

文章编号: 1001-0920(2009)11-1729-06

## 一体化中包计划模型与算法

董红宇<sup>a,b</sup>, 黄敏<sup>a,b</sup>, 王兴伟<sup>a</sup>, 郑秉霖<sup>b</sup>

(东北大学 a. 信息科学与工程学院, b. 流程工业综合自动化教育部重点实验室, 沈阳 110004)

**摘要:** 描述了一体化中包计划问题, 归纳了炼钢-连铸-热轧及下游工序的一体化工艺规程, 建立了以优化中包数、工艺附加成本和各流向产能平衡为目标的多目标优化模型. 基于策略和加权和方法处理多目标优化问题, 针对模型设计了基于 7 种邻域结构和局部迭代搜索方法的改进型变邻域深度搜索算法和改进型简化变邻域搜索算法. 通过实际数据仿真, 将两种算法与启发式算法进行对比, 同时对两种算法进行性能分析, 其结果验证了所提出模型和算法的有效性.

**关键词:** 一体化中包计划; 变邻域深度搜索算法; 多目标优化; 奖励机制

**中图分类号:** TP301

**文献标识码:** A

## Model and algorithm for integrated tundish planning

DONG Hong-yu<sup>a,b</sup>, HUANG Min<sup>a,b</sup>, WANG Xing-wei<sup>a</sup>, ZHENG Bing-lin<sup>b</sup>

(a. College of Information Science and Engineering, b. Key Laboratory of Integrated Automation of Process Industry of Ministry of Education, Northeastern University, Shenyang 110004, China. Correspondent: DONG Hong-yu, E-mail: hongyu.dong@gmail.com)

**Abstract:** The integrated tundish planning problem is described firstly, and the steelmaking-continuous casting-hot rolling and other downstream integrated technical constraints are summarized. Then, a multi-objective optimization model is proposed to optimize the number of tundish, the additional cost of technical operations and the throughput balance to each flow. The heuristic strategy and weighted sum method are introduced to deal with the multi-objective optimization model. The improved variable neighborhood descent (VND) search method and reduced variable neighborhood search (RVNS) method are proposed based on seven kinds of new neighborhoods and iterated local search (ILS). Finally, the two algorithms are compared with heuristic method based on numeral analysis to the actual data, and their performance are analyzed. The results show the effectiveness of the model and algorithm.

**Key words:** Integrated tundish planning; Variable neighborhood descent search method; Multi-objective optimization; Reward principle

### 1 引言

钢铁企业生产计划与调度问题一直受广大学者关注<sup>[1-6]</sup>. 由于钢铁生产的半连续、多阶段、多工序规程等问题, 大多学者都将钢铁企业的计划问题划分为多个阶段单独处理. 然而, 随着一体化生产工艺的出现, 迫切需要制定出与之相适应的一体化生产计划, Tang 等<sup>[2]</sup>提出了一体化生产计划研究的必要条件, 即每个独立的计划都要考虑各个工序阶段的工艺规程; Lee 等<sup>[1]</sup>较早提出应优化中间包寿命, 但缺乏对中间包优化的模型研究; 文献<sup>[3]</sup>考虑了中间

包寿命问题, 同时考虑了钢级、宽度和交货期等因素, 建立了单目标优化模型, 并采用启发式算法对 15 个炉次的实际生产数据进行了有效求解, 但没有以优化中间包为目的, 也没有考虑其他工艺阶段规程. 钢铁企业生产计划的研究大多集中在热轧计划<sup>[5,6]</sup>, 而以优化中间包设备为目标的研究文献则很少. 本文从一体化角度出发(考虑各个阶段工艺规程), 在保证产能平衡的前提下优化设备的使用效率、生产成本和潜在库存.

连铸机作业率是衡量连铸机生产水平的一个重

收稿日期: 2008-10-24; 修回日期: 2009-01-05.

基金项目: 国家自然科学基金项目(70671020, 70931001, 70721001, 60673159); 新世纪优秀人才支持计划项目(NCET-05-0295, NCET-05-0289); 高等学校博士学科点专项科研基金项目(20070145017, 20060145012); 国家高技术研究发展计划项目(2007AA041201).

