

# 差异分批模式下的联合成本优化问题及算法<sup>①</sup>

程八一<sup>1,2</sup>, 黄小曼<sup>1,2</sup>, 杨艳艳<sup>1,2</sup>, 胡笑旋<sup>1,2</sup>

(1. 合肥工业大学管理学院, 合肥 230009; 2. 过程优化与智能决策教育部重点实验室, 合肥 230009)

**摘要:** 提出了一类制造企业的联合成本优化问题, 将企业的产品制造环节和配送环节进行协同运作, 实现供应链环境下的联合调度. 在生产环节, 考虑一类典型的差异分批制造模式, 即待加工的作业尺寸有差异, 而批处理设备的容量确定, 设备环境为多台并行设备; 在配送环节, 企业采用自有车辆进行运输, 车辆具有相同的运输能力; 若完工的作业在当前无可用车辆进行配送, 则转入产成品库存; 联合成本为生产、库存和配送三阶段的总成本. 本文首先构造了基于整数规划的数学模型, 证明了联合成本的最小化问题是强 NP-hard 问题; 然后设计了多项式时间的近似算法, 分析了算法的时间复杂性, 并证明了算法的求解性能.

**关键词:** 供应链调度; 联合成本; 差异作业; 近似算法

**中图分类号:** TP301      **文献标识码:** A      **文章编号:** 1007-9807(2016)08-0102-11

## 0 引 言

在当前激烈的市场竞争环境下, 制造型企业不再单纯的追求大批量、高速度的生产方式, 而是将重点放在提高作业质量、改善客户服务水平等方面, 统筹企业的采购、库存、生产与配送环节, 获得经济效益的总体优化, 从而提升企业的综合竞争力. 供应链调度便是从制造企业所在供应链的视角出发, 采用精确调度的方法, 考虑面向供应链的多环节优化与调度, 包括制造企业的原材料采购、生产、库存和配送过程. 其思想由 Hall 和 Potts<sup>[1]</sup> 两位学者提出, 他们的研究表明, 在传统的制造模式下, 通过供应链调度的方法, 能够将制造企业的总成本降低 20% 以上.

近年来供应链调度的研究有了较快的进展, 很多学者在供应链调度模型的构建以及优化算法的设计方面做了大量的工作. 按照供应链调度所涉及的范围可以分为两阶段供应链调度和三阶段调度, 其中两阶段调度问题涉及到供应商 - 制造

企业和制造企业 - 客户这两类模型, 三阶段调度则为供应商 - 制造企业 - 客户模型. 在供应商 - 制造企业模型中, Chen 和 Hall<sup>[2]</sup> 和 Agnetis 等<sup>[3]</sup> 研究了供应商和制造企业最优方案的冲突, 分别提出了一类兼顾二者利益的合作机制; Torabi 等<sup>[4]</sup> 考虑了供应商和制造企业的原材料库存和配送总成本, 提出了优化总成本的智能算法. 两阶段供应链调度问题的另一种模型便是制造企业 - 客户模型, Chen 和 Vairaktarakis<sup>[5]</sup> 考虑了古典生产模式下的生产与配送调度问题, 证明了该模式下优化制造跨度和成本问题均为 NP-hard 问题, 提出了几类启发式算法; 由于这类问题的计算复杂性, 后期的研究集中在近似算法<sup>[6-7]</sup> 和智能算法方面<sup>[8-9]</sup>, 另外, 学者们考虑了两阶段供应链调度问题的现实约束并设计了优化算法, 这些约束包括作业保质期极短<sup>[10]</sup> 和生产能力有限<sup>[11]</sup> 等. 另一方面, 三阶段供应链调度问题, 则涉及到原材料设计、生产加工和成品配送三个部分, Hall 和 Potts<sup>[1]</sup> 系统的构造了该类问题在古典生产模式下

① 收稿日期: 2013-06-04; 修订日期: 2015-10-30.

基金项目: 国家自然科学基金资助项目(71202048; 71471052; 71521001); 教育部人文社会科学基金资助项目(13YJC630051).

作者简介: 程八一(1981-), 男, 安徽安庆人, 博士, 副教授. Email: bycheng@mail.ustc.edu.cn

的模型,并设计了基于动态规划的优化算法; Selvarajah和 Steiner<sup>[12]</sup>设计了一类优化配送-库存联合成本的近似算法,最坏性能比为 1.5; Sawik<sup>[13]</sup>考虑了一类优化库存、生产和配送总成本的问题,且作业具有很长的生命周期; Yimer 和 Demirli<sup>[14]</sup>则提出了一类分解算法,将三阶段问题分解为生产和运输两个子问题,并采用遗传算法进行优化;后期的研究中学者们也考虑了工业中的实际约束,包括带时间窗<sup>[15]</sup>和同步原料补给<sup>[16]</sup>的供应链调度问题。

但上述的供应链调度研究均在古典生产模式<sup>[17]</sup>下展开,这类模式的基本假定,就是加工设备同时生产一个作业或者固定数量个作业,前者称为传统生产模式,后者称为批生产模式。但在现实工业中,大量的生产问题并不属于这类生产模式,例如半导体芯片制造业、陶瓷制造业、食品加工业、金属冶炼业等,以芯片制造业为例,芯片出厂前的最后一道工序是芯片的高温预烧,工序的目的是通过高温处理发现有质量问题的芯片,从而保证出厂芯片的质量。预烧的处理时间长,是其他工序的几倍乃至几十倍,芯片有不同的尺寸和预烧时间,均由客户指定,而预烧炉的容积是确定的,在预烧时,芯片可以分批放在炉中进行,每一批芯片的总体积不能超过炉子的容积,同时,芯片的预烧时间可以大于客户要求的时间,但不能小于这个时间,因此一批芯片的处理时间等于其中芯片的最长处理时间。称这类生产模式为差异分批模式(production with batch-processing machines with arbitrary-size jobs),在食品加工业、金属冶炼业中,同样有大量的该类问题。1994年,Uzsoy<sup>[18]</sup>提出了差异分批模式的单设备调度问题,证明了单设备的制造跨度最优化问题是强 NP-hard 问题,总完工时间问题是 NP-hard 问题,并分别设计了启发式算法; Zhang 等<sup>[19]</sup>分析了 Uzsoy 提出的算法的性能,并进一步设计了一种最坏性能比为 7/4 的 MLPT-FF 算法; Tang 等<sup>[20]</sup>针对一类考虑原材料库存的差异作业批调度问题设计了近似算法,程八一等<sup>[21]</sup>和 Tang 等<sup>[22]</sup>分别提出了优化流水车间问题的近似算法;文献[23,24]采用了动态规划方法对该问题进行了优化;文献[25]采用了聚类算法进行求解。除近似算法、动态规划算法和聚类算法之外,一些性能优异的智能算法也被

