

考虑加热炉生产因素的热轧板坯轧制计划模型与算法

张卓伦^{1,2}, 张文新^{1,2}, 李铁克^{1,2}, 王柏琳^{1,2†}

(1. 北京科技大学 经济管理学院, 北京 100083; 2. 钢铁生产制造执行系统技术教育部工程研究中心, 北京 100083)

摘要: 加热炉生产是影响热轧机组利用率和轧制计划质量的重要环节之一。通过分析加热炉对热轧生产的影响, 抽取板坯标准在炉时间和出炉温度这两个关键因素, 建立了热轧板坯轧制计划的整数规划模型, 并提出了自适应邻域搜索算法。在算法中设计了约束满足策略、自适应搜索策略和反向学习邻域搜索策略, 其中约束满足策略针对目标特征和加热炉因素设计了两种值选择规则, 用于生成高质量初始解; 自适应搜索策略能够自主选择邻域结构和终止邻域搜索, 有效优化邻域结构选择方式和算法收敛速度; 反向学习邻域搜索策略基于反向学习技术增强解空间多样性, 提高全局搜索能力。基于实际生产数据设计了多种规模的实验, 验证了算法的有效性。

关键词: 轧制计划; 热轧板坯; 加热炉; 变邻域搜索; 自适应

中图分类号: TP301

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0183

Model and Algorithm for Rolling Planning of Hot-rolled Slab with Reheating Furnace Production Factors

ZHANG Zhuolun^{1,2}, ZHANG Wenxin^{1,2}, LI Tieke^{1,2}, WANG Bailin^{1,2†}

(1. School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China; 2. Engineering Research Center of MES Technology for Iron & Steel Production, Ministry of Education, Beijing 100083, China)

Abstract: Reheating furnace production is one of the important procedures that affect the utilization rate of hot rolling mills and the quality of rolling plans. By analyzing the influence of reheating furnace on hot rolling production, two key factors of slab, standard time in furnace and discharge temperature, are extracted. The integer programming model of hot-rolled slab rolling plan is established, and an adaptive neighborhood search algorithm is proposed. Constraint satisfaction strategy, adaptive search strategy and reverse learning neighborhood search strategy are designed in the algorithm. Two value selection rules of the constraint satisfaction strategy are designed for target characteristics and furnace factors to generate high-quality initial solutions; By using the adaptive search strategy, neighborhood structure can be autonomously selected and neighborhood search can be terminated autonomously, the neighborhood structure selection process and algorithm convergence speed are effectively optimized; the reverse learning neighborhood search strategy is based on the reverse learning technology to enhance the diversity of the solution space and improve the global search ability. Based on actual production data, experiments of various scales are designed to verify the effectiveness of the algorithm.

Keywords: Rolling planning; Hot-rolled Slab; Reheating furnace; Variable neighborhood search; Adaptive

0 引言

热轧板坯作为一种重要的钢铁产品, 被广泛应用于各大领域, 通常按照板坯钢种、厚度、宽度等属性将其划分为不同的品种。随着客户需求日渐趋于多品种小批量, 热轧板坯生产管理的难度也不断增大。轧制工序是热轧板坯生产过程中最后一道关

键工序, 编制轧制计划不仅需要考虑到本阶段的轧制约束, 还要兼顾其他阶段的工艺约束, 计划的质量受多方面因素的影响。因此, 如何在满足基本轧制工艺约束的前提下兼顾其他工序的影响, 特别是加热炉阶段的生产要求, 编制高质量的轧制计划, 成为了热轧板坯生产管理研究的重点。

收稿日期: 2021-01-29; 修回日期: 2021-03-30.

基金项目: 国家自然科学基金资助项目 (71701016, 71231001); 教育部人文社会科学研究青年基金项目资助 (17YJC630143); 北京市自然科学基金项目 (9174038); 中央高校基本科研业务费资助项目 (FRF-BD-20-16A)

†通讯作者. E-mail: wangbl@ustb.edu

针对轧制计划编制问题,不少学者已结合实际生产背景针对带钢、板坯、中厚板等产品生产线进行了科学研究。Chen等^[1]考虑热轧带钢生产线中的轧制工艺,将轧制计划问题划分成板坯组批和批内排序两个子问题进行求解。针对板坯轧制计划问题,Liu等^[2]将问题归结为多目标旅行商问题,Jia等^[3]则将问题归结为奖金收集车辆路径问题,张春生等^[4]从一体化计划层面,考虑炼钢工序及连铸坯传输方式等因素对轧制批量计划进行协调优化。郑忠等^[5]对比分析了带材和中厚板的轧制规程,兼顾连铸机、加热炉和轧机三大工序的生产目标和工艺约束对轧制计划编制问题进行了研究。王宸等^[6]和Hu等^[7]从电价能耗方面优化轧制计划,分别研究了考虑使用时间电价的热轧批量计划编制问题、考虑板坯间电耗变化的多目标热轧计划问题。在设备能耗和检修方面,董广静等^[8]深入研究了轧辊车削成本对轧制计划编制和调整过程的影响,汪洋等^[9]研究了周期性机器柔性检修环境下的热轧批量调度问题并证明了闲置时间最小化性质。上述研究为轧制计划相关研究打下了很好的理论基础,但与本文问题考虑的影响因素存在差异。在热轧生产线中,加热炉是铸轧工序间重要的缓冲环节,是保障热轧产品质量稳定性的关键环节,且具有高能耗的特点,是影响轧制计划编制的关键因素之一,而上述研究并未考虑加热炉生产因素对轧制计划编制的影响,其模型与方法不能直接用到本文的研究中。

近年来,加热炉的生产因素在热轧生产计划与调度管理中的重要性逐渐受到研究者的关注。杨业建等^[10]和谢金兰等^[11]在轧制计划已知的前提下,以坯料在炉时间最短为优化目标进行加热炉区调度,潘瑞林等^[12]和刘健等^[13]考虑板坯在加热炉的出炉温度、加热炉温度变化两个关键因素,对轧制计划调度问题展开研究,Li等^[14]考虑加热炉与轧机的衔接,将加热炉工艺约束引入到轧制计划编制过程中,对板坯组批和排序。尽管加热炉阶段和轧制阶段两大关键工序的生产计划与调度研究已取得一定的成果,但是很少有文献同时兼顾加热炉生产因素和轧制工艺约束。在实际生产中,加热炉虽然是辅助工序,但它是跟热轧无缝衔接的,加热炉与热轧之间无等待,热轧不允许停机待料,而且加热炉又是高耗能设备,节能降耗多是从加热炉的优化开始着手的,所以加热炉的影响不容忽视。本文通过分析影响热轧板坯轧制的加热炉生产因素,抽取关键因素引入到轧制计划编制问题模型中,并结合问题

