加工时间离散可控作业车间调度问题分解方法

牛刚刚^{①②*}, 孙树栋^①, Lafon Pascal^②, 杨宏安^①

① 西北工业大学系统集成与工程管理研究所, 现代设计与集成制造技术教育部重点实验室, 西安 710072;

② 特鲁瓦工程技术大学 LASMIS 实验室, 特鲁瓦 10010, 法国

* E-mail: ganggang.niu@hotmail.com

收稿日期: 2010-05-04; 接受日期: 2010-11-26

国家自然科学基金(批准号: 51075337, 50705076, 50705077)和陕西省自然科学基金(批准号: 2009JQ9002)资助项目

摘要 给出了加工时间离散可控的作业车间调度问题(job-shop scheduling problem with discretely controllable processing times, JSP-DCPT) 基于析取图的模型及其数学描述. 提出一种三步分解方法, 使得 JSP-DCPT 可以通过求解一个作业车间调度问题(job-shop scheduling problem, JSP)以及一系列离散时间-成本权衡问题得到解决. 为简化分解方法, 引入时间-成本相平面描述离散时间-成本权衡问题的权衡点, 并详细阐释了一种基于极限模式的集合支配理论以确定用于消减 JSP-DCPT 分解所得离散时间-成本权衡问题的一个上界. 随后提出一种基于极限模式的集合支配分解算法 EMSDDA (extreme mode-based set dominant decomposition algorithm). 基于 JSP 标准算例 FT10 构造了测试算例 JSPDCPT_FT10, 实验仿真验证了提出的理论及分解方法的有效性.

关键词

作业车间调度
离散可控加工时间
时间-成本权衡
三步分解方法

研究领域大多数确定性调度问题的工序加工时间都是常数, 然而, 在实际生产系统中, 工序加工时间受多种资源(如资金、加班、电力、转包、以及人力等)分配数量的影响和控制, 工序加工时间往往具有有限数量的多个可能取值^[1].

加工时间离散可控的作业车间调度问题 (JSP with discretely controllable processing times, JSP-DCPT) 是经典 JSP 的一类扩展问题, 增加有限且离散可分的加工资源, 工序加工时间能够缩短. JSP-DCPT 的解由两部分构成, 一是 JSP 排序方案, 即确定每台机器上工件的排序; 二是控制方案, 即确定具有离散可控加工时间的相关工序的加工模式. JSP-DCPT 解的优劣通过基于时间与成本综合指标的最小化程度来评价. JSP-DCPT 至少有一道工序的加工模式多于一种. 令 x 表示所有工序不同加工模式的组

合数, 则 JSP-DCPT 可行解空间的大小是对应 JSP 解空间大小的 x 倍. JSP-DCPT 的复杂度至少与 JSP 的复杂度相同, 而后者复杂度为 NP-hard^[2], 因此, 寻找 JSP-DCPT 的最优解非常困难.

考虑到工程实际对高效求解 NP-hard 问题的需求, 一种被广泛应用求解 JSP-DCPT 的方法是多项式时间近似 (polynomial time approximation scheme, PTAS). PTAS 是一种近似方法, 通过求解对应原优化问题的一个近似问题, 在多项式时间里所产生的最优解是原问题最优解的 ε ($\varepsilon > 0$) 倍. 关于 PTAS 的研究文献大多针对 JSP-DCPT 的特定简化问题, 如速度可控双机混合车间调度^[3], 速度可控双机流水车间调度^[4], 加工时间离散可控的单机调度^[5], 加工时间离散可控的非相关平行机调度^[6]. Jansen 等人^[7]提出一种能在线性时间内求解 JSP-DCPT 的 PTAS, 但计算

代价很大. 近似方法并非总能适用于工程实际, 原因是这类方法通常用到整数规划、线性规划、半定性求解工具、以及复杂数据结构或者复杂算法, 所有这些都导致实施困难. 另外, 随着 ε 的减小 PTAS 的多项式级数急速增加, 尽管某些近似方法能在多项式时间里求解问题, 但问题求解时间远超出实际允许.

分解方法因能有效降低求解难度, 提高求解效率, 已有将分解方法与启发式方法结合求解多种复杂 JSP 的相关研究^[8~10]. 对包含人力资源安排的复杂 JSP, Artigues 等人^[8]通过分析调度解支配关系, 展示了基于分解的问题表示方法相对于直接表示方法在理论上的创新性. 对于包含平行机的作业车间柔性制造系统调度, Lee 等人^[9]将问题分解为路径选择和 JSP 两个子问题, 并给出了 2 种求解算法. Agnetis 等人^[10]在求解柔性制造单元工件工具联合调度问题时发现, 给定两台机器上的工件排序, 工具调度问题能转换为两工件 JSP 问题求得最优解, 受此启发, 问题分解策略被用于工件工具联合调度问题求解.

本文结构安排如下: 第 1 节给出了 JSP-DCPT 的模型及其数学描述; 第 2 节提出一种三步分解方法求解 JSP-DCPT, 并就该方法的简化进行了分析, 得到几个理论结果; 第 3 节提出一种基于极限模式的集合支配分解算法(EMSDDA)求解 JSP-DCPT; 第 4 节基于 JSP 标准算例 FT10 构造测试算例, 验证了本文提出的有关理论和方法; 第 5 节对全文进行了总结.