作者简介: 董红宇(1982—), 男, 长春人, 博士生, 从事生产计划与调度、metaheuristic 算法等研究; 黄敏(1968—), 女, 福建长乐人, 教授, 博士生导师, 从事计算机集成制造系统、生产计划与调度等研究.

要指标. 所谓连铸机的作业率就是要求连铸机工作时间多而“睡觉”时间少,它是衡量连铸机生产水平的一个重要指标. 影响这个指标的因素很多,其中中间包设备的使用情况是直接影响该指标的重要因素之一.

中间包作为钢水承载设备,在多炉连浇更换钢包时不减拉速,并为多炉连浇创造条件. 因此,中间包起到连接炼钢与连铸工序的桥梁作用,有效地利用中间包生命周期可以实现多炉连浇炉数、缩短操作时间、提高产能.

一个中包可以定义为一个中间包允许连续浇注的一组炉次. 一体化中包计划(ITP)是在一体化条件下对炉次的组批过程,即在满足一体化工艺规程条件下,使得所组中包尽量少,各个流向上产能平衡且偏差小,炉次客户重量尽量充分利用,且组在同一炉次中的所有板坯的属性应尽量相同或相近.

本文从一体化角度出发,建立了多目标优化模型,并将其转化为单目标优化问题. 设计了基于局部迭代搜索算法(ILS),改进了变邻域深度搜索算法(VND)以及简化变邻域搜索算法(RVNS). 基于实际数据与启发式算法进行对比,对比结果表明,改进的VND和RVNS是高效的. 最后通过变多目标权重系数以及迭代局部搜索算法的局部迭代次数(NLI)对两种算法进行了性能分析.

## 2 炉次属性构成与一体化工艺规程

### 2.1 炉次属性构成

ITP以炉次为基本单位,炉次的属性如下:

$$s_i = [SG_i, C_i^{\text{code}}, W_i^{\min}, W_i^{\max}, WT_i, \text{PRI}_i, F_i, T_i^R, W_i^R, N_i^{\text{c-no}}].$$

其中: $s_i$ 表示炉次 $i$ 的多维属性, $i \in \{1, \dots, n\}$ ;  $SG_i$ 为炉次 $i$ 的钢级(化学成分);  $C_i^{\text{code}}$ 为炉次 $i$ 的连浇组代码(连铸标识);  $[W_i^{\min}, W_i^{\max}]$ 为炉次 $i$ 的宽度窗;  $WT_i$ 为炉次 $i$ 的重量;  $\text{PRI}_i$ 为炉次 $i$ 的优先级(由交货期、客户等级等决定);  $F_i$ 为炉次 $i$ 的流向代码集(炉次内板坯轧制后的去向标识);  $T_i$ 为炉次 $i$ 的厚度集;  $W_i^R$ 为炉次 $i$ 的轧制宽度(炉次 $i$ 中所有板坯中最小轧制宽度);  $N_i^{\text{c-no}}$ 为炉次的序号.

### 2.2 一体化工艺规程

为实现一体化中包计划,中包计划需满足一体化工艺规程,具体分为如下的硬规程和软规程.

硬规程如下:

**规程 1** 同一中包中炉次必须有同一连浇组代码;

**规程 2** 中包的生命周期为 6 个炉次,即最多可以连续浇注 6 炉,实际生产中至少 3 炉次;

**规程 3** 同一中包中炉次间组合在一起宽度跳

跃不可超过 100 mm,跳跃次数每 3 ~ 6 个炉次最多调宽 2 次;

**规程 4** 各个流向产能要平衡,即在允许的生产能力范围内生产.

软规程如下:

**规程 5** 同一中包中炉次的钢级尽量相同;

**规程 6** 同一中包中炉次宽度窗尽量有交集,且越大越好;

**规程 7** 同一中包中炉次的轧制厚度尽量接近;

**规程 8** 同一中包中炉次的轧制宽度尽量接近;

**规程 9** 同一中包中炉次优先级(交货期、客户等级)接近;

**规程 10** 客户有效重量需要充分利用;

**规程 11** 同一流向的炉次尽量组在同一中包中;

**规程 12** 流向上产能与该流向目标产能尽量接近.

规程 1 ~ 规程 3,规程 5 和规程 6 为连铸阶段规程; 规程 7 和规程 8 为热轧阶段规程; 规程 9) 和规程 10 为板坯库规程; 规程 4,规程 11 和规程 12 为其他工序规程.