特征设计自适应邻域搜索算法。

1 问题描述

本文针对一类典型的热轧板坯生产线展开研究,该产线包含一台加热炉和一台轧制机组。在生产过程中,待轧制的板坯为了达到指定的轧制温度,必须先进入加热炉加热,达到标准在炉时间后才允许出炉,其中,标准在炉时间是指允许板坯出炉的最早时间,板坯出炉后立即经过辊道进入轧机,由于生产工艺要求,轧机具有很强的加工连续性,不允许停机等待。此外,由于加热炉能力有限,当板坯出炉温度变化过大时,会产生待温现象,即板坯等待加热炉升温或降温到合适温度再进炉加热,进而导致能耗增加。因此,为了进一步优化轧制计划质量、提高加热炉和轧机的利用率,有必要将加热炉的生产因素考虑到热轧板坯轧制计划编制过程中。将加热炉阶段对轧制过程的影响作如下分析:

1) 板坯的标准在炉时间影响轧机利用率,轧机生产连续性很强,若前序板坯已完成轧制,而后序板坯尚未达到标准在炉时间,此时轧机只能空转等待板坯,甚至出现停机待料情况。因此,相邻板坯的标准在炉时间若有明显差异,则不易保证热轧的连续性,从而导致轧机利用率下降;

2) 板坯的标准在炉时间影响板坯的轧制开始时间,板坯必须在达到标准在炉时间且前序板坯完成轧制的情况下才能出炉,因此板坯的轧制开始时间由标准在炉时间和前序板坯轧制完成时间决定;

3) 板坯的出炉温度影响加热炉利用率和热轧能耗水平,不同钢种的板坯有不同的出炉温度,在加热炉中如果前后两块板坯的温差过大,后序板坯则需要等待加热炉变化到合适温度后才能入炉,这会降低加热炉利用率,增大能耗,所以轧制计划在编制时会要求考虑板坯的出炉温度。

综上所述,本文问题可以描述为:已知待加工板坯集和每个板坯的属性值,需要在满足板坯宽度、厚度、同宽连轧等基础约束的前提下,充分考虑板坯在加热炉阶段的标准在炉时间和出炉温度的影响,以优化轧制计划质量、满足交货期、提高加热炉和轧机利用率为目标,对板坯集进行分组和排序。结合加热炉和轧机两阶段的工艺规程,将优化目标和约束条件描述如下。

1) 优化目标,主要从订单要求、工艺要求以及设备利用率三方面考虑,具体包括:最小化板坯轧制提前期和拖期;最小化轧制单元内板坯间的属性值(宽度、厚度、出炉温度、标准在炉时间)跳跃;

最小化轧机空转时间。

2) 约束条件, 主要考虑两方面的工艺约束: 轧制工艺基本约束, 包括板坯间的属性(宽度、厚度)跳跃度限制、同宽公里数长度限制、轧制单元的总长度限制; 加热炉工艺的相关约束, 包括板坯出炉顺序和轧制顺序必须一致、板坯在加热炉达到标准在炉时间且前序板坯已完成轧制时才可出炉、相邻板坯出炉温度区间要有交集。

2 问题建模

2.1 问题假设

针对考虑加热炉生产因素的热轧板坯轧制计划编制问题, 结合问题背景作以下假设: 1) 轧制单元总数已知, 待轧板坯所有属性值及属性值跳跃限制已知, 其中宽度、厚度跳跃不区分正反向, 其他轧制工艺约束参数均已知; 2) 不考虑轧机换辊、异常中断、设备检修以及设备间的运输时间; 3) 加热炉与热轧工序间无等待时间, 板坯的入炉顺序、出炉顺序和轧制顺序均相同; 4) 本文考虑的板坯交货期不是合同交期, 是通过订单排程得到的板坯期望完工时间, 订单排程计划是根据合同交期在设备上进行的粗排, 编制轧制计划时需要尽可能的满足板坯期望完工时间, 模型中的板坯最早和最晚轧制时间, 是根据板坯轧制时长和板坯期望完工时间计算得到的; 5) 本文研究对象为板坯轧制计划编制问题, 加热炉因素是作为计划层面的影响因素来考虑的, 主要体现在板坯的出炉顺序及时间、出炉温度等重要因素上, 入炉时间作为非重要影响参数, 这里仅基于板坯出炉时间和标准在炉时间对板坯轧制开始时间进行倒推估算, 不再进行更为细节的调度优化。

2.2 符号定义

为了方便描述模型, 将符号定义如下。

1) 索引与集合。 i 表示板坯序号; k 表示轧制单元序号, $k = 1, 2, \dots, m$, m 表示轧制单元的数量; N 表示待排板坯集合, $N = 0, 1, 2, \dots, n$, n 为板坯总数量, $i = 0$ 对应虚拟板坯。

2) 参数定义。 Q_{max} 表示轧制单元长度上限; Q_k 表示轧制单元 k 中所有板坯长度之和; R 表示同宽连轧长度上限; L_i 表示板坯 i 的长度; ML 表示一个极大的正数; W_i 表示板坯 i 的宽度, W_{max} 为相邻板坯宽度跳跃最大值; TK_i 表示板坯 i 的厚度, TK_{max} 为相邻板坯厚度跳跃最大值; TE_i 表示出炉温度, δ_i 为出炉温度偏差; TL_i 表示标准在炉时间, TL_{max} 为标准在炉时间跳跃上限; p_i 为板坯 i 的轧制

时长, a_i 和 b_i 分别为最早和最晚轧制开始时间。

3) 决策变量。 Ω_i 表示板坯 i 后连续同宽连轧的累积轧制长度; t_i 为轧制开始时间, 亦即板坯出炉时间, t_i^0 为入炉时间; 0-1 变量有

$$x_{ijk} = \begin{cases} 1 & \text{轧制单元 } k \text{ 内板坯 } j \text{ 为 } i \text{ 的直接后继;} \\ 0 & \text{否则。} \end{cases}$$

$$y_{ik} = \begin{cases} 1 & \text{轧制计划 } k \text{ 包含板坯 } i; \\ 0 & \text{否则。} \end{cases}$$

$$z_{ij} = \begin{cases} 1 & \text{板坯 } i \text{ 与板坯 } j \text{ 同宽;} \\ 0 & \text{否则。} \end{cases}, z_{i0} = z_{0i} = 0.$$

4) 针对单个板坯的优化目标: t_i 为提前和拖期轧制的惩罚值, β_1 和 β_2 分别为提前和拖期惩罚系数, 计算公式为:

$$P_i^t = \beta_1 \sum_i^n \max(0, a_i - t_i) + \beta_2 \sum_i^n \max(0, t_i - b_i) \quad (1)$$

5) 针对相邻板坯的优化目标:

P_{ij}^W 表示板坯 i 到板坯 j 的宽度跳跃惩罚值, 惩罚系数为 λ^W , 计算公式为

$$P_{ij}^W = |W_i - W_j| \times \lambda^W \quad (2)$$

P_{ij}^{TK} 表示板坯 i 到板坯 j 的厚度跳跃惩罚值, 惩罚系数为 λ^{TK} , 计算公式为

$$P_{ij}^{TK} = |TK_i - TK_j| \times \lambda^{TK} \quad (3)$$

P_{ij}^{TE} 表示板坯 i 到板坯 j 的出炉温度跳跃惩罚值, 惩罚系数为 λ^{TE} , 计算公式为

$$P_{ij}^{TE} = |TE_i - TE_j| \times \lambda^{TE} \quad (4)$$

P_{ij}^{TL} 表示板坯 i 到板坯 j 的标准在炉时间跳跃惩罚值, 惩罚系数为 λ^{TL} , 计算公式为

$$P_{ij}^{TL} = |TL_i - TL_j| \times \lambda^{TL} \quad (5)$$

P_{ij}^{tv} 表示板坯 i 到板坯 j 的轧机空转惩罚值, 惩罚系数为 λ^{tv} , 计算公式为

$$P_{ij}^{tv} = \max(0, t_j - t_i - p_i) \times \lambda^{tv} \quad (6)$$

综上, 两个相邻板坯间的惩罚值可以表示为

$$c_{ij} = P_{ij}^W + P_{ij}^{TK} + P_{ij}^{TL} + P_{ij}^{TE} + P_{ij}^{tv} \quad (7)$$

2.3 数学模型

考虑加热炉生产因素的热轧板坯轧制计划的整数规划模型如下:

$$\min Z = \sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ijk} + \sum_{i=1}^n P_i^t \quad (8)$$

s.t. 式 (1) - (7)

$$\sum_{k=1}^m y_{ik} = 1, i = 1, 2, \dots, n \quad (9)$$

$$\sum_{j=1}^n x_{ijk} = y_{ik}, i = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (10)$$

$$\sum_{j=1}^n x_{jik} = y_{ik}, i = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (11)$$

$$y_{0k} = 1, k = 1, 2, \dots, m \quad (12)$$

$$\sum_{i=1}^n x_{i0k} = 1, k = 1, 2, \dots, m \quad (13)$$

$$\sum_{i=1}^n x_{0ik} = 1, k = 1, 2, \dots, m \quad (14)$$

$$x_{0ik} + x_{i0k} \leq 1, i = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (15)$$

$$\sum_{i=1}^n y_{ik} L_i \leq Q_{\max}, k = 1, 2, \dots, m \quad (16)$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} y_{ik} y_{jk} = \sum_{i=1}^n y_{ik}, k = 1, 2, \dots, m \quad (17)$$

$$\omega_j + ML(1 - z_{ij} x_{ijk}) \geq \omega_i + L_j, \quad i = 1, 2, \dots, n, j = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (18)$$

$$L_i \leq \omega_i \leq R, i = 1, 2, \dots, n \quad (19)$$

$$|TL_i - TL_j| x_{ijk} \leq TL_{\max}, \quad i, j = 1, 2, \dots, n, i \neq j, k = 1, 2, \dots, m \quad (20)$$

$$|TE_i - TE_j| x_{ijk} \leq \delta_i + \delta_j, \quad i, j = 1, 2, \dots, n, i \neq j, k = 1, 2, \dots, m \quad (21)$$

$$|W_i - W_j| x_{ijk} \leq W_{\max}, \quad i, j = 1, 2, \dots, n, i \neq j, k = 1, 2, \dots, m \quad (22)$$

$$|TK_i - TK_j| x_{ijk} \leq TK_{\max}, \quad i, j = 1, 2, \dots, n, i \neq j, k = 1, 2, \dots, m \quad (23)$$

$$x_{ijk}(t_j - t_i - p_i) \geq 0, \quad i, j = 1, 2, \dots, n, i \neq j, k = 1, 2, \dots, m \quad (24)$$

$$t_j - TL_j \geq t_j^0, j = 1, 2, \dots, n \quad (25)$$

$$t_j^0 \geq 0, j = 1, 2, \dots, n \quad (26)$$

其中, 目标函数 (8) 表示最小化总惩罚值, 由相邻板坯间的惩罚值和单个板坯的提前/拖期惩罚所构成; 式 (1) - (7) 为惩罚值定义; 约束 (9) 表示所有板坯均被分配且仅能被分配到一个轧制单元内; 约束 (10) 和 (11) 分别表示所有板坯有且

仅有一个前序板坯和后继板坯; 约束 (12) 表示虚拟板坯0必须分配到每个轧制单元中; 约束 (13) (15) 表示保证构成回路; 约束 (16) 表示轧制单元最大长度限制; 约束 (17) 表示避免出现子回路; 约束 (18) 和 (19) 表示同宽连轧长度限制; 约束 (20) 表示相邻板坯的标准在炉时间跳跃最大值限制; 约束 (21) 表示相邻板坯的出炉温度区间需要有交集; 约束 (22) 表示相邻板坯宽度跳跃上限; 约束 (23) 表示相邻板坯厚度跳跃上限; 约束 (24) 表示板坯的轧制开始时间不早于其紧邻前序板坯的轧制完成时间; 约束 (25) 表示板坯在炉加热时长不能小于标准在炉时长; 约束 (26) 为板坯的入炉时间取值约束。

3 求解算法

3.1 算法思路

轧制计划作为钢铁生产管理的重要组成部分, 一直备受关注, 不少学者对带钢和板坯的轧制计划问题求解方法进行了研究, 提出了多种智能优化算法, 如基于规则的启发式算法^[15]、基于多目标优化策略改进的进化算法^[16]、混合果蝇优化策略的变邻域搜索算法^[17]等。上述求解算法基于经典的启发式算法框架进行设计, 通过引入多种求解策略或其他算法的寻优特点来提高主算法的性能。其中, 变邻域搜索算法具有结构简单、灵活性强、求解速度快的特点, 被作为基本框架广泛应用于算法设计中。本文设计自适应变邻域搜索算法 (Adaptive variable neighborhood search, AVNS) 求解模型, 求解框架如图1所示。算法首先采用基于值选择规则的约束满足算法获得初始解, 然后引入自适应搜索策略和精英反向学习策略优化变邻域搜索过程。

