

侯艳,任丙飞,滕少华,等.带罐约束的多目标短期炼油调度优化研究[J].江西师范大学学报(自然科学版),2023,47(3):307-316.

HOU Yan,REN Bingfei,TENG Shaohua,et al. The multi-objective short-term scheduling optimization with charging-tank-switch-overlap constraint in refinery [J]. Journal of jiangxi Normal university(Natural Science),2023,47(3):307-316.

文章编号:1000-5862(2023)03-0307-10

带罐约束的多目标短期炼油调度优化研究

侯 艳,任丙飞,滕少华,朱清华

(广东工业大学计算机学院,广东 广州 510006)

摘要:短期炼油调度问题不仅涉及多个优化目标,而且存在复杂的约束条件.由于带罐约束加剧了蒸馏塔对供油罐资源的争夺,所以这将导致调度解空间呈指数级膨胀,供油速度联调下协同优化效应急剧恶化.基于此,为保证任意供油速率下调度决策的可靠性和效率,该文提出了一种带罐约束的短期炼油多目标调度优化算法 NSGA-III-CPS.首先,任给一组炼油任务,在带罐约束下通过随机设置不同供油速度,使得算法能够获取在每种速度下可行调度方案、参数以及策略.其次,通过炼油成本计算与多目标评估,选择获取优化调度方案及策略.此外,该算法改进了交叉算子,通过优化交叉操作的父代选择策略,对父代种群进行快速非支配排序,支配等级高的父代赋予更高的被选中概率,使父代优良基因能更好地遗传到子代.因此,该算法综合考虑了供油罐使用成本与切换成本、原油在管道中的混合成本与在罐底的混合成本等多维炼油调度目标,在带罐约束下获取了可变供油速度下的炼油任务优化调度策略.最后,通过与4种经典多目标进化算法进行比较,对比结果表明:应用新算法以及策略在炼油成本上降低了10.0%~64.3%.

关键词:炼油;生产计划和调度;带罐约束;NSGA-III-CPS 算法

中图分类号:TP 301 **文献标志码:**A **DOI:**10.16357/j.cnki.issn1000-5862.2023.03.11

0 引言

随着国家对环保问题的重视,相关部门对全国原油一次加工能力提出了进一步的限制.由于炼油厂的原油一次加工能力与其产能挂钩,所以在加工能力受限后,节约生产成本成为生产利润最大化的主要途径.有研究表明:优化后的炼油调度计划可以大大节省原油一次加工过程中产生的额外成本^[1].但是原油一次加工过程是一个包含离散事件和连续过程的混合系统,不仅包含大量的操作约束、资源约束和时间约束,而且还具有装置多、生产规模庞大以及工艺复杂等特点,目前炼油短期调度问题已被证实为 NP-hard 问题^[2].如何优化炼油短

期调度计划成为近年来众多学者的研究热点之一.

大部分学者采用传统的数学规划方法对问题进行建模优化. L. D. de Assis 等^[3]基于离散时间变量,构建了混合整数线性规划模型(mixed integer linear programming, MILP),在该模型中每个时间槽内只能存在一个离散事件,用决策变量来决定该离散事件是否发生,但是随着问题复杂程度的提升,需要更细致地划分时间,使得时间槽缩短,从而导致决策变量随之急剧增加.为了解决上述问题, G. A. Schneider 等^[4]基于连续时间提出了一个 MILP 模型,利用该模型对炼油厂内的油罐调度问题进行了优化. F. Bayu 等^[5]基于连续时间和状态任务网络(state-task net, STN)对原油调度问题建模,在该模型中原油脱盐操作被视为一个单独的任务,原油

收稿日期:2022-10-19

基金项目:国家自然科学基金(61603100,61972102)和广东省重点领域研发计划课题(2020B010166006)资助项目.

作者简介:侯 艳(1977—),女,湖北公安人,副教授,博士,主要从事离散事件系统、生产计划与调度优化研究. E-mail:houyan@gdut.edu.cn

脱盐成本被视为其中一个优化目标. 虽然上述基于连续时间的模型减少了决策变量的数量, 但是随着调度周期的增大, 以及解决问题的复杂性提升, 引入连续时间变量将不可避免产生非线性约束, 从而使得该问题的数学模型变为混合整数非线性规划模型(mixed integer non-linear programming, MINLP)^[6-8].

由于炼油短期调度问题具有 NP-hard 属性, 所以很难在有限时间内求解一个关于此类问题的 MINLP 模型. 为了更高效地生成短期调度计划, 一些学者提出了一种基于仿真建模的方法解决该问题. Wu Naiqi 等^[9-10]将原油调度过程分为上下 2 层, 在上层中采用线性规划方法以最大化生产利润为目标制定出每个蒸馏塔的炼油计划, 在下层中则运用 Petri-Net 相关理论对问题进行仿真建模, 从而得出在不同情况下的可调度性条件, 并基于这些可调度性条件生成可行的炼油短期详细调度计划来满足上层的生产安排. 但是该方法只能保证获得调度计划的可行性, 并没有对调度计划进行优化. 目前已有大量研究表明进化算法在优化调度问题时有着优越的表现^[11-12]. 为了获得优化的炼油短期详细调度计划, 文献[13-14]将炼油短期调度问题转换为一种资源指派问题, 视供油罐为一种资源, 并将其指派给蒸馏塔, 从而使得该问题可以运用遗传算法进行优化求解. 侯艳等^[15-16]在假设上层炼油计划已知的情况下, 在下层中利用智能算法获得了优化的原油详细调度计划, 但是以上优化过程均没有考虑带罐约束. 文献[10]在考虑带罐约束的前提下, 分析了炼油短期调度问题的可调度性条件, 但是并没有进一步对该问题进行优化. 综上可知, 目前仍没有相关研究对包含带罐约束炼油短期调度问题进行优化, 而忽略带罐约束对问题进行优化将会导致获得的短期调度计划不能直接应用于实际生产. 因此, 本文首先基于文献[13]的指派思想, 分析带罐约束的特性, 从保证调度决策可行的角度出发, 提出了一种新的带罐约束处理策略, 使其转换为可用多目标进化算法优化的多目标调度问题. 为了提高进化算法在该问题上的优化性能, 对 NSGA-III (non-dominated sorting genetic algorithm-III) 算法^[17]进行改进, 提出了 NSGA-III-CPS 算法优化短期详细生产调度计划.

1 原油一次加工和短期调度问题

在介绍如何得到优化的短期详细生产调度计

划之前, 先简单介绍原油一次加工过程及其短期详细生产调度计划(见图 1).