1 模型

JSP-DCPT 的可行解由 JSP 排序和工序加工模式向量确定. 令 σ , δ 分别表示 JSP 排序方案和工序加工模式的有序集合, n, m 分别表示 JSP-DCPT 的工件数和机器数, O_{ij} ($i = 1, \dots, n; j = 1, \dots, m$) 表示工件 J_i 在机器 M_j 上的加工工序, x_{ij} ($i = 1, \dots, n; j = 1, \dots, m$) 表示工序 O_{ij} 所选的加工模式.

本文结合两种方法描述 JSP-DCPT 模型: ①描述 JSP 的析取图 (Disjunctive graph)^[11]; ②描述项目网络图的“结点-活动”表示法 (Activity-on-node representation)^[12]. 图 1 给出了 JSP-DCPT 的模型示意图.

图 1 中, O_{ij} 的加工模式 (即 x_{ij}) 标注在 O_{ij} 对应结点的右上角. 从而, JSP-DCPT 可行解的两部分 (即 JSP 调度方案 σ 和工序加工模式的有序集合 δ), 可以分别由各台机器上析取弧的方向以及由一个包括所有工序加工模式的集合来描述.

在对 JSP-DCPT 进行数学描述之前, 先将要用到的符号定义如下:

- σ JSP 排序方案;
- δ 所有工序的加工模式的有序集合;
- $T(\sigma, \delta)$ 由 σ 和 δ 确定的 JSP-DCPT 解的 makespan;
- $C(\delta)$ 由 δ 确定的 JSP-DCPT 解的控制成本;

图 1 JSP-DCPT 模型示意图

n	工件数;
m	机器数;
J_i	工件 i ;
M_j	机器 j ;
O_{ij}	工件 J_i 在机器 M_j 上的加工工序;
s_{ij}	工序 O_{ij} 的开始加工时间;
x_{ij}	工序 O_{ij} 的加工模式;
l_{ij}	工序 O_{ij} 不同加工模式的数量;
$t_{ij}(x_{ij})$	工序 O_{ij} 在模式 x_{ij} 的加工时间;
$c_{ij}(x_{ij})$	工序 O_{ij} 在模式 x_{ij} 的控制成本;
$\mu_{ij,1}$	工序 O_{ij} 的正常加工时间;
$\mu_{ij,l_{ij}}$	工序 O_{ij} 的极限压缩加工时间.

本文研究时间-成本双目标优化 JSP-DCPT, 优化目标为 $T(\sigma, \delta)$ 和 $C(\delta)$. 通过扩展 Graham 等人^[13]提出的调度问题的 3 元组表示方法 $\alpha|\beta|\gamma$, 本文所研究的 JSP-DCPT 表示为 $J_m|DCPT|(T(\sigma, \delta), C(\delta))$. 这里, J_m 表示作业车间, $DCPT$ 是离散可控加工时间的缩写, $(T(\sigma, \delta), C(\delta))$ 是要最小化的目标. JSP-DCPT 的数学描述如下:

$$\min(\max(s_{ij} + t_{ij}(x_{ij})), \sum c_{ij}(x_{ij})). \quad (1)$$

约束条件:

$$s_{ij} \geq s_{ik} + t_{ik}, \quad (2)$$

for all $i = 1, \dots, n; j, k = 1, \dots, m; \text{ and } j \neq k.$

$$s_{ij} \geq s_{hj} + t_{hj}, \quad (3)$$

for all $i, h = 1, \dots, n; j = 1, \dots, m; \text{ and } i \neq h.$

$$t_{ij}(x_{ij}) = \mu_{ij,x_{ij}}, \mu_{ij,x_{ij}} \in \{\mu_{ij,1}, \mu_{ij,2}, \dots, \mu_{ij,l_{ij}}\},$$

$$\text{and } \mu_{ij,1} > \mu_{ij,2} > \dots > \mu_{ij,l_{ij}} \geq 0, \quad (4)$$

for all $i = 1, \dots, n; j = 1, \dots, m; \text{ and } x_{ij} = 1, 2, \dots, l_{ij}.$

(1)式给出了 JSP-DCPT 的优化目标, 最小化 makespan (即 $\max(s_{ij} + t_{ij}(x_{ij}))$) 和加工控制成本 (即 $\sum c_{ij}(x_{ij})$); (2)式保证了工件 J_i 在机器 M_j, M_k 上加工工序的顺序约束, 即工序 O_{ik} 先于其工件紧后工序 O_{ij} 加工; (3)式保证了机器 M_j 上加工工件的顺序约束, 即设备 M_j 上的工序 O_{hj} 先于其设备紧后工序 O_{ij} 加工; (4)式保证了 O_{ij} 的加工时间由其所选加工模式 x_{ij} 确定, 且仅能是给定离散加工时间排序集合中的一个值.

2 方法

JSP-DCPT 的可行解由 JSP 排序和工序加工模式的有序集合确定的特点启发了一种求解 JSP-DCPT 的三步分解方法.

Step1. 设定 JSP-DCPT(记作 P)的所有工序为极限压缩加工模式, 求解相应的 JSP(记作 PI), 得到 JSP 的全部解.