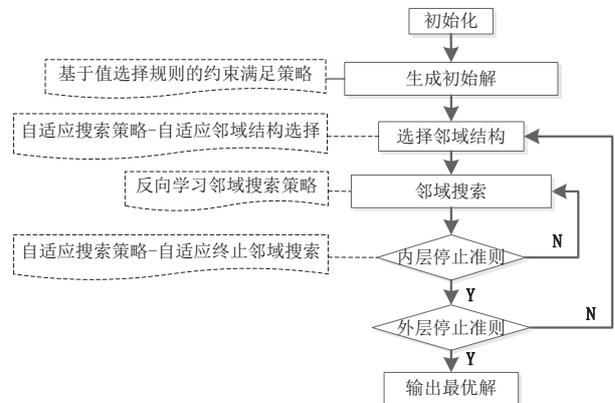


图1 自适应变邻域搜索算法AVNS求解框架

3.2 构造初始解

本文基于约束满足算法生成初始解。首先将模型映射为约束满足问题, 方法如下: 将模型中用于

确定板坯间轧制顺序的决策变量 x_{ijk} 进行转换, 定义变量 s_i 表示板坯 i 之后轧制的下一块板坯的序号, 变量取值范围映射为值域 $D(s_i)$; 将模型中确定轧制单元 k 是否包含板坯 i 的变量 y_{ik} 进行转换, 定义集合 V_k 表示轧制单元 k 包含的板坯集合; 将模型中确定相邻板坯是否同宽的变量 z_{ij} 进行转换, 定义 r_{ik} 表示第 k 个轧制单元内与板坯 i 连续同宽的板坯集合; 模型中的约束映射为约束集合 C , 轧制计划编制问题则可以被描述为 $CSP = (V, D, C)$ 。进而, 本文结合问题特点, 设计了两种值选择规则。

1) 最小惩罚值和自由度优先规则。基于贪婪策略优先选择惩罚值小的板坯, 容易导致后排的方案质量较差。为了弱化贪婪策略对后排方案的影响, 在选择板坯时同时考虑板坯的惩罚值和自由度:

①假设 i 为轧制单元 k 当前的末位板坯, j 为待选板坯, 板坯轧制提前/逾期惩罚值为 P_j^t , 按照式(1)计算, 若将 j 排到板坯 i 后面, 产生的惩罚值为 c_{ij} , 按照式(7)计算。另外, 为统一评价方式, 对同宽连轧长度超过上限设置惩罚值 c_j^r , 计算公式为:

$$c_j^r = \begin{cases} 0 & \text{if } (W_j \neq W_i) \\ & \text{or } (W_j = W_i \text{ and } L_j + \sum_{i' \in r_{ik}} L_{i'} \leq R) \\ (L_j + \sum_{i' \in r_{ik}} L_{i'} - R) \times \mu & \text{otherwise} \end{cases} \quad (27)$$

其中, i' 为与板坯 i 同宽连轧板坯集合中的任意板坯, μ 表示惩罚系数。综上, 选择板坯 j 产生的总惩罚值可以表示为:

$$C_j = c_{ij} + c_j^r + P_j^t \quad (28)$$

②自由度是指满足与板坯相邻约束的板坯数量, 用 $freedom$ 表示。假设 j 为待选板坯, $j \in N$, 则 j 的自由度 $freedom(j)$ 为: 在待排板坯集 N 中, 与 j 的宽度、厚度、标准在炉时间、出炉温度的差值不超过最大跳跃值的板坯数量。

在值选择过程中, 采用如下方法优化板坯选择: 为惩罚值和自由度分别设立波动区间, 波动区间内的所有板坯都可以参与下一个规则的筛选。惩罚值波动系数用 ε_c 表示, 自由度波动系数用 ε_f 表示, 对于任意待选板坯 j , 若用 $\min(c_j)$ 表示最小惩罚值, 用 $\min(freedom(j))$ 表示最小自由度, 则波动区间为 $[\min(c_j), \min(c_j) + \varepsilon_c]$ 和 $[\min(freedom(j)), \min(freedom(j)) + \varepsilon_f]$ 。

2) 最大标准在炉时间优先规则。根据问题特点可知, 若前序板坯标准在炉时间比后序板坯小很多, 当前序板坯轧制完成时, 后序板坯仍未达到标准在炉时间, 此时轧机需要空转等待, 这是导致轧机利

用率低的主要原因。如果不考虑其他约束, 将板坯按照标准在炉时间由大到小的顺序加工即可避免轧机空闲。因此, 在值选择过程中, 优先选择标准在炉时间最大的板坯, 一定程度上可以降低轧机空闲时间。

基于上述两个规则, 由约束满足算法生成初始解的过程描述如下:

step1: 初始化待排板坯集合 N , 轧制单元序号 $k = 0$, 轧制单元 k 包含的板坯集合 V_k 为空集, 初始化变量 s_i 的值域 $D(s_i) = \{0, 1, 2, \dots, n\}$ 。根据约束对每个 $s_i (i \in N)$ 的值域 $D(s_i)$ 做一致性处理;

step2: 若 $N = \phi$, 求解过程结束; 否则 $k = k + 1$, 令 $i = 0$, $V_k = V_k \cup 0$, $Q_k = 0$, 从 N 中选择宽度最大的板坯 i , 令 $s_0 = i$, $N = N \setminus i$, $V_k = V_k \cup i$, $Q_k = Q_k + L_i$;

step3: 从变量 s_i 值域 $D(s_i)$ 中搜索合适的板坯 j , 如果 $Q_{\max} - Q_k \leq L_j$, 转step2; 否则, 从符合轧制单元长度限制的板坯中选择惩罚值在 $[\min(c_j), \min(c_j) + \varepsilon_c]$ 区间内的板坯, 再从满足轧制惩罚区间的板坯中, 选择自由度在 $[\min(freedom(j)), \min(freedom(j)) + \varepsilon_f]$ 区间内的板坯, 最后从满足自由度区间的板坯中, 选择标准在炉时间最大的板坯 j , 令 $s_i = j$, $N = N \setminus j$, $V_k = V_k \cup j$, $Q_k = Q_k + L_j$;

step4: 实施赋值触发约束传播, 更新待排变量(板坯)的值域;

step5: 将轧制单元 k 中的末位板坯作为下一个待排变量 s_i , 若 $D(s_i) \neq \phi$, 转step3; 否则转step2。

3.3 构造邻域结构

邻域结构是变邻域搜索算法的核心内容, 其作用是拓展解的搜索空间, 降低算法陷入局部最优的风险, 提高算法全局寻优能力, 邻域结构产生解的方式, 决定了算法是否在有效时间内获得最优解。结合问题特点, 将编码方案和邻域结构设计如下。

(1) 编码方案设计

本文设计二维自然数编码表示解方案, 每个编码代表一个轧制计划, 它由多行元素构成, 每一行代表一个轧制单元, 每个元素为一个板坯。每行的元素数量(即每个轧制单元包含的板坯数)不一定相同, 在邻域搜索过程中不允许改变。图2(a)给出了一个编码的示例, 可以看出, 这个轧制计划包含了2个轧制单元, 每个轧制单元有5个板坯。其中, S1和S2分别表示第一个和第二个轧制单元, 板坯轧制顺序分别为: $\{5, 8, 7, 1, 3\}$, $\{6, 2, 10, 4, 9\}$ 。

(2) 邻域结构设计

在实际生产中，待排板坯集合中通常存在大量的同规格板坯，基于点操作的邻域结构搜索效率会偏低，因此本文引入基于块操作的邻域结构，将两类邻域结构结合使用，具体的邻域结构如下：