引入到差异分批模式的优化调度中,包括遗传算法<sup>[26]</sup>、模拟退火算法<sup>[27]</sup>、蚁群算法<sup>[28-30]</sup>和微粒群算法<sup>[31]</sup>等。在作业配送方面,Sahin 等<sup>[32]</sup>从经济分析的角度入手,提出了公路、铁路、海上运输等方式的优化组合方法; Rieksts 等<sup>[33]</sup>和 Janeiro 等<sup>[34]</sup>将配送和库存环节联合起来,提出了基于库存的配送策略以获得最低的配送成本; Koca 和 Yildirim<sup>[35]</sup>考虑了一类特殊的成本结构,采用分级方法优化配送成本,Ghosei 和 Ghannadpour<sup>[36]</sup>则采用了智能优化方法对配送成本进行优化,获得了较理想的实验结果。

但上述差异分批生产模式的研究仅限于车间内的优化调度,配送研究也仅仅针对作业配送环节,并没有立足制造企业的经营决策高度对两者进行协调优化。生产环节的研究主要通过设计生产方案来获得最低的生产成本,配送环节的研究主要是通过对车辆和产成品的获得最低的配送成本,由于两个环节内的优化目标不同,因此两者的简单相加必然不是制造企业的最优策略,这两种优化方案对制造企业的借鉴和参考价值是有限的。尤其是在当前强调企业服务水平的环境下,应对制造企业的生产、库存和配送进行集成思考和系统设计,从企业经营决策的高度给出生产-库存-配送的联合调度策略,获得作业生命周期内从安排加工直至送达客户的一体化运作方案,从而为制造企业提供全面、精细的运作支持。本文采用一典型的差异分批制造模式,从制造企业的决策角度出发,设计集成化的分批、生产、库存和配送方案,来优化企业的供应点调度总成本。对制造企业来说,生产成本主要包括生产过程的能耗和操作人员工资费用,这些均取决于生产时间的长短;在库存环节,成本包括库存运行成本和人员工资等变动成本,库存成本取决于库存产品总量和库存时间;而在配送环节,成本则主要包括运输中的燃油费用和驾驶员工资,这些均取决于配送路线的长度和运输的次数。首先建立基于上述声明的生产-库存-配送联合成本的供应链调度模型,分析成本优化问题的计算复杂性,然后设计有效的多项式时间算法,实现联合成本的优化,以期为制造企业的运作提供直接有效的借鉴和参考。

### 1 整数规划模型与计算复杂性

制造企业有  $m$  台批处理设备, 设备集为  $M = \{M_1, \dots, M_m\}$ , 每台设备的最大容量为  $U$ , 要加工的客户的作业集为  $J = \{1, 2, \dots, n\}$ , 其中作业  $j$  的体积为  $u_j$ , 客户要求  $j$  的加工时间为  $t_j$ , 作业经分批之后得到批集合  $\Phi = \{b_{11}, \dots, b_{1K(1)}; b_{21}, \dots, b_{2K(2)}; \dots, b_{m1}, \dots, b_{mK(m)}\}$ , 其中  $K(i)$  表示设备  $M_i$  上加工的批数,  $K$  为所有设备上的总批数,  $b_{ik}$  表示设备  $M_i$  上处理的第  $k$  批作业,  $b_{ik}$  的处理时间为  $T_{ik}$ , 完工时间为  $C_{ik}$ . 作业加工完毕之后, 制造企业利用自有车辆进行配送, 车辆台数为  $v$ , 每台车的最大容量为  $V$ , 在配送时, 车辆从制造企业行驶至客户所在地的时间为  $t_0$ , 需要的车次数为  $L$ , 车次的集合记为  $D = \{d_1, d_2, \dots, d_l\}$ , 其中  $d_l$  为子作业集, 且  $\sum_{l=1}^L |d_l| = |D| = n = |J|$ . 在生产阶段, 由于生产成本和处理时间正相关, 因此本文中令生产阶段的成本为  $PC = \lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} + \mu_1$ , 其中  $\lambda_1 > 0$ ,  $\lambda_1$  和  $\mu_1$  均为常数; 在配送阶段, 每一次的配送成本固定, 配送总成本和配送的车次数正相关, 因此令配送成本  $DC = \lambda_2 L + \mu_2$ , 其中  $\lambda_2 > 0$ ,  $\lambda_2$  和  $\mu_2$  均为常数; 由于车辆有限, 并非所有完工作业都能及时配送, 当前不能配送的完工作业转入库存, 库存成本包括库存运行成本和变动成本. 其中运行成本包括仓库设备的折旧以及水电等能耗成本, 从第一个作业入库至最后一个作业出库的时间记为  $IS$ , 则运行成本与  $IS$  正相关; 变动成本为作业出入库等相关的作业费用, 作业  $j$  的库存时间为  $I_j$ , 则变动成本与  $\sum_{j=1}^n I_j s_j$  正相关, 因此库存环节的成本为  $IC = \lambda_3 IS + \lambda_4 \sum_{j=1}^n I_j s_j + \mu_3$ , 其中  $\lambda_3, \lambda_4 > 0$ . 对于供应链调度问题, 采用 Chen<sup>[37]</sup> 的五区间表示方法, 即  $\alpha_1 | \alpha_2 | \alpha_3 | \alpha_4 | \alpha_5$ , 其中  $\alpha_1$  表示制造企业的加工设备配置,  $\alpha_2$  表示生产加工中的约束,  $\alpha_3$  表示客户数量,  $\alpha_4$  表示配送车辆配置,  $\alpha_5$  表示供应链调度的优化目标. 基于上述表示方法, 本文研究的供应链调度问题可以表示为  $Pm | U, u_j, batch | 1 | v, V | TC$ , 其中, “ $Pm$ ”表示

加工设备为  $m$  台相同的并行设备; 每台设备的最大容量为  $U$ , 作业  $j$  的体积为  $u_j$ , 作业分批处理且处理工程不允许中断. “1”表示客户数为 1; “ $v, V$ ”表示制造企业有  $v$  台可用车辆, 每辆车的最大容积为  $V$ . “ $TC$ ”表示供应链调度的优化目标是生产和配送中的总成本.