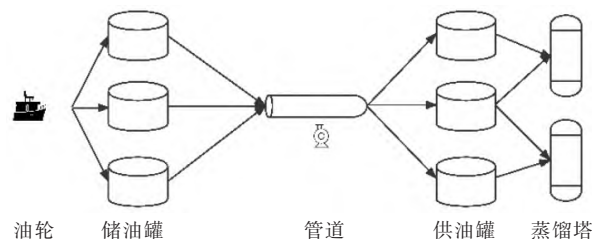


图1 原油一次加工过程

图1为炼油厂原油一次加工过程示意图, 该过程可以分为3个阶段: 1) 在装载原油的油轮到达港口后, 在油轮中的原油被卸载到港口的储油罐中; 2) 根据炼油厂内的炼油需求, 以管道运输的方式将原油从港口储油罐转运至厂区供油罐中; 3) 按照炼油计划, 在供油罐中的原油被源源不断地输送到蒸馏塔内进行炼制. 在以上过程中涉及以下优化目标: (i) 油轮卸载原油的成本; (ii) 油轮的等待成本; (iii) 储油罐的使用成本; (iv) 最大化生产率; (v) 由于在供油罐供油时不能完全压出罐内原油, 所以不同类型的原油会在供油罐的罐底混合, 从而产生罐底混合成本; (vi) 不同类型的原油在管道内的混合成本; (vii) 在蒸馏塔更换供油罐时出现的供油罐切换成本; (viii) 供油罐的使用成本. 其中基于控制理论目标(i)~(iv)已经在上层中用线性规划方法进行了优化, 本文将优化目标(v)~(viii).

1.1 短期炼油调度问题

如文献[10-11]所述, 生成详细的短期调度生产计划分2个阶段. 首先利用线性规划方法在上层中生成蒸馏塔炼制计划, 然后基于 Petri-Net 模型得到的可调度性条件, 在下层中生成详细的短期调度计划, 从而实现上层的蒸馏塔炼油计划.

本文工作将基于上层得到的蒸馏塔炼制计划, 在满足带罐约束的前提下, 在下层中得到优化的短期详细调度生产计划. 为了方便地描述短期详细调度生产计划, 首先给出原油调度决策的定义.

定义1 原油调度决策用6元组 $\{\ell, I, O, \xi, \alpha, \beta\}$ 表示, 其中 ℓ 表示调度决策需要操作的原油类型, I 和 O 分别表示在该调度决策中原油的来源与原油需要送往的目的地, ξ 表示在调度决策中需要处理的原油量, α 和 β 分别表示调度决策执行的起始时间和结束时间.

在原油转运过程中, 调度决策一共分为3种类型: 1) U 代表原油从油轮卸载到港口储油罐的卸油决策; 2) T 代表原油从港口储油罐通过管道运输至

炼油厂内供油罐的转运决策;3) F 代表原油从炼油厂内供油罐到蒸馏塔的供油决策. 详细调度计划就是要通过制定一系列的调度决策来实现给定的蒸馏塔炼油计划. 因此一个短期详细原油调度计划 P 可以表示为

$P = \{U_1, U_2, \dots, U_x, T_1, T_2, \dots, T_y, F_1, F_2, \dots, F_z\}$, 其中 x 、 y 和 z 分别表示制定一个短期详细调度计划所需要 U 、 T 和 F 的数量.

在实际生产中, 炼油厂为了降低炼油成本, 通常会在油价较低时大量购入足够用于生产的原油, 因此本文假设在储油罐中原油足够, 不考虑 U , 只对 T 和 F 做优化. 接下来给出该问题的优化目标以及在优化过程中需要满足的约束条件.

1.2 约束条件与优化目标

令 \mathcal{E}_g 和 \mathcal{E}_h 分别代表原油 g 和 h , 原油类型集合为 $\{1, 2, \dots, e\}$. M^p 为原油在管道内的混合成本系数矩阵, $M_{g,h}^p$ 代表 \mathcal{E}_g 和 \mathcal{E}_h 在管道内的混合成本系数, $\tau_{g,h}$ 代表在调度周期内 \mathcal{E}_g 和 \mathcal{E}_h 在管道内的混合次数; M^T 为原油在供油罐底部混合的成本系数矩阵, $M_{g,h}^T$ 代表 \mathcal{E}_g 和 \mathcal{E}_h 在供油罐底部的混合成本系数, $\varepsilon_{g,h}$ 代表在调度周期内 \mathcal{E}_g 和 \mathcal{E}_h 在供油罐底部的混合次数; Q 表示单个供油罐的使用成本系数, u 代表在调度期内使用的供油罐总个数; N 代表蒸馏塔单次切换供油罐的成本系数, s 代表在调度期内切换供油罐的次数. 优化目标函数如下所示:

$$\begin{cases} \min J_1 = \sum_{g=1}^e \sum_{h=1}^e \tau_{g,h} M_{g,h}^p, \\ \min J_2 = \sum_{g=1}^e \sum_{h=1}^e \varepsilon_{g,h} M_{g,h}^T, \\ \min J_3 = uQ, \\ \min J_4 = sN, \end{cases}$$

其中 J_1 表示原油在管道中的混合成本, J_2 表示原油在罐底的混合成本, J_3 表示供油罐使用成本, J_4 表示供油罐切换成本.

令 D 代表蒸馏塔集合, m 为集合 D 中的元素个数, d_l 代表 D 中的第 l 个蒸馏塔, $l \in \{1, 2, \dots, m\}$; 令 K 代表供油罐集合, n 代表集合 K 中的元素个数, k_h 代表 K 中的第 h 个供油罐, $h \in \{1, 2, \dots, n\}$; 调度周期为 Γ ; $I_p(y, t)$ 代表在时刻 t 时设备 y 作为输入接口连接其他设备的数量; $O_p(y, t)$ 代表在时刻 t 时设备 y 作为输出接口连接其他设备的数量; $T_N(y, t)$ 代表在时刻 t 时设备 y 存储原油类型的数量; $I_T(y)$ 代表设备 y 停止接收原油的时间; $O_T(y)$ 代表设备 y 开始输出原油的时间.

整个调度周期涉及的约束如下:

$$I_p(d_l, t) = 1, t \in \Gamma, l \in \{1, 2, \dots, m\}, \quad (1)$$

$$I_p(k_h, t) \leq 1, t \in \Gamma, h \in \{1, 2, \dots, n\}, \quad (2)$$

$$O_p(k_h, t) \leq 1, t \in \Gamma, h \in \{1, 2, \dots, n\}, \quad (3)$$

$$I_p(k_h, t) + O_p(k_h, t) \leq 1, t \in \Gamma, h \in \{1, 2, \dots, n\}, \quad (4)$$

$$T_N(k_h, t) \leq 1, t \in \Gamma, h \in \{1, 2, \dots, n\}, \quad (5)$$

$$O_T(k_h) - I_T(k_h) \geq \psi, h \in \{1, 2, \dots, n\}. \quad (6)$$

式(1)表示在任何时刻时蒸馏塔只能从一个供油罐接收原油, 且蒸馏塔都必须处于连续工作的状态; 式(2)表示在任何时刻时供油罐只能从一条管道接收原油; 式(3)表示在任何时刻时一个供油罐至多只能给一个蒸馏塔供油; 式(4)表示在任何时刻时任何一个供油罐都不可以同时被充油和供油; 式(5)表示在任何时刻时每个供油罐内最多只能存放一种原油; 式(6)表示在原油被转运至供油罐后, 罐内原油在输送至蒸馏塔之前, 必须在罐内驻留一段时间使得原油与盐水分离, 这段时间为驻留约束时间, 用 ψ 表示.

