



Research  
Smart Process Manufacturing—Article

## 非线性调合调度问题的全局优化

Pedro A. Castillo Castillo<sup>a</sup>, Pedro M. Castro<sup>b</sup>, Vladimir Mahalec<sup>a,\*</sup>

<sup>a</sup> Department of Chemical Engineering, McMaster University, Hamilton, ON L8S 4L8, Canada

<sup>b</sup> Center for Mathematics, Fundamental Applications and Operations Research, Faculty of Sciences, University of Lisbon, Lisbon 1749-016, Portugal

### ARTICLE INFO

#### Article history:

Received 7 December 2016

Revised 16 February 2017

Accepted 20 February 2017

Available online 28 March 2017

#### 关键词

全局优化

非线性汽油调合

连续时间调合模型

分段线性松弛

### 摘要

汽油调合调度是炼油工业中的一个重要问题。一方面, 该问题具有调度问题本身所具有的组合优化特性; 另一方面, 由于调合各种具有不同品质特性的物料, 导致该优化问题的非凸性。本文提出一种新型的全局优化算法, 用于求解基于连续时间汽油调合调度模型的混合整数非线性规划问题。该模型包含调合配方优化、分配问题及若干操作特性和约束; 算法上采用分段 McCormick 松弛 (PMCR) 和规范多参数解聚 (NMDT), 计算全局最优解的估计值, 其松弛技术将双线性项中的一个变量值域进行分割, 进而在每一个分段上产生凸松弛; 通过增加分段数和缩减变量的值域, 提高对全局最优解的估计。本文利用该算法求解四个案例, 并与两个商业全局优化求解器和两个启发式算法进行比较, 结果表明, 本文提出的全局优化算法与商业求解器具有同等水平, 但是在计算速度上稍逊于启发式算法。

© 2017 THE AUTHORS. Published by Elsevier LTD on behalf of the Chinese Academy of Engineering and Higher Education Press Limited Company. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. 引言

最优生产调度策略的计算和实施可以降低操作运行成本、增加收益、避免违反实际环境中的约束[1]。但是, 复杂的工业装置通常具有多种产品、库存、分配子系统, 多种不同的原料、中间产物和最终产品, 以及这些元素之间错综复杂的联系, 从而导致调度问题成为一个困难的决策问题。

调度问题通常涉及四个主要的决策[1]。①定义所需任务以满足相应的目标、要求和(或)需求指标; ②设定每个任务在网络中可利用的处理单元或资源; ③定义任务执行序列; ④决定每个任务的开始和终止时间(图1)。最优调度的决策是最大或最小化所需的目标, 如利润、

总成本、提前期等。基于数学规划的调度软件和工具在实际应用中越来越常见。

由于汽油在炼油厂的总收益中占60%~70%, 因此汽油调合的调度是一个非常重要的工业问题[2-4]。在汽油调合系统中, 来自于专用供应罐的成分在调合罐中混合或在线调合后, 被输送到成品罐。调合罐具有输入流或输出流(图2), 类似于间歇操作。与之相反, 在线调合器属于连续操作(图3)。除了上述四个决策, 调合调度操作还包含调合配方的确定, 即保证产品质量指标的各调合成分的量。

本文的汽油调合系统的描述见图3。汽油调合通过一个或多个连续调合器实现, 每一个调合器连接到调合成分的来源上。在炼油装置的配置中, 调合后的物料可

\* Corresponding author.

E-mail address: [mahalec@mcmaster.ca](mailto:mahalec@mcmaster.ca)

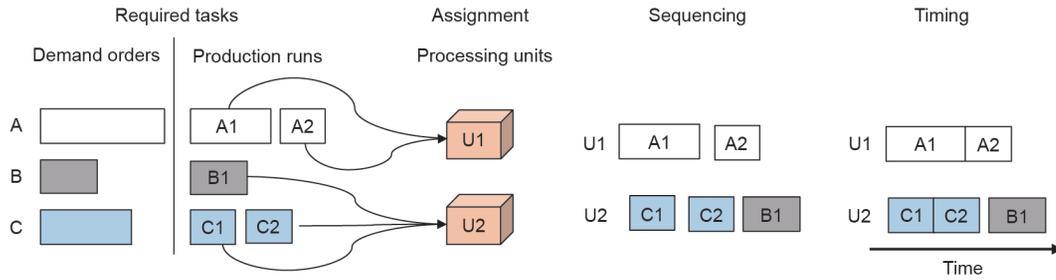


图1. 主要的调度决策变量。

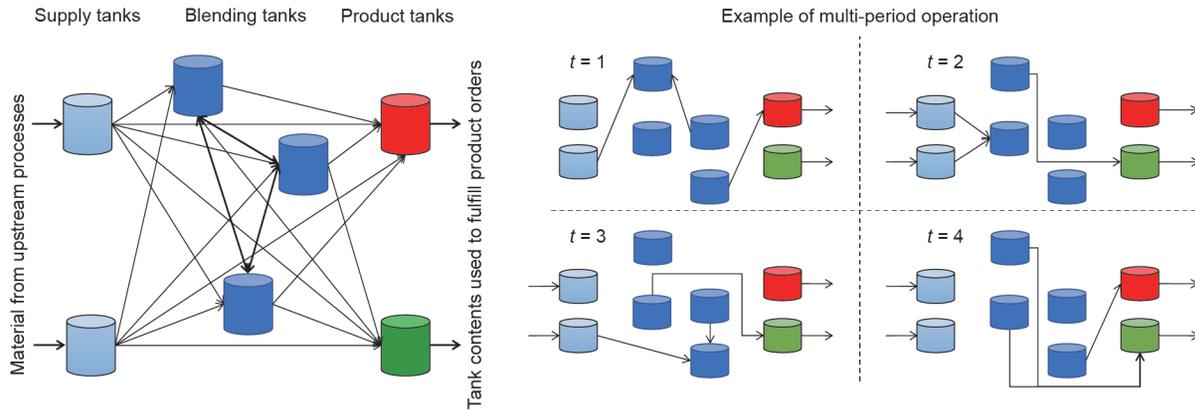


图2. 间歇调合系统。

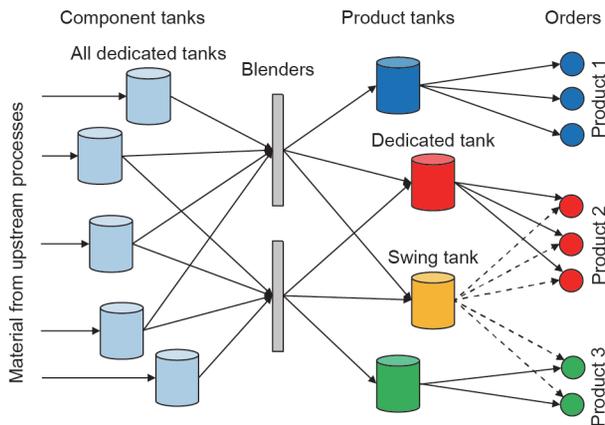


图3. 连续汽油调合系统。

以流向储罐也可以直接进入管道。由于需要生产多种不同牌号的汽油产品(如常规、中级、高级), 调合器需要从调合一种牌号切换到另一种。每次切换需要全部或部分重新排列调合进料的管道, 导致调合生产能力的损失。而且, 切换到不同的品质属性区间需要重新设置或重新校正分析仪表。

汽油最重要的几个品质属性包括研究法辛烷值(RON)、马达法辛烷值(MON)、瑞德蒸气压(RVP)、密度(D)、硫含量(S)、芳烃(A)和烯烃(O)含量。RON、MON和RVP的调合并不是线性的, 因此, 在调合模型中对这些品质属性考虑非线性的调合可以增加所得解的