## 3 多目标一体化中包计划优化模型

### 3.1 符号说明

用于优化模型中的参数和变量描述如下:

参数: $p$ 为中包总数,中包集 $\Omega_p = \{1, 2, \dots, p\}$ ;  $n$ 为炉次总数,炉次集 $\Omega = \{1, 2, \dots, n\}$ ;  $m$ 为流向数,流向 $m$ 的炉次集为 $\Omega_m$ ;  $|M|$ 为流向总数, $M$ 为流向集.

决策变量

$$x_{ik} = \begin{cases} 1, & \text{中包 } k \text{ 包含炉次 } i; \\ 0, & \text{中包 } k \text{ 不包含炉次 } i. \end{cases}$$

辅助变量: $I_k$ 为第 $k$ 个中包所包含炉次的集合, $I_k = \{i'_k \mid x_{i'_k k} = 1, i'_k \in \Omega, k \in \Omega_p\}$ ;  $y_k = \{x_{i'_k k} \mid x_{i'_k k} = 1, i'_k \in I_k, k \in \Omega_p\}$ .

为了有效地处理软规程,这里采用一种奖励机制来量化炉次间属性差异. 采用“高分优选”策略,即对于较优的方案(指派炉次到中包)奖励较高的分数,从而最优的计划等价于分数最高的方案组合.

$S_{ik}(y_k)$ 为第 $i$ 个炉次组到第 $k$ 个中包中的奖励分数(函数)之和,由钢级、宽度、有效炉次重量(不包含无委材的炉次重量)、优先级、轧制宽度、轧制厚度和流向 7 部分构成,定义如下:

$$S_{ik}(y_k) = S_{ik}^{\text{SG}}(y_k) + S_{ik}^{\text{W}}(y_k) + s_{ik}^{\text{WT}} + S_{ik}^{\text{PRI}}(y_k) +$$

$$S_{ik}^{WR}(y_k) + S_{ik}^{TR}(y_k) + S_{ik}^F(y_k).$$

其中： $S_{ik}^{SG}(y_k)$  处理一体化工艺流程 5； $S_{ik}^W(y_k)$  处理一体化工艺流程 6； $S_{ij}^T(y_j)$  处理一体化工艺流程 7； $S_{ik}^{WR}(y_k)$  处理一体化工艺流程 8； $S_{ik}^{PRI}(y_k)$  处理一体化工艺流程 9； $s_{ik}^{WT}$  处理一体化工艺流程 10； $S_{ik}^{Fm}(y_k)$  处理一体化工艺流程 11。

上述奖励函数为复杂的线性和非线性函数。

### 3.2 多目标优化模型

基于上述参数和变量,优化模型建立如下:

$$\min p, \tag{1}$$

$$\max \sum_{i \in \Omega} \sum_{k \in \Omega_p} S_{ik}(y_k) x_{ik}, \tag{2}$$

$$\min \sum_{m=1}^{|M|} \left| \sum_{i \in \Omega_m} \sum_{k \in \Omega_p} \omega t_{ik} x_{ik} - WT_{obj}^m \right|. \tag{3}$$

s. t.

$$\omega t_{\min}^m \leq \sum_{i \in \Omega_m} \sum_{k \in \Omega_p} \omega t_{ik} x_{ik} \leq \omega t_{\max}^m, \forall m \in M; \tag{4}$$

$$|C_i^{\text{code}} - C_{\min(y_k)}^{\text{code}}| x_{ik} = 0, \forall i \in \Omega, k \in \Omega_p; \tag{5}$$

$$\sum_{k \in \Omega_p} x_{ik} \leq 1, \forall i \in \Omega; \tag{6}$$

$$n_k^{\min} \leq \sum_{i \in \Omega} x_{ik} \leq n_k^{\max}, \forall k \in \Omega_p; \tag{7}$$