定义下述决策变量

$$x_{ik} = \begin{cases} 1 & \text{若建批 } b_{ik} \\ 0 & \text{若否} \end{cases}; y_{jik} = \begin{cases} 1 & \text{若 } j \in b_{ik} \\ 0 & \text{若否} \end{cases};$$

$$w_l = \begin{cases} 1 & \text{若建 } d_l \\ 0 & \text{若否} \end{cases}; z_{jl} = \begin{cases} 1 & \text{若 } j \subset d_l \\ 0 & \text{若否} \end{cases}$$

根据上述的问题描述, 建立整数规划模型如下

$$\min TC = \lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} + \lambda_2 L + \lambda_3 IS + \lambda_4 \sum_{j=1}^n I_j s_j + \mu_1 + \mu_2 + \mu_3 \quad (1)$$

$$\text{s. t. } \sum_{j=1}^n x_{ik} y_{jik} u_j \leq U \quad i = 1, \dots, m; k = 1, \dots, K_i \quad (2)$$

$$T_{ik} = \max\{t_j : y_{jik} = 1\} \quad i = 1, \dots, m; k = 1, \dots, K_i \quad (3)$$

$$C_{i, h+1} - T_{i, h+1} \geq C_{ih} \quad i = 1, \dots, m; h = 1, \dots, K_i - 1 \quad (4)$$

$$n \geq K_{LB} \geq \lceil \sum_{j=1}^n u_j / U \rceil \quad (5)$$

$$K(i) = \sum_{k=1}^n x_{ik} \quad i = 1, \dots, m \quad (6)$$

$$K = \sum_{i=1}^m K(i) \quad (7)$$

$$n \geq L_{LB} \geq \lceil \sum_{j=1}^n u_j / V \rceil \quad (8)$$

$$L = \sum_{l=1}^n w_l \quad (9)$$

$$\sum_{j=1}^n w_l z_{jl} u_j \leq V \quad l = 1, \dots, L \quad (10)$$

$$x_{ik}, y_{jik}, w_l, z_{jl} \in \{0, 1\} \quad \forall i, j, k \quad (11)$$

模型中  $K_{LB}$  和  $L_{LB}$  分别表示总批数  $K$  和总配送次数  $L$  的下界. 目标函数 (1) 给出了供应链调度总成本的表达式; 约束条件 (2) 要求同一批中的所有作业总体积不能超过设备的容量, 式 (3) 和式 (4) 要求批的加工不允许中断; 式 (5) 和式 (7) 分别给出了总批数的上下界以及计算方法,

其中  $\lceil a \rceil$  表示不小于  $a$  的最小整数; 式 (6) 利用决策变量求出  $K(i)$ ; 式 (8) 给出了运输中车次数量的下界, 式 (9) 则利用决策变量  $w_l$  求出车次数; 式 (10) 要求每一车作业的总体积不能超过车辆的运输能力; 式 (11) 声明模型中的决策变量均为 0-1 变量.

$Pm|U, u_j, batch|1|v, V|TC$  的计算复杂性可用下述的特殊情形得出. 考虑  $t_0 = 0, \lambda_3 = \lambda_4 = \mu_3 = 0, m = 1, t_1 = t_2 = \dots = t_n$ , 即: ① 配送时间为零; ② 库存成本为零; ③ 只有一台生产设备; ④ 每个作业的处理时间相同. 由①、②可知作业加工完毕后不需安排库存和配送, 故供应链调度问题转化为传统的生产调度问题, 最小化  $TC$  等价于最小化  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$ ; 由③知, 由于仅有一台生产设备, 因此问题等价于最小化制造跨度, 也即  $1|U, u_j, batch|C_{max}$ ; 而由④知, 每个作业处理时间相同, 因此最小化  $C_{max}$  等价于最小化总批数  $K$ , 此时, 作如下转换, 问题等价于著名的装箱问题 (bin packing problem):

优化目标: 最小化箱子的数量;

箱子容积: 1;

物品  $j$  的体积:  $u_j/U$ .

而上述装箱问题是强 NP-hard 问题, 因此, 有下述结论.

推论 1  $Pm|U, u_j, batch|1|v, V|TC$  是强 NP-hard 问题.

## 2 优化算法及其时间复杂性

令  $\pi^*$  是  $Pm|U, u_j, batch|1|v, V|TC$  的最优解, 其相关的参数均加上星标表示, 例如  $TC^*, L^*$  等. 考察目标函数  $TC = \lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} + \lambda_2 L + \lambda_3 IS + \lambda_4 \sum_{j=1}^n I_j s_j + \mu_1 + \mu_2 + \mu_3$ , 其中  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$  表示所有设备加工时间的总和, 显然等价于单台设备加工  $\Phi$  中所有批的时间, 因此最小化  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$  等价于最小化单设备问题的制造跨度, 为后续表述的方

便, 将  $\{b_{ik}: i = 1, \dots, m; k = 1, \dots, K_i\}$  等价表示为  $\{B_k: k = 1, \dots, K\}$ , 其中  $B_k$  的处理时间为  $P_k$ .

由于  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$  为所有批次的处理时间求和, 故与批之间的前后顺序无关. 相应的, 最优解  $\pi^*$  中, 所有批的集合为  $\{B_k^*: k = 1, \dots, K^*\}$ , 其中  $B_k^*$  的处理时间为  $P_k^*$ .

下面给出本文的算法 A.

### 算法 A

步骤 1 将所有作业按照加工时间非增的顺序排序, 并重新编号为  $1, 2, \dots, n$ , 即有  $t_1 \geq t_2 \geq \dots \geq t_n$ .

步骤 2 新建一个空批  $B_1$ , 将作业 1 放入  $B_1$ , 并将作业 1 从作业集  $J$  中删除; 然后从  $u_2$  开始, 逐个将作业  $j$  加入  $B_1$  中, 如果总体积仍然不超过  $U$ , 则  $j \in B_1$ , 若  $j$  加入后总体积大于  $U$ , 则  $j \notin B_1$ , 直至所有作业均判断完毕. 若此时  $J \neq \phi$ , 则新建空批  $B_2$ , 按上述方法组批. 重复上述组批过程, 直到  $J = \phi$  为止. 此时得到一个可行的批集合  $\Phi = \{B_1, B_2, \dots, B_K\}$ .

步骤 3 将  $B_1, B_2, \dots, B_K$  按处理时间非增顺序排序, 从处理时间最长的批开始, 逐个分别安排到  $M_1, M_2, \dots, M_m$  上加工, 同时将这些批从  $\Phi$  中删除. 若  $\Phi \neq \phi$ , 则等到任意设备上有一批作业处理完毕时, 将  $\Phi$  中处理时间最长的批安排到该设备上加工, 直至  $\Phi = \phi$  为止, 此时所有批次均安排完毕.

步骤 4 将完工的作业按下述方法分组配送. 将  $B_k (k = 1, \dots, K)$  中的作业按任意顺序排列, 得到完整的作业序列  $1, 2, \dots, n$ . 首先安排第一辆车配送的作业集  $d_1$ : 将作业 1 放入其中, 对后续的所有作业, 逐个放入  $d_1$ , 如果总容量不超过  $V$ , 则  $j \in d_1$ , 否则  $j \notin d_1$ . 然后按照该分组方法, 依次建立  $d_2, d_3, \dots, d_L$ , 直至所有完工作业均安排配送为止. 此时得到一个可行的配送方案  $\{d_1, d_2, \dots, d_L\}$ . 若当前批已经完工, 且无可用车辆进行配送, 则将完工作业转入库存.