准确性、降低品质过剩[4,5]。

自20世纪60年代以来, 研究者在推导“调合指数”上做了大量工作。在调合成分的实际品质属性的非线性转换的基础上, 通过线性组合预测调合的品质。即使应用这种方法, 仍然存在两个问题:

(1)如果在罐中进行产品调合, 总是存在一些先前调合残留的物料(称之为罐底)。罐中物料的属性必须包含在新的调合配方的计算中, 因而即使在使用调合指数的情况下, 用于多周期调度的调合模型也是非线性的。

(2)利用这些指数计算的调合物性不是100%准确的, 年累计品质过剩会导致极大的成本增加, 因而有必要使用非线性调合模型用于计算辛烷值在内的物性。

调度问题的数学形式一般表示为混合整数线性规划(MILP)问题。但是, 由于汽油调合问题本质上的非线性, 为了保证精度, 有必要采用混合整数非线性规划(MINLP)形式。大部分的非线性项是非凸的, 在凸优化策略无效的情况下, 全局优化算法有其必要性。在描述本文所提全局优化算法之前, 首先对前人的研究做一简要综述。

### 1.1. 炼油装置调度问题综述

基于对时间域的处理方式, 调度模型主要可以分为

两类：离散时间和连续时间形式。在离散时间形式中，时间域被分为若干个固定起始和终止时间的时长，而在连续时间模型中，时间域被分为由优化算法确定的时间片。尽管连续时间模型比对应的离散时间模型产生更少的离散变量，但是由于其较差的松弛[6]和对计算性能的平衡，会导致形式更为复杂且产生较多“大M”形式的约束。更多关于调度问题形式的综述见文献[1,7-9]。

由于汽油调合问题在商业上的重要性，诸多学者对其进行了研究，而且其非凸特性，使其成为测试不同形式和算法的合适对象。汽油调合问题中的运行约束与这些因素有关：多用途罐、不同的调合器、不同的储罐策略(如是否允许同时接收和交付物料)、实际问题(如最小化调合运行和设置时间)。这些因素在公开文献的调度模型中并没有被同时考虑。在某些情况下，假设调合配方是固定的，即不能被优化。下游的分配问题或运输问题(即定时交货任务以满足需求)有时也会作为调度问题的一部分。

Méndez等[2]提出了离散时间和连续时间MILP模型用于调度汽油调合操作，采用迭代法处理非线性调合规则的同时保证模型的线性，但多个关键的操作约束和分配问题都没有被考虑。

Jia和Ierapetritou[10]提出了连续时间MILP模型用于同时调度汽油调合任务和分配操作，通过采用指定的首选配方保证了模型的线性，他们的模型随后被扩展到炼油装置主要处理单元的调度操作中[11]。

Glismann和Gruhn[12]采用基于离散时间模型的双层方法，首先通过非线性规划问题计算调合配方和生产目标，然后利用这些调合配方和生产指标求解MILP形式的短期调度问题。调度模型基于资源-任务网络(RTN)形式，并且没有考虑多用途罐或交货调度的问题。

Li等[13]提出了特性化所有装置(即调合器和储罐)的公共时间网络的连续时间MILP模型，Li和Karimi[3]通过使用特定装置时间网格及包含大部分实际过程中的运行约束，扩展和改进了这一模型，两个模型都利用调合指数优化调合配方。基于这两项工作，Li等[4]提出了一个特定装置连续时间MINLP形式，其中，非线性项来源于强制调合速度为定值。

Castillo和Mahalec[14]提出了三层分解算法，配方的优化可以通过线性和(或)非线性调合规则实现，他们的工作考虑了分配问题、调合尺寸阈值约束、并行非等效调合器、中间储罐、产品相关设置时间。每一层均实现一个离散时间模型，第一层优化调合配方，第二层近似生产调度，第三层计算详细的调合-交货调度。由于

第三层全时域调度模型的规模较大，在各子区间内进行求解。与以往研究[3,13]相比较，通过该方法能获得更好的解，且在大规模问题的执行时间上要少两个数量级。在随后的工作中，Castillo和Mahalec[15]提出了与Li和Karimi[3]的研究相比具有较大修改的连续时间调度模型(具有更少的二进制整数变量[16])，用于处理第三层的问题，在非线性调合调度问题上可以以较短的时间获得非常接近全局最优的解。

Lotero等[17]提出了多周期汇集问题的新的形式，他们将这种离散时间MINLP形式称作是source-based模型(类似于Castro的split-fraction模型[18])和concentration-based模型的混合模型[19]。通过增加冗余约束改进线性松弛形式，利用二段式MILP-NLP方法求解该模型，其中，MILP模型为原MINLP模型的松弛形式，NLP模型通过固定原模型中的整数变量获得。该算法在每个迭代中给MILP模型添加整数、优化性或可行性的切割，并且当MILP和NLP解的差异小于一个预先指定的容差时停止。

Cerdá等[20]提出了连续时间MILP形式的模型，在求解时利用浮动时间片动态分配到时间段，该模型包含了实际中大部分的操作约束。Cerdá等[21]随后扩展了该模型用于处理非线性调合规则，进而形成了连续时间MINLP模型，通过线性调合指数替换非线性调合规则导出近似的MILP形式。MILP中计算的二进制整数变量的值在原始MINLP问题中固定，因而形成NLP问题，用于获得原问题的近似最优解。

## 1.2. 全局优化文献综述

在过去的30年中，非线性非凸问题的全局优化研究获得了大量的关注。即使是最强大的商业求解器已被开发出来[22,23]，该领域依然有持续的发展。

全局优化算法的共同点是产生问题的凸松弛，以提供目标函数值的下界(LB)及产生可行解(上界, UB)的方式。此外，通过使上、下界接近的方法，使得优化距离可以降低到 $\epsilon$ -容差内。

计算紧的下界是非常重要的，通过保留可行域空间但具有健壮松弛的等效问题替换原问题形式(汇集问题的不同形式是一个众所周知的例子[24])。另一个选项是重新组织模型的约束并增加其他约束增强松弛，尽管在原始空间中可能是冗余的，这一方法被称为重建线性化技术[25]，其缺点是没有系统的方法获得可以增强松弛的操作。

一种保证收敛到全局最优解的广泛采用的方法是空间分支定界[26,27],是商业求解器如BARON[22]、ANTIGONE[23]、COUENNE[28]和SCIP[29]不可或缺的部分。空间分支定界通过迭代的逐个降低变量的值域,进而增强凸松弛的质量。考虑到如果初始松弛质量太差,在存在较多非凸项的情况下,收敛速度将会非常慢,因此具有较好的分支策略和边界紧缩策略是非常重要的,基于优化性的边界紧缩(OBBT)将在下文介绍。尽管OBBT通常应用在根节点或达到一定深度之后[28],但最近的研究表明,在每个节点应用OBBT可以获得显著降低的优化性距离[30]。OBBT包含对每个变量求解一个最小化和一个最大化问题(在非凸项中的变量),以产生更紧的下界和上界。OBBT可以以序贯方式求解——具有可以产生更紧的变量边界的优点和导致计算量较大的缺点,也可以采用并行方式[31]。

双线性项在过程系统工程中是主要的非凸性的来源,它可以通过考虑变量完整边界[32]或者分段降低值域[33,34]的McCormick包线进行松弛。同步的值域分割包含增加新的二进制整数变量,在无限的分段数的限制下保证全局优化性,可以避免采用空间分支定界。其中一个严重的问题是分段数紧密相关的问题规模,先前的分段线性策略[33]中模型规模线性增加,最近的形式使得规模呈对数增加,后者举例如下。