$$0 \leq A_{ik}(y_k) x_{ik} \leq T_k^A, \forall i \in \Omega, k \in \Omega_p; \tag{8}$$

$$x_{ik} \in \{0, 1\}, \forall i \in \Omega, k \in \Omega_p. \tag{9}$$

目标函数(1)表示最小化中包数;式(2)表示最大化所有中包中炉次奖励分数(即最小化附加成本);式(3)为最小化各流向上目标产能偏差之和。

约束条件(4)表示流向产能限制;式(5)表示同一中包中炉次要具有相同连浇组代码(即一同连浇);式(6)表示一个炉次至多属于一个中包;式(7)表示中包的背包约束,即容量约束;式(8)表示同一中包内炉次之间调宽次数限制。这里中包调宽次数

$$A_{ik}(y_k) = \begin{cases} \text{ceil}(-\Delta/\delta^W), & \Delta < 0; \\ 0, & \Delta \geq 0. \end{cases}$$

$$\Delta = \min\{W_{i_k}^{\max}\}, W_i^{\max}\} - \max\{W_{i_k}^{\min}\}, W_i^{\min}\}.$$

## 4 模型求解策略设计

解决多目标优化问题较传统的方法有加权和方法、加权指数和、目标规划法和目标函数定界法等,最常用的方法为加权和方法<sup>[7]</sup>。多目标问题的处理方法比较依赖于原问题,对于实际问题则需决策者根据经验对各个目标作出偏好选择或处理。

### 4.1 多目标优化问题转化为单目标优化问题

从实际角度出发,本文采用如下策略将多目标优化问题转化为单目标优化问题:

目标函数(1)与约束条件(7)综合考虑,基于策

略方法对问题进行初始化,其依据是在保证约束(7)的条件下,考虑目标函数(1)而得到一组初始解。目标函数(2)和目标函数(3)采用加权和方法转化为单目标函数,目标函数(3)采用如下奖励策略处理:

$$f_3(x_{ik}) = \lambda_{obj}^{F_{obj}^m} / \max\left\{1, \sum_{m=1}^{|M|} \left| \sum_{i \in \Omega_m} \sum_{k \in \Omega_p} \omega t_{ik} x_{ik} - WT_{obj}^m \right| \right\}.$$

将转化后的单目标函数作为问题的适应值,即

$$Z = \omega_1 \left( \sum_{i \in \Omega} \sum_{k \in \Omega_p} S_{ik}(y_k) x_{ik} \right) + \omega_2 f_3(x_{ik}). \tag{10}$$

其中： $\omega_1 + \omega_2 = 1; \omega_1, \omega_2 \geq 0$ 。

因为采用了目标函数奖励机制,从而会将尽量多的炉次组到中包中,间接地保证了目标函数(1)。其他约束条件将在算法实现上予以考虑。

原问题转化为单目标优化模型如下:

$$\begin{aligned} \max Z = & \\ & \omega_1 \left( \sum_{i \in \Omega} \sum_{k \in \Omega_p} S_{ik}(y_k) x_{ik} \right) + \omega_2 f_3(x_{ik}); \end{aligned} \tag{11}$$

s. t. 式(4) ~ (9)。

### 4.2 改进的 VND 和 RVNS

ITP 是以批的模式将相同或相似的炉次尽量组合的组合优化问题,是 NP-hard 问题<sup>[4,8]</sup>。为实现大规模问题的有效求解,本文采用改进的 VND 和 RVNS。1997 年, Mladenović 和 Hansen 提出一种新颖的元启发式算法,即 VNS<sup>[9]</sup>,并于 2001 年欧洲运筹学期刊上发表特邀综述<sup>[10]</sup>,给出了改进的 VNS 版本,并在实际问题中与其他算法作了对比分析。由于 VNS 概念简单、操作容易,在短短几年时间便得到迅速发展,若干改进版已成功地应用于典型的 NP-hard 问题<sup>[10,11]</sup>。VNS 的改进版有 VND, RVNS 和偏态变邻域搜索方法(SVNS)等。