上述为在不带罐约束条件下的短期炼油调度优化模型, 本文将在上述约束条件的基础上, 考虑带罐约束, 同时优化上述多个目标. 接下来将详细介绍带罐约束条件, 并对带罐约束过程进行分析, 给出合理的带罐约束处理方式.

2 带罐约束原油优化调度

2.1 带罐约束条件

假设蒸馏塔 d_l 的供油决策集合为 $\{F_{l,1}, F_{l,2}, \dots, F_{l,Z_l}\}$, 其中 Z_l 代表在调度期内 d_l 的供油决策个数, $F_{l,i}$ 代表 d_l 的供油决策集合中的第 i 个供油决策, $i \in \{1, 2, \dots, Z_l\}$. 令 $\alpha_{l,i}^F$ 和 $\beta_{l,i}^F$ 分别代表 $F_{l,i}$ 的开始时间和结束时间, $\alpha_{l,i+1}^F$ 和 $\beta_{l,i+1}^F$ 分别代表 $F_{l,i+1}$ 的开始时间和结束时间, 带罐约束时长用 Φ 表示, 则带罐约束描述为

$$\begin{cases} IP(d_l, t) = 2, t \in [\alpha_{l,i+1}^F, \beta_{l,i}^F], \\ IP(d_l, t) = 1, \text{其他}, \end{cases} \quad (7)$$

$$\beta_{l,i}^F - \alpha_{l,i+1}^F = \Phi, \quad (8)$$

其中 $l \in \{1, 2, \dots, m\}$, $i \in \{1, 2, \dots, Z_l - 1\}$. 不难看出, 在带罐约束条件引入后, 约束(1)同时需要做出调整, 其中式(7)代表在带罐约束期间内需要 2 个供油罐同时给蒸馏塔供油, 也就是说当一个供油罐即将供油结束面临切换供油罐供油时, 需要另一个供油罐与之同时向该蒸馏塔供油, 这样做是为了更

加平稳地切换供油罐,避免因供油罐切换而使得蒸馏塔产生波动现象.带罐约束时长 Φ 一般设置为常量.

2.2 带罐约束分析

Hou Yan 等^[13]首先根据油罐初始存储状态对系统进行初始化得到初始状态 S_1 ,根据初始状态 S_1 生成空闲供油罐到蒸馏塔的指派,根据指派以及当前系统状态,获得决策的原油类型以及可以转运的原油量,从而得到对应的 T 和 F ,到达下一个新的状态 S_2 ,重复以上步骤,直至所有蒸馏塔炼制计划完成.本文基于以上指派思想对问题进行转换.

如图2所示,图2(a)和图2(b)分别为不考虑带罐约束和考虑带罐约束的情况.当不考虑带罐约束时,令 $t_{l,i}$ 表示 $F_{l,i}$ 的结束时间,同时也是 $F_{l,i+1}$ 的开始时间.在不考虑带罐约束的情况下执行决策 $F_{l,i}$ 时,该决策的供油速率始终等于蒸馏塔 d_l 的炼油速率,因此只要确定了 $F_{l,i}$ 的炼油量,时间点 $t_{l,i}$ 就随之确定.当引入带罐约束时,从图2(b)不难看出在时间区间 $[\alpha_{l,i+1}^F, \beta_{l,i}^F]$ 内有2个供油决策同时执行,此时 $F_{l,i}$ 的供油速率小于蒸馏塔 d_l 的炼油速率,且该供油速率从一个常量变为了在调度过程中的待优化变量,此时即使 $F_{l,i}$ 的炼油量确定了, $F_{l,i}$ 的开始时间和结束时间也难以确定.

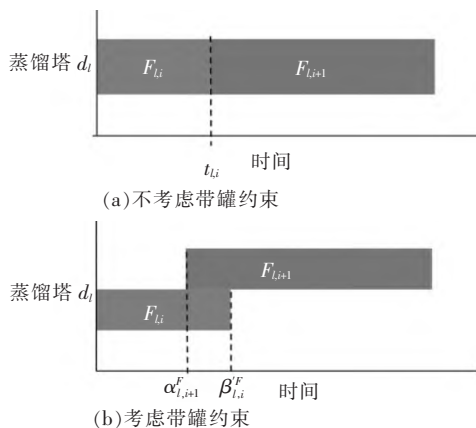


图2 不考虑带罐约束与考虑带罐约束的对比图

由带罐约束的描述可知:供油决策 $F_{l,i}$ 对应的供油罐必须在时间点 $\beta_{l,i}^F$ 之后才可以向其充油, $F_{l,i+1}$ 对应的供油罐内的原油必须在时间点 $\alpha_{l,i+1}^F$ 之前完成驻留.也就是说,当考虑带罐约束时,由于 $\alpha_{l,i+1}^F$ 和 $\beta_{l,i}^F$ 受供油速率影响,即使在时间区间 $[\alpha_{l,i+1}^F, \beta_{l,i}^F]$ 内 $F_{l,i}$ 和 $F_{l,i+1}$ 的供油速率满足了带罐约束,若供油速率选择不合适,则也会使得获得的调度违反其他约束条件.因此,在指派过程中优化带罐期间的供油速率,同时保证调度不违反其他约束条件为本研究的主要难点之一.

2.3 带罐约束处理

2.3.1 系统初始化阶段 在初始化过程以及后续的指派过程中供油决策 F 均会生成,首先介绍初始化过程.系统初始化过程也就是要将初始存储在供油罐中的原油分配给蒸馏塔的过程.由于蒸馏塔炼油计划以及在初始时供油罐存储原油信息均已知,所以根据供油罐内存储原油的型号可以找到需要炼制此类原油的蒸馏塔,然后将该罐内的原油分配给此蒸馏塔.令 ζ_i 代表在初始时供油罐 k_i 的空间使用率, V_i 代表在供油罐 k_i 内存储原油的油量, A_i 代表供油罐 k_i 的容量,则 $\zeta_i = V_i/A_i, i \in \{j | V_j \neq 0, j \in \{1, 2, \dots, n\}\}$.为了提升供油罐的空间利用率,优先选择 ζ_i 较小的供油罐对蒸馏塔供油,直到所有初始存储在供油罐内的原油均被分配给蒸馏塔,此时一系列 F 随之生成.

接下来介绍如何在生成供油决策的过程中对其进行处理,使之在满足带罐约束的同时保证不会违反其他约束条件.

由图2可知,3个时间点的关系为 $\alpha_{l,i+1}^F < t_{l,i} < \beta_{l,i}^F$.首先在不考虑带罐约束的情况下,根据调度完成情况可以获得每次供油决策分配给蒸馏塔 d_l 的原油量以及 d_l 的炼油速率,从而可以获得 $t_{l,i}$,其计算过程为

$$t_{l,i} = (\xi_{l,i,1}^F + \xi_{l,i,2}^F + \dots + \xi_{l,i,i}^F) / f_l, \quad (9)$$

其中 f_l 为 d_l 的炼油速率, $\xi_{l,i}^F$ 为供油决策 $F_{l,i}$ 分配给蒸馏塔 d_l 的原油量.