Misener等[35]提出了一个全局优化算法用于标准的汇集问题并指出对数策略在分段数超过8时具有更多的优势。Kolodziej等[19]提出了MINLP形式的多周期汇集问题,其中将储罐的动态库存作为调合成分导致了非线性。采用基数离散技术将每个双线性项中的一个变量分段获得MILP松弛,该松弛技术被称为多参数解聚技术[36]。Castro[37]提出了规范多参数解聚技术(NMDT),通过将变量的上下界内的区间进行离散(属于 $[0,1]$ ),该方法的优点在于即使每个离散变量的值域不同,其分段数目仍保持一致。NMDT成功应用于求解多周期调合问题获得全局最优解,包括单独的方法[18,38,39]或集成在分支定界策略中[30]。

总之,上文所引文献中的工作使得在一定规模内汽油调合调度问题的最优解可以保证。但是,处理大规模问题时,计算量和全局最优解的验证依然是一个困难的挑战。

### 1.3. 本文创新点

本文提出新型的确定性全局优化算法用于求解非凸

MINLP或NLP模型,由于该模型包含双线性(和)二次项导致的非线性,因此该算法的主要特点有:

- (1) 利用不同的线性或分段线性松弛技术产生原非凸问题的凸松弛;
- (2) 凸松弛的解集为局部非线性求解器提供初始点以获得原问题的可行解;
- (3) 分段线性松弛的分段数量动态增加;
- (4) 通过OBBT降低非线性项中变量的值域;
- (5) 可行解求解和OBBT计算通过并行计算策略完成。

该算法在不同的汽油调合调度案例上进行测试。对于这类问题,结果显示所提算法优于两个商业全局优化求解器,或与之相当。

本文结构如下:第2节描述问题形式和模型假设;第3节对本文采用的调度模型进行综述,并给出用于辛烷值调合的非线性方程;第4节简单介绍用于估计全局最优解的分段线性松弛技术;第5节描述用于降低非线性项中变量值域的OBBT方法;第6节提出了全局优化算法的步骤;第7节包含描述测试案例的数据;第8节给出利用所提算法所得的结果及与其他方法的比较;第9节给出本文的结论。

## 2. 问题描述

给定一个调合系统(即储料罐、调合头以及它们之间的连接部分,如图3所示),一个调度周期内的一组调合组分以及对应的供应量和质量特性数据,一组产品以及对应的最大、最小产品性质参数,每种产品的交货单,初始库存水平,需要决定调合配方、生产和交货顺序、所有罐的库存配置,同时最小化参调原料成本、切换成本(即参调原料流量、交付相同订单的储罐个数和中间储罐中产品转变)和滞期成本(即延迟交货)。

考虑如下约束:

- (1) 如果一个调合器生产一种产品,必须有一个最小生产量;
- (2) 一个调合器在同一时间最多只能生产一种产品,一旦开始调合,调合器必须工作一段时间才能切换到生产另一种产品;
- (3) 切换产品时每个调合器都需要一个最小启动时间;
- (4) 同一时间每个调合器的产品最多只能进入一个产品罐(行业惯例);

(5) 同一时间产品罐只能存储一种产品;

(6) 产品罐不能同时接收和交付物料。

本文考虑如下假设:

(1) 来自上游的每种组分的流量是分段连续的;

(2) 每种组分的特性是分段连续的;

(3) 每个调合器中都能达到理想混合;

(4) 每种调合组分都只有一个储罐;

(5) 只有中间储罐可以变更产品存储种类(即从储存一种产品变为储存另一种产品);

(6) 对于中间储罐, 不同产品之间的转变次数可以忽略;

(7) 对于每一个调合器, 产品调合之间的转换次数取决于产品, 但与顺序无关;

(8) 每个订单仅包含一种产品(每个包含不同产品的原始订单都能被分解成单独的产品订单);

(9) 所有订单都在调度周期内完成。

总的来说, 该问题考虑调合调度和交货操作、配方决策, 以及一个调度周期内产品在多功能储罐中的分配。

### 3. 汽油调合调度模型

本文采用了Castillo和Mahalec[16]提出的调度模型。该模型采用连续时间公式, 考虑非线性调合方程, 不允许产品罐同时接收和交付物料。这是一个非凸的混合整数非线性规划模型, 被表示为模型P(或问题P)。本节将重点描述该调度模型的主要特征。

该调度模型使用不同长度的特定时间间隔来确定何时需要在每个单元(这里指调合器和储罐)中执行特定任务。我们分配足够多的时间间隔, 这很可能会高于调合每个等级所需的时间间隔数目。这保证有足够的自由度(足够的可用切换)来满足不同的产品交付计划。

一个单位时隙的开始时间等于上一个片段的结束时间。第一个单位时隙开始于调度周期的起点, 最后一个单位时隙恰好匹配于周期的终点。调合计划开始于每一个时间片段的起点, 但可以终止于片段终点之前。产品储罐的交货任务可以开始和终止在相应的时间片段之内。假设组分罐以一些特定速度连续接收原料(即供应曲线)。时段用来描述当调合原料供应率和(或)质量发生变化时的点。时隙分配给这些时段。时隙必须终止于所指定的时段内。然而, 对于组分罐, 一个时段内最后一个时隙必须恰好终止在时段边界上[以此遵守组分的供

应率和(或)质量变化]。图4给出了一个调合系统的具体时隙, 这个系统包括两个参调组分罐(CT1、CT2)、一个调合器和两个产品罐。单位时隙1和2预先分配给时段1, 时隙3和4分配给时段2。注意, 通过优化已经确定了CT2中的时隙3和PT1中的时隙4为零长度。

调度模型的目标是最小化调合成本(即原料成本)、与调合批次有关的切换成本、多功能储罐中的原料转变、交付批次的数目(即用来从给定的罐中交付指定订单的时隙数), 以及滞期成本。在交货调度中增加相应的惩罚项, 以避免同时从多个储罐交付同一订单, 并减少从同一储罐间隙性交付同一订单的情况。

在模型中采用二进制变量来确定每个时隙下的离散决策, 内容如下:

(1) 调合器出口进入哪个产品罐(每个变量代表一种调合器-产品罐连接关系);

(2) 每个产品罐中存储的是什么牌号的汽油(每个变量代表一种牌号产品罐);

(3) 每个产品罐部分或全部对应哪个订单(每个变量代表一种产品罐-订单连接关系)。

利用这些二进制变量, 其他离散决策可以用0-1连续变量建模, 例如:

(1) 每个调合器生产什么牌号的汽油;

(2) 一个调合器的当前状态(运行还是空闲);

(3) 调合器从运行过渡到空闲, 或者从空闲过渡到运行;

(4) 何时新调合器开始运行;

(5) 调合器中的产品转换;