1) Point-Insert，对于轧制单元B中的板坯a，将其从原位置删除并随机插入到B中；

2) Point-Exchange，对于轧制单元B中的板坯a和b，交换a和b位置；

3) Block-Insert，对于轧制单元B中的连续的多个板坯C，将其从原位置删除并随机插入到B中；

4) Block-Exchange，对于轧制单元A和B，从A和B中分别随机选择由连续的多个板坯组成的板坯块C和D，交换C和D。

基于上述四种邻域结构产生的邻域解，难免会出现严重违反约束条件的情况，即产生无效解，为了保证解的有效性并维持解空间的多样性，本文在求解过程中将无效解的惩罚值调至为极大数，令其选择概率趋于零。

3.4 自适应搜索策略

为了提高算法效率，本文设计自适应搜索策略优化变邻域搜索过程，为每个邻域结构设置选择概率，并在邻域搜索中基于自适应终止条件结束搜索以避免无效迭代。自适应搜索策略主要包括自适应邻域结构选择策略和自适应邻域搜索终止准则，具体内容如下。

(1) 自适应邻域结构选择策略

自适应的邻域结构选择方式是指在迭代过程中记录每个邻域结构的使用次数和搜索成功次数，将成功搜索率作为评价邻域结构寻优效果的指标，并基于成功搜索率计算选择概率，利用轮盘赌策略选择邻域结构。其中搜索成功是指迭代搜索后当前最优解得到了改善。若 PS 表示选择概率， PN 表示邻域结构的成功搜索率，则 PS 的计算公式如下：

$$PS_u = \frac{PN_u}{\sum_{u \in U} PN_u} \quad (29)$$

其中 $PN_u = num_u / NUM_u + \rho$ ， NUM 和 num 分别表示邻域结构使用次数和搜索成功次数， u 表示邻域结构序号， U 为邻域结构集合， ρ 是为了避免初始选择概率为0而设置的极小数。

在算法初期令每个邻域结构的初始选择概率相同，以后每次迭代结束时更新选择概率，并基于轮盘赌策略选择下一个邻域结构。观察公式(29)可知，随着算法不断迭代，搜索效果越好的邻域结构，其

选择概率也会越大，有助于提高寻优速度。

(2) 自适应邻域搜索终止准则

若邻域结构迭代搜索的次数是固定值，那么当某一邻域结构在迭代搜索过程中已经无法再改进解的时候，算法依旧要完成规定的迭代次数。为了避免不必要的迭代，设计自适应终止邻域搜索机制如下：在迭代过程中，规定每个邻域结构迭代次数上下限，若当前累积迭代次数介于上下限之间时，判断每次迭代的改善比率，连续多次未能改进解则终止搜索，更换下一个领域结构。用 nm_u 表示邻域结构 u 当前已迭代的次数， nm_Max 、 nm_Min 为最大、最小迭代次数， θ 为终止阈值。邻域搜索自适应终止判断过程如下：

1) 当 $nm_u > nm_Max$ 时，邻域搜索终止；

2) 当 $nm_Min < nm_u < nm_Max$ 且邻域搜索连续 θ 代未能改善解时，邻域搜索终止。

在最小和最大迭代次数内，算法根据搜索效果自适应地终止迭代，既能充分利用邻域结构的搜索能力，又能适时更换邻域结构。

3.5 反向学习邻域搜索策略

反向学习技术是由Tizhoosh^[18]提出的一种新的智能优化技术，被广泛用于改进各种智能优化算法，主要是通过综合评价解及其反向解来提高解空间的广泛性，优化搜索过程。反向解定义如下^[18-19]：假设 x 为区间 $[a, b]$ 上的任意实数，则 x 的反向数可以定义为 $x^* = a + b - x$ ，那么对于可行解 $x = (x_1, x_2, \dots, x_j)$ ， x_j 为第 j 维度上的值， $x_j \in [a_j, b_j]$ ， x_j^* 是 x_j 的反向数，则定义 $x^* = (x_1^*, x_2^*, \dots, x_j^*)$ 为 x 的反向解， $x_j^* = \eta(a_j + b_j) - x_j$ ， η 为一般化系数，取值范围是0到1之间的随机数。

由于 η 为小数，而本文问题的解是二维整数编码，在根据反向解定义生成反向解的过程中会出现无效解，因此设计如下策略修复反向解：

将无效解所有维度取绝对值并向上取整，遍历反向解 x^* 的所有维度，记录 x_j^* 数值重复或超出值域的无效维度，将未出现在 x^* 中的 x_j ，即丢失的个体，进行随机排序并依次赋值给无效的维度。

Wang^[20]从概率理论的角度证明了，当前解有50%的概率比其反向解远离最优解。因此若在搜索过程中同时考虑当前解空间和其反向解空间，可提高寻得最优解的可能性。反向学习邻域搜索策略具体步骤如下：

step 1: 基于当前邻域结构和当前最优解生成 $candidateNum$ 个邻域解，邻域解集记为 RS ；

step 2: 基于反向解的定义计算 RS 中所有邻域解的反向解, 反向解集记为 RS^* ;

step 3: 利用反向解修复策略修复 RS^* 中的解;

step 4: 将 RS 和 RS^* 中所有解带入式(28)计算惩罚值, 选择惩罚值最小的解作为当前最优解。

3.6 算法步骤

AVNS算法的求解步骤如下, 其中 NM 表示算法外层迭代次数, 即选择邻域结构的次数, NM_Max 为 NM 上限:

step 1: 约束满足获取初始解(详见3.2节);

step 2: 初始化参数, 包括算法外层迭代次数 NM 、外层最大迭代次数 NM_Max 、内层迭代次数 nm 、内层最大最小迭代次数 nm_Max 和 nm_Min 、邻域搜索生成候选解个数 $candidateNum$, 当前算法最优 $result^*$, 令各种邻域结构的初始选择概率相等;

step 3: 判断外层迭代终止条件, 若 $NM > NM_Max$, 算法终止, 否则利用轮盘赌方法选择邻域结构;

step 4: 基于当前邻域结构, 利用反向学习邻域搜索策略进行邻域搜索(详见3.5节), 更新当前邻域最好解, $nm = nm + 1$, 更新邻域结构使用次数 NUM 以及累积搜索成功次数 num ;

step 5: 根据解的质量和邻域迭代次数 nm 判断是否满足邻域搜索自适应终止条件, 若不满足则返回step 4, 若满足则更新算法当前最优解 $result^*$;

