

连铸—轧制混流生产模式下轧批调度问题的分支—定价算法

汪恭书¹ 刘静宜² 唐立新¹

摘 要 研究了连铸—轧制在热装、温装和冷装混流生产模式下的一类新型轧批调度问题. 以最小化温装钢坯 (热钢锭) 缓冷 (等待) 导致的热能损失和连轧机架切换带来的产能损失为目标, 建立了整数规划模型. 由于商业优化软件难以在有限时间内直接求得模型的最优解甚至可行解, 提出利用 Dantzig-Wolfe 分解技术将原模型分解为主问题和子问题, 采用列生成算法对主问题和子问题进行迭代求解得到原问题的紧下界, 最后以列生成算法作为定界机制嵌入分支—定界框架中形成分支—定价算法, 执行分支搜索过程以获得整数最优解. 本文还从影响分支—定价算法性能的要素出发提出改进策略. 针对主问题, 提出列生成和拉格朗日松弛混合求解策略来抑制单一列生成算法的尾效应. 针对价格子问题, 在动态规划算法中提出了基于占优规则和标号下界计算方法来及早消除无效状态空间, 加速求解过程. 以钢铁企业的实际生产数据和扩展的随机算例进行了数值实验, 结果显示所提出改进策略能够突破求解能力的限制, 使分支—定价算法在可接受计算时间内求得工业规模问题的最优解.

关键词 连铸—轧制, 混流生产, 列生成, 分支—定价, 动态规划

引用格式 汪恭书, 刘静宜, 唐立新. 连铸—轧制混流生产模式下轧批调度问题的分支—定价算法. 自动化学报, 2017, 43(7): 1178–1189

DOI 10.16383/j.aas.2017.c160316

Branch-and-price Algorithm for Rolling Batch Scheduling Problem in Continuous-casting and Rolling Processes with Hybrid Production Mode

WANG Gong-Shu¹ LIU Jing-Yi² TANG Li-Xin¹

Abstract This paper studies a new type of rolling batch scheduling problem arising in continuous casting and rolling processes which are linked by the hybrid production mode of direct hot charge rolling (DHCR), hot charge rolling (HCR) and cold charge rolling (CCR). The problem is formulated as an integer programming model with the objective of minimizing heat energy loss due to cooling (Waiting) of HCR slabs (Hot ingots) and productivity loss caused by changeover of rolling shelves. Since the commercial optimization software is difficult to obtain an optimal solution or even a feasible solution of the model within a limited CPU time, the model is decomposed into a master problem and a set of subproblems using the Dantzig-Wolfe decomposition. The master problem and subproblems are iteratively solved through column generation algorithm to obtain a tighter lower bound of the original problem. Finally, the column generation algorithm acting as a bounding mechanism is embedded into the branch-and-bound framework to form a branch-and-price algorithm which performs the branch search process to obtain an integer optimal solution. Furthermore, key factors affecting the performance of the algorithm are analyzed and corresponding improvement strategies are proposed. For the master problem, a solution strategy of hybridizing column generation and Lagrangian relaxation is proposed to restrain the tailing-off effect of column generation. For the pricing subproblem, a dominance rule and label lower bound calculation based method is proposed to eliminate non-promising state space early in the dynamic programming algorithm such that the solution procedure is speeded up. Numerical experiments are carried out on actual production data provided by a steel company and random instances extended from actual production data. The results show that the proposed improvement strategies can break through the limitation of the solving ability and enable the branch-and-price algorithm to solve industrial scale problem optimality within an acceptable CPU time.

Key words Continuous-casting and rolling, hybrid production, column generation, branch-and-price, dynamic programming

Citation Wang Gong-Shu, Liu Jing-Yi, Tang Li-Xin. Branch-and-price algorithm for rolling batch scheduling problem in continuous-casting and rolling processes with hybrid production mode. *Acta Automatica Sinica*, 2017, 43(7): 1178–1189

收稿日期 2016-04-08 录用日期 2016-11-08
Manuscript received April 8, 2016; accepted November 8, 2016
国家自然科学基金 (71672032, 71621061, 71202151), 国家重点研发计划 (2017YFB0304100) 资助
Supported by National Natural Science Foundation of China (71672032, 71621061, 71202151) and National Key Research and Development Program of China (2017YFB0304100)
本文责任编辑 赵千川

热装模式是通过辊道把连铸线下来的热坯直接

Recommended by Associate Editor ZHAO Qian-Chuan

1. 东北大学工业与系统工程研究所 沈阳 110819 2. 东北大学数学系 沈阳 110819

1. Institute of Industrial and Systems Engineering, Northeastern University, Shenyang 110819 2. Department of Mathematics, Northeastern University, Shenyang 110819

装炉加热, 然后送往轧制机组进行轧制. 同温装(连铸钢坯先进入保温坑, 再装炉加热后轧制)和冷装(连铸钢坯先下线到原料库, 再上线装炉加热, 最后轧制)模式相比, 热装使生产过程衔接更紧凑, 能源损失更小. 随着钢铁工业的发展, 能源问题已经成为制约钢铁工业发展的重要因素, 如何节能降耗成为钢铁企业进一步发展的关键问题. 在连铸-轧制工序中, 提高钢坯热送率是一个非常有效的节能手段. 热装技术对连铸-轧制工序的节能降耗具有重要意义, 但是由于受到产品质量和工艺过程的限制, 并不是所有钢材产品都适用热装技术. 在初轧产线, 轴承钢, 工具钢由连铸机浇注后, 一般都需要到保温坑进行缓冷后再加热轧制, 否则会由于产生裂纹等质量问题而报废. 部分连铸钢坯如异钢种交接坯, 中间包首尾坯, 由于可能存在质量问题需要下线检测和清理也不适用热装方式. 此外, 连铸和轧制机组对批量生产准则不一致以及机组间产能不匹配也导致钢坯难以全部实现热装. 因此, 热装、温装和冷装混流生产是衔接连铸和轧制工序的常见方式.

轧制工序生产运作管理问题一直是研究的热点, 但以往的研究重点主要集中在单一生产模式的计划与调度问题. 文献 [1] 研究了冷装模式下热轧批量计划问题, 建立 TSP (Traveling salesman problem) 模型并提出禁忌搜索算法. 文献 [2] 针对热装生产计划问题, 提出一种遗传算法. 文献 [3] 为冷装模式下热轧批量计划问题开发了柔性决策支持系统. 文献 [4] 提出两阶段法求解热轧调度问题, 第一阶段对组批问题建立 VRPTW (Vehicle routing problem with time windows) 模型并提出改进的单亲遗传算法, 第二阶段采用智能搜索方法确定轧批的排序以提高热送率. 文献 [5] 将大规模轧批调度问题建模为多目标路径问题, 通过定义层次费用结构将原模型分解为一系列子问题进行求解.

由于热装对工序节能的重要性, 针对热装模式下的连铸-轧制集成生产计划与调度近年来也引起一些研究者的关注. 文献 [6] 对炼钢-连铸-热轧集成计划与调度进行了定义, 分析了建模要素, 但没有给出具体的数学模型. 文献 [7] 研究了连铸-轧制集成计划问题, 提出了虚拟板坯的概念来衔接前后工序的计划, 建立了混合网络模型, 提出了数学规划和启发式混合求解策略. 文献 [8] 对炼钢-连铸-热轧集成计划建模和优化方法进行了综述. 文献 [9] 针对薄板坯连铸连轧生产组织中出现的问题进行了仿真分析, 对企业如何选取生产组织方式给出建议. 文献 [10] 从节能降耗的角度研究了炼钢-连铸-热轧一体化管理中的炉次计划和轧制计划协调问题. 文献 [11] 研究了炉次在连铸及轧制阶段的组批及批排序问题, 同时考虑连铸和轧制阶段对组批及批排序的

要求, 还考虑下游工序精整机组负荷均衡生产的要求. 上述这些研究主要都是侧重于单一模式下的连铸-轧制集成调度问题, 很少考虑混流模式下的调度问题, 并且这些问题主要都侧重于连铸工序的决策及其对轧制工序的影响. 本文以初轧产线生产过程为背景, 研究连铸-轧制在热装, 温装和冷装混流生产模式下的轧批调度问题, 主要侧重于轧制工序的优化决策. 同以往研究的问题^[1-11]相比, 本文研究的问题具有以下显著特点: 1) 以保障热装模式的轧批在连铸-轧制工序上紧凑衔接为主要工艺约束, 决策温装和冷装模式的轧批在轧制机组上的交叉生产顺序和时间; 2) 考虑温装钢坯(热钢锭)由于过度缓冷(等待)带来的热能损失; 3) 考虑连续轧制不同坯型规格轧批时机架切换带来的产能损失.