(6) 多功能罐中的产品转变。

该调度模型也考虑了可变调合率、可变交付率、调合器和产品的启动时间(即清洗或传感器重校带来的空闲时间), 从待调组分罐到调合器的输送速度最大, 调

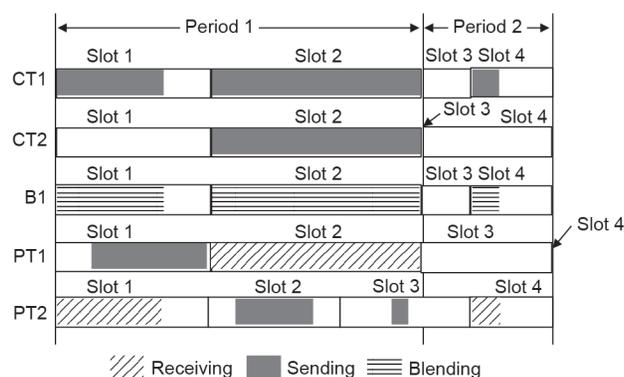


图4. 调度模型时隙示例。

合器尺寸最小以及每个调合器和产品的运行时间最小。其他约束包括物料平衡、产品组分规格、产品质量规格以及线性和(或)非线性混合方程。

求解该模型的全局最优解的困难来自于以下因素：

- (1) 大量离散决策，主要关系到时隙数目、汽油牌号、调合器、产品罐和订单(问题的组合)；
- (2) 非线性混合方程(问题的非凸性)；
- (3) 所有考虑到的操作约束。

Castillo和Mahalec[16]发现引入关于最小调合成本和最小切换成本的约束可以提高求解能力并降低中小规模问题的求解时间。最小调合成本的计算方法参考Castillo等[40]提出的步骤。

非线性混合公式描述见下文，它们被重写为非线性项仅为双线性或平方的形式。

### 非线性调合公式

公式(1)~(19)是对辛烷值调合ethyl RT-70模型[5,41]进行重构的形式。公式(1)、(13)、(14)、(18)中存在二进制项。公式(15)、(16)、(17)、(19)中存在平方项。下面介绍主要集合、下标、变量和参数。参调组分构成集合 $\mathbf{I} = \{i\}$ ， $\mathbf{BL} = \{bl\}$ 由调合器组成， $\mathbf{N1} = \{n\}$ 是时隙，集合 $\mathbf{QN} = \{(\theta, n)\}$ 代表与每个质量特性 $\theta$ 相关的时隙。变量 $V_{\text{comp}}(i, bl, n)$ 代表在时隙 $n$ 期间待调组分 $i$ 进入调合器 $bl$ 的体积。变量 $V_{\text{blend}}(bl, n)$ 为时隙 $n$ 期间通过调合器 $bl$ 的物料总体积。变量 $r(i, bl, n)$ 代表时隙 $n$ 期间组分 $i$ 进入调合器 $bl$ 的体积分数。参数 $Q_{bc}(i, e, \theta)$ 表示待调组分 $i$ 的质量属性 $e$ 和质量概况 $\theta$ 的值。 $sens(i, \theta)$ 是一个代表辛烷值灵敏度的参数，它和辛烷值RON-MON不同。Ethyl RT-70模型中的系数参考Singh等[5]如下：

$$V_{\text{comp}}(i, bl, n) = r(i, bl, n)V_{\text{blend}}(bl, n) \quad \forall i, bl, n \in \mathbf{N1} \quad (1)$$

$$r_{\text{avg}}^{\text{RON}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta) \quad \forall e = \text{"RON"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (2)$$

$$r_{\text{avg}}^{\text{MON}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta) \quad \forall e = \text{"MON"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (3)$$

$$Ol_{\text{avg}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta) \quad \forall e = \text{"Ol"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (4)$$

$$Ar_{\text{avg}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta) \quad \forall e = \text{"Ar"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (5)$$

$$Ol_{\text{avg}}^{\text{sq}}(bl, n) = \sum_i r(i, bl, n) \left[ Q_{bc}(i, e, \theta)^2 \right] \quad \forall e = \text{"Ol"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (6)$$

$$Ar_{\text{avg}}^{\text{sq}}(bl, n) = \sum_i r(i, bl, n) \left[ Q_{bc}(i, e, \theta)^2 \right] \quad \forall e = \text{"Ar"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (7)$$

$$sens_{\text{avg}}(bl, n) = \sum_i r(i, bl, n)sens(i, \theta) \quad \forall i, bl, n \in \mathbf{N1} \quad (8)$$

$$sens_{\text{avg}}^{\text{RON}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta)sens(i, \theta) \quad \forall e = \text{"RON"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (9)$$

$$sens_{\text{avg}}^{\text{MON}}(bl, n) = \sum_i r(i, bl, n)Q_{bc}(i, e, \theta)sens(i, \theta) \quad \forall e = \text{"MON"}, bl, n \in \mathbf{N1}, \theta : (\theta, n) \in \mathbf{QN} \quad (10)$$

$$Q_{\text{pr}}(bl, e, n) = r_{\text{avg}}^{\text{RON}}(bl, n) + a_1 \left[ sens_{\text{avg}}^{\text{RON}}(bl, n) - r_{\text{avg}}^{\text{RON}}(bl, n) \right] + a_2 \left[ Ol_{\text{avg}}^{\text{sq}}(bl, n) - Ol_{\text{avg}}(bl, n) \right] + a_3 \left[ Ar_{\text{avg}}^{\text{sq}}(bl, n) - 2 \cdot Ar_{\text{avg}}(bl, n) + Ar_{\text{avg}}(bl, n) \right] \quad \forall e = \text{"RON"}, bl, n \in \mathbf{N1} \quad (11)$$

$$Q_{\text{pr}}(bl, e, n) = r_{\text{avg}}^{\text{MON}}(bl, n) + a_4 \left[ sens_{\text{avg}}^{\text{MON}}(bl, n) - r_{\text{avg}}^{\text{MON}}(bl, n) \right] + a_5 \left[ Ol_{\text{avg}}^{\text{sq}}(bl, n) - Ol_{\text{avg}}(bl, n) \right] + a_6 \left[ Ar_{\text{avg}}^{\text{sq}}(bl, n) - 2 \cdot Ar_{\text{avg}}(bl, n) + Ar_{\text{avg}}(bl, n) \right] \quad \forall e = \text{"MON"}, bl, n \in \mathbf{N1} \quad (12)$$

$$r_{\text{avg}}^{\text{RON}}(bl, n) = r_{\text{avg}}^{\text{RON}}(bl, n)sens_{\text{avg}}(bl, n) \quad \forall bl, n \in \mathbf{N1} \quad (13)$$

$$r_{\text{avg}}^{\text{MON}}(bl, n) = r_{\text{avg}}^{\text{MON}}(bl, n)sens_{\text{avg}}(bl, n) \quad \forall bl, n \in \mathbf{N1} \quad (14)$$

$$Ar_{\text{avg}}^2(bl, n) = Ar_{\text{avg}}(bl, n)^2 \quad \forall bl, n \in \mathbf{N1} \quad (15)$$

$$Ol_{\text{avg}}^2(bl, n) = Ol_{\text{avg}}(bl, n)^2 \quad \forall bl, n \in \mathbf{N1} \quad (16)$$

$$Ar_{\text{avg}}^{\text{sq}}(bl, n) = Ar_{\text{avg}}^{\text{sq}}(bl, n)^2 \quad \forall bl, n \in \mathbf{N1} \quad (17)$$

$$Ar_{\text{avg}}^3(bl, n) = Ar_{\text{avg}}^{\text{sq}}(bl, n)Ar_{\text{avg}}(bl, n) \quad \forall bl, n \in \mathbf{N1} \quad (18)$$

$$Ar_{\text{avg}}^4(bl, n) = Ar_{\text{avg}}^2(bl, n)^2 \quad \forall bl, n \in \mathbf{N1} \quad (19)$$