step 6: 根据式(29)计算每个邻域结构的选择概率, $NM = NM + 1$, 转step 3。

4 数据实验

4.1 实验设计

为验证AVNS算法的求解质量, 采用某大型钢铁企业生产车间的实际生产数据对算法进行测试。选取板坯数量分别为50, 80, 120, 180, 240, 300, 350, 400的8种规模数据, 每组样本主要的属性信息包括: 板坯编号、宽度、厚度、长度、轧制时长、数量等, 为了检验数学模型的正确性, 本文利用CPLEX对模型进行编译, 并对坯料数量为10的小规模算例进行求解, 经验证模型是正确的。实验中所有算法均采用Microsoft Visual C#编程, 实验环境为Corei5/1.8GHz/8GB/Windows7, 采用相对偏差率(Percentage Relative Difference, PRD)作为衡量算法有效性的指标, 计算公式为

$$PRD = \frac{C(A) - C^{best}}{C^{best}} \times 100 \quad (30)$$

式中: $C(A)$ 表示算法A获取的目标值, C^{best} 为五种算法获得的最好解的目标值。

根据实际生产工艺将轧制约束相关参数设置为: $Q_{max} = 1200$ 、 $R = 600$ 、 $W_{max} = 500$ 、 $TK_{max} = 70$ 、 $TL_{max} = 26$ 。将板坯规格跳跃惩罚相关系数设置为: $\lambda^W = 0.8$ 、 $\lambda^{TK} = 0.8$ 、 $\lambda^{TL} = 0.9$ 、 $\lambda^{TE} = 0.9$ 、 $\lambda^{tv} = 1$ 、 $\mu = 2$ 。将提前和拖期惩罚系数设置为 $\beta_1 = 0.5$ 、 $\beta_2 = 0.8$ 。惩罚值波动系数 ε_c 和自由度波动系数 ε_f 为算法参数, 利用约束满足求解策略对算例进行预测试, 将根据测试效果将参数设置为 $\varepsilon_c = 500$ 、 $\varepsilon_f = 50$ 。

4.2 对比算法设置

轧制计划问题已有不少相关研究, 为了测试本文算法的性能, 实验中将本文算法与相关文献中的算法进行对比, 主要包括以下四种算法:

1) 单亲遗传算法 (Partheno-genetic algorithm, SGA) [13]。与文献中的单亲遗传算法对比, 以验证AVNS算法的有效性。

2) 约束满足算法 (Constraint Satisfaction Problem, CSP)。将AVNS算法中的约束满足策略与其他算法进行对比, 以验证本文基于值选择策略获得初始解的有效性。

3) 变邻域搜索算法 (Variable neighborhood search, VNS)。基于变邻域搜索算法的框架, 采用与AVNS算法相同初始解生成策略, 利用本文设计的邻域结构和自适应搜索策略进行搜索, 与CSP对比验证自适应搜索策略的有效性, 与AVNS对比验证反向学习邻域搜索策略的有效性。

4) 混合变邻域搜索算法 (Hybrid variable neighborhood search, HVNS) [17]。文献基于果蝇搜索策略设计了混合变邻域搜索算法, 并对邻域的切换方式做了改进。为了使AVNS算法与HVNS对比更加合理, HVNS采用与AVNS算法相同的初始解获取策略, 以验证AVNS算法的有效性。

在对比实验时, 除CSP其他四种算法的最大迭代次数均设置为 $NM_Max = 400$ 次。为了合理有效地设定算法参数, 将自适应搜索策略嵌入到经典的VNS搜索框架中, 进行多次测试实验, 最终将自适应搜索策略涉及的参数设置如下: 令自适应终止阈值 $\theta = 5$, 自适应选择概率中的 $\rho = 0.05$, 每种邻域结构对邻域搜索最小、最大搜索次数分别为 $nm_Min = 5$ 、 $nm_Max = 50$; 邻域解数量由数据规模而定, $n < 100$ 时 $candidateNum = 50$, $100 \leq n < 200$ 时 $candidateNum = 150$,

$200 \leq n < 300$ 时 $candidateNum = 250$, $300 \leq n$ 时 $candidateNum = 400$ 。VNS算法采用与AVNS相同的参数, SGA和HVNS均采用所引文献中的参数。

4.3 算法对比实验

针对8种规模的问题实例, 每种规模的样本随机生成20个测试算例, 用上述五种算法进行求解。计算每种算法求得惩罚值的PRD均值和标准差, 并进行方差分析(analysis of variance, ANOVA), 显著性水平 $\alpha = 0.05$ 。实验结果如表1所示。

表1 五种算法的PRD均值、标准差和方差分析统计结果(终止准则: 迭代次数)

n	AVNS		SGA		CSP		VNS		HVNS		ANOVA($\alpha = 0.05$)	
	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	F	P-value
50	0.24	0.32	8.74	0.54	8.58	0.96	5.95	0.65	0.87	0.76	363.26	1.28E-33
80	0.27	0.44	8.12	0.57	7.33	0.64	5.75	0.41	0.82	0.61	662.86	1.27E-36
120	0.48	0.62	11.71	0.56	11.15	1.71	6.64	0.73	1.06	0.92	538.49	5.97E-30
180	0.37	0.79	16.28	0.90	16.50	1.36	8.42	1.04	2.57	1.49	427.67	3.61E-35
240	0.59	1.02	19.55	1.76	22.86	2.12	13.51	1.34	2.97	2.19	319.48	2.09E-32
300	0.51	1.12	23.94	1.13	19.68	3.39	15.72	0.85	6.34	3.39	177.04	6.50E-27
350	0.57	1.46	23.37	2.3	23.51	4.21	10.96	2.39	5.49	8.76	282.93	2.89E-31
400	0.62	1.03	29.89	2.84	29.77	2.22	16.58	1.71	4.52	3.25	341.40	4.94E-33
50	0.24	0.32	8.74	0.54	8.58	0.96	5.95	0.65	0.87	0.76	363.26	1.28E-33

3) 基于本文算法初始解、邻域结构和自适应搜索策略设计的VNS算法, 其PRD均值比CSP平均降低37.55%, 比SGA平均降低39.45%, 这说明本文设计的自适应搜索策略是有效的, 能够较大幅度改进初始解。但是与AVNS相比, VNS的PRD均值是AVNS的平均23倍, 最高达到31倍, 这表明本文所提出的反向学习邻域搜索策略是有效的, 能够进一步扩大解空间, 搜索到更好的解。

4) 基于本文设计的反向学习邻域搜索策略改进的HVNS算法, 其PRD均值最高是AVNS的12倍, 最低约为2倍, 这表明本文基于自适应搜索策略和反向学习邻域搜索策略设计的AVNS是有效的, 相对于文献[17]中的算法, 具有更好的寻优能力。此外AVNS的PRD均值的标准差比HVNS平均降低了53.36%, 这说明AVNS算法求解性能更加稳定。

问题规模是影响算法求解效果的重要因素, 因此本文进一步丰富数据规模, 分别将五种算法对15种规模的算例进行测试(30、50、80、100、120、150、180、200、240、270、300、320、350、380、400), 结果如图2所示。由图2可知, 当数据规模增大时, 各种算法的PRD均值都会趋于增大, 但是AVNS与其他四种算法相比, 其变化曲线最平稳,