在算法 A 中, 步骤 1-步骤 2 完成作业的分批, 步骤 3 完成批次的生产加工, 步骤 4 则获得完工作业的配送方法, 故最终获得的可行解  $\pi$  是一个集成化的方案, 给出了作业分批、生产和配送的完整过程. 下面首先分析算法 A 的时间性能.

定理1 算法A的时间复杂度为  $O(mn + n \ln n)$ .

证明 A的时间复杂性可以通过分析4个步骤得到.

步骤1对  $n$  个作业排序,时间复杂性为  $O(n \times \ln n)$ .

步骤2等价于 First Fit 方法<sup>[18]</sup> 求解装箱问题,装箱问题如推论1之前内容所示,由于 First Fit 算法的时间复杂度为  $O(n \ln n)$ ,故步骤2的时间复杂度为  $O(n \ln n)$ .

步骤3中,对批排序的时间复杂度为  $O(K \times \ln K)$ ,由于  $K \leq n$  故时间复杂度小于  $O(n \ln n)$ ;在安排加工时,安排前  $m$  个批的时间为  $O(m)$ ,后  $(K - m)$  个批中每个批搜索设备的时间为  $O(m)$ ,故步骤3的时间复杂度为  $O(m(K - m))$ ,显然不超过  $O(mn)$ .

步骤4中的配送方法等价于 First Fit 算法求解下述的装箱问题:物品数量为  $n$ ;物品  $j$  体积为  $u_j/V$ ;箱子的容积为1.而库存算法由生产、配送方案直接产生,其时间复杂度为  $O(n)$ ,因此,步骤4的时间复杂度为  $O(n \ln n)$ .

综上,通过四个步骤的时间复杂度可知,算法A的时间复杂度为  $O(mn + n \ln n)$ . 证毕.

下面给出具体的算例来说明算法A的求解过程.

### 3 算例

某陶瓷加工企业,一批订单中共有40件瓷器需要进行生产和配送,客户要求的瓷器尺寸和烧制时间如表1所示,显然在本问题中,每件胚体可视为一个作业,每个瓷窑视为一台批处理设备.表1中第一行和第四行为作业的编号,第二行和第五行为作业的体积,第三行和第六行为作业的烧制时间.已知设备的容量为  $U = 50$ ,设备台数  $m = 3$ ,配送车辆的容积为  $V = 100$ ,车辆数  $v = 3$ ;烧制成本  $PC = 20T + 50$ ,其中  $T$  为烧制时间;配送成本  $DC = 25L + 100$ ,其中  $L$  为配送次数;库存成本  $IC = 6IS + 0.5 \sum_{j=1}^n I_j s_j + 100$ ,其中  $IS$  为库存运行时间,  $I_j$  为完工作业  $j$  的库存时间,从制造企业到客户的单程配送时间为10.

那么根据A的步骤1,将作业按照  $t_j$  的非增关系排序,可获得排序后的作业序列如表2所示,表的结构与表1相同.

表1 作业尺寸和加工时间

Table 1 Processing time of jobs

$j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$u_j$	13	9	25	7	15	24	3	21	8	4	15	34	2	15	10	9	11	5	3	17
$t_j$	20	14	11	16	9	10	7	20	7	19	6	2	8	4	10	15	16	8	2	9
$j$	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
$u_j$	3	12	7	10	7	15	9	10	7	6	7	34	5	15	6	4	5	12	13	8
$t_j$	6	13	19	3	20	8	2	13	15	3	11	14	10	17	8	16	3	6	10	4

表2 按  $t_j$  排序后的序列

Table 2 Job list ordered by  $t_j$

$j$	1	8	25	23	10	34	4	17	36	16	29	2	32	22	28	3	31	6	15	33
$u_j$	13	21	7	7	4	15	7	11	4	9	7	9	34	12	10	25	7	24	10	5
$t_j$	20	20	20	19	19	17	16	16	16	15	15	14	14	13	13	11	11	10	10	10
$j$	39	5	20	13	18	26	35	7	9	11	21	38	14	40	24	30	37	19	27	12
$u_j$	13	15	17	2	5	15	6	3	8	15	3	10	15	8	10	6	5	3	9	34
$t_j$	10	9	9	8	8	8	8	7	7	6	6	13	4	4	3	3	3	2	2	2

根据步骤2和步骤3,可以获得如表3所示的分批结果.第一列表示所得的批序列,这些批尚未安排到设备上,记为  $e_i$ ;第二列表示分配到设备

上之后的加工情况,可知在设备  $M_1$  上加工的批集合为  $\{e_1, e_6, e_9\}$ ,  $M_2$  上加工的批集合为  $\{e_2, e_5, e_8\}$ ,  $M_3$  上加工的批集合为  $\{e_3, e_4, e_7\}$ ;第三列为

每一个批中的作业集; 第四列为每个批中作业的总 体积, 易见  $b_{13}$ 、 $b_{22}$  和  $b_{33}$  对设备容量的利用率为 98%, 而其他 6 批的利用率则均为 100%; 第五列表示每个批次的处理时间.

于是, 根据  $PC$  的计算方法可以获得生产总

成本如表 4 所示. 其中第一列表示设备编号, 第二列 ~ 第七列分别表示设备上的每个批次及完工时间, 每台设备上的最后一批作业的完工时间即为设备的运行时间; 第八列为每台设备的生产总成本.

表 3 分批结果

Table 3 Obtained batches

批次	设备加工编号	批中作业	作业总体积	批处理时间
$e_1$	$b_{11}$	1, 8, 25, 23, 13	50	20
$e_2$	$b_{21}$	10, 34, 4, 17, 36, 16	50	19
$e_3$	$b_{31}$	29, 2, 32	50	15
$e_4$	$b_{32}$	22, 28, 3, 7	50	13
$e_5$	$b_{22}$	31, 6, 15, 33, 19	49	11
$e_6$	$b_{12}$	39, 5, 20, 18	50	10
$e_7$	$b_{33}$	26, 35, 9, 11, 21	49	8
$e_8$	$b_{23}$	38, 14, 40, 24, 39	50	6
$e_9$	$b_{13}$	30, 27, 12	49	3