Step2. 对于每个 JSP 解 σ_i , 求解对应的离散时间-成本权衡问题(记作 $PII(\sigma_i)$), 得到 $PII(\sigma_i)$ 的最优加工模式的有序集合.

Step3. 组合 JSP 排序方案 σ_i 与相应的最优加工模式的有序集合, 得到 JSP-DCPT 的解集; 寻找所得解集中的全部非支配解, 即得到 JSP-DCPT 的帕累托解集.

考虑到大规模 JSP-DCPT 的解空间巨大且结构复杂, 高额计算成本会导致上述分解方法在实际中不能很好应用. 下面对 JSP-DCPT 三步分解法的简化展开分析.

给定 PI 的一个解 σ_i , 对应的离散时间-成本权衡问题记作 $PII(\sigma_i)$. s_j 表示 $PII(\sigma_i)$ 的任意一个权衡点, 特别地, s_0 和 s_k 分别表示所有工序均设置为正常模式和极限压缩模式时的权衡点; δ_j 是对应权衡点 s_j 的所有工序的加工模式的有序集合; t_j 和 c_j 分别表示 $PII(\sigma_i)$ 对应权衡点 s_j 的 makespan 与成本.

引入时间-成本相平面来描述 $PII(\sigma_i)$ 的权衡点. 时间-成本相平面上的点 $P_j(c_j, t_j)$ 对应 $PII(\sigma_i)$ 的权衡点 s_j . 特别地, $P_0(c_0, t_0)$ 和 $P_k(c_k, t_k)$ 分别对应 $PII(\sigma_i)$ 的权衡点 s_0 和 s_k . 对于正常模式控制成本 c_0 , 由于不同的初始值只影响到时间-成本权衡问题的帕累托边界(Pareto frontier)在时间-成本相平面上的水平偏移量, 而对帕累托边界上最优解的数量及分布没有影响. 为减少问题变量, 文中假设 $c_0=0$.

图 2 是时间-成本相平面上一个连续权衡过程 $s_0, s_1, \dots, s_k$ 的示意图.

定义 1. 工序压缩(松弛)比 $r_{o_{ij}}(x_{ij}, x'_{ij})$.

对于工序 O_{ij} 从加工模式 x_{ij} 到加工模式 x'_{ij} 的一个压缩(松弛)过程, 工序压缩(松弛)比定义为

$$r_{o_{ij}}(x_{ij}, x'_{ij}) = \frac{t_{ij}(x_{ij}) - t_{ij}(x'_{ij})}{c_{ij}(x_{ij}) - c_{ij}(x'_{ij})},$$

图2 时间-成本相平面上一个连续权衡过程的示意图

$i=1,2,\dots,n; j=1,2,\dots,m; \text{且 } x_{ij} \neq x'_{ij}.$

本文假设对任意工序 O_{ij} , $r_{O_{ij}}(x_{ij}, x'_{ij}) \equiv \alpha$.

定义 2. 调度方案压缩(松弛)比 $r_{\sigma_i}(\delta_u, \delta_v)$.

对于调度方案 σ_i 从权衡点 s_u 到 s_v 的一个连续压缩(松弛)过程, 调度方案压缩(松弛)比定义为

$$r_{\sigma_i}(\delta_u, \delta_v) := \frac{|t_u - t_v|}{|c_u - c_v|}.$$

这里, δ_u 和 δ_v 分别是对应权衡点 s_u 和 s_v 的所有工序的加工模式的有序集合, $\delta_u, \delta_v = \{x_{ij} | i=1, \dots, n; j=1, \dots, m\}$, 且 $\delta_u \neq \delta_v$. t_u 和 t_v 分别是 $PII(\sigma_i)$ 对应权衡点 s_u 和 s_v 的 makespan, c_u 和 c_v 分别是 $PII(\sigma_i)$ 对应权衡点 s_u 和 s_v 的控制成本.

考虑到仅当压缩(松弛)关键工序(关键路径上的工序)且关键工序压缩(松弛)后关键路径不发生改变时, 调度方案压缩(松弛)比等于工序压缩(松弛)比; 而对于其他情形, 调度方案压缩(松弛)比恒小于工序压缩(松弛)比. 因此, 得到推论如下.

推论 1. $r_{\sigma_i}(\delta_u, \delta_v) \leq \alpha$.

考虑时间-成本权衡问题 $PII(\sigma_i)$, s_0 和 s_k 分别是所有工序均设置为正常模式和极限压缩模式的权衡点, t_{σ_i} 和 t'_{σ_i} 分别表示 $PII(\sigma_i)$ 对应权衡点 s_k 和 s_0 的 makespan, c_{σ_i} 表示 $PII(\sigma_i)$ 对应权衡点 s_k 的控制成本. 令 s'_k 表示满足如下条件的一个虚拟权衡点: $PII(\sigma_i)$ 对应 s'_k 的 makespan 等于 t_{σ_i} , 且调度方案 σ_i 从权衡点

s_0 到 s'_k 的压缩过程对应的调度压缩比等于工序压缩比.

定义 3. 最小化 makespan 的成本下界 $c_{t_{\sigma_i}}$.