由表1的统计结果可以看出:

1) 所有规模测试中, ANOVA分析得到的P值均趋于0, 明显小于 α , 说明四种算法的求解效果存在明显差异。对于所有规模的问题, AVNS的PRD均值均为最小, 这说明AVNS求解效果更好。

2) 所有实验中, 基于邻域搜索的算法PRD均值都小于SGA, 这说明邻域搜索框架求解本文问题的效果更好, 优于SGA。此外, CSP的PRD均值平均比SGA降低了2%, 这说明本文设计的约束满足策略是有效的, 能够为邻域搜索提供较好的初始解。

这说明随着问题规模增大, 本文算法的求解优势也更加明显。

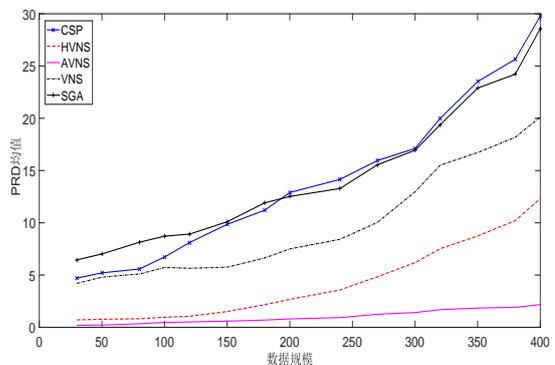


图2 PRD均值随数据规模变化曲线

为了分析算法的收敛性, 随机选择一个规模为240的算例, 利用五种算法进行求解, 图3为收敛曲线对比图。由图3可知, 本文设计的AVNS算法寻优速度最快, 在180代以后趋于收敛, 在搜索过程中获得解的质量明显高于其他四种算法, HVNS和VNS虽然与AVNS采用相同的初始解, 但是后序的寻优速度较慢, 且质量较差, 在350代左右才趋于收敛。SGA的求解质量远不如其他算法, 在150代左右才能达到与CSP相同的求解效果, 而且在450代左右才收敛。因此算法AVNS的收敛速度和求解质

量优于其他四种算法。

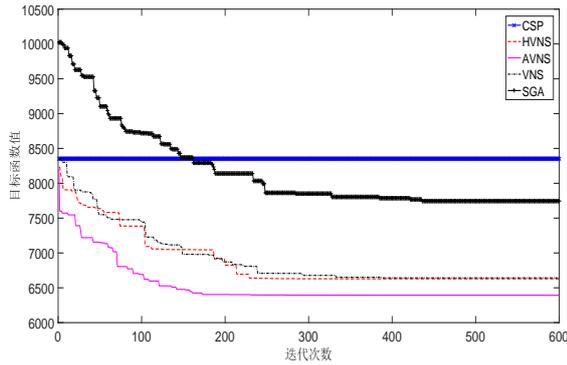


图3 收敛曲线对比图

在算法求解效率方面, 本文将除CSP之外的其他四种算法进行对比, 分别对80、240、400三种规模的算例进行测试, 表2为算例的平均CPU时间。

表2 算法的CPU时间 /s

n	SGA	VNS	HVNS	AVNS
80	26.20	15.07	14.32	13.14
240	225.29	134.25	114.35	128.79
400	538.74	492.05	487.67	452.61
均值	263.41	213.79	205.45	198.18

由表2可知: SGA的计算时间最长, 主要是因为遗传操作过程比变邻域搜索复杂; VNS计算时间比HVNS和AVNS长, 是因为采用了固定的邻域搜索次数, 而HVNS和AVNS都是通过搜索效果自适应确定邻域搜索次数。此外, AVNS计算时间略少于HVNS, 主要是因为AVNS基于反向学习的邻域搜索过程较为简便。综上所述, AVNS算法可以在有效时间内获得较高质量的解, 可满足应用需要。

4.4 加热炉因素的影响分析

为了进一步探究加热炉因素对轧制计划的影响, 分别对标准在炉时间和出炉温度进行参数分析。对于标准在炉时间的实验设计如下: 生成6个规模为80的算例, 算例的标准在炉时间方差存在明显差异且呈递增趋势, 保证其他参数的方差区别很小, 利用五种算法求解算例, 观察当标准在炉时间方差递增时目标函数值的变化情况。对于出炉温度的实验采用相同的方式。目标函数随方差增加的变化情况如图4、图5所示。由图4可以看出, 保证算例其他参数方差几乎不变的情况下, 五种算法求得的目标函数值都会随着标准在炉时间方差增大而增大, 这说明算例中板坯间的标准在炉时间差异过大, 会增加惩罚值, 降低轧制计划的质量。

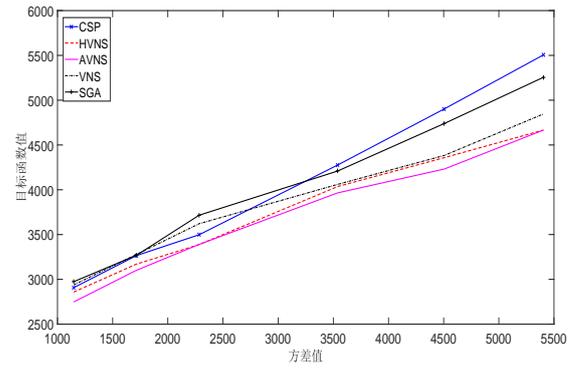


图4 目标函数随方差变化曲线(标准在炉时间)

观察图5, 同理可得, 算例中板坯间的出炉温度如果波动越大, 惩罚值越差, 轧制计划质量越差。

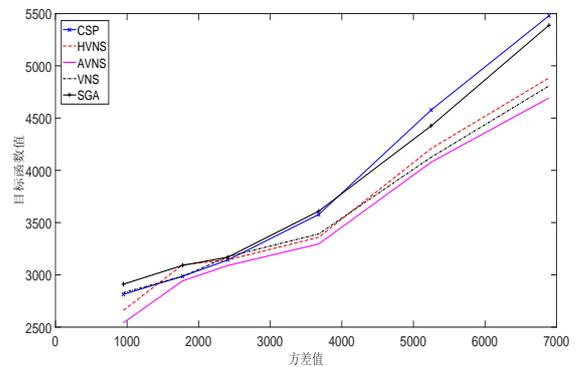


图5 目标函数随方差变化曲线(出炉温度)

5 结论

本文结合钢厂的实际情况, 研究了考虑加热炉生产因素的热轧板坯轧制计划编制问题。首先描述了热轧板坯生产线中加热炉和轧机的特点, 通过分析板坯在加热炉阶段的生产过程, 将标准在炉时间和出炉温度作为影响轧制计划的重要因素, 综合考虑宽度、厚度、同宽连轧等轧制工艺约束, 建立了以最小化规格跳跃惩罚、轧制提前/拖期惩罚、轧机空闲惩罚为目标的数学模型。结合模型特征提出了一种自适应变邻域搜索算法, 基于二维自然数编码提出了四种邻域结构, 并设计了约束满足策略、自适应搜索策略和反向学习邻域搜索策略来优化算法性能。在约束满足算法中提出了两种值选择规则, 用于生成高质量的初始解。自适应搜索策略优化了变邻域搜索算法的邻域选择和收敛速度, 反向学习邻域搜索策略则提高了解空间多样性和全局最优性。采用多种规模的生产数据测试算法, 实验结果表明本文提出的AVNS算法具有很好的求解性能。