## 4. 分段线性松弛

第1节中提到，由于MILP求解器十分成熟，因此使

用分段线性松弛变得越来越普遍。本文将采用分段McCormick松弛(PMCR)和标准多参数分解技术(NMDT)。通过这两种方法,模型P中的每个二进制项都被单一变量所替代,从而将相应的方程线性化。单一变量受到多种线性约束,给模型增加了额外的连续二进制变量。如果等于1,这些额外的二进制变量将激活双线性项中的一个变量的域(即区间)的特定间隔。区间的数目表示为 $NP$ ,并且假设所有离散变量都有相同数目的分区。PMCR方法在 $NP$ 和每个离散化变量所需的额外二进制变量数之间构建了线性关系,同时NMDT呈对数关系。对于这些方法更详尽的解释,读者可以参考文献[16,37]。

由此产生的MILP模型表示为PR模型,它是问题P的松弛。这代表模型PR的最优解是问题P全局最优解的有效估计(在最小化情况下,这将是一个下限,  $LB$ )。此外,模型PR的最优可行解的估计是问题P全局最优解的有效估计。因此,即使模型PR在给定时间内未能通过MILP求解器找到最优解,仍然可以找到一个新全局最优解的估计。分区数越多,模型PR和模型P越接近;图5给出了一个涉及单个离散变量的例子。

如果松弛比较紧,那么其最优解将非常接近原始最优解。因此,寻找原始问题P(在最小化情况下,这将是一个上限,  $UB$ )的可行解的策略是用模型PR的最优解对P进行初始化。一些MILP求解器,如CPLEX,能够为MILP问题存储多个可行解,由于初值不同可能导致P有不同的解出现,因此本文以并行方式使用多启动策略。注意,由于一些与商用求解器求解速度和鲁棒性相关的实际原因,求解NLP模型比MINLP更便捷。这是为什么在将问题P(MINLP)转化为PF(NLP)时二进制变量采用固定值的原因。模型P、PR和PF的紧缩记法如下:

模型P:

$$\begin{aligned} & \min f_0(x, y) \\ \text{s.t. } & f_m(x, y) \leq 0 \quad \forall m \in \mathbf{M} \setminus \{0\} \\ & f_m(x, y) = \sum_{(i,j) \in \mathbf{BLT}} a_{ijm} x_i x_j + B_m x + C_m y + d_m \quad \forall m \in \mathbf{M} \\ & 0 \leq x^L \leq x \leq x^U \\ & x \in \mathbb{R}^d, y \in \{0, 1\}^{l_y} \end{aligned}$$

模型PR:

$$\begin{aligned} & \min f_0^R(x, y) \\ \text{s.t. } & f_m^R(x, y) \leq 0 \quad \forall m \in \mathbf{M} \setminus \{0\} \\ & f_m^R(x, y) = \sum_{(i,j) \in \mathbf{BLT}} a_{ijm} w_{ij} + B_m x + C_m y + d_m \quad \forall m \in \mathbf{M} \\ & g_n^R(x, w, v, z) \leq 0 \quad \forall n \in \mathbf{N} \\ & g_n^R(x, w, v, z) = H'_n x + A'_n w + B'_n v + C'_n z + d'_n \quad \forall n \in \mathbf{N} \\ & 0 \leq x^L \leq x \leq x^U \end{aligned}$$

模型PF:

$$\begin{aligned} & \min f_0^F(x) \\ \text{s.t. } & f_m^F(x) \leq 0 \quad \forall m \in \mathbf{M} \setminus \{0\} \\ & f_m^F(x) = \sum_{(i,j) \in \mathbf{BLT}} a_{ijm} x_i x_j + B_m x + C_m \hat{y} + d_m \quad \forall m \in \mathbf{M} \\ & 0 \leq x^L \leq x \leq x^U \\ & x \in \mathbb{R}^d \end{aligned}$$

注意,在本节中,集合 $\mathbf{M} = \{m\}$ 代表所有初始约束,集合 $\mathbf{N} = \{n\}$ 代表分段线性松弛所需的所有约束,集合 $\mathbf{BLT} = \{(i, j)\}$ 代表所有的双线性项。变量 $x$ 和 $y$ 分别是初始连续和二进制变量,松弛策略引入的变量 $v$ 和 $z$ 分别是额外的连续和二进制变量。标量 $l_x, l_y, l_w, l_v$ 和 $l_z$ 分别代表向量 $x, y, w, v$ 和 $z$ 的长度。参数 $x^L$ 和 $x^U$ 分别是变量 $x$ 的上下界。注意二次项可以视为双线性项。

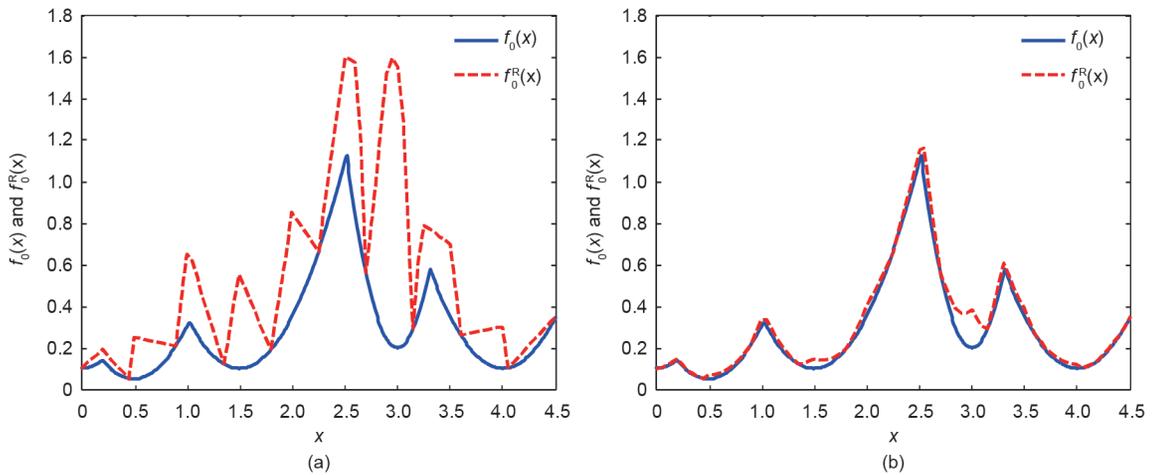


图5. 增加分段数, 松弛模型( $f_0^R$ )的准确性与原问题( $f_0$ )的比较。(a) 10个分段数; (b) 100个分段数。

## 5. 变量的紧缩界限

随着离散变量分区数的增加，**PR**模型变得越来越紧(即接近模型**P**)。然而，分区数增加将导致模型**PR**的规模增大，当分区数大到一定程度，**PR**模型的求解将变得十分困难。因此，需要一种新的方法避免构建大量分区。本文采用OBBT方法[34,42]。该方法通过求解两个优化问题(即每个变量的最大化和最小化问题)来计算这些变量的新的边界条件以减少涉及非线性项的变量域。这一步是在求得**P**的新的较优可行解之后进行的。减少了变量的域，**PR**模型在不增加分区的情况下会更接近**P**，如图6所示。

OBBT中使用的数学模型记为**PRB**模型，它被构造为**P**的松弛形式，但是具有不同的目标函数和额外的约束。为了计算变量 $x_h$ 的下界 $x_h^L$ ，目标函数被用来计算该变量的最小值。为了计算一个变量 $x_h$ 的上界 $x_h^U$ ，目标函数被用来计算该变量的最大值。为了计算新的边界，通过对原始目标函数增加额外的约束来进行松弛之后的目标函数 $f_0^R(x, y)$ 的值必须至少与当前最优可行解一样。