$PII(\sigma_i)$ 在权衡点 s'_k 的控制成本为其最小化 makespan 的成本下界, 记作 $c_{t_{\sigma_i}}$.

令 s'_0 表示满足如下条件的另一个虚拟权衡点:

$PII(\sigma_i)$ 对应 s'_0 的 makespan 等于 t'_{σ_i} , 且调度方案 σ_i

从权衡点 s_k 到 s'_0 的松弛过程对应的调度方案松弛比等于工序松弛比. Δc 表示这一松弛过程减少的成本, $c_{t_{\sigma_i}}$ 表示 $PII(\sigma_i)$ 对应 s_k 的控制成本.

定义 4. 最大化 makespan 的成本上界 $c'_{t_{\sigma_i}}$.

$c'_{t_{\sigma_i}}$ 为 $PII(\sigma_i)$ 最大化 makespan 的成本上界,

$$c'_{t_{\sigma_i}} = c_{\sigma_i} - \Delta c.$$

给定 $PII(\sigma_i)$, 最小化 makespan 的成本下界($c_{t_{\sigma_i}}$)和最大化 makespan 的成本上界($c'_{t_{\sigma_i}}$)在时间-成本相平面上标示如图 3. 此外, 相平面上的点 $P_0(0, t'_{\sigma_i})$ 和 $P_k(c_{\sigma_i}, t_{\sigma_i})$ 分别对应极限权衡点 s_0 和 s_k . $P'_0(c'_{t_{\sigma_i}}, t'_{t_{\sigma_i}})$ 对应一个虚拟极限权衡点, 其 makespan 等于 $PII(\sigma_i)$ 在权衡点 s_0 的 makespan, 控制成本等于 $PII(\sigma_i)$ 最大化 makespan 的成本上界. $P'_k(c_{t_{\sigma_i}}, t_{\sigma_i})$ 对应另一个虚拟极限权衡点, 其 makespan 等于 $PII(\sigma_i)$ 在权衡点 s_k 的

图3 时间-成本相平面上的极限权衡点示意图

makespan, 控制成本等于 $PII(\sigma_i)$ 最小化 makespan 的成本下界. 因此, $P_0(0, t_{\sigma_i}')$ 、 $P_0'(c'_{t_{\sigma_i}}, t_{t_{\sigma_i}}')$ 、 $P_k(c_{\sigma_i}, t_{\sigma_i})$ 和 $P_k'(c_{t_{\sigma_i}}, t_{\sigma_i})$ 构成平行四边形 $\square P_0 P_0' P_k P_k'$. 上述极限权衡点在时间-成本相平面上的位置见图 3.

令 $\mathbb{S}_i$ 表示 $PII(\sigma_i)$ 的权衡空间. 下面提出一个引理, 以确定 $PII(\sigma_i)$ 的权衡空间 $\mathbb{S}_i$ 的边界.

引理 1. $\square P_0 P_0' P_k P_k'$ 构成 $\mathbb{S}_i$ 的边界.

下面结合时间-成本相平面上的几何分析(见图 4)完成引理 1 的证明.

证明. 点 $P_j(c_j, t_j)$ 对应权衡空间 $\mathbb{S}_i$ 中的权衡点 s_j . $\mathbb{A}$, $\mathbb{B}$, $\mathbb{C}$ 和 $\mathbb{D}$ 分别表示 $\square P_0 P_0' P_k P_k'$, $\triangle P_0 P_k' A$, $\triangle P_0' D P_k$ 和 $\square P_0 D P_k A$ 中所有点的集合.

由 $0 \leq c_j \leq c_{\sigma_i}$ 和 $t_{\sigma_i} \leq t_j \leq t_{\sigma_i}'$, 得

$$P_j(c_j, t_j) \in \mathbb{D}. \quad (5)$$

δ_0 和 δ_j 分别是对应权衡点 s_0 和 s_j 的工序加工模式的有序集合. 在图 4 中, $\tan \theta_2 = r_{\sigma_i}(\delta_0, \delta_j)$, $\tan \theta_1 = \alpha$. 由推论 1, $r_{\sigma_i}(\delta_0, \delta_j) \leq \alpha$. 因此, $\tan \theta_2 \leq \tan \theta_1$, 进而得到 $\theta_2 \leq \theta_1$. 所以

$$P_j(c_j, t_j) \notin \mathbb{B}. \quad (6)$$

接下来应用反证法证明 $P_j(c_j, t_j) \notin \mathbb{C}$.

假设 $P_j(c_j, t_j) \in \mathbb{C}$ 且 $P_j(c_j, t_j)$ 与 P_j' 所处位置重合. 由于调度方案可以从任意的权衡点 $P_j(c_j, t_j)$ 压缩到极限压缩权衡点 $P_k(c_{\sigma_i}, t_{\sigma_i})$, 因此, 由图 4 可得

图 4 引理 1 证明用图

$r_{\sigma_i}(\delta_j, \delta_k) = \tan \theta_4 > \tan \theta_3 = \alpha$. 进一步得到 $r_{\sigma_i}(\delta_j, \delta_k) > \alpha$. 此结论显然与推论 1 的 $r_{\sigma_i}(\delta_j, \delta_k) \leq \alpha$ 矛盾. 因此,

$$P_j(c_j, t_j) \notin \mathbb{C}. \quad (7)$$

由 $\mathbb{D} = \mathbb{A} \cup \mathbb{B} \cup \mathbb{C}$ 以及 (5), (6) 和 (7) 式得 $P_j(c_j, t_j) \in \mathbb{A}$, 即 $P_j(c_j, t_j) \in \square P_0 P_0' P_k P_k'$. 因此, $\square P_0 P_0' P_k P_k'$ 构成 $\mathbb{S}_i$ 的边界.