参考文献(References)

- [1] Chen Q, Pan Q, Zhang B, et al. Effective Hot Rolling Batch Scheduling Algorithms in Compact Strip Production[J]. IEEE Transactions on Automation

- Science and Engineering, 2019:1-19.
- [2] Liu L, Liu L, Wan X, et al. Research on modelling and optimization of hot rolling scheduling[J]. Journal of Ambient Intelligence and Humanized Computing, 2019,10(3):1201-1216.
- [3] Jia S J, Yi J, Yang G K, et al. A multi-objective optimisation algorithm for the hot rolling batch scheduling problem[J]. International Journal of Production Research, 2013,51(3):667-681.
- [4] 张春生, 李铁克. 炼钢与热轧批量计划协调优化模型与算法[J]. 信息与控制, 2017,46(01):122-128.
(Zhang C S, Li T K. Optimization Model and Algorithm for Lot Planning Coordination between Steelmaking and Hot Rolling. Information and Control, 2017,46(01):122-128.)
- [5] 郑忠, 王永周, 卢义, 等. 考虑铸轧协调的炼钢-连铸-热轧一体化生产计划[J]. 钢铁, 2020,55(12):107-113.
(Zheng Z, Wang Y Z, Lu Y, et al. Integrated production planning of steelmaking-continuous casting-hot rolling considering coordination between casting and rolling[J]. Iron and Steel, 2020,55(12):107-113.)
- [6] 王宸, 杨洋, 周学良, 等. 基于HPCVRP的多目标热轧计划排程研究[J]. 现代制造工程, 2017(4):102-109.
(Wang C, Yang Y, Zhou X L, et al. Research on multi-objective hot milling planning based on HPCVRP[J]. Modern Manufacturing Engineering, 2017(4):102-109.)
- [7] Hu Z, He D, Song W, et al. Model and Algorithm for Planning Hot-Rolled Batch Processing under Time-of-Use Electricity Pricing[J]. Processes, 2020,8(1):42.
- [8] 董广静, 李铁克, 王柏琳, 等. 考虑轧辊成本的轧制计划编制与调整[J]. 控制与决策, 2017,32(1):63-70.
(Dong G J, Li T K, Wang B L, et al. Rolling planning and adjusting problem with roller cost[J]. Control and Decision, 2017,32(1):63-70.)
- [9] 汪洋, 吴子轩, 李铁克, 等. 考虑柔性机器检修的钢管热轧批量调度方法[J]. 控制理论与应用, 2019,36(9):1536-1544.
(Wang Y, Wu Z X, Li T K, et al. Methods of hot-rolled batch scheduling for seamless steel tube with flexible machine maintenance[J]. Control Theory & Applications, 2019,36(9):1536-1544.)
- [10] 杨业建, 姜泽毅, 张欣欣. 钢坯热轧加热炉区生产调度模型与算法[J]. 北京科技大学学报, 2012,34(07):841-846.
(Yang Y J, Jiang Z Y, Zhang X X. Model and algorithm of furnace area production scheduling in slab hot rolling[J]. Journal of University of Science and Technology Beijing, 2012,34(07):841-846.)
- [11] 谢金兰, 谭园园, 刘士新. 热轧生产过程加热炉优化调度模型及算法[J]. 辽宁科技大学学报, 2012,35(03):251-255.
(Xie J L, Tan Y Y, Liu S X. Model and algorithm for scheduling problem of reheating furnace[J]. Journal of University of Science and Technology Liaoning, 2012,35(03):251-255.)
- [12] 潘瑞林, 胡邦国, 曹建华, 等. 考虑加热温度曲线的热轧批次计划[J]. 计算机集成制造系统, 2014,20(11):2826-2835.
(Pan R L, Hu B G, Cao J H, et al. Hot-rolling batch plan considering heating temperature curve[J]. Computer Integrated Manufacturing Systems, 2014,20(11):2826-2835.)
- [13] 刘健, 王彬, 王柏琳, 等. 板坯热轧计划编制的优化模型与算法[J]. 计算机集成制造系统, 2015,21(12):3263-3270.
(Liu J, Wang B, Wang B L, et al. Optimization model and algorithm for slab hot rolling planning problem[J]. Computer Integrated Manufacturing Systems, 2015,21(12):3263-3270.)
- [14] Li F, Zhang Y, Wei H, et al. Integrated problem of soaking pit heating and hot rolling scheduling in steel plants[J]. Computers & operations research, 2019,108:238-246.
- [15] 吴子轩, 李铁克, 张文新, 等. 考虑机器检修的热轧钢管批量计划方法[J]. 控制理论与应用, 2017,34(09):1250-1259.
(Wu Z X, Li T K, Zhang W X, et al. Methods of hot-rolled batch planning for seamless steel tube with machine maintenance[J]. Control Theory & Applications, 2017, 34(09):1250-1259.)
- [16] Pan Q, Gao L, Wang L. A multi-objective hot-rolling scheduling problem in the compact strip production[J]. Applied Mathematical Modelling, 2019,73:327-348.
- [17] Zhang B, Pan Q, Gao L, et al. A hybrid variable neighborhood search algorithm for the hot rolling batch scheduling problem in compact strip production[J]. Computers & Industrial Engineering, 2018,116:22-36.
- [18] Tizhoosh H R. Opposition-Based Learning: A New Scheme for Machine Intelligence[C]. International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria:IEEE, 2005,695-701.
- [19] Rahnamayan S, Tizhoosh H R, Salama M M A. Opposition-Based differential evolution[J]. IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, 2009,2(13):398-417.
- [20] Wang H, Liu Y, Zeng S Y. Opposition-based particle swarm algorithm with cauchy mutation[C]. Proceedings of the IEEE Congress on Evolutionary Computation, Singapore: IEEE, 2007: 4750-4756.

作者简介

张卓伦(1996—), 男, 博士生, 从事生产计划与调度、智能算法的研究, E-mail: zhangzl0036@163.com;
 张文新(1966—), 男, 副教授, 从事管理信息系统、生产调度与计划等研究, E-mail: zhangwx@manage.ustb.edu.cn;
 李铁克(1958—), 男, 教授, 博士生导师, 从事生产调度与计划、先进制造管理等研究, E-mail: tieke@ustb.edu.cn;
 王柏琳(1983—), 女, 副教授, 从事智能优化算法等研究, E-mail:wangbl@ustb.edu.