除研究问题具有新特征外, 本文还提出了基于数学规划的精确算法用于求解实际问题. 以往针对实际计划调度问题的研究一般建立数学模型用以描述问题特征, 在求解上主要采用启发式或智能优化等近似算法. 近似算法虽然易于实现, 但无法评价解的质量. 本文则实现数学模型和最优化求解的完全融合. 首先在分析问题特征的基础上建立整数规划模型; 接着使用 Dantzig-Wolfe 分解技术将原问题分解为主问题和子问题; 然后采用列生成算法在主问题和子问题之间进行迭代得到主问题线性松弛的最优解(为原问题提供下界); 最后将列生成算法作为定界机制嵌入分支-定界框架中形成分支-定价算法, 执行分支搜索过程以获得整数最优解. 列生成求解过程中需要保持子问题解的整数性要求, 这就使得主问题线性松弛能够提供比原问题线性松弛更紧的下界. 因此所提出的分支-定价算法也能够实现自我评价, 即将得到的可行解同下界比较得到对偶间隙来评价解的质量. 本文还从影响分支-定价算法性能的元素出发, 提出改进策略. 由于主问题模型的退化导致在列生成迭代过程中对偶解存在严重振荡现象, 进而导致在迭代过程后阶段收敛速度慢, 这一现象称为列生成算法的尾效应 (Tailing-off effect). 因此, 在主问题求解方面, 提出了列生成和拉格朗日松弛混合求解策略, 即在列生成算法的迭代过程中基于拉格朗日松弛估算主问题的下界, 以避免尾效应. 由于子问题的 NP-难属性导致常规的动态规划算法复杂度高、计算效率低. 因此, 在子问题求解方面, 不同于以往文献通过松弛子问题来寻求高效动态规划算法的策略^[12-13], 本文在不影响子问题最优解空间的前提下, 通过挖掘问题本身特性, 提出了基于占优规则和标号下界计算来及早消除动态规划算法中的无效状态空间, 加速求解过程. 为获取整数最优解, 在分支策略方面, 通过探究原模型变量和主问题变量之间的关系, 提出基于原模型变量

的分支策略,并且通过证明一个引理来说明所提出的分支策略总能有效划分空间而又不会割舍整数最优解.以钢铁企业实际生产数据和扩展的随机算例进行了数值实验,结果显示所提出改进策略能够突破求解能力的限制,使得分支一定价算法在可接受计算时间内求得工业规模问题的最优解,计算性能显著优于商业优化软件和手工排产方法.

本文结构如下:第1节描述问题特征并建立整数规划模型;第2节从模型分解、列生成主问题和子问题的求解方法、分支策略等方面详细阐述分支一定价算法的关键要素及改进策略;第3节设计了数值计算实验;第4节给出结论.

1 问题描述及建模

初轧产线主要生产大方坯、小方坯和圆坯等产品.同连铸钢坯相比,初轧钢坯由于经过初轧大压缩比的轧制使得组织结构更致密,内部缺陷较少,使用性能更高.初轧产线工艺流程如图1所示,主要完成连铸大方坯的缓冷、加热和轧制过程.根据最终产品的质量和工艺要求不同,部分连铸大方坯采用热装模式,由保温台车直接送入加热炉;另一部分连铸大方坯采用温装模式,即先进入保温坑进行缓冷后再进入加热炉;还有一部分连铸大方坯由于质量或计划原因需要下线到连铸堆场冷却后再上线进入加热炉,即冷装模式.另外,由于初轧机组产能一般大于连铸机组产能,初轧机组在空闲时间还可以轧制部分热钢锭及外购冷钢锭以提高设备利用率.连铸大方坯(钢锭)在加热炉(均热炉)中加热后,在初轧机上被轧制为初轧大方坯.部分初轧大方坯经空冷床空冷后进入大方坯库,其他初轧大方坯由中间辊道送至连轧机组,轧制后进入圆坯库或小方坯库.

在初轧生产中,为减少高温钢坯的能源损耗,钢铁企业提出以热装为核心的生产组织思想,即要保证热装生产模式下的钢坯在连铸和轧制工序上实现紧凑衔接.为实现这一要求,热装生产模式下的钢坯在轧制机组上的顺序与它们在连铸机上的顺序必

须相同.在给定连铸机生产计划的前提下,热装钢坯在轧制工序的开始和结束加工时间是确定的.将不同批次的热装钢坯在轧制工序上不连续加工所产出的间隙定义为时间槽.对于温装和冷装生产模式下的钢坯以及热钢锭和冷钢锭,则需要充分利用时间槽进行加工.由于不同来源的钢坯和钢锭在轧制机组上交替生产,因此整个生产过程也称为混流生产.在初轧产线,轧批是批量生产的基本单位,定义为在轧制机组上连续生产且被轧制为相同坯型规格最终产品的一组钢坯(或钢锭)组合.一炉钢水铸造出的钢坯或钢锭通常在轧制机组上被轧制为相同坯型规格的最终产品,对应一个轧批.另外,连轧机共有六台机架,在连轧加工时,大方坯要依次经过这六个机架中的部分或者全部机架.当前后轧批的坯型规格不同时,要更换连轧机的机架,由此将耗费时间,影响产能发挥.因此,在混流生产模式下,轧批调度问题可以描述如下:给定一组时间槽和一组待加工的轧批(包括温装和冷装生产模式下的钢坯以及热钢锭和冷钢锭),决策如何将轧批分配到时间槽及轧批在时间槽内的轧制顺序,以达到减少温装钢坯(热钢锭)由于缓冷(等待)带来的热能损失和减少机架切换带来的产能损失的目的.

基于上述的问题描述及特征分析,下面将建立问题的整数规划模型,模型中使用的符号定义如下.

1) 模型参数

N : 轧批集合;

T : 时间槽集合;

l_t : 时间槽 t 的长度;

p_i : 轧批 i 的轧制时间;

N_t : 允许分配到时间槽 t 的候选轧批子集;

T_i : 轧批 i 可分配的候选时间槽子集;

c_{it} : 轧批 i 分配到时间槽 t 的能损费用;对温装钢坯和热钢锭 $c_{it} > 0$,对其他钢坯和钢锭 $c_{it} = 0$;

μ_{ij} : 轧批 i 和 j 时由于坯型规格不同带来的机架切换时间;

w : 轧制机组单位时间的产能损失费用.

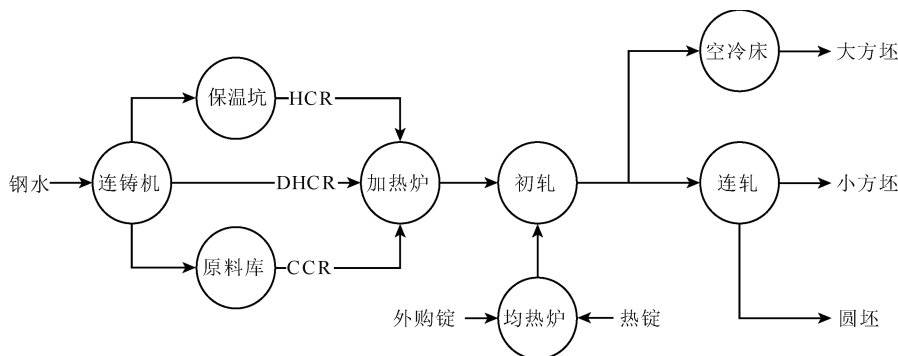


图1 初轧生产工艺流程图

Fig. 1 The production process of primary rolling

2) 决策变量

y_{it} : 轧批 i 分配到时间槽 t 时, 取 1; 否则, 取 0;

x_{ijt} : 轧批 i 和 j 都安排在时间槽 t 内且轧批 j 紧接 i 之后轧制时, 取 1; 否则, 取 0.

采用上述定义的参数和变量, 问题的整数规划 (Integer programming, IP) 模型可表示如下:

$$\min \alpha \sum_{t \in T} \sum_{i \in N_t} c_{it} y_{it} + (1 - \alpha) \sum_{t \in T} \sum_{i \neq j \in N_t} w \mu_{ij} x_{ijt} \quad (1)$$

$$\text{s.t.} \quad \sum_{t \in T_i} y_{it} = 1, \quad \forall i \in N \quad (2)$$

$$y_{it} = \sum_{j \in N_t \cup \{0\} \setminus \{i\}} x_{ijt} = \sum_{j \in N_t \cup \{0\} \setminus \{i\}} x_{jit}, \quad \forall t \in T, i \in N_t \quad (3)$$

$$\sum_{j \in N_t} x_{0jt} = \sum_{j \in N_t} x_{j0t} \leq 1, \quad \forall t \in T \quad (4)$$

$$\sum_{i \in N_t} p_i y_{it} + \sum_{i \neq j \in N_t} \mu_{ij} x_{ijt} \leq l_t, \quad \forall t \in T \quad (5)$$

$$\sum_{i \neq j \in S} x_{ijt} \leq |S| - 1, \quad \forall t \in T, S \subseteq N_t, |S| \geq 2 \quad (6)$$

$$y_{it} \in \{0, 1\}, x_{ijt} \in \{0, 1\}, \quad \forall t \in T, i \neq j \in N_t \cup \{0\} \quad (7)$$

目标函数 (1) 包含两项, 即最小化温装钢坯和热钢锭热能损失和最小化轧制机组产能损失的加权和, 其中 $\alpha \in [0, 1]$ 为目标的权重系数. 约束 (2) 表示轧批 i 只能分配到一个时间槽. 约束 (3) 表示如果轧批 i 包含在时间槽 t 内, 则在该时间槽内, 轧批 i 有且仅有一个紧前和紧后轧批; 特别当轧批 i 处于第一个位置或最后一个位置时, 用变量 x_{0it} 和 x_{i0t} 表示. 约束 (4) 表示每个时间槽内最多只能安排一个轧批序列. 约束 (5) 为时间槽的容量约束, 即安排到时间槽内轧批的加工时间以及相邻轧批之间的机架切换时间之和不能超过时间槽长度. 约束 (6) 限制了同一时间槽内轧批排序的可行性, 即要求被分配到给定时间槽 t 的任意轧批子集 S 都不会排序成回环. 约束 (7) 给出决策变量的取值范围.