本文将 ILS 引入标准版的 VND 和 RVNS 中。ILS 是一种简单有效的随机局部搜索算法,已被证明是求解 TSP 问题的有效算法。其基本思想是:先局部应用随机的办法进行寻优,寻优过程中通过不断扰动,直到满足局部接受准则;基于这个局部最优进一步扰动寻优;反复上述过程<sup>[12]</sup>。ILS 通常将算法迭代数或运行时间作为停止准则。该方法能够突破局部最优限制,使邻域朝着全局最优方向搜索。

目前 VNS 和 ILS 大多是针对 TSP 等问题的应用,尚未应用于钢铁生产计划领域。本文构造了适合 ITP 的改进算法,即将 ILS 与 VND 和 RVNS 相结合。

改进的 VND(IVND) 步骤如下:

Step1: 初始化。选择搜索过程中邻域结构集

$N_k, k = 1, \dots, k_{\max}$ , 给出初始解  $x$ , 选择停止准则.

Step2: 重复如下步骤, 直到满足停止准则:

Step2.1: 设置  $k = 1$ .

Step2.2: 直到  $k = k_{\max}$ , 重复如下步骤:

Step2.2.1: 探测邻域. 寻找  $x$  的最好的邻居, 从中随机产生  $x', x' \in N'_k(x)$ .

Step2.2.2: 更新. 如果  $x'$  对应的适应值优于当前  $x$  对应的适应值, 则设置  $x = x'$ , 局部迭代次数计数置 1, 并继续在邻域  $N_1$  内搜索; 否则, 更新局部迭代次数计数, 重复 Step2.2.1 和 Step2.2.2, 直到达到 NLI, 设置  $k = k + 1$ .

标准版的 VND 可以在较短时间内获得问题的理想解, 但其“爬山”能力较差. 因此, 本文设计了 IVND, 通过引入迭代局部搜索方法, 即对每个邻域的多次重复迭代搜索, 从理论上保证了最好邻居的选择, 提高了“爬山”能力.

改进的 RVNS(IRVNS) 步骤如下:

Step1: 初始化. 选择搜索过程中邻域结构集  $N_k, k = 1, \dots, k_{\max}$ , 给出初始解  $x$ , 选择停止准则.

Step2: 重复如下步骤, 直到满足停止准则:

Step2.1: 设置  $k = 1$ .

Step2.2: 直到  $k = k_{\max}$ , 重复如下步骤:

Step2.2.1: 随机搜索. 随机选择  $x$  的一个中包, 在其第  $k$  个邻域中产生  $x', x' \in N_k(x)$ .

Step2.2.2: 更新. 如果局部最优值优于当前最优值, 则设置  $x = x'$ , 设置局部迭代次数计数为 1, 并继续在邻域  $N_1$  内搜索; 否则, 更新局部迭代次数计数, 重复 Step2.2.1 和 Step2.2.2, 直到达到 NLI, 设置  $k = k + 1$ .

标准版的 RVNS 是对标准版 VNS 的简化, 目的是节省邻域搜索时间, 即以牺牲解的质量来换取时间. RVNS 省略了 VNS 中局部搜索, 适用于局域搜索耗时的大规模计算问题. 而本文设计的 IRVNS, 引入了迭代局部搜索方法, 增强了原有 RVNS 的广域搜索能力和鲁棒性.

#### 4.2.1 初始解构造策略

模型中所考虑的约束条件较多, 故存在初始可行解构造困难的问题. 这里从原问题角度出发, 首先对数据进行预处理. 数据预处理规则如下:

1) 收池炉次信息, 按连浇组代码分组, 并使各流向上炉次重量满足生产能力的下限, 即满足约束条件(4)的下限和约束条件(5);

2) 根据流向代码, 按轧制宽度由大到小对订单手册中数据分组排序, 标记炉次号  $N_i^{c-no}$ .

以流向能力为主线构造初始解, 流程见图 1.