注意，**PR**模型和**PRB**模型可以采用不同的松弛策略。本文中**PRB**模型采用标准McCormick包线，舍弃对变量 $y$ 的完整性要求，以此将**PRB**简化为线性规划(LP)。变量 $x_h$ 的上下界通过对应LP模型的最优解来更新。对于一个最小化问题的**PRB**模型，紧凑记法如下所示：

模型**PRB**:

$$\begin{aligned} & x_h^L = \min x_h \quad (x_h^U = \max x_h) \\ \text{s.t. } & f_0^R(x, y) \leq UB \\ & f_m^R(x, y) \leq 0 \quad \forall m \in \mathbf{M} \setminus \{0\} \\ & f_m^R(x, y) = \sum_{(i,j) \in \text{BLT}} a_{ijm} w_{ij} + B_m x + C_m y + d_m \quad \forall m \in \mathbf{M} \end{aligned}$$

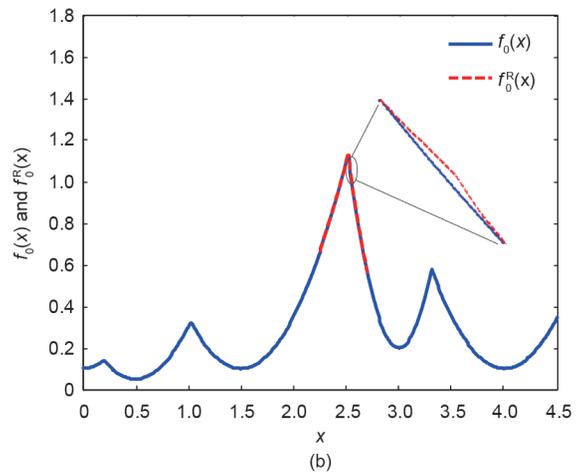
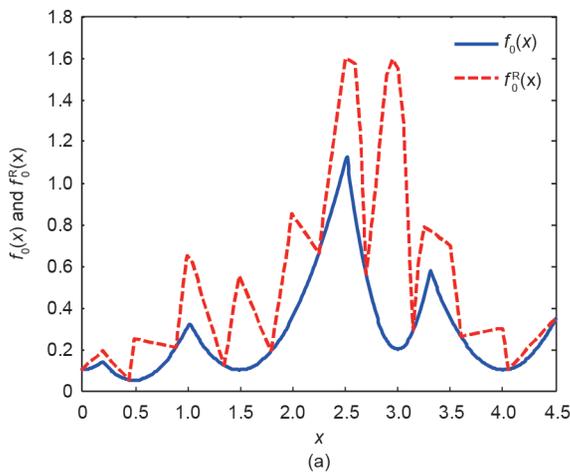


图6. 减少变量域，松弛模型( $f_0^R$ )的准确性与原问题( $f_0$ )的比较。(a) 在 $x \in [0, 4.5]$ 区间上，分为10段；(b) 在 $x \in [2.25, 2.7]$ 区间上，分为10段。

$$\begin{aligned} & g_k^{\text{RB}}(x, w) \leq 0 \quad \forall k \in \mathbf{K} \\ & g_k^{\text{RB}}(x, w) = H'_k x + A'_k w + d'_k \quad \forall k \in \mathbf{K} \\ & 0 \leq x^L \leq x \leq x^U \\ & x \in \mathbb{R}^x, y \in [0, 1]^b, w \in \mathbb{R}^{lw} \end{aligned}$$

OBBT为了减少计算时间，在并行框架下求解所有涉及非线性项的变量的LP模型。因此，边界会弱于顺序求解。由于要求解的例子非常多，这些例子在不同块中被求解。这些块由并行解决的最大问题数目定义。求解完一个块后，更新相应的边界，然后求解下一个块。图7给出了OBBT方法的流程图。注意每个变量OBBT只用一次。

## 6. 全局优化算法

下面针对一个最小化问题给出了本文提出的全局优化算法步骤，图8是相应的流程图。注意该算法可以应用于任何非线性项为双线性或二次形式的MINLP问题。

(1) 算法初始化。定义要使用的分区数 $\{NP_1, NP_2, \dots, NP_{\text{last}}\}$ 和集合 $NP = NP_1$ 。设置下界 $LB = -\infty$ ，上界 $UB = +\infty$ ，迭代计数器 $IT_{\text{total}} = 1$ ，与代数相同的分区 $IT_{\text{sameNP}} = 1$ ，最大迭代次数 $IT_{\text{total}}^{\text{max}}$ ，与最大代数相同的分区 $IT_{\text{sameNP}}^{\text{max}}$ ，最大计算时间 $TIME_{\text{total}}^{\text{max}}$ ，最小相对容差 $\varepsilon$ 。

(2) 下界计算。使用CPLEX求解器求解**PR**模型MILP问题，启用求解器的并行选项和解库选项。通过CPLEX求解器计算出最优可能解，若值大于当前 $LB$ ，则更新 $LB$ 。

(3) 上界计算。使用存储在CPLEX解库中的解作为模型**PF**的NLP问题初值。并行求解模型**PF**实例并使用局部非线性求解器。若计算结果中任一解为可行解且比

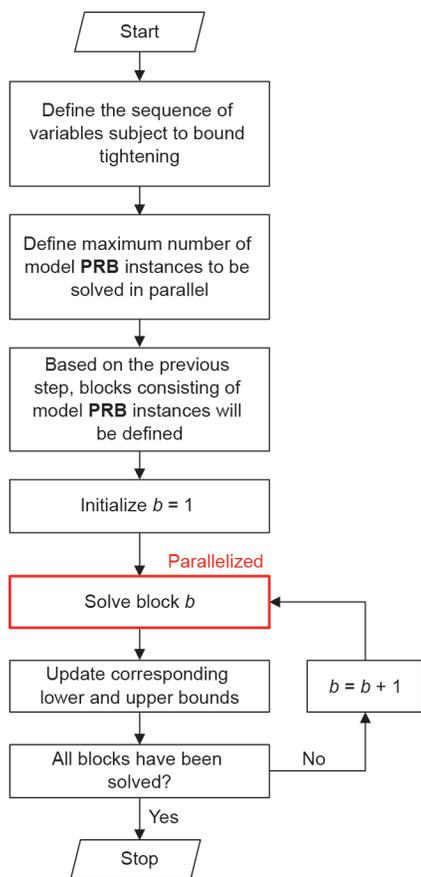


图7. OBBT方法流程图。

当前 $UB$ 有更小的目标函数值, 则更新 $UB$ 。

(4) 更新最优上下界距离。这一步中采用的公式为:  
 $OptGap = (UB - LB)/LB \times 100$ 。

(5) 检查终止条件。当 $OptGap \leq \varepsilon$ ,  $IT_{total} = IT_{total}^{max}$ , 计算时间等于或大于 $TIME_{total}^{max}$ , 区间数达到 $NP_{last}$ 时, 算法停止, 否则转到步骤(6)。

(6) 当步骤(3)中上界无法继续提高, 或者当 $IT_{sameNP} = IT_{sameNP}^{max}$ , 转到步骤(7)。否则, 使用第5节中的OBBT方法减少涉及非线性项的变量的域; 设置 $IT_{total} = IT_{total} + 1$ ,  $IT_{sameNP} = IT_{sameNP} + 1$ , 然后转到步骤(2)。

(7) 将分区数增加到下一个指定值。设置 $IT_{total} = IT_{total} + 1$ , 然后转向步骤(2)。

虽然算法的主要部分都已经被提出(如PMCR、NMDT、OBBT), 本文的创新性体现在具体的实施方式上。具体表现在: ①CPLEX求解器的解库被用来存储PF模型的初值; ②PF模型的实例采用并行方式求解; ③OBBT应用于变量块和并行框架中; ④没有采用分支策略。