以某大型钢铁企业实际生产计划数据为例, 考虑计划期有 12 个时间槽, 50 个轧坯, 每个时间槽内候选轧批数平均为 15 个, 则该模型包括约 3 000 个 0-1 变量, 30 000 个约束, 是一个大规模整数规划模型, 使用商业优化软件如 CPLEX 很难在有限的时间内直接求得最优解甚至可行解. 因此, 需要在探究问题结构特征的基础上, 定制化设计高效求解算法.

2 求解算法

列生成是求解大规模线性规划问题的一个有效方法. 基本思想源于 Dantzig 和 Wolfe 基于凸规划理论对一类具有角形结构的线性规划模型提出一种分解策略, 即 Dantzig-Wolfe 分解^[14]. Gilmore 等^[15] 首次将列生成技术应用于求解一维切割问题. Desrochers 等^[16] 将列生成嵌入分支-定界框架中, 构成分支-定价算法, 用以求解带有时间窗口的路径问题. 伴随着计算机硬件以及高性能商业线性规划软件的发展, 基于列生成的分支-定价算法已成功应用于许多复杂的组合优化问题, 例如任务调度问题^[17-18]、批量计划问题^[19-20]、车辆路线问题^[21-22]、切割/装箱问题^[23-24] 等.

本节首先使用 Dantzig-Wolfe 分解技术将原问题分解为主问题和子问题, 然后提出分支-定价算法框架, 最后从影响算法性能的关键要素出发, 针对主问题求解方法、价格子问题求解方法和分支策略进行设计并提出改进策略.

2.1 Dantzig-Wolfe 分解

应用 Dantzig-Wolfe 分解技术, 将 IP 模型分解为主问题和子问题. 主问题包含目标函数 (1) 和约束 (2), 子问题包含约束 (3)~(7) 及一个辅助目标函数.

定义 ω 为一个可行子调度, 描述为在一个时间槽内满足子问题约束 (3)~(7) 的轧批序列. 令 Ω_t 为时间槽 t 的所有可行子调度集. 时间槽 $t \in T$ 的可行子调度 $\omega \in \Omega_t$ 由二元常量 η_{ijt}^ω 描述, 即 $\eta_{ijt}^\omega = 1$ 表示轧批 i 和 j 同时包含在可行子调度 ω 中且轧批 j 紧接 i 之后轧制, 否则 $\eta_{ijt}^\omega = 0$. 特别地, $\eta_{0jt}^\omega = 1$ 和 $\eta_{j0t}^\omega = 1$ 分别表示轧批 j 是可行子调度 ω 内第一个和最后一个轧批. 按 Vanderbeck 提出的离散化 Dantzig-Wolfe 分解法^[25], IP 模型的变量 x_{ijt} 和 y_{it} 可表示为所有可行子调度的二元凸组合, 即:

$$x_{ijt} = \sum_{\omega \in \Omega_t} \eta_{ijt}^\omega \lambda_\omega, \quad \forall t \in T, i \neq j \in N_t \cup \{0\} \quad (8)$$

$$y_{it} = \sum_{\omega \in \Omega_t} \sum_{j \in N_t \cup \{0\} \setminus \{i\}} \eta_{ijt}^\omega \lambda_\omega, \quad \forall t \in T, i \in N_t \quad (9)$$

$$\sum_{\omega \in \Omega_t} \lambda_\omega = 1, \quad \forall t \in T \quad (10)$$

$$\lambda_\omega \in \{0, 1\}, \quad \forall t \in T, \omega \in \Omega_t \quad (11)$$

将式 (8)~(11) 代入式 (1) 和式 (2), 整理后得到集划分 (Set partitioning) 模型, 称为主问题 (Main problem, MP):

$$\min \sum_{t \in T} \sum_{\omega \in \Omega_t} c_\omega \lambda_\omega \quad (12)$$

$$\text{s.t. } \sum_{t \in T} \sum_{\omega \in \Omega_t} a_{i\omega} \lambda_{\omega} = 1, \quad \forall i \in N \quad (13)$$

$$\sum_{\omega \in \Omega_t} \lambda_{\omega} = 1, \quad \forall t \in T \quad (14)$$

$$\lambda_{\omega} \in \{0, 1\}, \quad \forall t \in T, \omega \in \Omega_t \quad (15)$$

其中, 二元变量 λ_{ω} 决定子调度 ω 是否被选择, $a_{i\omega} = \sum_{j \in N_t \cup \{0\} \setminus \{i\}} \eta_{ijt}^{\omega}$ 表示轧批 i 在子调度 ω 中出现的次数, $c_{\omega} = \alpha \sum_{i \in N_t} c_{it} a_{i\omega} + (1-\alpha) \sum_{i \neq j \in N_t} w \mu_{ij} \eta_{ijt}^{\omega}$ 表示子调度 ω 包含的轧批总能耗损失费用与子调度 ω 中相邻轧批所需的机架切换导致的产能损失费用的加权和. 由于不包含任何轧批的空子调度也是任意时间槽 t 的可行子调度, 因此, 约束 (14) 的 “=” 约束可以改为 “ $\leq$ ” 约束. 约束 (14) 取严格不等式时, 表示时间槽 t 选择了空子调度.

从结构上看, IP 模型比 MP 模型更简单, 但 MP 仍然难以直接求解. 这是由于对每个时间槽 t 而言, 可行子调度集 Ω_t 的基数为问题规模的指数形式, MP 为超大规模整数规划问题. 松弛 λ 变量的整数性约束, 可获得 MP 的线性松弛, 记为 LMP (Linear-relaxation of master problem). 对 LMP 而言, 列生成算法是一个很好的选择, 它从一个包含部分列变量的限制主问题 (Restricted linear-relaxation of master problem, RLMP) 出发, 通过 RLMP 和价格子问题 (Subproblem, SP) 之间的迭代来获得 LMP 的最优解.

设 π 和 σ 分别是 RLMP 中对应约束 (9) 和约束 (10) 的对偶解, 为寻找负消减费用列, 需要求解以下价格子问题 (SP).

$$\min \left\{ c_{\omega} - \sum_{i \in N_t} a_{i\omega} \pi_i - \sigma_t \mid t \in T, \omega \in \Omega_t \right\} \quad (16)$$

SP 可以按时间槽分解为 $|T|$ 个子问题, 每个子问题记为 SP_t .

$$\min \left\{ c_{\omega} - \sum_{i \in N_t} a_{i\omega} \pi_i - \sigma_t \mid \omega \in \Omega_t \right\} \quad (17)$$

将 c_{ω} 表达式代入并整理得到:

$$\min \left\{ \sum_{i \in N_t} (\alpha c_{it} - \pi_i) a_{i\omega} + (1-\alpha) \sum_{i \neq j \in N_t} w \mu_{ij} \eta_{ijt}^{\omega} - \sigma_t \mid \omega \in \Omega_t \right\} \quad (18)$$

在列生成迭代中, 需要依次求解 $|T|$ 个子问题. 如果对任意时间槽 t 都有 $\text{opt}(SP_t) \geq 0$, 则说明已经求得 LMP 问题的最优解, 否则, 需要将 SP_t 最优解对应的列加入 RLMP.

2.2 分支-定价算法

当 LMP 的最优解是整数时, 就获得了原问题的最优解. 而大多数时候 LMP 的最优解不满足变量的整数性要求, 这时仅获得了原问题的下界, 但 LMP 能够提供比 IP 线性松弛 (记为 LIP) 更紧的下界. 这是因为 LMP 仅松弛了 λ 变量, 却保持子问题解 η 的整数性要求, 而 LIP 完全松弛了 x, y 变量. 由式 (8) 和式 (9) 可知, x, y 变量可等价表达为 λ 与 η 的乘积和, 由此可以看出 LMP 比 LIP 松弛的信息要少, 因此求解 LMP 能够为 MP 提供紧的下界. 这表明列生成算法可以作为一种高效的定界机制, 执行分支搜索过程以获得 MP 的整数最优解.

