

基于 DEA 混合算法的模糊车间 作业计划问题的研究*

卢冰原¹, 程八一²

(1. 南京工程学院 经济管理学院, 南京 211167; 2. 合肥工业大学 管理学院, 合肥 230009)

摘要: 针对以最小化制造跨度为目标, 具有模糊加工时间的车间作业计划问题, 采用梯形模糊数来表征时间参数, 并应用可能性理论, 在此基础上构建车间作业计划问题目标函数。为了对模糊环境下的车间作业计划问题进行有效求解, 给出了一种 DEA-GA 混合求解算法, 混合算法采用了 DNA 进化算法的分裂、变异和水平选择算子, 然后利用遗传算法的交叉算子实现个体之间的交互, 避免早熟收敛。仿真实验表明, 该算法高效可行, 与 GA 等优化算法相比, 具有更快的收敛速度。

关键词: 车间作业计划; 模糊环境; DNA 进化算法; 遗传算法; 组合优化

中图分类号: TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2010)08-2933-03

doi:10.3969/j.issn.1001-3695.2010.08.033

Research of fuzzy job-shop scheduling problem based on hybrid DEA-GA algorithm

LU Bing-yuan¹, CHENG Ba-yi²

(1. School of Economy & Management, Nanjing Institute of Technology, Nanjing 211167, China; 2. School of Management, Hefei University of Technology, Hefei 230009, China)

Abstract: This paper studied the job-shop scheduling problem which had fuzzy operation time and aimed at minimizing makespan. For this problem, it introduced trapezoidal fuzzy number to denote time parameters, on which the aim function was constructed. After that, proposed a hybrid DNA evolutionary algorithm integrating the mechanism of DNA evolutionary algorithm with genetic algorithm to get perfect scheduling scheme. To escape immature convergence, combined the crossover operator of genetic algorithm with the operators in DNA evolutionary algorithm, including the division, level selection, mutation operators. Moreover, simulation results show that this algorithm is feasible and effective. Compared with other optimization algorithms such as GA, it has rapid convergence ability.

Key words: job-shop scheduling (JSS); fuzzy environment; DNA evolutionary algorithm; genetic algorithm; combinatorial optimization

车间作业计划 (JSS) 问题是众多实际生产调度问题的一种简化模型, 多年来学者们对此进行了大量的研究。目前, 针对 JSS 问题的研究大多集中在理想的不确定性环境下, 相关参数为确定值, 然而现实生产过程中, 由于人力等因素的影响, 往往存在着大量的模糊因素。本文针对具有模糊时间参数的 JSS 问题, 用梯形模糊数 (trapezoidal fuzzy number, TrFN) 来表征时间参数, 并在此基础上构建模糊环境下的 JSS 问题模型。

在 JSS 问题求解方面, 研究人员先后提出了包括遗传算法在内的多种求解方法^[1,2]。针对遗传算法在应用中出现的收敛速度慢和早熟问题, 余文等人^[3]基于单细胞分裂过程中 DNA 分子的演变和进化机制, 提出了 DNA 进化算法 (DNA evolutionary algorithm, DEA)。DEA 首先被应用于连续优化问题的求解, 之后牛群等人^[4,5]利用 DEA 方法求解 flow shop 和 job shop 问题, 将算法应用到经典调度问题中。针对实际应用中 DEA 可能存在的早熟收敛问题, 本文将传统的遗传算法与 DEA 的进化算子有效结合, 在 DEA 迭代过程中增加了交叉算

子, 给出了一种 DEA 混合算法, 使得个体的进化在局部区域深度搜索的同时, 也能跳出局部极值模态, 实现有效的全局优化。本文采用这种 DEA 混合算法对模糊环境下的 JSS 问题进行求解, 并通过实例对算法的有效性进行了验证。