## 7. 案例研究

本文中的测试由Li和Karimi[3]中的案例4、8、12和

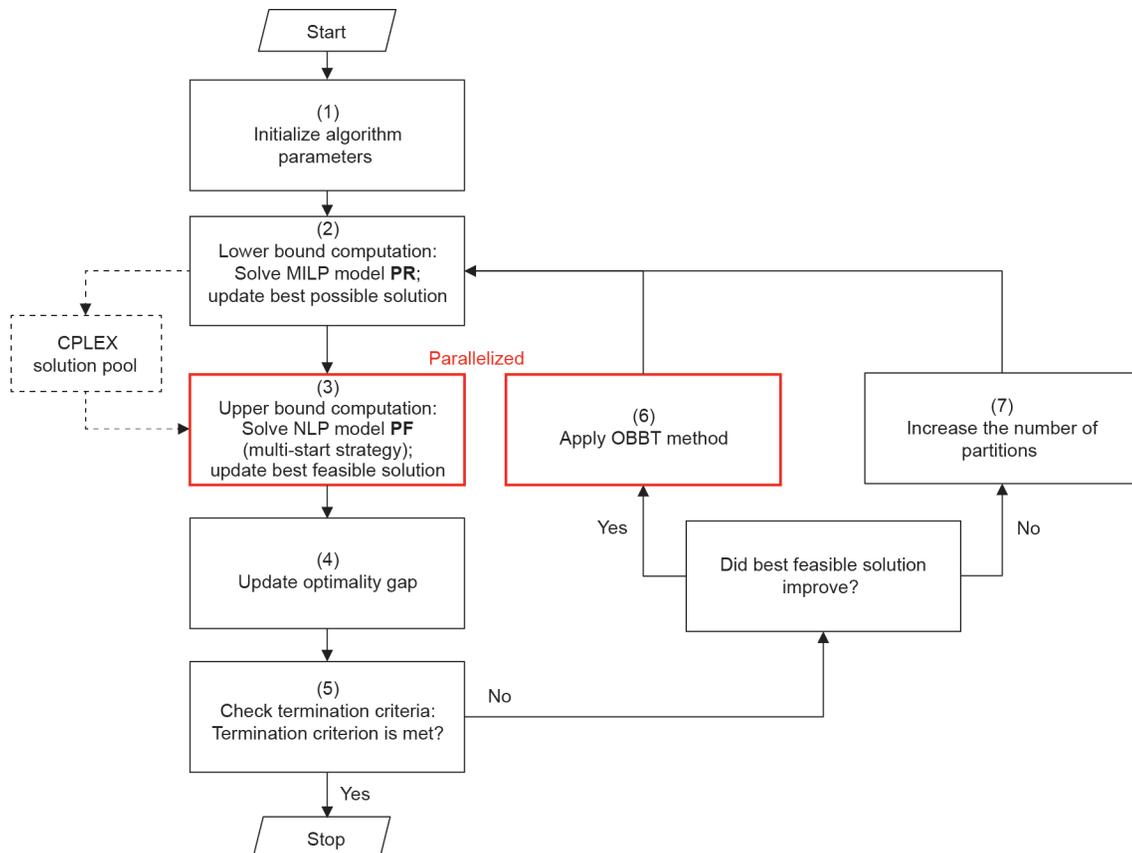


图8. 全局优化算法流程图。

14组成。本文中的不同之处在于ethyl RT-70模型被用来调合RON和MON性质(见第3节)而不是调合指标。RON指标的相关性参考了Li等[13], 见公式(20)和(21), 用这两个公式通过Li和Karimi[3]给出的调合指标来计算实际的RON。RBN代表研究法辛烷值的混合指数。MON的值在本文中给定, 相应的最小产品规格设定为零; 因此, MON的约束在最优解中不会被激活。参调组分的质量特性被认为是已知的(见第2节); 因此, 公式(20)和(21)仅用来在优化操作之前将参调组分的RBN值转化为RON值(即这两个公式在优化问题中并不出现)。

$$RBN = RON + 11.5 \quad 0 \leq RON \leq 85 \quad (20)$$

$$RBN = \exp(0.0135RON + 3.422042) \quad RON > 85 \quad (21)$$

表1对研究案例的调合系统进行了介绍。表2中给出了周期、周期长度、相关时隙和在该周期内可交付的订单。RON和MON值以及各自的规格在表3中给出。表4列出了不考虑最小调合和切换成本约束情况下模型P的规模统计。当考虑这两个约束时, 将四个等式约束加入模型中(最小化调合成本、最小化交付次数、最小化调

表1 研究案例概述

Example ID	Number of blenders	Number of orders	Number of products	Number of product tanks	Number of quality properties
4	1	15	4	11	9
8	2	20	4	11	9
12	2	35	5	11	9
14	3	45	5	11	9

表2 各案例参数

Example ID	Period	Duration (h)	Slots	Orders that can be delivered
4	1	100	1, 2	O1-O7, O12-O15
	2	92	3, 4	O8-O11
8	1	80	1, 2	O1-O7, O12-O19
	2	70	3, 4	O8
	3	42	5, 6	O8-O11, O20
12	1	50	1-3	O1-O7, O12, O13, O15, O19, O33
	2	50	4-6	O14-O18, O27, O28, O33
	3	50	7-9	O8, O21, O24, O29-O32, O34, O35
	4	42	10-12	O8-O11, O20, O22, O23, O25, O26
14	1	50	1-3	O1-O7, O12, O13, O15, O19, O26
	2	50	4-7	O14-O18, O26
	3	50	8-10	O8, O21, O24, O27-O31, O45
	4	42	11-13	O8-O11, O20, O22, O23, O25, O32-O44

表3 调合原料性质及产品指标

Property	Blend components									Product specifications [min, max]				
	C1	C2	C3	C4	C5	C6	C7	C8	C9	P1	P2	P3	P4	P5
RON	75	90.3	95.6	97.3	83	100	115	118	81	[95, 200]	[96, 200]	[94, 200]	[90, 200]	[98, 200]
MON	66	80.8	80.5	91.7	74	100	109	100	72	[0, 200]	[0, 200]	[0, 200]	[0, 200]	[0, 200]

表4 模型复杂性统计

Example ID	Number of equations	Number of variables	Number of binary variables	Number of bilinear terms
4	6 207	2 503	433	168
8	9 297	3 323	553	336
12	23 087	8 170	1 317	672
14	32 574	10 693	1 628	1 092

合次数、最小化在多功能罐中产品改变的次数)。注意, 调合系统的规模和相关调度模型从案例4到案例14逐渐增加。

## 8. 结果

本文计算所采用的计算机配置为Intel® Core™ i7-4710HQ CPU, 2.50 GHz, 12 GB RAM, 操作系统为Windows10(8-core)。全局优化算法在Python 2.7中实现。Python脚本生成GAMS(数学规划和优化的高级建模系统)文件和相关数学模型, 这些模型随后使用GAMS-Python API(应用程序界面)进行求解。本文采用CPLEX 12.6求解器计算PR模型和PRB模型, 采用CONOPT 3求解器计算PF模型。

全局优化算法的终止条件为: 最优上下界距离0.01%或者3600 s(1 h)。最大迭代次数不设限制, 相同分区数的迭代次数也不设限制。PR模型采用PMCR时分区数为{2, 4, 8, 16, 32}, 采用NMDT时为{10, 100, 1000}。

