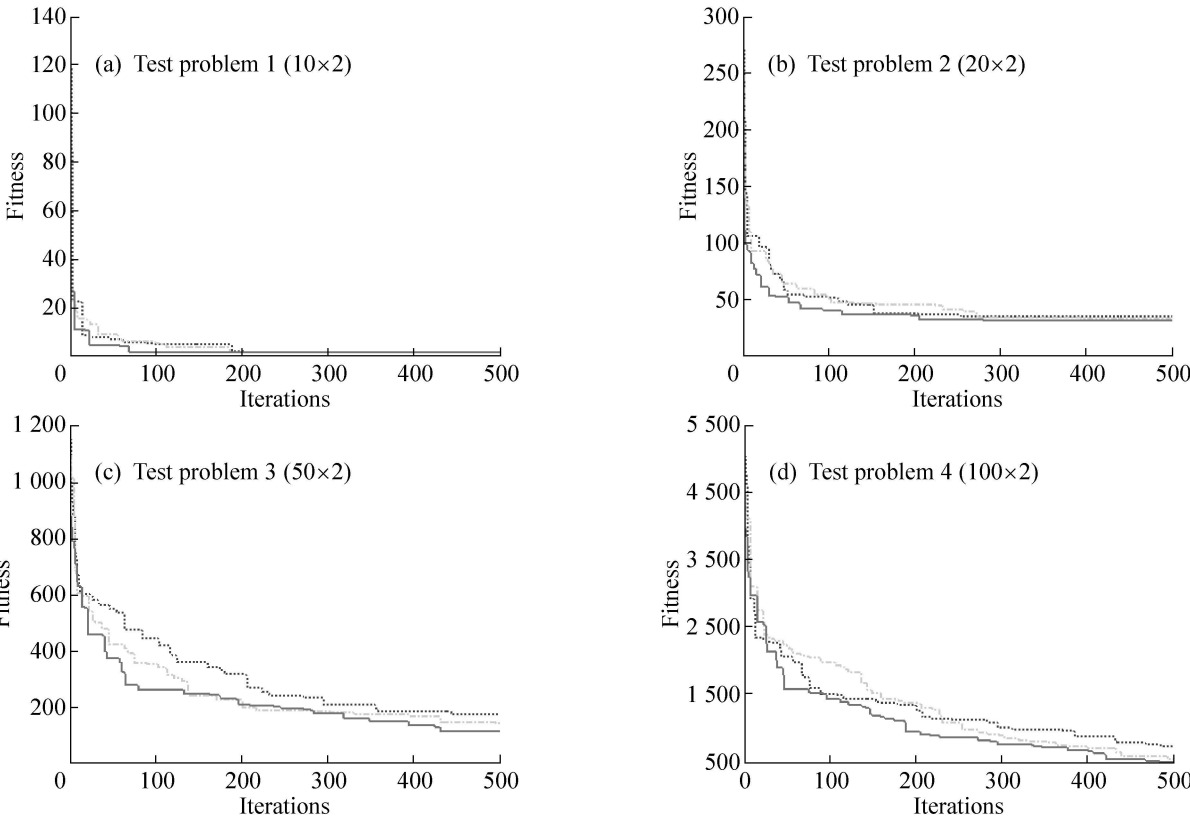


图 4 不同参数下的 HQEP 算法的最优值演化曲线

Fig. 4 Optimal evolution curves of HQEP with different parameters for test problems

.....HQEP1; ----HQEP2; —HQEP3

求解的分布图。从图中可以看出,用 SEP 算法求解的算例,解的波动范围较大,容易早熟;而 HQEP 算法所得的解,整体波动范围较小,平均值优于 SEP 算法。

对比不同参数下 HQEP 算法的搜索性能,如图 4 所示,其显示了算法求解算例的最优值的进化过程。在求解小规模问题时, HQEP 在 200 代左右就能找到最优解,不同的种群数和 q 值对于问题的求解影响较小,由于量子个体的多状态表示方式,使得即使小种群也能以较快的收敛速度找到解。当工件数增大到 50 时,就体现出参数的设置对于算法求解问题的影响了。适当的种群数能增加个体的多样性,量子变异和随机 q 值可以防止早熟现象。

4 结束语

本文提出了一种新的混合量子进化规划算法并应用于并行机调度问题,引入量子个体来表示工件的排序,由此增加解的多样性,并定义了新的量子旋转角度查询表来进行并行机调度问题的工件排序的量子变异。算法结合了量子进化算法的探索、开发能力和进化规划的随机选择机制,有效的避免了早熟问题,提高了算法的搜索能力。

将量子计算与其他智能优化方法结合是目前的研究热点,其在调度问题上的应用研究较少。今后的研究方向可以考虑寻找更为适合调度问题的量子编码方式和量子旋转角的定义。

参考文献:

- [1] Funda Sivrikaya-Serifoglu, Gunduz Ulusoy. Parallel machine scheduling with earliness and tardiness penalties[J]. Computers & Operations Research, 1999, **26**: 773-787.
- [2] 牛群,顾幸生.基于启发式规则的新型进化算法在流水车间调度中的应用[J].华东理工大学学报:自然科学版,2006, **32** (12): 1472-1477.
- [3] Lian Zhigang, Gu Xingsheng, Jiao Bin. A similar particle swarm optimization algorithm for permutation flowshop scheduling to minimize makespan[J]. Applied Mathematics and Computation, 2006, **175**(1): 773-785.
- [4] 徐震浩,顾幸生.用混合算法求解 Flow Shop 调度问题[J].华东理工大学学报:自然科学版,2004, **30**(2): 234-238.
- [5] Narayanan A, Moore M. Quantum-inspired genetic algorithms [C]//Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC96). N ogaya, Japan: IEEE Press, 1996: 61-66.
- [6] Han K H, Kim J H. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization[J]. IEEE Trans Evol Comput, 2002, **6**: 580-593.
- [7] Feynman R P. Simulating physics with computers[J]. International Journal of Theoretical Physics, 1982, **21**: 133-153.
- [8] Shor P W. Algorithms for quantum computation: Discrete logarithms and factoring[C]//Proceedings of the 35th Annual Symposium on the Foundation of Computer Sciences. Santa Fe, NW: IEEE Computer Society Press, 1994: 124-134.
- [9] Grover L K. A fast quantum mechanical algorithm for database searches[C]//Proceedings of the 28th ACM Symposium on the Theory of Computing. Philadelphia, Pennsylvania: ACM Press, 1996: 212-215.
- [10] Wang Ling, Wu Hao, Zheng Da-zhong. A quantum-inspired genetic algorithm for scheduling problems[J]. Lecture Notes in Computer Science, 2005, **3612**: 417-423.
- [11] Fogel D B. Applying evolutionary programming to selected traveling salesman problems[J]. Cybernetics and Systems, 1993, **24**(1): 27-36.
- [12] Liu Bin, Wu Cheng. Scheduling algorithm based on evolutionary computing in identical parallel machine production line[J]. Robotics and Computer Integrated Manufacturing, 2003, **19**: 401-407.
- [13] Shim Sang-Oh, Kim Yeong-Dae. Scheduling on parallel identical machines to minimize total tardiness[J]. European Journal of Operational Research, 2007, **177**: 135-146.