接着在考虑带罐约束的情况下,令 $r'_{l,i}$ 代表在 $[\alpha_{l,i+1}^F, \beta_{l,i}^F]$ 时间段内 $F_{l,i}$ 决策的供油速率,则 $f_l - r'_{l,i}$ 为在 $[\alpha_{l,i+1}^F, \beta_{l,i}^F]$ 时间段内 $F_{l,i+1}$ 决策的供油速率.在此需要强调, $r'_{l,i}$ 与 f_l 呈线性相关,即 $r'_{l,i} = cf_l$,其中比例系数 c 随机生成,且 $c \in (0, 1)$.带罐约束时间与上述时间点的关系为

$$(t_{l,i} - \alpha_{l,i+1}^F) f_l = \Phi r'_{l,i}. \quad (10)$$

首先根据式(9)计算获得 $t_{l,i}$,在 $r'_{l,i}$ 确定后,根据式(10)计算获得 $\alpha_{l,i+1}^F$,最后根据式(8)计算得到 $\beta_{l,i}^F$.在系统初始化过程中,按照上述方法对生成的供油决策的开始时间和结束时间进行处理,使其满足带罐约束.

2.3.2 指派阶段 在系统初始化完成后,进入指派阶段,即指派空罐给蒸馏塔从而生成转运决策以及供油决策,此时需要确定每次决策操作的原油量.在此需要强调的是:由于指派阶段的转运决策和供油决策为一一对应关系,转运决策转运的原油用于

供油决策向蒸馏塔供油,所以这2种决策操作的油量是相等的.从带罐约束分析可知,虽然在带罐约束期间内的供油速率满足了带罐约束条件,但是仍然会出现违反其他约束的情况,其原因是一开始计算获得的转运原油量可能不合适.因此本文首先随机设定每次指派对应的在带罐约束期间内的供油速率,然后根据该速率计算出满足约束条件的转运油量,进而在带罐约束期间内的供油速率便可以使用进化算法寻优得到.由此可知,接下来最重要的问题就是如何确定向供油罐转运的原油量.

为了使得转运决策尽可能安全地转运更多的原油,其转运原油量按照最大可转运油量计算.假设本次指派生成的供油决策为图2中的 $F_{l,i+1}$,对应的转运决策为 $T_{l,i+1}$,其中本次指派需要转运原油类型为 ℓ_g ,对应的供油罐为 k_h ,蒸馏塔为 d_l ,且 $F_{l,i}$ 决策已知,转运决策 $T_{l,i+1}$ 的开始时间已知,只需计算该指派生成的转运决策的原油转运量,其计算公式为

$$\xi = \min(\Lambda_h, \xi_{l,g}, (\beta_{l,i}^F - \alpha_{l,i+1}^T - \psi - \Phi)f_{pip}),$$

其中 ξ 代表本次指派可以转运 ℓ_g 的最大油量, Λ_h 为本次指派所选供油罐 k_h 的容量, $\xi_{l,g}$ 为所选蒸馏塔 d_l 完成炼油计划当前还需要原油 ℓ_g 的油量, $(\beta_{l,i}^F - \alpha_{l,i+1}^T - \psi - \Phi)f_{pip}$ 代表在不违背驻留约束以及带罐约束的前提下可以通过管道运输的最大油量, f_{pip} 为管道运输速率, $\alpha_{l,i+1}^T$ 为 $T_{l,i+1}$ 的开始时间.通过以上方式便能根据每次指派获得对应的可行详细调度决策,接下来给出问题的优化思路.

3 改进的NSGA-III算法及其优化过程

从上述的分析可以看出,比例系数 c 的设置会影响供油罐的带罐时间区间,由于供油罐与时间均为非常重要的资源,所以合理设置比例系数 c 能够提高供油罐的利用率,从而降低罐底混合成本、用罐成本以及原油在管道内的混合成本.炼油短期调度问题已被证实为NP-hard问题^[2],同时在上述带罐约束处理过程中,确定 c 的过程没有规律可循,因此本文首先随机生成 c ,然后使用多目标优化算法进行优化.在前期的研究中, c 设置为一个常数,而本研究中 c 的取值范围为连续区间,这意味着需要在更大的求解空间内寻优,使得算法寻优困难增大,因此本研究将对NSGA-III算法的交叉算子进行

改进,提升该算法在较大解空间内寻优的性能.

NSGA-III算法基于快速非支配排序遗传算法的框架,提出了一种新的选择算子,该算子是基于参考点选择新的个体组成下一代父代种群的.虽然该方法提高了种群的多样性,但在解决本文研究的问题时该算法的交叉算子仍有较大的改进空间.

交叉算子的作用是将较优个体的优良基因遗传给下一代个体.在优化过程中,父代个体的选择通常使用随机方式,因此较优个体被选择为交叉父代的概率等同于较优个体在种群中的占比.但是在本文研究的问题中,由于解空间变大,所以在较优个体数量增加的同时,被较优个体所支配的数量也会大大增加,且后者增加的比例更多,从而使得较优个体被选择为交叉父代的概率变小,优良基因遗传给子代个体的概率也随之变小.因此本文提出一种新的选择策略(CPS)来选择用于交叉操作的父代个体,以便提高较优个体被选中做父代的概率,并基于NSGA-III提出了一种新算法NSGA-III-CPS.

3.1 染色体结构

根据前面的分析,染色体结构定义如下: $\{a_1, a_2, \dots, a_\lambda, b_1, b_2, \dots, b_\lambda, c_1, c_2, \dots, c_\omega, c_{\omega+1}, \dots, c_{\omega+\lambda}\}$.首先初始化系统,生成 $F_1, F_2, \dots, F_\omega$. $F_1, F_2, \dots, F_\omega$ 由在带罐约束时间内的供油速率 $r'_1, r'_2, \dots, r'_\omega$ 通过解码 $c_1, c_2, \dots, c_\omega$ 获得,其中 ω 为当前系统非空供油罐的数量.在系统初始化完成后,得到初始状态 S_1 .令 K'_i 和 D'_i 分别代表在状态 S_i 下可用的供油罐集合和蒸馏塔集合,则 a_i 和 b_i 分别用于解码获得从 K'_i 和 D'_i 所选中的供油罐和蒸馏塔,在该状态下得到的供油决策 $F_{\omega+i}$ 由在带罐约束时间内的供油速率 $r'_{\omega+i}$ 通过解码 $c_{\omega+i}$ 获得.