分支-定价本质上就是列生成求下界加上分支搜索寻找整数最优解的过程, 算法流程如图 2 所示, 其中定价是指以对偶解为输入生成新列的过程. 分支-定价算法的计算复杂性包含三个层次求解过程的复杂性, 第 1 层是分支树搜索过程的复杂性, 第 2 层是在分支节点上采用列生成算法求解原问题线性松弛的复杂性, 第 3 层是列生成一次迭代中求解子问题的复杂性. 由于第 2 层的列生成算法本质上是单纯形算法的一种扩展, 在最坏情况下单纯形算法的计算复杂性是指数形式, 因此在理论上分支-定价算法的计算复杂性也是关于问题规模的指数形式. 虽然单纯形算法具有指数形式计算复杂性, 但对于实际线性规划问题, 单纯形算法几乎不会遇到最坏情况, 通常只需要 $4m \sim 6m$ 的迭代步数来完成计算, 其中, m 表示线性规划问题的约束个数. 类似地, 分支-定价算法在实际应用中也几乎不会遇到最坏情况, 因此算法性能需要通过数值实验来验证. 通过上述复杂性分析可以发现, 影响分支-定价算法性能的要害主要包括分支树搜索过程 (分支策略和节点选择策略)、主问题和价格子问题的求解方法. 常见节点选择策略有深度优先、广度优先和最好下界, 本文采用深度优先和最好下界相结合的方式, 即当深度优先不可继续进行, 回溯到最好下界节点. 下文针对主问题求解方法、价格子问题求解方法、分支策略分别进行重点阐述并提出相应的改进策略.

2.3 主问题的求解方法

列生成是求解 LMP 模型的一种精确算法. 但是由于 LMP 模型的退化导致在列生成迭代过程中对偶解存在严重振荡现象, 进而导致在迭代过程的后阶段收敛速度慢, 这一现象称为列生成算法的尾效应 (Tailing-off effect). 因此, 在主问题求解方面为避免单一列生成算法的尾效应, 提出列生成和拉格朗日松弛混合求解策略, 即在列生成算法的迭代过程中基于拉格朗日松弛来估算原问题的下界, 试图在列生成算法满足终止条件前, 找到原问题有效的

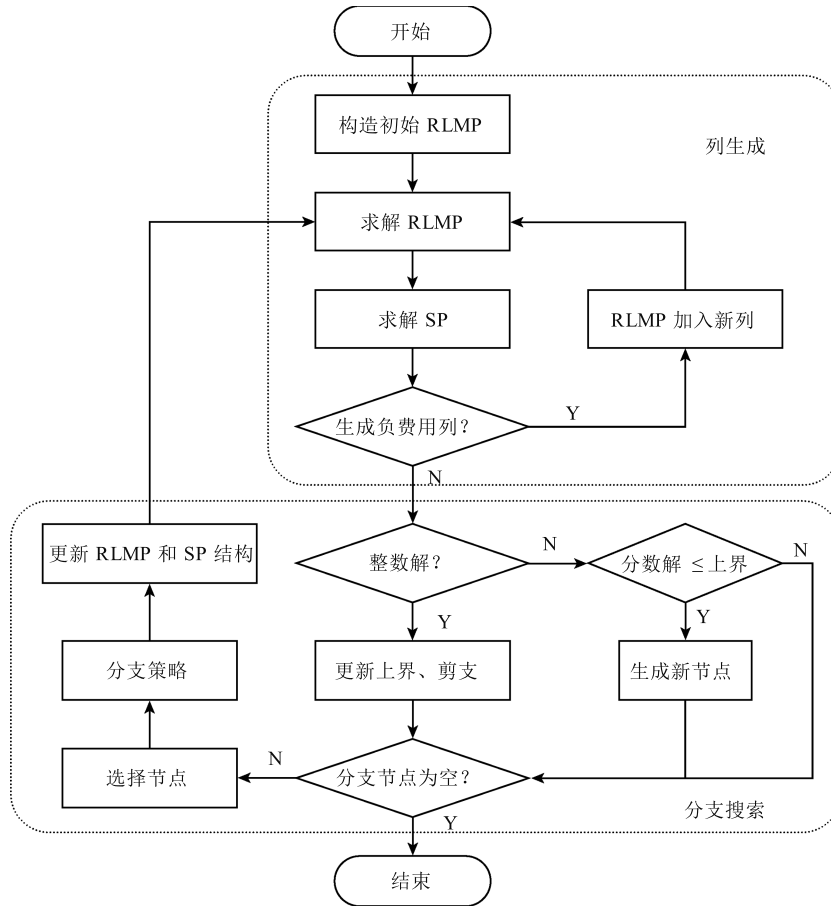


图2 分支-定价算法流程图

Fig. 2 The flow chart of branch-and-price algorithm

下界, 从而在分支节点上加速下界的计算过程.

引入乘子向量 $\boldsymbol{\eta} = \{\eta_1, \eta_2, \dots, \eta_n\} \in \mathbf{R}^n$, 其中 η_i 对应于约束 (13) 中的第 i 行, 将约束 (13) 松弛到目标函数 (12) 中, 形成下面的 LR (Lagrangian relaxation) 问题:

$$\begin{aligned} \min & \sum_{t \in T} \sum_{\omega \in \Omega_t} \left(c_{\omega} - \sum_{i \in N} a_{i\omega} \eta_i \right) \lambda_{\omega} + \sum_{i \in N} \eta_i \\ \text{s.t.} & (14), (15) \end{aligned} \quad (19)$$

LR 问题可以分解为 $|T|$ 个子问题, 每个对应一个时间槽 t , 记为 LR_t

$$\delta_t(\boldsymbol{\eta}) = \min \left\{ c_{\omega} - \sum_{i \in N} a_{i\omega} \eta_i \mid \omega \in \Omega_t \right\} \quad (20)$$

令 LR 乘子 $\boldsymbol{\eta}$ 等于 RLMP 的对偶解 $\boldsymbol{\pi}$, 则可得 MP 问题的有效下界:

$$LB = \sum_{t \in T} \min(\delta_t(\boldsymbol{\pi}), 0) + \sum_{i \in N} \pi_i \leq \text{opt}(\text{MP}) \quad (21)$$

不难发现, 当 LR 乘子为对偶解 $\boldsymbol{\pi}$ 时, 除常数 σ_t 外, LR_t 与 SP_t 等价. 因此, 在列生成算法的每一步迭代中, 都可通过式 (21) 计算出 MP 的下界.

在分支树根节点处, 当 RLMP 最优值与由式 (21) 计算出的 LB 之间的间隙小于一个较小的预定值 (如 0.1%) 时, 可将 LB 看作下界, 提前终止列生成算法, 从而避免尾效应. 在分支树其他节点处, 同样在列生成算法迭代过程中由式 (21) 计算下界 LB , 当 LB 大于或等于当前最好整数解的目标值时, 也可提前终止列生成算法, 并将该节点剪支.

2.4 价格子问题的求解方法

由式 (18) 可以看出, 针对每个时间槽 t , 价格子问题 SP_t 本质上是从 N_t 中寻找一组轧批序列, 使得轧批序列中所有轧批的加工时间以及相邻轧批之间的机架切换时间之和不能超过时间槽长度, 目标是使得轧批序列对应的削减费用最小. 为求解价格子问题, 将其等价转换为一个网络优化问题, 然后设计求解网络优化问题的动态规划 (Dynamic programming, DP) 算法, 并通过挖掘子问题本身特性, 提出

了基于占优规则和标号下界计算来消除无效状态空间, 加速求解过程.

由于轧批 $i \in N \setminus N_t$ 不允许被分配到时间槽 t , 显然有 $a_{i\omega} = 0, \forall i \in N \setminus N_t, \omega \in \Omega_t$. 因此在求解 SP_t 时仅需考虑轧批 N_t . 为了便于表达, 下文仍使用全部轧批 N 来描述子问题的算法, 但在实现算法的编码中, 对任意子问题已令 $N = N_t$. 定义赋权有向图 $G = (V, A)$, 其中, 点集 $V = N \cup \{0, n+1\}$, 0 表示起点, $n+1$ 表示终点, 其他顶点对应于轧批集 N , 弧集 $A = \{(0, i), (i, n+1) | i \in N\} \cup \{(i, j) | i \neq j \in N\}$. 对任意弧 $(i, j) \in A$, 定义 τ_{ij} 和 θ_{ij} 分别为从顶点 i 到 j 的时间和费用, 其中, $\tau_{0i} = p_i, \theta_{0i} = \alpha c_{it} - \pi_i, \tau_{i, n+1} = 0, \theta_{i, n+1} = 0, \tau_{ij} = p_i + \mu_{ij}, \theta_{ij} = \alpha c_{jt} + (1 - \alpha)w\mu_{ij} - \pi_j, \forall i \neq j \in N$. 令 $(0, i_1, i_2, \dots, i_s, 0)$ 是网络优化问题的一条可行路径, 则按 θ_{ij} 的定义可得路径总费用为 $\sum_{k=1}^s (\alpha c_{i_k t} - \pi_{i_k}) + (1 - \alpha) \sum_{k=1}^{s-1} w\mu_{i_k i_{k+1}}$. 由式 (18) 不难看出, 轧批序列 $i_1, i_2, \dots, i_s$ 对应的子调度削减费用也为 $\sum_{k=1}^s (\alpha c_{i_k t} - \pi_{i_k}) + (1 - \alpha) \times \sum_{k=1}^{s-1} w\mu_{i_k i_{k+1}}$. 因此, SP_t 可以等价归结为以下网络优化问题: 在 G 上找到一条从起点 0 到终点 $n+1$ 的初等路径, 使得总访问时间不超过 l_t , 并且所经弧的费用之和最小.