定义 5. 点支配关系 $\succ$.

给定点 $P_1(c_1, t_1)$ 和点 $P_2(c_2, t_2)$, 如果满足 $c_1 < c_2$, $t_1 \leq t_2$ 或者 $c_1 \leq c_2$, $t_1 < t_2$, 则 P_1 完全支配 P_2 , 记作 $P_1 \succ P_2$.

定义 6. 集合支配关系 $\succ_s$.

给定集合 $\mathbb{A}_1$ 和集合 $\mathbb{A}_2$, 如果满足 $\forall P_2 \in \mathbb{A}_2$, $\exists P_1 \in \mathbb{A}_1$, $P_1 \succ P_2$, 并且 $\forall P_1 \in \mathbb{A}_1$, $\nexists P_2 \in \mathbb{A}_2$, $P_2 \succ P_1$, 则 $\mathbb{A}_1$ 完全支配 $\mathbb{A}_2$, 记作 $\mathbb{A}_1 \succ_s \mathbb{A}_2$.

下面提出一个引理, 给出集合支配关系成立的充分条件. 如前所述, $\mathbb{S}_i$ 表示 $PII(\sigma_i)$ 的权衡空间, s_0 和 s_k 分别表示所有工序均为正常模式与极限压缩模式的权衡点, t_{σ_i} 和 t_{σ_i}' 分别是 $PII(\sigma_i)$ 对应 s_k 和 s_0 的 makespan.

引理 2. 对于 $PII(\sigma_1)$ 和 $PII(\sigma_2)$, 如果 $t_{\sigma_1}' < t_{\sigma_2}$, 则 $\mathbb{S}_1 \succ_s \mathbb{S}_2$.

下面结合时间-成本相平面上的几何分析(见图 5)完成引理 2 的证明.

证明. $A_1(0, t_{\sigma_1}')$, $B_1(c'_{t_{\sigma_1}}, t'_{t_{\sigma_1}})$, $C_1(c_{\sigma_1}, t_{\sigma_1})$ 和 $D_1(c_{t_{\sigma_1}}, t_{\sigma_1})$ 构成平行四边形 $\square A_1 B_1 C_1 D_1$. $A_2(0, t'_{\sigma_2})$, $B_2(c'_{t_{\sigma_2}}, t'_{t_{\sigma_2}})$, $C_2(c_{\sigma_2}, t_{\sigma_2})$ 和 $D_2(c_{t_{\sigma_2}}, t_{\sigma_2})$ 构成平行四边形 $\square A_2 B_2 C_2 D_2$. 由引理 1, $\square A_1 B_1 C_1 D_1$ 和 $\square A_2 B_2 C_2 D_2$ 分别构成 $\mathbb{S}_1$ 和 $\mathbb{S}_2$ 的边界.

$\forall P_i(c_i, t_i) \in \square A_2 B_2 C_2 D_2$, $0 < c_i$ 和 $t_{\sigma_1}' < t_i$ 成立. 由定义 5, $A_1(t_{\sigma_1}', 0) \succ P_i(t_i, c_i)$. 即

$$\begin{aligned} \forall P_i(c_i, t_i) \in \square A_2 B_2 C_2 D_2, \exists A_1(0, t_{\sigma_1}') \in \square A_1 B_1 C_1 D_1, \\ A_1(0, t_{\sigma_1}') \succ P_i(c_i, t_i). \end{aligned} \quad (8)$$

因为 $t_{\sigma_1}' < t_{\sigma_2}$, 所以 $\forall P_i(t_i, c_i) \in \square A_2 B_2 C_2 D_2$

图5 引理2证明用图

and $\forall P_j(t_j, c_j) \in \square A_1 B_1 C_1 D_1$, $t_i > t_j$. 也就是说, $\forall P_j(t_j, c_j) \in \square A_1 B_1 C_1 D_1$, $\nexists P_i(t_i, c_i) \in \square A_2 B_2 C_2 D_2$, $P(t_i, c_i)_i \succ P_j(t_j, c_j)$. (9)

由(8)和(9)式, 以及定义6, 得 $S_1 \succ_s S_2$.

定义7. 基于极限模式的集合支配关系 $\succ_e$.

定义满足引理2的集合支配关系为基于极限模式的集合支配关系, 记作 $\succ_e$.

令 A 表示求解 JSP-DCPT 对应 JSP PI 得到的一个排序集合, $A = \{\sigma_i | i = 1, 2, \dots, u, \dots, v\}$, v 是集合 A 所

有元素的个数. t_{σ_i} 和 t_{σ_i}' 分别是 $PII(\sigma_i)$ 对应权衡点 s_k 和 s_0 的 makespan, S_i 是 $PII(\sigma_i)$ 的权衡空间. 令 $F(S_i)$ 表示权衡空间 S_i 的帕累托边界, 亦即 S_i 中所有非支配权衡点的集合.