由于蒸馏塔为不间断工作模式,在每次生成指派时,至少会有 m 个供油罐正在给蒸馏塔供油,所以可用的供油罐数量最多为 $n-m$ 个,在此 a_i 和 b_i 均用整数编码,且 $a_i \in [0, n-m]$, $b_i \in [1, m]$.另外 $r'_i = cf_i$, $i \in \{1, 2, \dots, \omega + \lambda\}$, c_i 采用实数编码,为了防止 r'_i 过大或过小, c_i 的取值区间设置为 $[0.1, 0.9]$,当 $a_i = 0$ 时,系统处于主动停运状态.

接下来确定染色体的长度,由于可能出现的停运状态,因此 λ 根据下式确定:

$$\lambda = \sum_{g=1}^e \lceil \xi_g / \Lambda_{\min} \rceil + \Pi, \quad (11)$$

其中 $\lceil * \rceil$ 为向上取整操作, ξ_g 代表为了完成所有蒸馏塔炼油计划所需转运原油 ℓ_g 的油量, Λ_{\min} 代表

在所有供油罐中容量最小的供油罐的容量, Π 为大于 0 的整数. 由于供油罐可能存在全都被占用的情况, 此时管道运输需要主动停运一段时间, 等待新的供油罐空出来, 所以考虑部分可行解会存在主动停运的情况, 式(11)需要引入正整数 Π , 以此来保证这一部分可行解不会丢失.

3.2 染色体解码

首先初始化系统, 根据基因 $c_1, c_2, \dots, c_\omega$, 按照本文介绍的初始化过程, 将原本存储在供油罐内的原油分配给蒸馏塔, 按照先后顺序生成 $F_1, F_2, \dots, F_\omega$. 在系统初始化完成后, 在状态 S_i 时, 需要解码 a_i 和 b_i 选择供油罐以及蒸馏塔, 根据供油罐到蒸馏塔的指派生成详细的调度决策 T_i 和 $F_{\omega+i}$ ($i \in \{1, 2, \dots, \lambda\}$). 首先, 根据在状态 S_i 时的蒸馏塔炼油计划完成情况, 可以获得此状态下可用供油罐的集合 K'_i 以及未完成炼油计划的蒸馏塔集合 D'_i . 若 $|K'_i| = 0$ 或者 $a_i = 0$, 则生成停运指派, 使系统处于停运状态, 直到有一个供油罐完成供油; 否则

$$j = \begin{cases} a_i \bmod |K'_i|, & a_i \bmod |K'_i| \neq 0, \\ |K'_i|, & \text{其他}, \end{cases}$$

$$l = \begin{cases} b_i \bmod |D'_i|, & b_i \bmod |D'_i| \neq 0, \\ |D'_i|, & \text{其他}, \end{cases}$$

其中 $i \in \{1, 2, \dots, \lambda\}$. 确定 K'_i 中的第 j 个供油罐被选中, D'_i 中的第 l 个蒸馏塔被选中. 在供油罐和蒸馏塔确定后, 根据 $c_{\omega+i}$ 以及前文所描述方法生成满足带罐约束的决策 T_i 和 $F_{\omega+i}$.

3.3 用于交叉的父代的选择策略(CPS)

在介绍 CPS 之前, 首先给出支配关系的相关定义.

对于一个 M 维的多目标最小化问题, 假设个体 X_1 对应解为 $\{f_1(X_1), f_2(X_1), \dots, f_M(X_1)\}$, 若任意 2 个解 X_1 与 X_2 满足条件

$$\begin{cases} f_i(X_1) \leq f_i(X_2), \forall i = 1, 2, \dots, M, \\ f_i(X_1) < f_i(X_2), \exists i = 1, 2, \dots, M, \end{cases}$$

则称个体 X_1 支配个体 X_2 , 用 $X_2 < X_1$ 表示; 对于任意 2 个集合 S_1 和 S_2 , 若其满足条件 $\forall j \in S_1, \exists i \in S_2$ 使得 $j < i$, 则称集合 S_2 支配集合 S_1 , 用 $S_2 >_d S_1$ 表示. CPS 如下所述:

1) 对父代种群进行快速非支配排序, 将父代种群划分为 R 个不同层次的非支配集, S_i 代表第 i 层非支配集, $i \in \{1, 2, \dots, R\}$;

2) 令 p_i 代表从 S_i 中选择个体作为交叉父代的

概率, 则 $p_i = (R + 1 - i) / (1 + 2 + \dots + R)$, 当 S_i 被选中时, 从该集合中随机选出一个个体作为交叉父代之一;

3) 重复步骤 2), 从父代种群中选出 2 个父代个体进行模拟二进制交叉操作, 得到 2 个子代个体.

3.4 NSGA-III-CPS 算法流程

NSGA-III-CPS 算法流程如下所示:

1) 种群初始化, 随机生成种群 P_0 , $|P_0| = E$, 设置 $G = 0$, E 为种群大小, G 为进化代数;

2) 交叉变异操作, 根据 CPS 策略从父代种群 P_i 中选择 $E/2$ 对父代个体交叉得到 E 个子代个体, 再对 E 个子代个体进行多项式变异操作, 得到子种群 Q_i ;

3) 选择操作, 令 I 代表归档集, 则 $I_i = P_i + Q_i$, 基于 NSGA-III 算法中的选择算子^[15], 从 I_i 中选出 E 个个体组成下一父代种群 P_{i+1} , $G = G + 1$;

4) 重复执行 2) 和 3), 直至 G 达到最大迭代次数.

4 算例分析

图 3 为中国某大型炼油企业的一个 10 d 炼油计划, 该炼油厂共有 3 个蒸馏塔 d_1, d_2, d_3 , 其炼油速率分别为 333.3、291.7、625.0 t/h, 管道转运速率为 1 250 t/h.

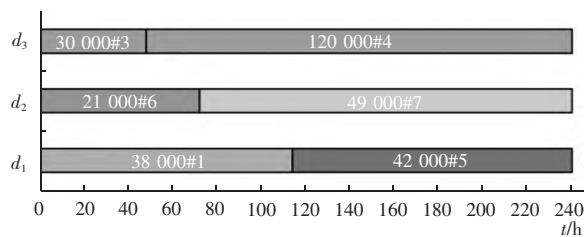


图 3 蒸馏塔炼油计划(10 d)