1 模糊车间作业计划问题模型

在一个规模为 $m \times n$ 的 JSS 问题中, 包括待加工的作业集合 $J = \{J_1, J_2, \dots, J_n\}$ 和机器集合 $M = \{M_1, M_2, \dots, M_m\}$ 。作业集合中第 i 个作业 J_i 包含 n_j 个工序, 表示为一组有序工序集合 $O_{ij}, i \in \{1, \dots, n_j\}, O_{ij}$ 表示第 j 个作业的第 i 个工序。工序 O_{ij} 在机器 M_k 上的加工时间为 T_{ijk} , 作业在加工期间不可以被中断, 并且在任一时刻, 一台机器至多只能处理一个作业。目标是发现一种具有最优性能指标的作业计划方案。本文针对加工时间 T_{ijk} 模糊不确定的情况, 用梯形模糊数来表征 T_{ijk} , 表示为四元组 $(t_{ijk}^o, t_{ijk}^{ml}, t_{ijk}^{mu}, t_{ijk}^p)$, 如图 1 所示。

收稿日期: 2009-12-27; **修回日期:** 2010-02-22 **基金项目:** 国家自然科学基金资助项目 (70671096); 江苏省教育厅高校哲学社会科学基金资助项目 (09SJD630036); 南京工程学院校级科研基金资助项目 (QKJA2009015)

作者简介: 卢冰原 (1977-), 男, 安徽阜南人, 副教授, 博士, 主要研究方向为商务智能 (qqiceland@sina.com); 程八一 (1981-), 男, 讲师, 博士, 主要研究方向为商务智能。

操作时间分布函数 $\mu_{T_{ijk}}(t)$ 表示 T_{ijk} 取值为 t 的能性程度, 位于区间 $[t_{ijk}^{ml}, t_{ijk}^{mu}]$ ($t | \mu_{T_{ijk}}(t) = 1$) 内的 t 被认为是 T_{ijk} 最可能的取值; T_{ijk} 在区间 $[t_{ijk}^o, t_{ijk}^p]$ ($t | \mu_{T_{ijk}}(t) = 0$) 以外的区域取值的可能性最小。本文中 JSS 问题以最小制造跨度 (makespan) 为目标, 表示为 $C_{max} = \max_k (\max_j \max_i (C_{ijk}))$, $j \in \{1, 2, \dots, n\}$, $i \in \{1, 2, \dots, n_j\}$, $k \in \{1, 2, \dots, m\}$ 。

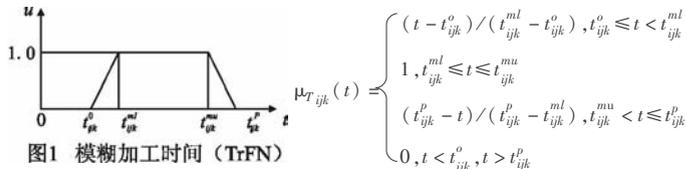


图 1 模糊加工时间 (TrFN)

$C_{ijk} = S_{ijk} + T_{ijk} \circ S_{ijk}$, T_{ijk} 和 C_{ijk} 分别表示 O_{ij} 在机器 M_k 上的开始时间, 加工时间和结束时间。 C_{ijk} , S_{ijk} 和 makespan 也同样表征为 TrFN, 相应数值可通过 TrFN 代数运算获得。对应梯形模糊数四元组中四个参数, 有 $C_{ijk}^d = S_{ijk}^d + T_{ijk}^d$, $C_{max}^d = \max_k (\max_j \max_i (C_{ijk}^d))$, $d \in \{1, 2, 3, 4\}$, 分别对应 TrFN 四元组中的各个分量。 $C_{max} = \bigvee_k \bigvee_j \bigvee_i (C_{ijk}^o, C_{ijk}^{ml}, C_{ijk}^{mu}, C_{ijk}^p)$ 。根据可能性理论^[6], 目标函数表示为

$$E[C_{max}] = \int_0^\infty Cr\{C_{max} \geq r\} dr - \int_{-\infty}^0 Cr\{C_{max} \leq r\} dr$$

2 DNA 进化算法

DEA 模拟自然界中单亲生物的进化过程行为, 是一种基于迭代的进化算法^[3]。个体在每次迭代中通过分裂算子 F , 变异算子 M , 选择算子 S_X, S_Y 这三类基本算子完成自身的演化。个体的 DNA 单链记为 v , 链长为 L , D 是 v 的全集, D 的全部子集构成的集类为 D^* 。各类算子的定义如下:

定义 1 若 F 是 $D \rightarrow D^*$ 的映射, 对 $\forall v \in D, F(v) = \{v, -v, v\bar{\cdot}, -v\bar{\cdot}\} \equiv [v \ -v \ \bar{v} \ -v\bar{\cdot}] \in D^*$, 则 F 是 $D \rightarrow D^*$ 上的分裂算子。通过分裂算子, 可以获得 v 的分裂集 $[v \ -v \ \bar{v} \ -v\bar{\cdot}]$, 分裂集中的四个元素分别对应 v 的恒等变换、反演变换、同互补变换和异互补变换。

定义 2 若 M 是 $D \rightarrow D^*$ 的映射, 对 $\forall v \in D, M(v) = \{v_1, v_2, \dots, v_L, v\} \equiv [m_1 \ m_2 \ \dots \ m_L \ m] \otimes v = [v_1 \ v_2 \ \dots \ v_L \ v]$, 则 M 为 $D \rightarrow D^*$ 上的变异算子, 其中 m_i 是位变异映射, 控制 v_i 的变异。

定义 3 若 $\forall D_x^* \subset D^*$, S_X 为 $D^* \times D^* \rightarrow D^*$ 的映射, 且 $\forall V = [v \ -v \ \bar{v} \ -v\bar{\cdot}] \in D^* \times D^*$, 有 $S_X(V) \equiv V_X, f(V_X) \equiv \max\{f(v), v \in V\}$, 则 S_X 为 $D^* \times D^* \rightarrow D^*$ 的水平选择算子, S_X 与 F 的乘积 E_X 为水平演化算子, 完成对个体的分裂和选择。

定义 4 V 的集类记为 D_V^* , S_Y 是 $D_V^* \rightarrow D$ 的映射, 若对 $\forall V_M \in D_V^*$, 有 $S_Y(V) \equiv V_M, f(V_M) \equiv \max\{f(v), v \in V_M\}$, 则 S_Y 为 $D_V^* \rightarrow D$ 上的垂直选择算子, S_Y 与 M 的乘积 E_Y 是垂直演化算子, E_Y 使个体发生变异, 并从变异之后的个体中间进行选择。

DEA 在解决离散优化问题时有较好的性能, 但 DEA 中也存在一定的问题。如果垂直选择采用确定性最优方法, 则 DEA 不再具有全局搜索性能, 个体会独立地收敛于局部极值模态。本文将遗传算法的交叉算子嵌入 DEA, 利用个体之间的交叉, 完成全局优化, 有效地解决了早熟收敛问题。

3 面向 JSS 问题的 HDEA 混合算法

3.1 DNA 单链的编码及初始化策略

编码是 DEA 算法实施的关键环节, 在求解 JSS 问题时,

DNA 单链表示一个完整的作业排序, 单链中的每个基因代表一个作业, 根据它们在单链中出现的顺序确定是该作业的第几道工序。例如, 一个 3×3 的 JSS 问题中随机产生了一个可行解 $[2 \ 1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 1]$, 第一个出现的 2 代表作业 2 的第 1 道工序。采用这种编码方式, 可以保证任意的随机排序都是可行解。

在初始种群构造方面, 首先要构造一个 DNA 单链, 单链中每个基因用作业号表示, 每个作业号出现的次数等于该作业包含的工序数, 单链长度为总工序数。初始化方法如下:

- a) 通过随机产生 $\text{rand} \in [1, n]$, 并不断将 rand 加入到单链中, 直至某个随机数出现次数和对应的作业中工序数一致并且单链长度等于总工序数, 从而构造出一个 DNA 单链。
- b) 循环进行步骤 a), 直至个体数达到种群规模 m_0 。