表 4 设备加工时间与生产成本

Table 4 Processing time and production cost of machines

设备编号	$b_{i1}$	$C_{i1}$	$b_{i2}$	$C_{i2}$	$b_{i3}$	$C_{i3}$	$PC_i$
$M_1$	$b_{11}$	20	$b_{12}$	30	$b_{13}$	33	710
$M_2$	$b_{21}$	19	$b_{22}$	30	$b_{23}$	36	770
$M_3$	$b_{31}$	15	$b_{32}$	28	$b_{33}$	36	770

然后根据算法 A 的步骤 4, 获得的配送以及库存方案如表 5 所示. 表中第一列和第二列分别为生产批次及加工编号; 第三、四、五列分别为批中的作业、作业总体积和完工时间; 第六列为每批作业配送的开始时间, 第七列为所在的配送次数, 第八列为批中作业的库存时间. 可见配送共需 5 次, 即五个车次, 其中第一次和第二次配送车辆均为满载, 其他三次车辆没有满载, 配送总成本  $DC = 225$ . 根据上述生产和配送方案, 可得库存方

案, 易见  $e_1 \sim e_6$  能够及时配送, 分别在前三次配送中完成.  $e_7 \sim e_9$  中的作业则需转入产品库存, 其中  $e_9$  最先完工, 但完工时所有车辆均在配送途中, 故须等到第一台车返回后和  $e_9$  中的作业一起安排配送, 出发时间为 39, 同理可知  $e_7$ 、 $e_8$  的库存情况.

由表可得  $IS = 12$ ,  $\sum_{j=1}^n I_j S_j = 12 \times 49 + 3 \times 50 + 6 \times 49 = 1032$ , 故库存成本  $IC = 6 \times 12 + 0.5 \times 1032 + 100 = 688$ .

表 5 库存方案

Table 5 Inventory plan

批次	设备加工编号	批中作业	作业总体积	完工时间	出发时间	配送序号	库存时间
$e_1$	$b_{11}$	1, 8, 25, 23, 13	50	20	20	$d_1$	0
$e_2$	$b_{21}$	10, 34, 4, 17, 36, 16	50	19	20	$d_1$	0
$e_3$	$b_{31}$	29, 2, 32	50	15	28	$d_2$	0
$e_4$	$b_{32}$	22, 28, 3, 7	50	28	28	$d_2$	0
$e_5$	$b_{22}$	31, 6, 15, 33, 19	49	30	30	$d_3$	0
$e_6$	$b_{12}$	39, 5, 20, 18	50	30	30	$d_3$	0
$e_7$	$b_{33}$	26, 35, 9, 11, 21	49	36	48	$d_5$	12
$e_8$	$b_{23}$	38, 14, 40, 24, 39	50	36	39	$d_4$	3
$e_9$	$b_{13}$	30, 27, 12	49	33	39	$d_4$	6

综上,算法 A 产生的联合调度方案总成本  $TC = PC + IC + DC = 3\ 163$ .

下面分析 A 的优化性能,主要指标是 A 的最坏性能比.

### 4 优化算法的求解性能

最坏性能比,也即多项式时间算法的近似比,用  $R$  表示,对最小化问题来说,算法得出的解与理论最优解的比值即为最坏性能比,比值越接近 1,说明算法的优化性能越好.在分析本文算法 A 的求解性能时,首先从生产环节展开分析.

引理 1 在生产环节,算法 A 产生的总批数不超过最优解的两倍.

证明 从 A 的运行过程可以看出,产生分批方案的是步骤 1 和 2.从步骤 1 和 2 得知,对任意的整数  $g, h \in [1, K]$  且  $g \neq h$ ,有

$$\sum_{j \in B_g} u_j + \sum_{j \in B_h} u_j > U \tag{12}$$

若否,则  $B_g$  和  $B_h$  中的作业可以合并到一个批中,不符合步骤 2 的要求.现从集合  $\{(g, h) \mid g + h = K + 1\}$  中选择整数对  $(g, h)$ ,根据式(12)可得

$$\sum_{g=1}^K (\sum_{j \in B_g} u_j + \sum_{j \in B_h} u_j) > KU \tag{13}$$

由于 A 产生的解  $\pi$  中,所有的作业均分到批中,因此有

$$\sum_{g=1}^K (\sum_{j \in B_g} u_j + \sum_{j \in B_h} u_j) = 2 \sum_{j=1}^n u_j \tag{14}$$

而在最优解  $\pi^*$  中,同理有

$$\sum_{k=1}^{K^*} \sum_{j \in B_k^*} u_j = \sum_{j=1}^n u_j \leq K^* U \tag{15}$$

根据式(13) - 式(15)得

$$KU < 2 \sum_{j=1}^n u_j \leq 2K^* U \tag{16}$$

也即  $K < 2K^*$ . 证毕.

根据引理 1,为了方便后续的证明,在 A 产生的可行解  $\pi$  中,增加  $(2K^* - K)$  个虚拟批,所谓虚拟批,即该批的处理时间和作业总体积均为零.这样在  $\pi$  中,生产环节的总批数为  $2K^*$  个.

引理 2 在生产环节,如果  $\pi$  中的作业  $r$  满

足  $\sum_{j=1}^{r-1} u_j \leq kU < \sum_{j=1}^r u_j$ ,那么作业  $1, 2, \dots, r$  均分配在批  $B_1, \dots, B_{2k}$  中.

证明 由步骤 1 知,作业  $1, 2, \dots, r$  为处理时间非增的顺序排列,且与剩下的  $(n - r)$  个作业相比  $1, 2, \dots, r$  将优先分批.考虑所需批数最大的情形  $\Psi_1$ :对任意的  $x, y \in [1, n]$  且  $x \neq y$ ,均有  $u_x + u_y > U$ .在  $\Psi_1$  情形下,任意两个作业都不能放到一个批中,故所需的批数最大.而  $\Psi_1$  情形下的总批数等于下述  $\Psi_2$  的总批数:  $u_1 = u_2 = \dots =$

$u_n = u_0$  且  $U/2 < u_0 \leq U$ . 根据条件  $\sum_{j=1}^{r-1} u_j \leq kU <$

$$\sum_{j=1}^r u_j, \text{ 有} \tag{17}$$

$$(k - 1)U < (r - 1)u_0 \leq kU \tag{17}$$

结合  $u_0 > U/2$  得到  $r - 1 < 2k$ ,也即

$$r < 2k + 1 \tag{18}$$

根据式(18),可知在  $\Psi_2$  下,作业  $r$  分配在了批  $B_1, \dots, B_{2k}$  中,由于  $\Psi_1$  和  $\Psi_2$  两种情形下的批结构相同,因此在最坏情形  $\Psi_1$  中,  $r$  分配在  $B_1, \dots, B_{2k}$  中.故对其他各种情形,  $r$  均能分配在前  $2k$  批中,引理 2 得证. 证毕.