本文设计求解上述网络优化问题的 DP 算法, 基本思想是从有向图 G 上的顶点 0 出发, 在顶点上通过标号更新和消除的方式逐步探寻满足条件的最小费用初等路. 在给出 DP 算法步骤之前, 首先提出一些定义和性质, 作用是限制子问题的无效搜索空间, 加速 DP 算法的收敛速度.

定义 1. 从起点 0 到顶点 i 的初等路 ω 的标号记为 (R_i^ω, C_i^ω) , 其中, $R_i^\omega = (\zeta^\omega, y_1^\omega, \dots, y_n^\omega)$ 为初等路 ω 中最后一个顶点 i 的访问时间, $(y_1^\omega, \dots, y_n^\omega)$ 表示顶点访问标识向量 (如果顶点 j 已被初等路 ω 访问, 取 $y_j^\omega = 1$; 否则, $y_j^\omega = 0$), C_i^ω 为初等路经过的顶点与弧的费用之和.

定义 2. 设 $(R_i^{\omega_1}, C_i^{\omega_1})$ 和 $(R_i^{\omega_2}, C_i^{\omega_2})$ 为顶点 i 上的两个标号, 当且仅当 $C_i^{\omega_1} \leq C_i^{\omega_2}, \zeta_i^{\omega_1} \leq \zeta_i^{\omega_2}, y_i^{\omega_1} \leq y_i^{\omega_2}, \forall j \in N$ 成立, 并至少存在一个严格不等式时, 则称 $(R_i^{\omega_1}, C_i^{\omega_1})$ 比 $(R_i^{\omega_2}, C_i^{\omega_2})$ 占优, 记为 $(R_i^{\omega_1}, C_i^{\omega_1}) \prec (R_i^{\omega_2}, C_i^{\omega_2})$.

为缩减 DP 算法的状态空间, 提出基于标号占优规则的性质来抑制顶点上标号的快速增长.

性质 1. 在 DP 算法的标号更新过程中, 删除被占优标号不会影响 SP_t 最优解的获得.

证明. 假设顶点 i 上 $(R_i^{\omega_1}, C_i^{\omega_1}) \prec (R_i^{\omega_2}, C_i^{\omega_2})$, 显然任何由 $(R_i^{\omega_2}, C_i^{\omega_2})$ 延展得到的新标号的费用都不比由 $(R_i^{\omega_1}, C_i^{\omega_1})$ 延展得到的新标号的费用大, 因此删除 $(R_i^{\omega_2}, C_i^{\omega_2})$ 不影响 SP_t 最优解的获得. $\square$

定义 2 是文献中常见的标号占优规则的定义, 下面提出与问题结构特征相关的占优规则的定义.

定义 3. 设 $(R_i^{\omega_1}, C_i^{\omega_1})$ 和 $(R_h^{\omega_2}, C_h^{\omega_2})$ 为顶点 i 和顶点 h 上标号, 当且仅当轧批 i 和 j 的坯型规格相同, $C_i^{\omega_1} \leq C_h^{\omega_2}, \zeta_i^{\omega_1} \leq \zeta_h^{\omega_2}, y_i^{\omega_1} \leq y_h^{\omega_2}, \forall j \in N$ 成立, 并至少存在一个严格不等式时, 则称 $(R_i^{\omega_1}, C_i^{\omega_1})$ 比 $(R_h^{\omega_2}, C_h^{\omega_2})$ 占优, 记为 $(R_i^{\omega_1}, C_i^{\omega_1}) \prec (R_h^{\omega_2}, C_h^{\omega_2})$.

定义 3 利用了参数 μ_{ij} 仅与坯型规格相关的性质. 基于定义 3 的占优规则也适用于性质 1, 并且使得更多的标号被消除. 除标号占优规则外, 本文提出标号下界计算方法来消减 DP 算法无效状态空间.

性质 2. 令 $U = \{j \in N | y_j^\omega = 0\}$ 为标号 (R_i^ω, C_i^ω) 对应的初等路 ω 中未访问的顶点集, 对 $\forall j \in U$, 令 $w_j = \min\{\tau_{ij} | (i, j) \in A\}$, $f_j = \min\{\theta_{ij} | (i, j) \in A\}$, 则 $C_i^\omega + \min\{\sum_{j \in U} z_j f_j | \sum_{j \in U} z_j w_j \leq l_t - \zeta^\omega, z_j \in [0, 1], \forall j \in U\}$ 是标号 (R_i^ω, C_i^ω) 的下界.

证明. 令标号 (R_j^ω, C_j^ω) 对应由标号 (R_i^ω, C_i^ω) 延展得到的可行初等路, 延展部分的子路记为 $(i = h_0, h_1, \dots, h_a = j)$, 则 $C_j^\omega = C_i^\omega + \sum_{s=1}^a \theta_{h_{s-1} h_s}$. 可行路径 $\bar{\omega}$ 显然满足 $\zeta^\omega = \zeta^\omega + \sum_{s=1}^a \tau_{h_{s-1} h_s} \leq l_t$. 由于 $\bar{\omega}$ 是初等路, 有 $h_s \in U, \forall s \in \{1, \dots, a\}$. 由 $f_j = \min\{\theta_{ij} | (i, j) \in A\}, \forall j \in U$ 可知, $C_i^\omega + \sum_{s=1}^a f_{h_s} \leq C_j^\omega$. 由 $w_j = \min\{\tau_{ij} | (i, j) \in A\}, \forall j \in U$ 可知, $\sum_{s=1}^a w_{h_s} \leq l_t - \zeta^\omega$. 因此, 对由标号 (R_i^ω, C_i^ω) 延展得到的任意初等路 $\bar{\omega}$, 都有 $C_j^\omega \geq C_i^\omega + \min\{\sum_{j \in U} z_j f_j | \sum_{j \in U} z_j w_j \leq l_t - \zeta^\omega, z_j \in [0, 1], \forall j \in U\}$. 考虑背包问题的线性松弛, 可以进一步得到 $C_j^\omega \geq C_i^\omega + \min\{\sum_{j \in U} z_j f_j | \sum_{j \in U} z_j w_j \leq l_t - \zeta^\omega, z_j \in [0, 1], \forall j \in U\}$. $\square$

性质 3. 在 DP 算法的标号更新过程中, 记标号 (R_i^ω, C_i^ω) 的下界为 $LB(R_i^\omega, C_i^\omega)$, 如果存在另外一个标号 $(R_j^{\omega^*}, C_j^{\omega^*})$, 使得 $\min\{C_j^{\omega^*}, 0\} < LB(R_i^\omega, C_i^\omega)$ 成立, 则删除标号 (R_i^ω, C_i^ω) 不影响列生成算法收敛.