3.2 个体适应度的计算

适应度是衡量种群中个体优劣的标准, 一般情况下, 将所研究问题的目标函数作为适应度函数, 针对 JSS 问题, 本文以制造跨度作为个体适应度, 将种群中个体按照对应的制造跨度从小到大排列, 选择具有最小制造跨度的个体作为最优值。针对梯形模糊数 $C_{max}(C_{ijk}^o, C_{ijk}^{ml}, C_{ijk}^{mu}, C_{ijk}^p)$, 根据可能性理论^[6], 调度目标函数

$$E[C_{max}] = \int_0^\infty Cr\{C_{max} \geq r\} dr - \int_{-\infty}^0 Cr\{C_{max} \leq r\} dr = \int_0^{C_{ijk}^o} Cr\{C_{max} \geq r\} dr + \int_{C_{ijk}^{ml}}^{C_{ijk}^{mu}} Cr\{C_{max} \geq r\} dr + \int_{C_{ijk}^p}^{C_{ijk}^o} Cr\{C_{max} \geq r\} dr + \int_{C_{ijk}^{mu}}^{C_{ijk}^p} Cr\{C_{max} \geq r\} dr - \int_{-\infty}^0 Cr\{C_{max} \leq r\} dr = C_{ijk}^o + 3(C_{ijk}^{ml} - C_{ijk}^o)/4 + (C_{ijk}^{mu} - C_{ijk}^{ml})/2 + (C_{ijk}^p - C_{ijk}^{mu})/4 = (C_{ijk}^o + C_{ijk}^{ml} + C_{ijk}^{mu} + C_{ijk}^p)/4$$

3.3 分裂和水平选择

分裂算子使个体完成自身的繁殖, 它包含恒等、反演、同互补和异互补四种变换, 根据这四种变换可以获得 v 的分裂集 $F(v) = [v \ -v \ v\bar{\cdot} \ -v\bar{\cdot}]$ 。其中: 反演变换 $-v$ 是 v 的逆序排列; 同互补变换 $v\bar{\cdot}$ 中的基因值由工件数 n 加上 1 减去工件号得到; 异互补变换是 $v\bar{\cdot}$ 的逆序排列。例如, 若 $v = [2 \ 1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 1]$, 则 $-v = [1 \ 3 \ 1 \ 2 \ 2 \ 3 \ 1 \ 2]$, $v\bar{\cdot} = [2 \ 3 \ 1 \ 2 \ 2 \ 3 \ 1 \ 3]$, $-v\bar{\cdot} = [3 \ 1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 2]$ 。在分裂集的四个体中, 水平选择算子选择其中适应度最小的个体, 进入变异和垂直选择阶段。

3.4 变异和垂直选择

变异算子可以产生与 v 差异较大的个体, 针对 JSS 问题的可行解数量多的特点, 本文采用双重变异的方法。首先从第一个基因开始, 依次交换 v 中的每一个基因和相邻的后面的基因; 然后随机产生 $1 \sim m_0$ 之间的两个整数, 交换它们的基因位。例如, 个体 $v = [2 \ 1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 1]$, 产生的随机数为 (2, 5), 则变异过程为 $[2 \ 1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 1] \rightarrow [1 \ 3 \ 2 \ 2 \ 1 \ 3 \ 1 \ 2] \rightarrow [1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 1 \ 2]$, 变异个体 $v_m = [1 \ 1 \ 2 \ 2 \ 3 \ 3 \ 1 \ 2]$ 。变异产生的个体在原来的个体上有着较大改进, 因此直接保留 v_m 。

3.5 交叉

采用单点交叉的方法, 对分裂和变异算子产生的个体进行交叉。对个体 v_1, v_2 来说, 先产生 $1 \sim n$ 的一个随机数 r , 将个体 v_1 中 r 左侧位的工件保持不变, 右侧的工件按照 v_2 中的顺序进行调整, 对 v_2 作相同的变换。

3.6 HDEA 混合算法流程

综上所述, 本文中 HDEA 混合算法步骤如下:

- a) 初始化算法的相关参数,如设置最大进化代数 N_{max} ,种群规模 m_0 等。
- b) 根据 3.1 节中的方法,对 JSS 问题进行编码,并构建初始种群。
- c) 根据 3.3 节中的方法,对上述群体进行分裂、水平选择操作。
- d) 根据 3.4 节中的方法,对上述群体进行变异和垂直选择操作。
- e) 根据 3.5 节中的方法,对上述群体进行交叉操作。
- f) 若 $N < N_{max}$ 的,则 $N = N + 1$,转步骤 c);若 $N = N_{max}$ 则终止算法。
- g) 输出最终解。

在 HDEA 的每次迭代中,个体的进化操作包括分裂、水平选择、变异、垂直选择和交叉五个环节,选择算子中个体适应度函数计算公式见 3.2 节。DEA 的优势在于个体自身的进化,使得个体在解空间的局部区域内进行深度搜索,而遗传算法的交叉算子,使个体能够跳出局部极值,有效地解决了早熟收敛的问题。

4 案例分析

本文给出一个 3×3 的模糊 JSS 问题的例子,每个工序在相应机器上的加工时间如表 1 所示。HDEA 的参数设置为: $m_0 = 50, N_{max} = 300, p_c = 0.5$,采用 VC#2005 编程,运行 10 次,在 190 代内都可以收敛并得到最优目标值,其中 7 次得到最优值。最小完工时间为 (16, 18, 22, 24),最优 $E[C_{max}] = 20.0$ 。在机器 M_1 上,各操作的执行顺序为 $O_{12} \rightarrow O_{22} \rightarrow O_{21}$;在机器 M_2 上,操作执行顺序为 $O_{11} \rightarrow O_{23}$;在机器 M_3 上,操作执行顺序为 $O_{13} \rightarrow O_{32} \rightarrow O_{31}$ 。

表 1 3×3 JSS 问题的模糊加工时间表

	M_1	M_2	M_3
J_1	O_{11} (1.6, 1.8, 2.2, 2.4)	O_{12} (4.8, 5.4, 6.6, 7.2)	O_{13} (11.2, 12, 15.4, 16.8)
	O_{21} (4.8, 5.4, 6.6, 7.2)	O_{22} (3.2, 3.6, 4.4, 4.8)	O_{23} (9.6, 10.8, 13.2, 14.4)
	O_{31} (9.6, 10.8, 13.2, 14.4)	O_{32} (1.6, 1.8, 2.2, 2.4)	O_{33} (6.4, 7.2, 8.8, 9.6)
J_2	O_{12} (1.6, 1.8, 2.2, 2.4)	O_{22} (8.0, 9.0, 11.0, 12.0)	O_{32} (4.8, 5.4, 6.6, 7.2)
	O_{22} (3.2, 3.6, 4.4, 4.8)	O_{23} (6.4, 7.2, 8.8, 9.6)	O_{31} (14.4, 16.2, 19.8, 21.6)
	O_{32} (14.4, 16.2, 19.8, 21.6)	O_{31} (1.6, 1.8, 2.2, 2.4)	O_{33} (3.2, 3.6, 4.4, 4.8)
J_3	O_{13} (1.6, 1.8, 2.2, 2.4)	O_{23} (14.4, 16.2, 19.8, 21.6)	O_{33} (4.8, 5.4, 6.6, 7.2)
	O_{23} (8.0, 9.0, 11.0, 12.0)	O_{33} (3.2, 3.6, 4.4, 4.8)	O_{31} (6.4, 7.2, 8.8, 9.6)

本文还选择了具有更大规模的 8×8 的 JSS 实例^[1],将其中的操作时间 t 由确定值调整为梯形模糊数 ($\delta \times t, \theta \times t, \psi \times t, \varphi \times t$),令 $\delta = 0.85, \theta = 0.95, \psi = 1.05, \varphi = 1.15$,从而构造出

8×8 模糊 JSS 问题。分别采用 HDEA 和 GA 进行求解,单点交叉概率 0.85,换位变异概率 0.09。为便于比较,设置 HDEA 与 GA 种群规模和最大进化代数相同,分别运行 20 次。实验表明,与标准 GA 相比,DEA-GA 具有较快的收敛速度,可以在 290 代内获得具有理想制造跨度的调度方案,同时不存在 GA 容易早熟的问题,如表 2 所示。