现在将  $\pi$  和  $\pi^*$  中的批均按照处理时间非增的顺序重新排列,记所得的批处理时间为  $\Gamma_1 \geq \Gamma_2 \geq \dots \geq \Gamma_K$  和  $\Gamma_1^* \geq \Gamma_2^* \geq \dots \geq \Gamma_{K^*}^*$ . 在目标函

数中  $TC = \lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} + \lambda_2 L + \mu_1 + \mu_2$  中的

$\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$  等价于所有批次在单台设备上加工时

的制造跨度,故  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}$  可表示为  $\sum_{k=1}^K T_k$ ,其中

$K$  为总批数.根据引理 1 和引理 2,可得下述定理.

定理 1 与最优解  $\pi^*$  相比,算法 A 产生的可行解  $\pi$  中,  $\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} \leq 2 \sum_{i=1}^m \sum_{k=1}^{K^*(i)} T_{ik}^*$ .

证明 先将  $\pi$  和  $\pi^*$  中的并行设备问题等价的转换为单设备制造跨度问题,即

$$\sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} = \sum_{k=1}^K T_k, \sum_{i=1}^m \sum_{k=1}^{K^*(i)} T_{ik}^* = \sum_{k=1}^K T_k^* \tag{19}$$

现在只需比较  $\sum_{k=1}^K T_k$  和  $\sum_{k=1}^K T_k^*$  的关系. 由于  $\Gamma_1, \dots, \Gamma_K$  和  $T_1, \dots, T_K$  仅仅是排序关系不同, 因此有

$$\sum_{k=1}^K T_k = \sum_{k=1}^K \Gamma_k \tag{20}$$

同理, 在最优解  $\pi^*$  中, 有

$$\sum_{k=1}^K T_k^* = \sum_{k=1}^K \Gamma_k^* \tag{21}$$

由式(20)、式(21)可知, 只需比较  $\sum_{k=1}^K \Gamma_k$  和  $\sum_{k=1}^K \Gamma_k^*$  的关系即可. 由引理 2 知

$$\Gamma_{2k+2} \leq \Gamma_{2k+1} \leq t_r \tag{22}$$

在最优解  $\pi^*$  中, 有  $\sum_{i=1}^k \sum_{j \in B_k^*} u_j \leq kU$ , 而根据引理

2,  $kU < \sum_{j=1}^r u_j$ , 故作业  $1, \dots, r$  不能全部分配到  $B_1, \dots, B_k$  中, 因此有

$$\Gamma_{k+1}^* > t_r \tag{23}$$

根据式(22)和式(23)得到

$$\Gamma_{2k+2} \leq \Gamma_{2k+1} < \Gamma_{k+1}^* \tag{24}$$

而对  $\Gamma_1^*$  和  $\Gamma_1$  有

$$\Gamma_1^* = \Gamma_1 = \max\{t_j; j = 1, \dots, n\} \tag{25}$$

故由式(24)和式(25)知对  $\forall k \in [1, K^*]$ ,  $\Gamma_{2k} \leq \Gamma_{2k-1} < \Gamma_k^*$ , 因此

$$\begin{aligned} \sum_{k=1}^K \Gamma_k &= \sum_{k=1}^{2K^*} \Gamma_k = \sum_{k=1}^{K^*} (\Gamma_{2k} + \Gamma_{2k-1}) \leq \\ &\sum_{k=1}^{K^*} 2\Gamma_k^* = 2 \sum_{k=1}^{K^*} \Gamma_k^* \end{aligned} \tag{26}$$

根据式(26), 结合式(19)~式(21)知, 定理 1 得证. 证毕.

下面分析配送阶段以及库存阶段. 根据算法 A 之前的声明, 最优解  $\pi^*$  中配送的车次数为  $L^*$ , 车次集合为  $\{d_1^*, d_2^*, \dots, d_{L^*}^*\}$ . 对于 A 产生的可行解  $\pi$ , 有下述引理.

**引理 3** 在配送环节, 算法 A 产生的配送成本  $L < 2L^*$ .

**证明** 算法 A 中, 配送方案的产生在步骤 4 中. 分析步骤 4 知, A 产生的可行车次集  $\{d_1, \dots, d_L\}$  满足

$$\sum_{j \in d_x} u_j + \sum_{j \in d_y} u_j > V$$

$$\forall x \neq y \text{ and } x, y = 1, \dots, L \tag{27}$$

从下述集合中选取整数对  $(x, y): \{(x, y) \mid x + y = L + 1\}$ , 可得

$$\sum_{x=1}^L (\sum_{j \in d_x} u_j + \sum_{j \in d_y} u_j) = 2 \sum_{j=1}^n u_j > LV \tag{28}$$

而最优解  $\pi^*$  中, 显然有

$$L^* V \geq \sum_{j=1}^n u_j \tag{29}$$

根据式(28)和式(29)得  $LV < 2 \sum_{j=1}^n u_j \leq 2L^* V$ , 即  $L < 2L^*$ . 证毕.

在库存环节, 令  $R_l$  表示  $d_l$  出发后至  $d_{l+1}$  出发前的时间内所有库存产品的成本,  $\tau$  为单程配送的时间, 则显然有  $IC = \lambda_3 IS + \lambda_4 \sum_{l=1}^L 2\tau R_l + \mu_3$ .

**引理 4** 算法 A 产生的库存方案中, 有  $IS <$

$$2IS^*, \sum_{j=1}^n I_j s_j < 2 \sum_{j=1}^n I_j^* s_j$$

**证明** 由引理 3 知  $L < 2L^*$ , 为便于后续证明, 将 A 的配送序列构造为  $\{d_1, d_2, \dots, d_L, d_{L+1}, \dots, d_{2L^*}\}$ . 其中  $d_{L+1}, \dots, d_{2L^*}$  为虚拟配送, 即  $d_{L+1} = \dots = d_{2L^*} = \phi$ . 在最优解  $\pi^*$  中, 有  $\sum_{j \in d_l^*} s_j \leq$

$V$ , 而在  $\pi$  中,  $\sum_{j \in d_{2l-1}} s_j + \sum_{j \in d_{2l}} s_j > V$ , 故有  $2 \sum_{j \in R_l^*} s_j > \sum_{j \in R_{2l-1}} s_j + \sum_{j \in R_{2l}} s_j$ , 即最优解中, 第  $l$  配送后剩余的库存产品体积均大于  $\pi$  中第  $(2l-1)$  和第  $2l$  次的库存产品体积, 因此有

$$\begin{aligned} 2 \sum_{j=1}^n I_j^* s_j &= \sum_{l=1}^{L^*} 4l\tau \sum_{j \in R_l^*} s_j > \sum_{l=1}^{L^*} 2l\tau \sum_{j \in R_{2l-1}} s_j + 2l\tau \sum_{j \in R_{2l}} s_j > \\ &\sum_{l=1}^{L^*} (2l-2)\tau \sum_{j \in R_{2l-1}} s_j + 2l\tau \sum_{j \in R_{2l}} s_j = \sum_{j=1}^n I_j s_j \end{aligned} \tag{30}$$

同时, 在库存运行时间方面, 有

$$2IS^* = 4\tau L^* > 2\tau L = IS \tag{31}$$

证毕.