证明. 当 $C_j^{\omega^*} \geq 0$ 时, 满足不等式 $\min\{C_j^{\omega^*}, 0\} < LB(R_i^\omega, C_i^\omega)$ 的标号 (R_i^ω, C_i^ω) 下界 $LB(R_i^\omega, C_i^\omega) > 0$, 说明由基于标号 (R_i^ω, C_i^ω) 延展所得到的任何新标号对应的初等路费用都大于 0, 那么其对应列的消减费用非负, 不必加入 RLMP. 当 $C_j^{\omega^*} < 0$ 时, 满足不等式 $\min\{C_j^{\omega^*}, 0\} < LB(R_i^\omega, C_i^\omega)$ 的标号 (R_i^ω, C_i^ω) 的下界 $LB(R_i^\omega, C_i^\omega) > C_j^{\omega^*}$, 因而 (R_i^ω, C_i^ω) 延展得到的所有标号对应的初等路费用都大于 $C_j^{\omega^*}$, 那么基于标号 (R_i^ω, C_i^ω) 延展的新路径不可能为 SP_t 最优解, 因此, 删除 (R_i^ω, C_i^ω) 也不影响列生成算法的收敛. $\square$

基于标号更新的 DP 算法的伪代码如下, 其中, P 表示待处理的顶点列表, Λ_i 是顶点 i 上的标号列表, C^{inc} 是最小标号费用.

```

1) Initialization
2)  $\Lambda_0 \leftarrow \{(0, 0, \dots, 0, 0)\}$ 
3) For all  $i \in N$  do
4)    $\Lambda_i \leftarrow \emptyset$ 
5) End for
6)  $C^{\text{inc}} \leftarrow \infty, P \leftarrow \{0\}$ 
7) While  $P \neq \emptyset$  do
8)    $i = \arg \min_{h \in P} \{\zeta^\omega | (R_h^\omega, C_h^\omega) \in \Lambda_h\}$ 
9)   For all  $j | (i, j) \in A$  do
10)    For all  $(R_j^\omega, C_j^\omega) \in \Lambda_i$  do
11)     If  $y_j^\omega = 0$  and  $\zeta^\omega + \tau_{ij} \leq l_t$  then
12)       $(R_j^\omega, C_j^\omega) \leftarrow (\zeta^\omega + \tau_{ij}, y_1^\omega, \dots, y_n^\omega, C_h^\omega)$ ,
        其中,  $y_j^\omega = 1, y_h^\omega = y_h^\omega, \forall h \in N \setminus \{j\}$ 
13)      If  $\text{LB}(R_j^\omega, C_j^\omega) < \min \{C^{\text{inc}}, 0\}$  then
14)        If  $(R_j^\omega, C_j^\omega)$  不被  $\Lambda_j$  中的其他标号占优 then
15)           $\Lambda_j \leftarrow \Lambda_j \cup \{(R_j^\omega, C_j^\omega)\}$ 
16)          删除  $\Lambda_j$  中被  $(R_j^\omega, C_j^\omega)$  占优的其他标号
17)           $C^{\text{inc}} \leftarrow \min \{C_j^\omega, C^{\text{inc}}\}$ 
18)        End if
19)      End if
20)    End if
21)   $\Lambda_i \leftarrow \Lambda_i \setminus (R_i^\omega, C_i^\omega)$ 
22) End for
23) If  $\Lambda_j \neq \emptyset$  then
24)    $P \leftarrow P \cup \{j\}$ 
25) End if
26) End for
27)  $P \leftarrow P \setminus \{i\}$ 
28) End while

```

所设计的 DP 算法及改进策略是直接用于求解由子问题等价归结的一类网络优化问题, 因此, 当任意其他问题的列生成子问题可归结为上述网络优化问题时, 本文所提出的 DP 算法及改进策略也是有效的, 说明改进策略具有很强的移植性。

2.5 分支策略

分支策略, 即分支变量的选择, 是分支-定价算法中一个重要的组成部分, 需要遵循两个原则: 1) 恰当的划分解空间来确保分支树的平衡; 2) 不增加价格子问题的求解难度. 在遵循这两个原则的基础上, 定义分支变量 β 如下:

$$\beta_{it} = \sum_{\omega \in \Omega_t} a_{i\omega} \lambda_\omega, \quad \forall i \in N, t \in T \quad (22)$$

由式 (22) 和式 (13) 可推导出如下关系:

$$\beta_{it} = \sum_{\omega \in \Omega_t} a_{i\omega} \lambda_\omega \leq \sum_{t \in T} \sum_{\omega \in \Omega_t} a_{i\omega} \lambda_\omega = 1, \quad \forall i \in N, t \in T \quad (23)$$

其中, β_{it} 表示轧批 i 到时间槽 t 的分配量, 式 (23) 表明 $\beta_{it} \in [0, 1]$.

在任意分支节点上, LMP 的最优解 λ 及对应的 β 变量存在以下四种可能情形:

情形 1. LMP 最优解 λ 对应的目标值不小于当前最好整数解. 在这种情形下, 不需对当前节点进行分支, 可以直接剪支.

情形 2. LMP 的最优解 λ 对应的目标值小于当前最好整数解, 并且为整数. 由式 (22) 不难得出任意 β_{it} 都等于 0 或 1. 在这种情形下, 更新当前最好解并剪支.

情形 3. LMP 的最优解 λ 对应的目标值小于当前最好整数解, 为分数且某些 β 变量也为分数. 在这种情形下, 需要对当前的节点进行分支. 首先按以下规则选取分支变量, 即 $|\beta_{i^*t^*} - 0.5| = \min |\beta_{it} - 0.5|$. 对于选定的分支变量 $\beta_{i^*t^*}$, 构建两个分支节点:

左节点上, 令 $\beta_{i^*t^*} = 0$, 表示轧批 i^* 不能分配到时间槽 t^* . 在该节点上求解时间槽 t^* 的子问题 SPP_{t^*} 时, 仅需要从有向图 G 删除顶点 i^* 以及与之关联的弧. 该节点的初始 RLMP 继承父节点 RLMP 中除满足关系 $a_{i^*\omega} = 1, \omega \in \Omega_{t^*}$ 之外的列.

右节点上, 令 $\beta_{i^*t^*} = 1$, 表示轧批 i^* 分配到时间槽 t^* . 在该节点上求解其他时间槽 ($t \neq t^*$) 的价格子问题 SP_t 时, 仅需要从有向图 G 删除顶点 i^* 以及与之关联的弧. 该节点的初始 RLMP 继承父节点 RLMP 中满足关系 $a_{i^*\omega} = 1, \omega \in \Omega_{t^*}$ 的列以及满足关系 $a_{i^*\omega} = 0, \omega \in \Omega_t, t \in T \setminus \{t^*\}$ 的列.

情形 4. LMP 的最优解 λ 对应的目标值小于当前最好整数解, λ 为分数且 β 为整数. 由以下引理可以证明这种情形不会发生.

引理 1. 如果在某个分支节点上所有 β 变量都是整数, 那么该分支节点的最优解也是为整数, 即所有 λ 变量都是整数.

证明. 由式 (23) 知, $\beta_{it} \in [0, 1]$, β 变量都是整数则说明 $\beta_{it} \in \{0, 1\}$. 令 $\Omega_t^* = \{\omega \in \Omega_t | \lambda_\omega > 0\}$, $N_t^* = \{i \in N | \beta_{it} = 1\}$. 对任意 $\omega \in \Omega_t^*, i \in N_t^*$, 由 $\beta_{it} = \sum_{\omega \in \Omega_t^*} a_{i\omega} \lambda_\omega = 1$ 和 $\sum_{\omega \in \Omega_t^*} \lambda_\omega \leq 1$ 可知, $a_{i\omega} = 1$. 这说明 Ω_t^* 中的所有子调度都包含相同的轧批. 由于 λ 是 LMP 的最优解, 因此 Ω_t^* 中的所有子调度的费用都相等, 进而说明 Ω_t^* 中的子调度完全同构. 由于 Ω_t 中的子调度互不相同, 显然 $\Omega_t^* \subseteq \Omega_t$ 最多只包含一个子调度, 这表明 λ 一定为整数. $\square$

引理 1 说明 β 作为分支变量可以划分整数解空间但又不会割舍最优解, 验证了分支策略的有效性.

3 计算实验

本文所提出的分支-定价算法使用 VC++6.0 语言编码, 在个人计算机 (Intel Core (TM)2 Quad 2.83 GHz CPU 和 3.25 GB 内存) 上进行性能测试.

3.1 实验数据

通过三类测试数据评价所提出算法的性能. 第 1 组和第 2 组是以国内某大型钢铁企业提供的实际生产数据进行扩展得到, 主要用于测试算法以及改进策略的求解能力和效果. 第 3 组以该企业的实际数据测试算法在实际应用中的效用. 在该企业, 计划编制周期为 7 天, 初轧产线每天约轧制 3 500 吨钢坯和钢锭, 约 70% 的钢坯采用热装. 轧批通常为炉钢水铸造出的钢坯, 约 150 吨. 一个计划期内初轧产线需要轧制约 163 个轧批, 其中热装轧批约 113 个, 温装、冷装及钢锭轧批约 50 个. 每天由热装钢坯产生 2~3 个时间槽, 每个时间槽的长度约为 3~5 小时. 经初轧和精轧机组轧制后, 最终坯型有大方坯 (厚度: 160~450 mm), 小方坯 (厚度: 90~142 mm), 圆坯 (直径: $\phi 75 \sim \phi 185$ mm). 不同坯型之间的机架切换时间为 0~60 分钟. 结合上述实际数据特征和数学模型可知影响问题复杂性的因素主要包括轧批数和时间槽数. 第 1 组小规模算例, 轧批数和时间槽数的组合分别取 $\{(20, 5), (25, 6), (30, 7), (35, 8), (40, 10)\}$, 用于比较商业优化软件 CPLEX 和分支一定价算法的求解能力. 第 2 组中规模算例, 轧批数 $\{50, 60, 80, 100\}$, 时间槽数 $\{12, 15, 18, 20\}$. 上述参数的不同组合产生了 16 种问题规模, 每种规模随机产生 20 个算例, 共测试了 320 个算例. 中规模算例与实际问题规模接近且覆盖情形更广, 用于测试分支一定价算法及改进策略的有效性和鲁棒性. 第 3 组算例包含 5 组实际数据, 轧批数和时间槽数组合分别为 $\{(52, 12), (56, 12), (55, 13), (56, 13), (58, 14)\}$.