定理1. 给定 JSP-DCPT 对应 JSP 的排序集合 $A = \{\sigma_i | i = 1, 2, \dots, u, \dots, v\}$, 如果 $t_{\sigma_1} < t_{\sigma_2} < \dots < t_{\sigma_u} < \dots < t_{\sigma_v}$ 且 $t_{\sigma_1}' < t_{\sigma_u}$, 则 $F(S_1 \cup S_2 \cup \dots \cup S_u \cup \dots \cup S_v) = F(S_1 \cup S_2 \cup \dots \cup S_{u-1})$.

证明. 由 $t_{\sigma_1}' < t_{\sigma_u}$ (已给出) 以及 $t_{\sigma_u} < t_{\sigma_{u+j}}$ (对 $j = 1, 2, \dots, v-u$), 得 $t_{\sigma_1}' < t_{\sigma_{u+j}}$ (对 $j = 1, 2, \dots, v-u$). 由引理2和定义7, 得

$$S_1 \succ_e S_{u+j} \quad (\text{for } j = 0, 1, 2, \dots, v-u). \quad (10)$$

由(10)式推出, S_{u+j} ($j = 0, 1, 2, \dots, v-u$) 中得任

意权衡点都被 S_1 中至少一个权衡点所支配. 因此, $S_1 \cup S_2 \cup \dots \cup S_u \cup \dots \cup S_v$ 的非支配权衡点与 $S_1 \cup S_2 \cup \dots \cup S_{u-1}$ 的非支配权衡点相同, 即 $F(S_1 \cup S_2 \cup \dots \cup S_u \cup \dots \cup S_v) = F(S_1 \cup S_2 \cup \dots \cup S_{u-1})$.

定理1表明, 应用本文提出的三步分解法求解 JSP-DCPT 时并非所有的离散时间-成本权衡问题都需要求解, 满足 $t_{\sigma_1} < t_{\sigma_u}$ 的所有离散时间-成本权衡问题 $PII(\sigma_i)$ 都能被消减, 从而简化 JSP-DCPT 求解过程. 随后给出两个推论.

推论2. 应用三步分解法求解 JSP-DCPT 时, 满足 $t_{\sigma_1} < t_{\sigma_u}$ 的所有离散时间-成本权衡问题 $PII(\sigma_i)$ 都能被消减.

推论3. t_{σ_1}' 是 t_{σ_u} 的一个上界.

推论3得自于仅考虑离散时间-成本权衡问题 $PII(\sigma_i)$ 的两个极限权衡点(即所有工序分别设置为正常模式与极限压缩模式对应的权衡点)的情形. 对于多于两个权衡点的情形, 由于新增的权衡点可能是非支配权衡点, 因此, $PII(\sigma_i)$ 包含所有非支配权衡点的权衡空间能够进一步缩小, 相应地, 所需求解的离散时间-成本权衡问题可以进一步消减.

3 算法

提出一种基于极限模式的集合支配分解算法 EMSDDA (extreme mode-based set dominant decomposition algorithm). 在不损失帕累托解的前提下通过减小问题权衡空间, EMSDDA 能够简化 JSP-DCPT 问题求解. EMSDDA 嵌套了一种禁忌搜索模拟退火混合算法 TSSA^[14]和一种快速精英保留非支配排序遗传算法 NSGA-II^[15], 以分别高效求解 JSP-DCPT 分解得到的 JSP 子问题和离散时间-成本权衡子问题. 关于 TSSA 与 NSGA-II 的算法设计, 详见文献[14, 15].

算法 EMSDDA.

Step1. 对 JSP-DCPT 问题, 设置全部工序为极限压缩模式, 应用 TSSA 算法求解对应的 JSP 问题 PI . 得到满足条件 $t_{\sigma_1} < t_{\sigma_2} < \dots < t_{\sigma_u} < \dots < t_{\sigma_v}$ 的集合 $A = \{\sigma_i | i = 1, 2, \dots, u, \dots, v\}$.

Step2. 设置整数 $i = 0$; 并设置集合 $B = \emptyset$.

Step3. Do:

Step3.1. $i = i + 1$.

Step3.2. 应用 NSGA-II 算法求解离散时间-成本权衡问题 $PII(\sigma_i)$. 将得到的非支配权衡点加入集合 B .

Step3.3. 应用点支配关系对集合 B 进行支配排序.

Step3.4. 删除集合 B 中的支配权衡点.

Until $t_{\sigma_i} > t_{\sigma_i}'$ 或者 $i = v$.

Step4. $t_{\sigma_u} = t_{\sigma_i}$. 返回集合 B , 得到 JSP-DCPT 的所有非支配权衡点.

4 算例

从 JSP 标准测试库 Fisher 和 Thompson 类^[16]中选取 JSP 算例 FT10 (问题规模为 $n \times m = 10 \times 10$), 构造相应 JSP-DCPT 算例 (记作 JSPDCPT_FT10) 以测试第 3 节中提出的算法 EMSDDA.