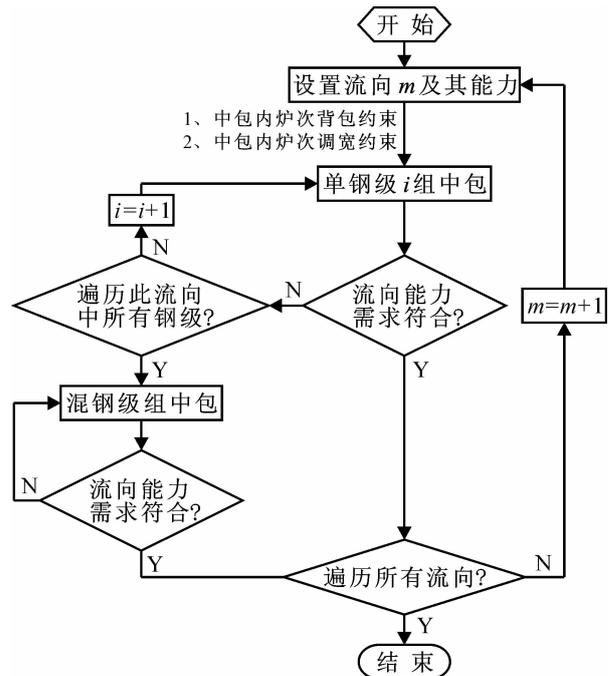


图 1 基于策略的解的初始化流程图

#### 4.2.2 邻域设计策略

对于两种算法而言, 邻域的设计将直接影响求解质量. 本文基于整数编码设计了 7 个邻域, 即  $k_{\max} = 7$ , 依次为插入 (insert), 替换 (instead), 转移 (shift), 交换 (exchange), 添加 (add), 分割 (divide), 合并 (combine). 其中: 插入邻域是将随机选择剩余炉次插入到合适的中包中; 替换邻域是将原有中包中的不合适的炉次替换为剩余炉次; 转移邻域是将原有中包某个或某些炉次转移到另一中包中; 交换邻域是交换两个不同中包中的炉次; 添加邻域是将几个剩余炉次组到中包中; 分割邻域是将一个中包分割成两个中包; 合并邻域是将两个中包合并为一个中包.

#### 4.2.3 探测邻域及随机搜索设计

本文两种算法中的 Step2.2 引入了新的机制, 即探测和搜索过程, 设置局部迭代次数 (NLI), 其值大小对于求解质量有较大影响.

#### 4.2.4 停止准则

IVND 与 IRVNS 都由两层循环组成, 外部循环停止准则为到达  $k_{\max}$  算法停止, 内部循环停止准则为达到 NLI 停止.

### 5 实例分析

#### 5.1 参数设计

随机选择某钢厂日生产数据, 由 949 块板坯作为原始数据, 组成 68 个炉次, 此为一体化中包计划的实例数据. 算法代码编写采用 Matlab 语言, 运行环境为 Pentium IV 3.0GHz/1G 台式机. 本文奖励

机制采用百分制,参数设计如表 1 所示.

表 1 基于奖励机制的参数设计统计表

参数	设定值	说 明
$\lambda_{obj}^{set}$	100	目标流向重量分数
$n_k^{min}$	3	中包容量下界
$n_k^{max}$	6	中包容量上界
$\delta^W$	100	宽度调节阈值
$T_k^A$	2	中包内最大调宽次数

5.2 评价指标定义

1) 中包平均利用率

$$\bar{R}^T = \sum_{k=1}^p \left( \left( \sum_{i=1}^n x_{ik} \right) / n_k^{max} \right) / p.$$

用于评价中间包的平均利用情况,其值越接近 1,中间包资源利用程度越高,从而可以间接节省生产成本.

2) 潜在库存水平

$$\bar{R}^{SL} = \sum_{k=1}^p \left( \sum_{i=1}^n WT x_{ik} - \sum_{i=1}^n WT_i / x_{ik} \right) / \left( \sum_{k=1}^p \sum_{i=1}^n WT x_{ik} \right).$$

用于评价未满足炉次添加无委材量满足情况,其值越接近 0,潜在库存越小,从而可以间接增加产量,节省资源.

3) 流向能力平均偏差率

$$\bar{R}^{FWT} = \sum_{m=1}^{|M|} \left| \sum_{i \in \Omega_m} \sum_{k \in \Omega_m} \tau wt_i x_{ik} - WT_{obj}^m \right| / \sum_{m=1}^{|M|} WT_{obj}^m.$$

用于评价各个流向上实际产能与目标产能之间的平均偏差,达到物流平衡.