表 1 为在炼油厂内供油罐的初始存储信息, 由图 3 及表 1 可知, 在调度期内仍需通过管道转运 #4 原油 90 000 t、#5 原油 26 000 t 和 #7 原油 49 000 t. 其中供油罐最小容量为 16 000 t, 令 $\Pi = 3$, 由此可以得到 $\lambda = \left(\left\lceil 90\,000/16\,000 \right\rceil + \left\lceil 26\,000/16\,000 \right\rceil + \left\lceil 49\,000/16\,000 \right\rceil + 3\right) = 15$. 由表 1 可知, 共有 7 个供油罐的原油可直接分配给蒸馏塔, 令 $\omega = 7$. 原油在管道中的混合成本系数和原油在罐底的混合成本系数分别用矩阵 M^p 和 M^T 表示, 驻留时间为 6 h, 带罐时间为 4 h, 其中

$$M^p = \begin{pmatrix} & \#1 & \#3 & \#4 & \#5 & \#6 & \#7 \\ \#1 & 0 & 11 & 12 & 13 & 7 & 15 \\ \#3 & 10 & 0 & 9 & 12 & 13 & 7 \\ \#4 & 13 & 8 & 0 & 7 & 12 & 13 \\ \#5 & 13 & 12 & 7 & 0 & 11 & 12 \\ \#6 & 7 & 13 & 12 & 11 & 0 & 11 \\ \#7 & 15 & 7 & 13 & 12 & 11 & 0 \end{pmatrix},$$
$$M^r = \begin{pmatrix} & \#1 & \#3 & \#4 & \#5 & \#6 & \#7 \\ \#1 & 0 & 11 & 12 & 13 & 10 & 15 \\ \#3 & 11 & 0 & 11 & 12 & 13 & 10 \\ \#4 & 12 & 11 & 0 & 10 & 12 & 13 \\ \#5 & 13 & 12 & 10 & 0 & 11 & 12 \\ \#6 & 10 & 13 & 12 & 11 & 0 & 11 \\ \#7 & 15 & 10 & 13 & 12 & 11 & 0 \end{pmatrix}.$$

表1 供油罐初始存储状态

供油罐	油罐容量/t	原油型号	原油油量/t
k122	16 000	#1	8 000
k129	34 000	#1	30 000
k115	34 000	#3	30 000
k116	34 000		
k127	34 000	#4	30 000
k182	16 000	#5	16 000
k124	20 000	#6	16 000
k181	16 000	#6	5 000
k180	16 000		
k125	30 000		

本文分别使用 MOEA/D^[18]、NSGA-Ⅲ^[17]、AWGA^[19] 以及 RVEA^[20] 算法对该问题进行优化求解, 以上算法均采用模拟二进制交叉以及多项式变异的方式更新种群. 交叉概率取 0.7, 变异概率取 0.5, 交叉算子和变异算子分布指数均取 20. 本文将通过 HV 指标^[21] 和 C 指标^[22] 对比不同算法在解决该问题时的求解性能, 对于每种算法均取 9 组不同的种群大小与迭代次数组合, 对于每组实验分别独立运行 30 次, HV 指标以及 C 指标取平均值. 实验基于 Python 编程, 实验环境为 Intel(R) Core(TM) CPU 2.6 GHz 4G 内存, Windows 10.

HV 指标为解集中所有点与参考点在目标空间内所围成的超立方体体积, 其值越大代表该算法的多样性和收敛性越好. 本文在设置参考点之前, 首先从所有实验数据中找到每个优化目标的最大值与最小值, 根据最大值与最小值对所有得到的解集做最大最小值归一化处理. 在归一化完成后, 设置

参考点为(1,1,1,1), 计算 HV 指标. 如图 4 所示, 横坐标代表了不同种群大小以及进化代数的组合, 50_100 代表种群大小为 50 且进化代数为 100. 从图 4 不难看出 NSGA-Ⅲ-CPS 性能最优, 其次是 RVEA 和 NSGA-Ⅲ; 随着评价次数的增多, 由图 4 可以看出 NSGA-Ⅲ 比 RVEA 稳定性更好.

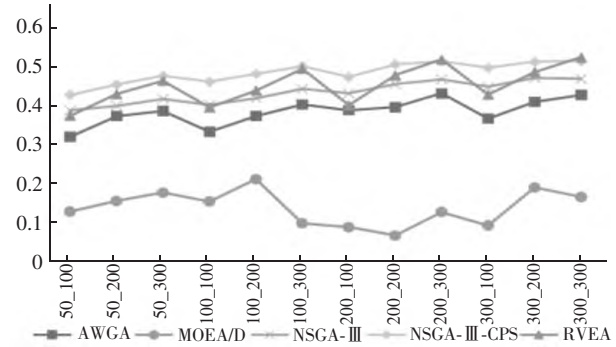


图4 HV 指标对比图

同时本文还采用 C 指标评估 NSGA-Ⅲ-CPS 算法获得解集的质量, C 指标是指 2 个解集之间的覆盖率, C 指标为

$$C(A,B) = \frac{|\{j \in B \mid \exists i \in A, j < i\}|}{|B|},$$
其中 $j < i$ 代表个体 i 支配个体 j . 由此可知, 当 $C(A,B) > C(B,A)$ 时, 解集 A 优于解集 B , 反之则解集 B 优于解集 A . 在这里需要强调, $C(A,B)$ 不等于 $1 - C(B,A)$, $C(A,B) = 1$ 代表 B 中的每一个个体至少被 A 中的一个个体支配, $C(A,B) = 0$ 代表 A 中个体均不支配 B 中个体.

令 A 代表 NSGA-Ⅲ-CPS 算法获得的解集, B 代表其他 4 种算法获得的解集, 实验结果如图 5 所示. 由图 5 可知 NSGA-Ⅲ-CPS 算法获得的解集质量明显优于其他 4 种算法获得的解集质量.

另外, 本文将 NSGA-Ⅲ-CPS 算法求解得到的结果与文献[10]的实验结果对比(见表 2). 方案 0 为文献[10]的结果, 方案 1 ~ 方案 4 为本文得到的部分结果, 其中 +13.3% 代表目标优化效果提升了 13.3%. 从表 2 可以看出: 通过优化在带罐时间内的供油速率, 以上 4 个方案的供油罐的切换成本 J_3 均提升了 13.3%. 方案 1 和方案 4 在用罐成本 J_4 不变的情况下, 管道原油混合成本 J_1 与罐底原油混合成本 J_2 均得到了较大的提升, 这说明当厂区内的供油罐资源较为充足时, 方案 1 和方案 4 是不错的选择. 方案 2 不仅能够进一步降低用罐成本 J_4 , 而且还能保证管道原油混合成本 J_1 与罐底原油混合成本 J_2 的优化效果. 方案 3 在方案 2 的基础上更进一步地降低了用罐成本 J_4 , 而代价仅仅是略微提高了罐底原油混合成本 J_2 . 当厂区内的供油罐资源较为紧张

时,可以考虑方案 2 和方案 3. 以上实验结果说明了 NSGA-III-CPS 算法在求解该问题上的有效性以及优越性,调度人员可以根据不同的情况进行择优选择.

图 6 为在优化过程中所获得的较优详细调度计划的甘特图.