模型中多目标的权重参数 α 的取值是通过预实验后与钢铁企业熟悉业务的计划员详细讨论而设定的. 在预实验中, 将 α 的取值分别设定为 0, 0.05, 0.1, $\dots$, 0.9, 0.95, 1, 然后采用分支一定价算法分别求得同组数据在不同模型参数设定下两项优化指标, 即能耗费用和机架切换时间. 对不同的取值下的能耗费用和机架切换时间的计算结果进行比对分析,

发现随着 α 取值的增加, 能耗费用逐渐减小, 在 $\alpha = 0.9$ 之后能耗费用变化趋近稳定. 由于在实际排产过程中能耗费用是首要指标, 机架切换时间是次要指标, 因此计划员认定 $\alpha = 0.9$ 能够客观反映实际所期望的优化效果.

3.2 实验结果与分析

将商业优化软件 CPLEX 直接求解 IP 模型作为 Benchmark 算法, 比较分支一定价算法和 CPLEX 对小规模算例的求解能力. 表 1 从线性松弛下界 (LMP/LIP)、最优解/上界 (OPT/UB)、整数间隙 (GAP)、计算时间 (CPU) 等方面比较二者的计算性能. 从表 1 可以看出, 前 4 个算例, 二者都能获得问题的最优解, 但分支一定价算法的计算时间有优势. 随着问题规模增大, CPLEX 计算时间快速增长, 对最后一个算例 CPLEX 在 2 小时内未能找到最优解, 而分支一定价算法快速获得最优解. 说明即使对小规模的复杂生产调度问题, 商业优化软件也难以有效求解, 需要针对问题特征定制化设计高效算法. 从线性松弛下界和整数间隙来看, LMP 能够获得比 LIP 更紧的下界, 这也是分支一定价算法能够快速获得最优解的主要原因.

为进一步分析分支一定价算法的求解能力, 针对中等规模算例进行测试. 计算结果如表 2 所示, 分别从整数间隙 (GAP), 分枝节点数 (Nodes), 计算时间 (CPU) 来描述算法性能. 从表 2 可以看出:

- 1) 所有测试实例的平均整数间隙不超过 1.5%, 最大整数间隙不超过 4.0%, 验证了 Dantzig-Wolfe 分解模型的线性松弛能为原问题提供紧的下界.
- 2) 分支一定价算法能够在合理计算时间内通过搜索有限节点来获得所测试实际规模问题的最优解, 说明分支一定价算法对求解这类问题的有效性, 且可以扩展到求解类似的组合优化问题.
- 3) 当轧批数固定时, 随着时间槽数增加, 计算时间减少. 该现象可以解释为: 对于固定的轧批, 时间槽数的增加意味着分配到每个时间槽的轧批平均数减少, 导致价格子问题更容易求解.

表 1 分支一定价算法与 CPLEX 求解小规模算例的计算结果比较

Table 1 Comparison of computational results obtained by branch-and-price and CPLEX for small scale instances

问题规模		分支一定价				CPLEX			
轧批数	时间槽数	LMP	OPT	GAP (%)	CPU (s)	LIP	UB	GAP (%)	CPU (s)
20	5	2 479.77	2 483.8	0.16	0.11	2 278.50	2 483.8	9.01	1.17
25	6	2 991.97	3 011.7	0.66	0.17	2 782.05	3 011.7	8.25	2.76
30	7	3 693.91	3 726.4	0.88	0.43	3 501.45	3 726.4	6.42	216.72
35	8	4 026.08	4 047.0	0.52	0.79	3 842.80	4 047.0	5.31	501.53
40	10	4 622.63	4 634.8	0.26	1.18	4 473.60	4 779.6	6.84	3 600.00

表 2 分支-定价算法求解中规模算例的计算结果

Table 2 Computational results of branch-and-price for solving medium scale instances

问题规模		GAP (%)		Nodes		CPU (s)	
轧批数	时间槽数	平均	最大	平均	最大	平均	最大
50	12	1.49	2.91	21	57	2.12	7.15
50	15	1.06	1.50	13	33	1.19	5.73
50	18	0.77	1.02	10	23	0.89	3.25
50	20	1.22	3.27	11	25	0.25	0.98
60	12	0.74	2.47	26	45	12.02	81.53
60	15	1.26	3.24	27	59	8.03	46.81
60	18	1.10	2.65	21	40	4.68	22.55
60	20	1.12	3.81	19	48	3.51	10.06
80	12	1.00	2.57	24	80	196.55	497.12
80	15	0.86	1.91	31	58	119.58	254.81
80	18	1.30	2.74	18	37	40.23	150.28
80	20	0.71	1.72	11	29	10.61	76.11
100	12	0.46	1.86	66	113	233.75	714.88
100	15	0.66	2.50	24	67	185.42	325.91
100	18	0.81	3.13	40	79	100.21	255.13
100	20	1.01	3.05	22	62	51.14	190.11

4) 随着问题规模的增大, 分枝节点增多, 计算时间也增多, 这是由于大规模算例的限制主问题和价格子问题的规模也变大, 因而更难求解。

针对主问题和子问题提出改进策略目的在于加速求解过程, 提升算法的求解能力。为了验证改进策略的效用, 本文实现了 4 个版本的分支-定价算法, 并分别对中等规模算例进行了测试, 其中, M1 表示未引入任何改进策略的基本算法, M2 表示针对主问题使用了列生成与拉格朗日松弛混合求解策略, M3 表示针对子问题使用了基于标号占优规则和标号下界计算的加速策略, M4 表示同时使用了针对主问题和子问题加速策略。表 3 对比了各版本算法的平均和最大计算时间。从表 3 中可以看出, M2 和 M3 在平均和最大计算时间都比 M1 要少, 说明针对主问题和子问题的加速策略都是有效的。对于部分 M1 在 2 小时内未找到最优解的算例, M3 能找到最优解, 说明针对子问题加速策略的效果更明显, 极大提升了算法的求解能力。另外, 对比 M4 与 M2、M3 计算时间可以看出, 双重加速策略要优于单一加速策略, 但改进效果主要来源于子问题加速策略。从本质上分析, 由于在列生成算法迭代的后期, 使用标号下界计算策略能够消除大量无效标号, 从而节约了标号的存储空间, 极大地节约了子问题动态规划算

法的计算时间。使用列生成与拉格朗日松弛混合策略求解主问题能够避免尾效应, 快速估界后对无效节点及早剪支, 也能节约分支搜索时间。

为验证分支-定价算法在实际应用方面的效用, 以 5 组实际数据进行实验。采用计算机模拟钢铁企业计划员手工排产方法, 将手工排产同分支-定价算法比较。两种方法的归一化结果如表 4 所示, 能耗费用对应目标函数第 1 项, 机架切换时间对应目标函数第 2 项。结果显示分支-定价算法优于手工排产方法, 目标值平均改进达 8.85%, 能耗费用节约 8.72%, 机架切换时间节约 17.86%。从实验中可以看出能耗费用的改进量与目标值的改进量接近。这是由于在实际排产中能耗费用是首要关注目标, 计划员设定能耗费用的权重系数较大, 导致能耗费用在目标值中比重较大。轧制工序能耗费用降低在以节能降耗为导向的生产管理中具有重要的积极意义。

4 结论

热装、温装和冷装混流生产是衔接连铸和轧制工序的常见方式。本文针对混流生产模式下的轧批调度问题, 建立了整数规划模型, 利用 Dantzig-Wolfe 分解技术将其转换为等价的主问题和子问题。

表 3 主问题和子问题改进策略的性能分析
Table 3 Performance analysis of improvement strategies for master problem and subproblem

问题规模		M1 (s)		M2 (s)		M3 (s)		M4 (s)	
轧批数	时间槽数	平均	最大	平均	最大	平均	最大	平均	最大
50	12	4.69	17.12	3.70	14.29	2.33	7.94	2.12	7.15
50	15	3.65	12.85	2.98	10.93	1.25	7.16	1.19	5.73
50	18	2.04	8.64	1.77	7.57	1.01	4.05	0.89	3.25
50	20	0.87	2.84	0.67	2.03	0.29	1.22	0.25	0.98
60	12	48.25	191.13	36.77	149.03	14.84	100.93	12.02	81.53
60	15	29.45	114.49	24.14	85.50	10.12	60.32	8.03	46.81
60	18	12.47	64.05	10.36	56.81	5.82	28.13	4.68	22.55
60	20	10.82	36.24	8.70	32.50	4.33	11.71	3.51	10.06
80	12	1 086.38	—	969.15	—	217.68	641.45	196.55	497.12
80	15	690.00	2 063.25	503.93	1 733.78	124.16	315.08	119.58	254.81
80	18	322.21	1 067.64	257.22	938.52	42.79	175.27	40.23	150.28
80	20	86.17	737.05	62.47	610.48	12.48	90.52	10.61	76.11
100	12	3 400.82	—	2 444.17	—	265.16	801.17	233.75	714.88
100	15	2 619.11	—	2 006.37	—	207.73	380.22	185.42	325.91
100	18	1 514.30	—	1 227.72	—	111.29	319.95	100.21	255.13
100	20	702.27	2 960.12	525.75	2 452.29	62.02	223.47	51.14	190.11