4) 中包平均分

$$R^{TS} = \sum_{i \in \Omega} \sum_{k \in \Omega} S_{ik} (y_k) x_{ik} / p.$$

用于评价中包中的炉次分配成一组工艺附加操作成本.

5.3 算法对比与性能分析

表 2 为 3 种算法的对比分析结果,其中  $\omega_1 = \omega_2 = 0.5, NLI = 4$ . 启发式算法是基于图 1 中初始解构造策略添加修复可行解步骤的算法. 表 2 中第 4,5,6 列分别对应原问题 3 个目标函数,第 11 列对应单目标函数的适应值. 从中间包利用率及平均分角度,IVND和IRVNS都优于启发式算法,即充分地

利用了中包的生命周期. 从流向产能平衡角度看,启发式算法的结果较接近目标流向产能,但其产量比 IVND 少了 4 个炉次,即 1200t 钢(日钢产量的 10%). 从潜在库存水平角度看,IRVNS 优于其余两种算法. 从整体求解质量看(适应值),IVND 优于另两种算法.

通过实例对 IVND 和 IRVNS 的性能进行分析. 影响算法性能的主要参数包括局部迭代次数(NLI),邻域结构及个数. 分析性能的指标为:算法运行 100 次(即全局迭代次数 NGI 为 100)的最佳值(Best);算法运行 100 次的最差值(Bad);达优率(Best rate)定义为 100 次运行达到最佳值的比率. 其中 CPU 时间是算法运行 1 次所用的时间.

IVND 和 IRVNS 随 NLI 变化的性能分析曲线分别如图 2 和图 3 所示. 对比分析结论如下:一方面,IVND 随着 NLI 的增加,达优率越来越高,NLI 从 5 开始便实现达优率 100%,且增长速度快. 因此,NLI 参数设置取 5 时,既能保证得到稳定的解,同时又能大大节省 CPU 时间;另一方面,IRVNS 随着 NLI 的增加也表现出达优率增长趋势,但增长速度及幅度都很小,NLI 从 20 到 60 达优率增长 10%,仿真显示此时算法 CPU 时间数量级为小时,

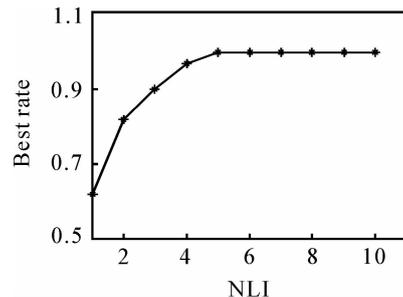


图 2 IVND 性能分析曲线

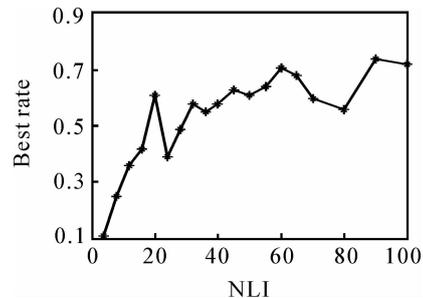


图 3 IRVNS 性能分析曲线

表 2 算法结果对比

算法	炉次数	组进炉次数	中包数	中包总分	流向能力偏差	中包平均利用率/%	流向能力偏差率/%	中包平均分	潜在库存水平/%	Z 值	CPU 时间/s
启发式	68	45	11	42371.24	1203.24	68.182	8.0332	3851.93	3.3559	21412.79	3.1090
IRVNS	68	46	11	42848.72	1494.68	69.697	9.8318	3895.34	3.3378	21626.70	7.1720
IVND	68	49	11	43170.88	1950.90	74.242	12.1457	3924.63	3.5051	21737.45	8.2350