表 2 HDEA、GA 求解 8×8 JSS 问题性能比较

算法	最差 $E[C_{max}]$	最优 C_{max}	最优 $E[C_{max}]$	最大收敛代数	最优值比率
HDEA	27.3	(18.7, 20.9, 23.2, 25.3)	22.1	290	68.3%
GA	28.1	(18.6, 21.3, 23.7, 25.5)	23.2	310	57.1%

5 结束语

本文对具有模糊加工时间的 JSS 问题进行了研究,用梯形模糊数来表征时间参数,并给出了相应的目标函数。在 JSS 问题求解方面,本文采用集成 GA 和 DEA 的混合进化算法,取得了较理想的结果。DEA 是单亲进化算法,其分裂算子和变异算子使个体的进化过程具备良好的连续性,适合局部搜索,而 GA 是基于种群的优化算法。本文的 HDEA 算法将单亲繁殖和种群交叉的优势结合起来,有效提高了算法的性能。仿真实验表明,与遗传算法相比,HDEA 有更好的全局性和鲁棒性,尤其在求解较大规模问题时,HDEA 收敛性能的优势更为明显。笔者还将对此作进一步的研究。

参考文献:

- [1] CACEM I, HAMMADI S. Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems [J]. IEEE Trans on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2002, 32(1): 1-13.
- [2] CHEN Hao-xun, LHLOW J, LEHMANN C. A genetic algorithm for flexible job shop scheduling [C]//International Conference on Robotics & Automation. 1999.
- [3] 余文,李人厚. 一种有效的双向进化算法 [J]. 小型微型计算机系统, 2003, 24(3): 527-530.
- [4] 牛群,顾幸生. 基于 DNA 进化算法的 Flow shop 生产调度问题 [J]. 上海大学学报, 2004, 10(S): 88-92.
- [5] 牛群,顾幸生. 基于 DNA 进化算法的车间作业调度问题研究 [J]. 控制与决策, 2005, 20(10): 1157-1160.
- [6] LIU Bao-ding, LIU Yian-kui. Expected value of fuzzy variable and fuzzy expected value model [J]. IEEE Trans on Fuzzy Systems, 2002, 10(4): 445-450.
- [7] 曾万聃,周绪波,戴勃. 关联规则挖掘的矩阵算法 [J]. 计算机工程, 2006, 32(2): 45-47.
- [8] 彭仪普,熊拥军. 关联规则挖掘 AorionTrid 算法优化研究 [J]. 计算机工程, 2006, 32(5): 55-57.
- [9] 刘勇国,李学明,张伟. 基于遗传算法的特征子集选择 [J]. 计算机工程, 2003, 29(6): 67-70.
- [10] 许国艳,史宇清. 遗传算法在关联规则挖掘中的应用 [J]. 计算机工程, 2002, 23(7): 122-124.
- [11] 张宗平. 一种更新关联规则的方法 [J]. 计算机工程, 2008, 34(1): 64-65, 68.
- [12] 陈熔. 数据挖掘技术在课程相关性中的应用研究 [J]. 西昌学院学报:自然科学版, 2007, 21(2): 67-69.
- [13] 刘红梅. 关联规则在学生成绩分析中的应用 [J]. 长江大学学报:自然科学版, 2008, 12(5): 357-359.

(上接第 2930 页)

- [3] PARK J S, CHEN M S. An effective hash based algorithm for mining association rules [C]//Proc of International Conference on Management of Data. New York: ACM Press, 1995: 175-186.
- [4] STAKOVIE J. Misconceptions about real-time computing; a serious problem for next generation system [J]. IEEE Computer, 1998, 21(10): 10-19.
- [5] BRIN S, MOTWANI R. Dynamic item set counting and implication rules for market basked data [C]//Proc of ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 1997: 255-264.
- [6] 赵方方,刘万军,陈芳元. 遗传算法在关联规则挖掘中的应用研究 [J]. 沈阳理工大学学报, 2006, 25(4): 51-54.