以上对算法的生产环节和配送环节均展开了分析, 下面给出算法的求解性能.

**定理 2** 在求解  $Pm \mid U, u_j, batch \mid 1 \mid v, V \mid TC$  问题时, 算法 A 的最坏性能比不大于 2.

证明 根据定理 1 和引理 3、引理 4 ,得到 A 的性能比如下

$$\begin{aligned}
R_A &= \frac{TC}{TC^*} = \frac{\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik} + \lambda_2 L + \lambda_3 IS + \lambda_4 \sum_{j=1}^n I_j s_j + \mu_1 + \mu_2 + \mu_3}{\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}^* + \lambda_2 L^* + \lambda_3 IS^* + \lambda_4 \sum_{j=1}^n I_j^* s_j + \mu_1 + \mu_2 + \mu_3} \\
&\leq \frac{2\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}^* + 2\lambda_2 L^* + 2\lambda_3 IS^* + 2\lambda_4 \sum_{j=1}^n I_j^* s_j + \mu_1 + \mu_2 + \mu_3}{\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}^* + \lambda_2 L^* + \lambda_3 IS^* + \lambda_4 \sum_{j=1}^n I_j^* s_j + \mu_1 + \mu_2 + \mu_3} \\
&\leq \frac{2\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}^* + 2\lambda_2 L^* + 2\lambda_3 IS^* + 2\lambda_4 \sum_{j=1}^n I_j^* s_j + 2\mu_1 + 2\mu_2 + 2\mu_3}{\lambda_1 \sum_{i=1}^m \sum_{k=1}^{K(i)} T_{ik}^* + \lambda_2 L^* + \lambda_3 IS^* + \lambda_4 \sum_{j=1}^n I_j^* s_j + \mu_1 + \mu_2 + \mu_3} \\
&= 2
\end{aligned}$$

定理 2 得证.

证毕.

小化问题是强 NP-hard 问题 ,然后提出了一类  $O(mn + n \ln n)$  时间的近似算法 ,最坏性能比不大于 2.

### 5 结束语

本文考虑了制造企业的一类供应链调度问题 ,在生产环节 ,制造企业的生产设备为多台并行批处理设备 ,每台设备有固定的容量 ,而加工的作业具有任意的尺寸和处理时间 ,加工时分批进行 ,每一批作业的加工不允许中断;在配送环节中作业的配送由企业的自有车辆进行运送 ,车辆具有相同的运输能力;作业加工完毕后 ,若不能立即配送 ,则转入产品库存.建立了生产 - 库存 - 配送联合成本优化问题的数学模型 ,证明联合成本的最

在后续研究中 ,本文的供应链调度问题可以根据实际生产扩展为其他设备模型 ,如流水作业模型、车间作业模型等 ,由于单设备的供应链调度问题已经是强 NP-hard 问题 ,因此流水作业、车间作业模型下的供应链调度问题是强 NP-hard 问题 ,对优化算法的研究应集中在近似算法和智能算法两个方面.另外 ,本文考虑的联合成本优化问题涉及到生产、库存和配送环节 ,后续研究可以扩展到采购环节 ,将采购、生产、库存和配送等环节综合考虑 ,进行系统化建模 ,并设计有效的优化算法.

### 参考文献:

[1]Hall N G , Potts C N. Supply chain scheduling: Batching and delivery [J]. Operations Research , 2003 , 51( 4) : 566 - 584.

[2]Chen Z L , Hall N G. Supply chain scheduling: Conflict and cooperation in assembly systems [J]. Operations Research , 2007 , 55: 1072 - 1089.

[3]Agnetis A , Hall N G , Pacciarelli D. Supply chain scheduling: Sequence coordination [J]. Discrete Applied Mathematics , 2006 , 154( 15) : 2044 - 2063.

[4]Torabi S A , Ghomi S M T , Karimi B. A hybrid genetic algorithm for the finite horizon economic lot and delivery scheduling in supply chains [J]. European Journal of Operational Research , 2006 , 173( 1) : 173 - 189.

[5]Chen Z L , Vairaktarakis G L. Integrated scheduling of production and distribution operations [J]. Management Science , 2005 , 51( 4) : 614 - 628.

[6]Selvarajah E , Steiner G. Batch scheduling in a two-level supply chain: A focus on the supplier [J]. European Journal of Operational Research , 2006 , 173( 1) : 226 - 240.

[7]Averbakh I , Xue Z. On-line supply chain scheduling problems with preemption [J]. European Journal of Operational Re-