表 3 加权因子变化条件下对比算法分析

算 法	权重系数		NGI	NLI	Best	Bad	Best rate/%	CPU 时间/s
	$\omega_1$	$\omega_2$						
IRVNS	0	1	100	4	454.35	454.35	100	6.6560
	0.1	0.9	100	4	4707.11	4646.04	26	6.9370
	0.3	0.7	100	4	13386.22	13029.42	8	7.2180
	0.5	0.5	100	4	22107.70	21412.79	7	6.8750
	0.7	0.3	100	4	30829.17	29796.17	10	7.6410
	0.9	0.1	100	4	39550.65	38179.55	9	6.9210
	1	0	100	4	43911.39	42371.24	7	7.1250
IVND	0	1	100	4	454.35	454.35	100	8.6720
	0.1	0.9	100	4	4646.04	4646.04	100	8.8600
	0.3	0.7	100	4	13164.07	13029.42	94	8.7970
	0.5	0.5	100	4	21737.45	21412.79	96	9.4540
	0.7	0.3	100	4	30310.82	29796.17	98	8.4690
	0.9	0.1	100	4	38884.20	38179.55	97	8.7190
	1	0	100	4	43170.88	42371.24	92	8.6090

因此 IRVNS 从达优率角度考虑性能劣于 IVND.

表 3 分析了多目标权重系数变化情况下两种算法的性能. IVND 的达优率都在 92% 以上, 而 IRVNS 的达优率比较低, 但在最佳解的质量上 IRVNS 优于 IVND. 可见, IVND 具有稳定的局部搜索能力, 短时间内可以找到高效稳定的最佳解, 而 IRVNS 则具有良好的爬山能力和高时效性, 但最佳解的个数少且波动性大. 因此, 对于其他组合优化问题, 需根据实际问题的特点, 选择适当的算法.

## 6 结 论

本文从一体化角度出发, 综合考虑了各阶段工艺规程, 以提高设备的利用率、减少工艺附加成本和优化各流向产能平衡为目标, 建立了多目标优化模型. 采用启发式策略将多目标优化问题转化为单目标优化问题, 针对该模型并结合 ILS 设计了 IVND 和 IRVNS. 通过处理实际生产数据, 将两种算法与启发式算法进行对比, 在产能和中间包利用率方面明显优于启发式算法, 验证了模型和算法的有效性. 同时, 对 IVND 和 IRVNS 进行了多目标权重变化的性能分析, 以及 NLI 变化的算法性能分析, 得到了两种算法 NLI 参数设置特点以及算法性能特点, 为两种算法的研究以及实际应用提供了科学依据.

## 参考文献 (References)

- [1] Lee H S, Murthy S S, Haider S W, et al. Primary production scheduling at steelmaking industries[J]. IBM J of Research and Development, 1996, 40(2): 231-252.  
 [2] Tang L X, Liu J Y, Rong A Y, et al. A review of

planning and scheduling systems and methods for integrated steel production [J]. European J of Operational Research, 2001, 133(1): 1-20.

- [3] 唐立新, 王梦光, 杨自厚. 炼钢-连铸对于浇次数未知的最优浇次计划模型与算法[J]. 钢铁, 1997, 32(7): 19-21.  
 (Tang L X, Wang M G, Yang Z H. Model and algorithm of cast plan with unknown number of cast for steelmaking-continuous casting scheduling[J]. Iron and Steel, 1997, 32(7): 19-21.)  
 [4] Chang S Y, Chang M R, Hong Y. A lot grouping algorithm for a continuous slab caster in an integrated steel mill[J]. Production Planning and Control, 2000, 11(4): 363-368.  
 [5] 刘士新, 宋健海, 周山长. 热轧带钢轧制批量计划优化模型及算法[J]. 控制理论与应用, 2007, 24(2): 243-248.  
 (Liu S X, Song J H, Zhou S C. Model and algorithm for solving hot strip rolling batch planning problems[J]. Control Theory and Applications, 2007, 24(2): 243-248.)  
 [6] 李耀华, 王伟, 徐乐江, 等. 热轧生产轧制计划模型与算法研究[J]. 控制与决策, 2005, 20(3): 275-279.  
 (Li Y H, Wang W, Xu L J, et al. Rolling plan model and algorithm in hot rolling plant [J]. Control and Decision, 2005, 20(3): 275-279.)  
 [7] Marler R T, Arora J S. Survey of multi-objective optimization methods for engineering[J]. Structural and Multidisciplinary Optimization, 2004, 26(6): 369-395.