表 2 本文结果与文献[10]的实验结果对比

方案编号	J_1	J_2	J_3	J_4
0	59	28	15	10
1	23(+ 61.0%)	20(+ 28.6%)	13(+ 13.3%)	10(0%)
2	34(+ 42.4%)	20(+ 28.6%)	13(+ 13.3%)	9(+ 10.0%)
3	34(+ 42.4%)	31(- 10.7%)	13(+ 13.3%)	8(+ 20.0%)
4	24(+ 59.3%)	10(+ 64.3%)	13(+ 13.3%)	10(0%)

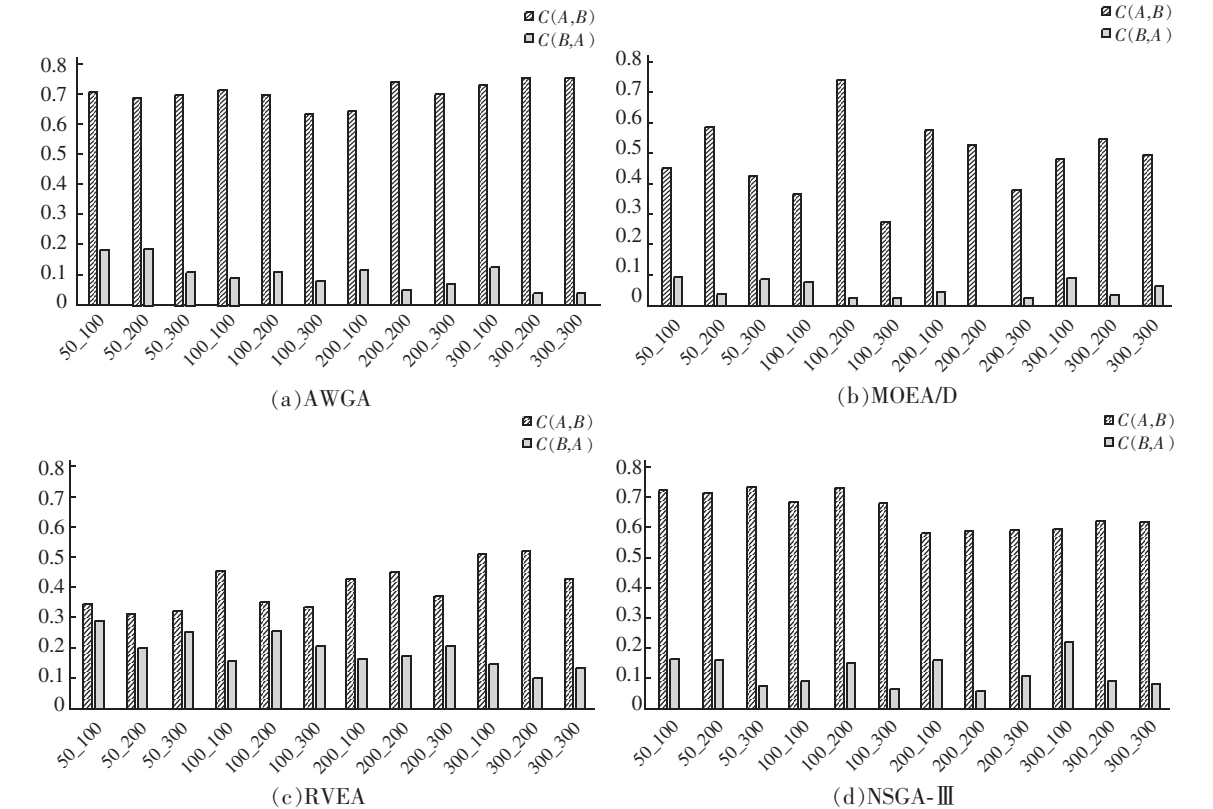


图 5 C 指标对比图

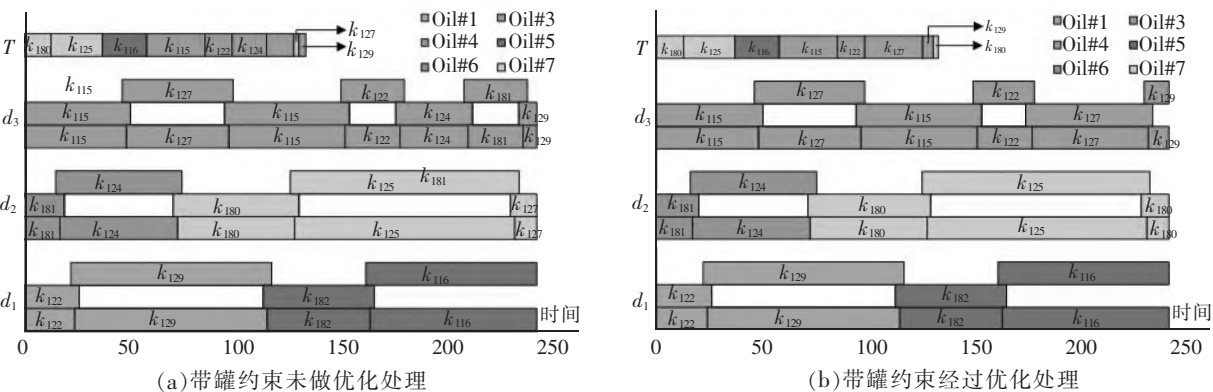


图 6 详细调度计划甘特图

从图 6(a) 可以看出:由于转运决策 T_6 的开始时间为 97.6 h,并且 d_3 中的第 2 个供油决策 $F_{3,2}$ 在带罐约束期间内的供油速率只是被简单处理为 $0.5f_3$,从而导致该决策的结束时间为 98.0 h,所以 k_{127} 无法被选择为 T_6 的供油罐,只能选择其他空罐,因此,罐底混合成本提高.而在图 6(b) 中的在带罐约束期间内的供油速率经过优化处理,决策 $F_{3,2}$ 在带罐约束期间内的供油速率被优化为 $0.6f_3$,使得

供油罐 k_{127} 可以在时间点 97.6 h 处得到提前释放, T_6 的供油罐也因此可以选择 k_{127} , 从而使得罐底混合成本降低了. 同时也因为在图 6(b) 中可以选择更大容量的 k_{127} 作为 $F_{3,5}$ 的供油罐, 所以整个调度减少了一次油罐切换, 即油罐切换成本降低了, 这还避免了在图 6(a) 中的 k_{181} 产生的罐底混合成本以及在管道内的一次原油混合. 因此, 以上例子可以充分说明: 通过优化在带罐约束期间内的供油速率, 在降低罐底混合成本和换罐成本的同时, 还能降低管道混合成本.

5 结论

本文基于已有文献, 将问题分为上下层, 并假设在上层蒸馏塔炼油计划已知的情况下, 对下层优化详细调度计划, 用以实现上层获得的蒸馏塔炼油计划. 由于大量约束条件的存在, 所以解决下层优化问题已经十分困难, 本文在原有优化问题的基础上考虑了带罐约束, 由于带罐约束特有的复杂性及不确定性, 所以下层的调度优化过程更加困难. 本文首先分析了带罐约束过程, 以随机决策的方式确定在 2 个供油罐共同供油时的供油速率, 然后从保证调度决策可行的角度出发, 基于指派思想得到新的带罐约束处理策略, 从而可以使用进化算法优化考虑带罐约束的原油详细调度问题. 同时为了解决引入带罐约束而带来的求解效率低下的问题, 本文改进了在交叉操作中父代的选择策略, 提出了一种新的 NSGA-III-CPS 算法. 为了验证该算法的有效性和优越性, 本文基于 C 指标和 HV 指标, 使用了 4 种经典的多目标进化算法与之比较, 实验结果表明 NSGA-III-CPS 算法在解决此问题时具有更好的性能; 本文还与已有研究的调度结果进行对比, 可以看出罐底混合成本、管道混合成本、供油罐切换成本以及用罐成本均得到了较大的优化. 但是按照最大转运量对原油进行转运, 可能导致部分不被紧急使用的原油提前转运到了厂区供油罐中, 这不仅占用了供油罐资源, 还可能会出现急需使用的原油无法按时转运的情况, 因此, 优化原油转运量、进一步降低成本是下一步研究方向.