注：表中“—”表示算法在 2 小时内未完成分支搜索过程。

表 4 分支-定价算法与手工计划的结果比较
Table 4 Comparison results between branch-and-price and the manual planning method

问题规模		目标值		能耗费用		机架切换时间	
轧批数	时间槽数	手工排产	分支-定价	手工排产	分支-定价	手工排产	分支-定价
51	12	1.0915	1.0000	1.0886	1.0000	1.3074	1.0000
56	12	1.0864	1.0000	1.0842	1.0000	1.2239	1.0000
55	13	1.0944	1.0000	1.0941	1.0000	1.1181	1.0000
56	13	1.0756	1.0000	1.0744	1.0000	1.1601	1.0000
58	14	1.0947	1.0000	1.0949	1.0000	1.0836	1.0000
	平均	1.0885	1.0000	1.0872	1.0000	1.1786	1.0000

采用列生成算法求解主问题的线性松弛为原问题提供紧的下界，并在分支树上通过设计有效的分支策略获取原问题的最优解。针对主问题，提出列生成和拉格朗日松弛混合求解策略以抑制单一列生成算法的尾效应。针对价格子问题，在动态规划算法中提出了基于占优规则和标号下界计算方法来及早消除无效状态空间，从而加速求解过程。实验结果显示所提出改进策略能够突破求解能力的限制，使得分支-定价算法在可接受计算时间内求得工业规模问题的最优解，计算性能显著优于商业优化软件和手工排产方法。由于针对主问题和子问题提出的改进策略具有较强的适用性，可扩展到其他类似优化问题。

References

1 Lopez L, Carter M W, Gendreau M. The hot strip mill production scheduling problem: a tabu search approach. *European Journal of Operational Research*, 1998, **106**(2-3): 317-335

2 Lv Zhi-Min, Xu Jin-Wu. Optimization method for hot charge rolling manufacture plan. *Journal of University of Science and Technology Beijing*, 2002, **24**(6): 675-678
(吕志民, 徐金梧. 一种适用于热送热装生产计划优化的方法. 北京科技大学学报, 2002, **24**(6): 675-678)

3 Cowling P. A flexible decision support system for steel hot rolling mill scheduling. *Computers and Industrial Engineering*, 2003, **45**(2): 307-321

4 Zhao J, Wang W, Liu Q L, Wang Z G, Shi P. A two-stage scheduling method for hot rolling and its application. *Control Engineering Practice*, 2009, **17**(6): 629-641

- 5 Pan C C, Yang G K. A method of solving a large-scale rolling batch scheduling problem in steel production using a variant of column generation. *Computers and Industrial Engineering*, 2009, **56**(1): 165–178
- 6 Lee H S, Murthy S S, Haider S W, Morse D V. Primary production scheduling at steelmaking industries. *IBM Journal of Research and Development*, 1996, **40**(2): 231–252
- 7 Cowling P, Rezi W. Integration of continuous caster and hot strip mill planning for steel production. *Journal of Scheduling*, 2000, **3**(4): 185–208
- 8 Tang L X, Liu J Y, Rong A Y, Yang Z H. A review of planning and scheduling systems and methods for integrated steel production. *European Journal of Operational Research*, 2001, **133**(1): 1–20
- 9 Lv Zhi-Min, Mu Wen-Heng, Xu Jian-Hua, Tang Di, Xu Jin-Wu. Production organization method and simulation of dual-line thin slab continuous casting and hot rolling. *Journal of University of Science and Technology Beijing*, 2005, **27**(3): 356–359
(吕志民, 牟文恒, 许剑桦, 唐荻, 徐金梧. 两流方式下薄板坯连铸连轧生产组织方法及仿真. 北京科技大学学报, 2005, **27**(3): 356–359)
- 10 Yu Gang, Tian Nai-Yuan, Xu An-Jun, He Dong-Feng. Optimization and coordination of steelmaking-hot rolling production plan. *Energy for Metallurgical Industry*, 2009, **28**(4): 6–9
(于港, 田乃媛, 徐安军, 贺东风. 炼钢-热轧生产计划的优化与协调. 冶金能源, 2009, **28**(4): 6–9)
- 11 Wang Gong-Shu, Tang Li-Xin. Modelling and optimization methods for the sequencing problem with batching decision in the continuous-casting and rolling production. *Acta Automatica Sinica*, 2012, **38**(10): 1713–1720
(汪恭书, 唐立新. 连铸-轧制生产中带有批决策的排序问题的建模与优化方法. 自动化学报, 2012, **38**(10): 1713–1720)
- 12 Chen Z L, Powell W B. Exact algorithms for scheduling multiple families of jobs on parallel machines. *Naval Research Logistics*, 2003, **50**(7): 823–840
- 13 Tang L X, Wang G S, Liu J Y. A branch-and-price algorithm to solve the molten iron allocation problem in iron and steel industry. *Computers and Operations Research*, 2007, **34**(10): 3001–3015
- 14 Dantzig G B, Wolfe P. Decomposition principle for linear programs. *Operations Research*, 1960, **8**(1): 101–111
- 15 Gilmore P C, Gomory R E. A linear programming approach to the cutting stock problem. *Operations Research*, 1961, **9**(6): 849–859
- 16 Desrochers M, Desrosiers J, Solomon M. A new optimization algorithm for the vehicle routing problem with time windows. *Operations Research*, 1992, **40**(2): 342–354
- 17 Moukrim A, Quilliot A, Toussaint H. An effective branch-and-price algorithm for the preemptive resource constrained project scheduling problem based on minimal interval order enumeration. *European Journal of Operational Research*, 2015, **244**(2): 360–368
- 18 Restrepo M I, Gendron B, Rousseau L M. Branch-and-price for personalized multiactivity tour scheduling. *INFORMS Journal on Computing*, 2016, **28**(2): 334–350
- 19 Fragkos I, Degraeve Z, De Reyck B. A horizon decomposition approach for the capacitated lot-sizing problem with setup times. *INFORMS Journal on Computing*, 2016, **28**(3): 465–482
- 20 Tang L X, Wang G S, Chen Z L. Integrated charge batching and casting width selection at baosteel. *Operations Research*, 2014, **62**(4): 772–787
- 21 Battarra M, Erdoğan G, Vigo D. Exact algorithms for the clustered vehicle routing problem. *Operations Research*, 2014, **62**(1): 58–71
- 22 Baldacci R, Mingozzi A, Roberti R, Wolfler Calvo R. An exact algorithm for the two-echelon capacitated vehicle routing problem. *Operations Research*, 2013, **61**(2): 298–314
- 23 Fleszar K. An exact algorithm for the two-dimensional stage-unrestricted guillotine cutting/packing decision problem. *INFORMS Journal on Computing*, 2016, **28**(4): 703–720
- 24 Gschwind T, Irnich S. Dual inequalities for stabilized column generation revisited. *INFORMS Journal on Computing*, 2016, **28**(1): 175–194
- 25 Vanderbeck F. On Dantzig-Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. *Operations Research*, 2000, **48**(1): 111–128

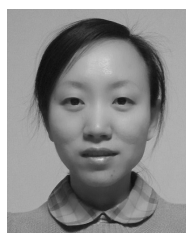


汪恭书 东北大学工业与系统工程研究所副教授. 主要研究方向为流程工业生产与物流调度, 最优化理论与方法, 决策支持系统开发.

E-mail: wanggongshu@ise.neu.edu.cn

(**WANG Gong-Shu** Associate professor at the Institute of Industrial and Systems Engineering, Northeastern

University. His research interest covers production and logistics scheduling in process industry, optimization theory and methodology, and development of decision support system (DSS).)



刘静宜 东北大学数学系讲师. 主要研究方向为数值计算方法与理论.

E-mail: liujingyi@mail.neu.edu.cn

(**LIU Jing-Yi** Lecturer in the Department of Mathematics, Northeastern University. Her research interest covers numerical computation methods and theory.)



唐立新 东北大学工业与系统工程研究所教授. 主要研究方向为生产调度, 物流与供应链管理和组合最优化. 本文通信作者.

E-mail: lixintang@mail.neu.edu.cn

(**TANG Li-Xin** Professor at the Institute of Industrial and Systems Engineering, Northeastern University. His

research interest covers production scheduling, logistics and supply chain management, and combinational optimization. Corresponding author of this paper.)