- search ,2007 ,181( 1) : 500 – 504.
- [8]Naso D , Surico M , Turchaiano B , et al. Genetic algorithms for supply-chain scheduling: A case study in the distribution of ready-mixed concrete [J]. *European Journal of Operational Research* ,2007 ,177( 3) : 2069 – 2099.
- [9]Zegordi S H , Abadi I N K , Nia M A B. A novel genetic algorithm for solving production and transportation scheduling in a two-stage supply chain [J]. *Computers & Industrial Engineering* ,2007 ,58( 3) : 373 – 381.
- [10]Chen Z L , Pundoor G. Order assignment and scheduling in a supply chain [J]. *Operations Research* ,2006 ,54( 3) : 555 – 572.
- [11]Hall N G , Liu Z. Capacity allocation in supply chains [J]. *Operations Research* ,2010 ,58( 6) : 1711 – 1725.
- [12]Selvarajah E , Steiner G. Approximation algorithms for the supplier's supply chain scheduling problem to minimize delivery and inventory holding costs [J]. *Operations Research* ,2009 ,57( 2) : 426 – 438.
- [13]Sawik T. Coordinated supply chain scheduling [J]. *International Journal of Production Economics* ,2009 ,120( 2) : 437 – 451.
- [14]Yimer A D , Demirli K. A genetic approach to two-phase optimization of dynamic supply chain scheduling [J]. *Computers & Industrial Engineering* ,2010 ,58( 3) : 411 – 422.
- [15]Yeung W K , Choi T M , Cheng T C E. Supply chain scheduling and coordination with dual delivery models and inventory storage cost [J]. *International Journal of Production Economics* ,2011 ,132( 2) : 223 – 229.
- [16]Osman H , Demirli K. Economic lot and delivery scheduling problems for multi-stage supply chains [J]. *International Journal of Production Economics* ,2012 ,136( 2) : 275 – 286.
- [17]Pinedo M. *Scheduling: Theory , Algorithms and Systems* [M]. 2nd ed. , New Jersey: Prentice-Hall ,2002.
- [18]Uzsoy R. Scheduling a single batch-processing machine with arbitrary job sizes [J]. *International Journal of Production Research* ,1994 ,32( 7) : 1615 – 1635.
- [19]Zhang G , Cai X , Lee C Y , et al. Minimizing makespan on a single batch-processing machine with nonidentical job sizes [J]. *Naval Research Logistics* ,2001 ,48( 3) : 226 – 240.
- [20]Tang L X , Gong Hua. A hybrid two-stage transportation and batch scheduling problem [J]. *Applied Mathematical Modelling* ,2008 ,32( 12) : 2467 – 2479.
- [21]程八一 , 胡笑旋. 差异作业批调度的流水车间问题及近似算法 [J]. *系统工程学报* ,2011 ,26( 3) : 393 – 399.  
Cheng Bayi , Hu Xiaoxuan. Approximation algorithm for scheduling batch processing machines with non-identical job sizes in flow shop [J]. *Journal of Systems Engineering* ,2011 ,26( 3) : 393 – 399. ( in Chinese)
- [22]Tang L X , Liu P. Two-machine flowshop scheduling problems involving a batching machine with transportation or deterioration consideration [J]. *Applied Mathematical Modelling* ,2009 ,33( 2) : 1187 – 1199.
- [23]Tang L X , Liu P. Minimizing makespan in a two-machine flowshop scheduling with batching and release time [J]. *Mathematical and Computer Modelling* ,2009 ,49( 5/6) : 1071 – 1077.
- [24]冯大光 , 唐立新. 一类新型批处理机调度问题的理论分析 [J]. *管理科学学报* ,2012 ,15( 6) : 33 – 48.  
Feng Daguang , Tang Lixin. Theoretical analysis of scheduling of a new batching machine [J]. *Journal of Management Sciences in China* ,2012 ,15( 6) : 33 – 48. ( in Chinese)
- [25]杜冰 , 陈华平 , 杨勃 , 等. 聚类视角下的差异工件平行机批调度问题 [J]. *管理科学学报* ,2011 ,14( 12) : 27 – 37.  
Du Bing , Chen Huaping , Yang Bo , et al. Scheduling parallel batching machines with non-identical job sizes from a clustering perspective [J]. *Journal of Management Sciences in China* ,2011 ,14( 12) : 27 – 37. ( in Chinese)
- [26]Damodaran P , Manjeshwar P K , Srihari K. Minimizing makespan on a batch-processing machine with arbitrary job sizes using genetic algorithm [J]. *International Journal of Production Economics* ,2006 ,103( 2) : 882 – 891.
- [27]Melouk S , Demodaran P , Chang P Y. Minimizing makespan for single machine batch-processing with arbitrary job sizes using simulated annealing [J]. *International Journal of Production Economics* ,2004 ,87( 2) : 141 – 147.
- [28]Cheng B Y , Li K , Chen B. Scheduling a single batch-processing machine with arbitrary job sizes in fuzzy environment using an improved ant colony optimization [J]. *Journal of Manufacturing Systems* ,2010 ,29( 1) : 29 – 34.
- [29]Cheng B Y , Wang Q , Yang S L , et al. An improved ant colony optimization for scheduling identical parallel batching ma-

- chines with arbitrary job sizes[J]. *Applied Soft Computing*, 2013, 13(2): 765–772.
- [30] 王栓狮, 陈华平, 程八一, 等. 一种差异工件单机批调度问题的蚁群优化算法[J]. *管理科学学报*, 2009, 12(6): 72–82.
- Wang Shuanshi, Chen Huaping, Cheng Bayi, et al. Minimizing makespan on a single batch processing machine with non-identical job sizes using ant colony optimization[J]. *Journal of Management Sciences in China*, 2009, 12(6): 72–82. (in Chinese)
- [31] 程八一, 陈华平, 王栓狮. 基于微粒群算法的单机不同尺寸工件批调度问题求解[J]. *中国管理科学*, 2008, 16(3): 84–88.
- Cheng Bayi, Chen Huaping, Wang Shuanshi. Scheduling a single batch-processing machine with non-identical job sizes based on particle swarm optimization[J]. *Chinese Journal of Management Science*, 2008, 16(3): 84–88. (in Chinese)
- [32] Sahin B, Yilmaz H, Ust Y, et al. An approach for analyzing transportation costs and a case study[J]. *European Journal of Operational Research*, 2009, 193(1): 1–11.
- [33] Rieksts B Q, Ventura J A. Two-stage inventory models with a bi-modal transportation cost[J]. *Computers & Operations Research*, 2010, 37(1): 20–31.
- [34] Janeiro M G, Jurado I G, Meca A, et al. A new cost allocation rule for inventory transportation systems[J]. *Operations Research Letters*, 2013, 41(5): 449–453.
- [35] Koca E, Yildirim E A. A hierarchical solution approach for a multicommodity distribution problem under a special cost structure[J]. *Computers & Operations Research*, 2012, 39(11): 2612–2624.
- [36] Ghosei K, Ghannadpour S F. Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm[J]. *Applied Soft Computing*, 2010, 10(4): 1096–1107.
- [37] Chen Z L. Integrated production and outbound distribution scheduling: Review and extensions[J]. *Operations Research*, 2010, 58(1): 130–148.

## Algorithm for minimizing joint cost for manufacturers with batching machines and arbitrary-size jobs

CHENG Ba-yi<sup>1,2</sup>, HUANG Xiao-man<sup>1,2</sup>, YANG Yan-yan<sup>1,2</sup>, HU Xiao-xuan<sup>1,2</sup>

1. School of Management, Hefei University of Technology, Hefei 230009, China;

2. Key Laboratory of Process Optimization and Intelligent Decision-making, Hefei 230009, China

**Abstract:** A class of problems for manufacturers are proposed to minimize joint cost. Production and outbound distribution are combined to achieve a joint scheduling in supply chain. In the production process, the manufacturers have identical parallel batching machines to process arbitrary-size jobs. The machines have a fixed capacity in size and the total size of jobs in a batch cannot exceed the machine capacity. In the distribution process, the manufacturers deliver the products using their own vehicles and the vehicles have identical transport capacities. If there are no available vehicles to deliver the products, they should be put in inventory. The total cost consists of the production cost, the distribution cost and inventory cost. An integer programming model of the problem is presented and the problem under investigation is shown to be NP-hard in the strong sense. Then a polynomial time algorithm is provided. The time complexity and performance guarantee of the proposed algorithm are analyzed.

**Key words:** supply chain scheduling; joint cost; arbitrary-size jobs; approximation algorithm