6 参考文献

- [1] 亓仁东, 王勇. 原油加工成本控制分析 [J]. 广东化工, 2013, 40(16): 118-119.
- [2] LI Fupei, YANG Minglei, DU Wenli, et al. Development and challenges of planning and scheduling for petroleum and petrochemical production [J]. Frontiers of Engineering Management, 2020, 7(3): 373-383.
- [3] DE ASSIS L S, CAMPONOGARA E. A MILP model for planning the trips of dynamic positioned tankers with variable travel time [J]. Transportation Research Part E: Logistics and Transportation Review, 2016, 93: 372-388.
- [4] SCHNEIDER G A, NEVES F, MAGATAO L, et al. A mathematical programming approach to optimize the scheduling of tanks in oil refineries [J]. IEEE Latin America Transactions, 2016, 14(2): 818-830.
- [5] BAYU F, SHAIK M A, RAMTEKE M. Scheduling of crude oil refinery operation with desalting as a separate task [J]. Asia-Pacific Journal of Chemical Engineering, 2020, 15(6): 2539.
- [6] 伍青青, 程辉. 连续汽油调和调度问题的建模与优化 [J]. 信息与控制, 2020, 49(5): 615-624, 632.
- [7] DA NEIRO S M, MURATA V V, JAHN B, et al. Dealing with multiple tank outflows and in-line blending in continuous-time crude oil scheduling problems [J]. Industrial & Engineering Chemistry Research, 2019, 58(11): 4495-4510.
- [8] DE ASSIS L S, CAMPONOGARA E, MENEZES B C, et al. An MINLP formulation for integrating the operational management of crude oil supply [J]. Computers & Chemical Engineering, 2019, 123: 110-125.
- [9] WU Naiqi, ZHOU Mengchu, LI Zhiwu. Short-term scheduling of crude-oil operations: enhancement of crude-oil operations scheduling using a Petri net-based control-theoretic approach [J]. IEEE Robotics & Automation Magazine, 2015, 22(2): 64-76.
- [10] WU Naiqi, CHU Chengbin, CHU Feng, et al. Schedulability analysis of short-term scheduling for crude oil operations in refinery with oil residency time and charging-tank-switch-overlap constraints [J]. IEEE Transactions on Automation Science and Engineering, 2011, 8(1): 190-204.
- [11] 于国龙, 崔忠伟, 左羽. 基于离散粒子群优化的 MPSoC 节能调度算法 [J]. 江西师范大学学报(自然科学版), 2016, 40(3): 307-311.
- [12] 庄景明, 彭昕昀. 基于改进遗传算法的新鲜农产品配送路线优化研究 [J]. 江西师范大学学报(自然科学版), 2012, 36(4): 399-402.
- [13] HOU Yan, WU Naiqi, ZHOU Mengchu, et al. Pareto-optimi-

- zation for scheduling of crude oil operations in refinery via genetic algorithm [J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017, 47(3): 517-530.
- [14] HOU Yan, WU Naiqi, LI Zhiwu, et al. Many-objective optimization for scheduling of crude oil operations based on NSGA-III with consideration of energy efficiency [J]. Swarm and Evolutionary Computation, 2020, 57: 100714.
- [15] 侯艳, 黄康焕, 张亿仙, 等. 原油一次加工过程的多目标调度优化 [J]. 工业工程, 2020, 23(4): 131-139
- [16] 侯艳, 伍乃骐, 白丽平. 原油处理过程短期生产计划优化 [J]. 工业工程, 2014, 17(3): 79-85.
- [17] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints [J]. IEEE Transactions on Evolutionary Computation, 2014, 18(4): 577-601.
- [18] ZHANG Qingfu, LI Hui. MOEA/D: A multi-objective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731.
- [19] KIM I Y, DE WECK O L. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation [J]. Structural and Multidisciplinary Optimization, 2005, 29(2): 149-158.
- [20] CHENG Ran, JIN Yaochu, OLHOFFER M, et al. A reference vector guided evolutionary algorithm for many-objective optimization [J]. IEEE Transactions on Evolutionary Computation, 2016, 20(5): 773-791.
- [21] EMMERICH M, BEUME N, NAUJOKS B. An EMO algorithm using the hypervolume measure as selection criterion [EB/OL]. [2022-03-19]. https://doi.org/10.1007/978-3-540-31880-4_5.
- [22] ZITZLER E, DEB K, THIELE L. Comparison of multi-objective evolutionary algorithms: empirical results [J]. Evolutionary Computation, 2000, 8(2): 173-195.

The Multi-Objective Short-Term Scheduling Optimization with Charging-Tank-Switch-Overlap Constraint in Refinery

HOU Yan, REN Bingfei, TENG Shaohua, ZHU Qinghua

(School of Computers, Guangdong University of Technology, Guangzhou Guangdong 510006, China)

Abstract: There are not only multiple optimization objectives, but also complex constraints in short-term refining scheduling problem. Because of the introduction of Charging-Tank-Switch-Overlap constraint, it will intensify the competition of distillation for charging tanks and lead to the exponential expansion of the scheduling solution space and the sharp deterioration of the collaborative optimization effect. Based on this, in order to ensure the reliability and efficiency of scheduling decisions at an arbitrary oil supply rate, the short-term refining multi-target scheduling optimization algorithm is proposed under Charging-Tank-Switch-Overlap constraints, called NSGA-III-CPS. Firstly, giving a group of refining tasks, the evolution algorithm in this paper can obtain feasible scheduling schemes, parameters, and strategies for each speed. Secondly, the optimized scheduling schemes and strategies through refining cost calculation and multi-objective evaluation can be selected. Moreover, the evolutionary algorithm in this paper improves the cross operator by optimizing the parent selection strategy, conducts rapid non-dominant ranking of the parent population, and the parents with a high dominance rank have a higher probability of being selected, so that the parent's good genes can be better inherited to the offspring. The algorithm considers the multi-dimensional refining scheduling objectives, such as the cost of the charging tanks and the charging tank switching, the mixing cost of crude oil in the pipeline and at the bottom of the tank, and obtains the refining task optimization scheduling strategy at the variable speed with consideration of Charging-Tank-Switch-Overlap constraint. Finally, by comparing with four classical multi-objective evolution algorithms, the experimental results show that it have a 10.0% to 64.3% improvement in the refining cost by applying the proposed algorithm and strategy.

Key word: refining; production planning and scheduling; charging-tank-switch-overlap constraint; NSGA-III-CPS

(责任编辑:冉小晓)