为简化示例, JSPDCPT_FT10 按如下方式构造: 假设每一道工序仅有两种加工模式: 正常模式和极限压缩模式. 且正常模式下工序加工时间是极限压缩模式下对应时间的 2 倍. 此外, 假设正常模式工序控制成本等于零, 以及工序 O_{ij} 的压缩成本在数值上等于因工序压缩而减少的加工时间数值.

为保证算法效率与效果之间的较好权衡, 采用正交试验设计确定算法 TSSA 和算法 NSGA-II 的参数.

对于 TSSA, 参数 L (禁忌列表长度), $ImproveIter$ (未改进连续迭代数), T (退火初温) 和 $TotIter$ (最大迭代数) 的设定见表 1.

对于 NSGA-II, 采用二进制编码(binary coding)对工序加工模式方案进行编码, 0 表示正常模式, 1 表示极限模式. 种群更新所用的 GA 算子为竞赛选择算子(tournament selection operator)、两点交叉算子(2-point crossover operator)、以及位反转变异算子(bit inversion mutation operator). 参数 $PopSize$ (种群数),

表 1 TSSA 参数设定

Item	Value
L	$10 + \text{jobNumber} / \text{machineNumber}$
$ImproveIter$	2500
T	$\text{bestMakespan} / 300$
$TotIter$	$\text{jobNumber} \times 50000$

$EliNum$ (精英个体数), $CroProb$ (交叉概率), $MutProb$ (变异概率), $Replter$ (终止迭代数)的设定见表 2.

为保证计算结果的充分多样化, 同样采用正交试验设计确定算法 EMSDDA 的参数, 参数 JSP_runs (应用 TSSA 求解子问题 JSP 的运行次数), $Maximal_JSP_solution_number$ (求解子问题 JSP 时可以存储的 JSP 解的最大数目), JSP_DCPT_runs (应用 EMSDDA 求解 JSP-DCPT 的运行次数)的设定见表 3.

EMSDDA 用 C++ 编程实现, 并在 PC (Pentium(R) 2.80 GHz CPU, 2.00G RAM) 上对算例 JSPDCPT_FT10 进行实验仿真. 仿真结果见表 4.

表 4 中, t_{σ_u} 是根据 JSP 标准算例的最优 makespan 计算得到的上界, 本算例对应 JSP 标准算例(FT10)的最优 makespan 为 930, 则对应此最优解的离散时间-成本权衡问题的时间分量的最大值(此时, 所有工序设定为正常模式)为 $930 \times 2 = 1860$, 即 $t_{\sigma_u} = 1860$. $T_{\max}$ 是应用 TSSA 算法求解 JSP 得到的最大 makespan. $Loc t_{\sigma_u}$ 和 $Loc T_{\max}$ 分别是集合 A 中 makespan 等于 t_{σ_u} 和 $T_{\max}$ 的 JSP 解在 A 中的位置. $Num t_{\sigma_u}$ 和 $Num T_{\max}$ 是选

表 2 NSGA-II 参数设定

Item	Value
PopSize	200
EliNum	100
CroProb	0.8
MutProb	0.1
Replter	4

表 3 EMSDDA 参数设定

Item	Value
JSP_runs	20
Maximal_JSP_solution_number	20000
JSP-DCPT_runs	10

表 4 JSPDCPT_FT10 的 10 次仿真结果

Run	t_{σ_u}	$T_{\max}$	$Loc t_{\sigma_u}$	$Loc T_{\max}$	$Num t_{\sigma_u}$	$Num T_{\max}$	Time (s)
1	1860	2208	7246	7271	84	84	5406.7
2	1860	2261	7719	7746	82	82	6041.9
3	1860	2332	7679	7688	83	83	6304.4
4	1860	2382	7680	7708	76	76	5992.5
5	1860	2229	7551	7592	83	83	5775.9
6	1860	2259	7444	7472	93	93	5765.1
7	1860	2198	7465	7486	88	88	5881.5
8	1860	2212	7643	7667	81	81	6142.6
9	1860	2293	7516	7543	93	93	6018.3
10	1860	2257	7548	7572	98	98	5800.6

代条件分别满足 $t_{\sigma_i} > t_{\sigma_u}$ 和 $t_{\sigma_i} > T_{\max}$ 时算法 EMSDDA 得到的累积帕累托解的个数. *Time* 列给出了算例运行时间, 单位为秒.

$Loc\sigma_i$ 表示 JSP 解 σ_i 在集合 A 中的位置, $Numt_{\sigma_i}$ 表示应用算法 EMSDDA 求解离散时间-成本权衡问题 $PII(\sigma_i)$ 后得到的累积帕累托解的个数. 图 6 展示了算例 JSPDCPT_FT10 10 次运行得到的累积帕累托解个数 $Numt_{\sigma_i}$ 随离散时间-成本权衡问题迭代求解数量 $Loc\sigma_i$ 的变化曲线.

对于满足条件 $Loc\sigma_u < Loc\sigma_i \leq LocT_{\max}$ 和 $Numt_{\sigma_u} = Numt_{\sigma_i} = NumT_{\max}$ 的 σ_i , 对应的 $PII(\sigma_i)$ 能够被消减而不损失最优权衡点. 因此, 为验证算法 EMSDDA 能够消减冗余的离散时间-成本权衡问题, 需要检验以下两个式子:

$$Loc\sigma_u < LocT_{\max}. \quad (11)$$

$$\begin{aligned} \forall \sigma_i \in \{ \sigma_i \mid Loc\sigma_u < Loc\sigma_i \leq LocT_{\max} \}, \\ Numt_{\sigma_u} = Numt_{\sigma_i} = NumT_{\max}. \end{aligned} \quad (12)$$

通过分析计算结果, 表 4 数据可验证(11)式. 图 6 可验证(12)式.

此外, 为检验所得帕累托边界的收敛性, 图 7 给出了应用 EMSDDA 10 次求解算例 JSPDCPT_FT10

图 6 JSPDCPT_FT10 的累积帕累托解数量随离散时间-成本权衡问题迭代求解数量的变化曲线

图 7 JSPDCPT_FT10 的帕累托边界

所得的全部帕累托最优权衡点.

由图 7 中帕累托最优权衡点的分布可以得出结论: 对算例 JSPDCPT_FT10, EMSDDA 能够得到收敛的帕累托边界.

5 结论

给出了 JSP-DCPT 问题的模型及数学描述. 提出一种三步分解方法, 使得 JSP-DCPT 可以通过求解一个相应的 JSP 问题以及一系列离散时间-成本权衡问题得到解决. 引入时间-成本相平面描述时间-成本权衡问题的权衡点. 通过将权衡空间进行分区并应用集合支配关系, 理论上证明了应用三步分解法时并不需要求解全部的离散时间-成本权衡问题, 准确地讲, 满足 $t_{\sigma_i} > t_{\sigma_u}$ 的所有离散时间-成本权衡问题

$PII(\sigma_i)$ 都能被消减, 且 t_{σ_i}' 是 t_{σ_u} 的一个上界. 通过融合 TSSA 和 NSGA-II 算法, 提出 EMSDDA 算法求解 JSP-DCPT. 对 JSPDCPT_FT10 算例的仿真结果展示了 EMSDDA 能够消减冗余的离散时间-成本权衡问题并得到收敛的帕累托边界.

对离散时间-成本权衡问题的最优消减取决于能否找到 t_{σ_u} 的最小上界. 本文所得上界 t_{σ_i}' 仅由对应最优 JSP 调度方案的离散时间-成本权衡问题的正常权衡点和极限压缩权衡点确定. 除两个极限权衡点外, 随着更多非支配权衡点的加入, 上界能够进一步减小. 因此, 寻找 t_{σ_u} 的最小上界, 亟待进一步研究.

参考文献

- 1 Shabtay D, Steiner G. A survey of scheduling with controllable processing times. *Discrete Appl Math*, 2007, 155: 1643–1666
- 2 Gonzalez T, Sahni S. Flowshop and jobshop schedules: Complexity and approximation. *Oper Res*, 1978, 26: 36–52
- 3 Ishii H, Masuda T, Nishida T. Two machine mixed shop scheduling problem with controllable machine speeds. *Discrete Appl Math*, 1987, 17: 29–38
- 4 Strusevich V A. Two machine flow shop scheduling problem with no wait in process: Controllable machine speeds. *Discrete Appl Math*, 1995, 59: 75–86
- 5 Chen Z, Lu Q, Tang G. Single machine scheduling with discretely controllable processing times. *Oper Res Lett*, 1997, 21: 69–76
- 6 Grigoriev A, Sviridenko M, Uetz M. LP rounding and an almost harmonic algorithm for scheduling with resource dependent processing times. In: *Approximation, Randomization, and Combinatorial Optimization Algorithms and Techniques*. Heidelberg: Springer Berlin, 2006. 140–151
- 7 Jansen K, Mastrolilli M, Solis-Oba R. Approximation schemes for job shop scheduling problems with controllable processing times. *Eur J Oper Res*, 2005, 167: 297–319
- 8 Artigues C, Gendreau M, Rousseau L M, et al. Solving an integrated employee timetabling and job-shop scheduling problem via hybrid branch-and-bound. *Comput Oper Res*, 2009, 36: 2330–2340
- 9 Lee D H, Yeong-Dae K. Scheduling algorithms for flexible manufacturing systems with partially grouped machines. *Special issue on scheduling: From Research Into Practice*, 1999, 18: 301–309
- 10 Agnetis A, Alfieri A, Brandimarte P, et al. Joint job/tool scheduling in a flexible manufacturing cell with no on-board tool magazine. *Comput Integr Manuf*, 1997, 10: 61–68
- 11 Balas E. Machine sequencing via disjunctive graphs: An implicit enumeration algorithm. *Oper Res*, 1969, 17: 941–957
- 12 Brucker P, Drexl A, Möhring R, et al. Resource-constrained project scheduling: Notation, classification, models, and methods. *Eur J Oper Res*, 1999, 112: 3–41
- 13 Graham R, Lawler E, Lenstra J, et al. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann Discrete Math*, 1979, 5: 287–326
- 14 Zhang C, Li P, Rao Y, et al. A very fast TS/SA algorithm for the job shop scheduling problem. *Comput Oper Res*, 2008, 35: 282–294
- 15 Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE T Evolut Comput*, 2002, 6: 182–197
- 16 Muth J F, Thompson G L. *Industrial Scheduling*. Englewood Cliffs, NJ: Prentice-Hall, 1963