MILP问题(如PR模型)的终止条件为: 最优上下界距离0.01%或者600 s。CPLEX求解器的并行选项启动(确定性模式), 同时最大线程设为8。另外, 启动CPLEX求解器的解库, 最大解库容量为30个, 启动替代选项产生不同解决方案。因此, 使用GAMS并行计算方式, 每次迭代最多可以求解30个PF模型实例。使用CONOPT 3默认终止条件。

LP问题(PRB模型实例)的终止条件为: 最优或60 s。使用GAMS并行求解方式并行解决最多100个LP问题。

为了方便比较, 采用相同的全局最优算法终止条件使用全局商用求解器BARON 15.9[33] 和ANTIGONE 1.1[34]来求解初始P模型。

8.1节介绍了不考虑最小化调合成本和切换成本约束条件的结果; 8.2节给出了考虑上述条件的结果; 8.3节给出了与一个已发表的启发式模型进行对此的结果。

### 8.1. 不考虑最小调合成本和切换成本的约束

表5对比了采用本文提出的优化算法和采用商用求解器得到的结果。当采用分段McCormick松弛策略构建PR模型时, 将本文提出的方法简记为GO-PMCR; 当采用标准化多参数分解技术时, 简记为GO-NMDT。

ANTIGONE、BARON和GO-PMCR为案例4和案例8计算相同的解。GO-NMDT为案例4计算相同的解, 但是案例8的计算结果稍高。GO-PMCR计算出来的结果在所有案例中都比GO-NMDT和ANTIGONE好。本研究是由于GO-PMCR能够使用的分区数(2, 4, 8, 16, 32)比GO-NMDT(10, 100, 1000)要多, 因此, GO-PMCR能通过MILP松弛产生更多的可行解。

BARON在1 h内不能为案例12和案例14找到可行解。根据商用求解器产生的日志文件可以看出, BARON更依赖于分支策略, 而ANTIGONE更侧重于MILP松弛和边界收紧策略(如本文中提出的算法)。调度问题的可行整数解只能在分支定界树的深层节点中才能找到[43], 当二进制变量个数很大时分支定界树的规模将会非常大。

在案例4和案例8中, 使用本文的算法和商用求解器计算得到了相近的最优上下界间隔。对于案例12和案例14, BARON无法得到最优间隔(没有找到可行解), 而GO-PMCR得到了一个比GO-NMDT和ANTIGONE更小的间隔。

对于这四个案例, 商业求解器和本文所提算法都不能在1 h内获得满足收敛容差的解。表5中给出的时间是采用不同算法得到的最优解所需的时间。GO-PMCR和GO-NMDT比商用求解器所需的求解时间少。在案例4这种规模较小的问题中GO-NMDT方法比GO-PMCR方法快。注意, 对于分区数选择, 采用GO-PMCR时MILP松弛的规模增长速度比采用GO-NMDT快。因此, 采用GO-PMCR求解松弛MILP能够更快达到最优, 但是需要更多的迭代次数。根据三个因素(最优解、最优上下界间隔以及最优解求解的速度)的考虑, GO-PMCR性能最佳。图9给出了当采用PMCR和NMDT时在松弛调度

表5 优化结果总结

Example ID	Best solution found (1000 USD)				Optimality gap (%)				Time to best solution (s)			
	AN-TIGONE	BARON	GO-PMCR	GO-NMDT	AN-TIGONE	BARON	GO-PMCR	GO-NMDT	AN-TIGONE	BARON	GO-PMCR	GO-NMDT
4	4 633	4 633	4 633	4 633	6.90	6.37	6.31	6.37	2 350	930	616	120
8	8 203	8 203	8 203	8 223	9.53	9.53	9.19	9.49	1 708	3 273	714	1 391
12	16 650	NF	15 408	15 440	20.51	NA	14.00	14.22	1 631	3 600	1 411	1 438
14	21 360	NF	21 316	31 639	12.50	NA	12.31	40.92	3 600	3 600	2 911	2 904

NF = not found; NA = not available.

模型(即PR模型)求解的每步迭代时二进制变量的总数。这表明对于指定的分区值, PMCR在前4~5次迭代时比NMDT需要的二进制变量个数少。这是由于在这几步迭代过程中, 采用PMCR算法的分区数是小于8的, 而NMDT算法分区数量是从10开始的。注意曲线中较平的部分表示采用OBBT算法而不是增加分区数。

本文所提算法与BARON算法的区别如下:

(1) 对双线性项采用分段松弛方法替代标准McCormick包线;

(2) 以动态增加分区数目的方法而不是采用分支策略来收紧MILP的松弛。

这些功能足够求解本文所提的调度问题, 但笔者认为, 对于其他类型问题, 采用本文的算法不一定总会比BARON好。

本文提出的算法与ANTIGONE算法性能相近, 但是有以下不同:

(1) 采用NMDT作为分段松弛策略;

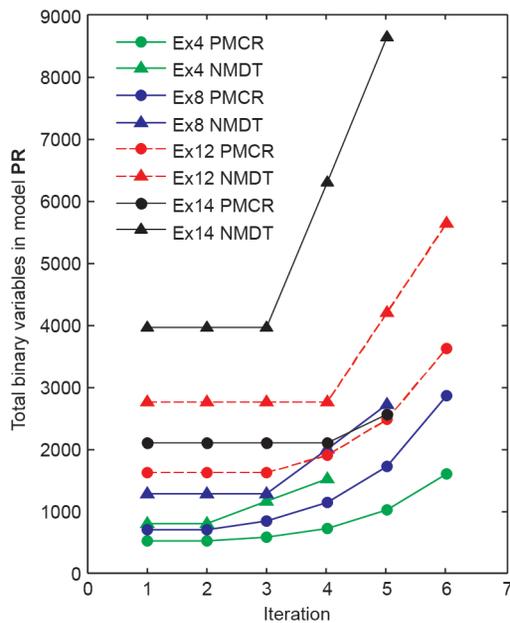


图9. 在算法各迭代周期PR模型中的二进制变量数目。

表6 优化结果总结

Example ID	Best solution found (1000 USD)				Optimality gap (%)				Time to best solution (s)			
	AN-TIGONE	BARON	GO-PMCR	GO-NMDT	AN-TIGONE	BARON	GO-PMCR	GO-NMDT	AN-TIGONE	BARON	GO-PMCR	GO-NMDT
4	4 633	4 633	4 633	4 633	0.01	0.01	0.01	0.01	26	296	30	14
8	8 207	8 204	8 206	8 204	0.05	0.02	0.04	0.02	557	1 218	103	140
12	23 590	NF	15 384	15 403	34.80	NA	0.01	0.13	3 333	3 600	2 674	742
14	23 520	NF	21 270	21 360	9.68	NA	0.13	0.55	1 636	3 600	2 574	2 845

NF = not found; NA = not available.

(2) 每步迭代中分区数量是如何变化的;

(3) 何时以及如何使用OBBT算法;

(4) 采用CPLEX解库存储MILP松弛问题的可行解, 并用这些可行解作为NLP问题的初值。

最后, ANTIGONE和BARON不仅可以解决双线性和二次项问题, 而且它们的应用可以提高其他数学算法的性能。

## 8.2. 考虑最小化调合成本和最小化切换成本带来的约束

在本节中, 采用本文提出的算法得到的计算结果与采用商用求解器得到的结果在表6中给出。与表5相比, 最显著的差异为在案例4和案例8中得到了更窄的最优上下界间隔和更短的求解时间。

对于案例4和案例8, 采用本文提出的算法和商用求解器都找到了相似的解。BARON在指定时间内依然没有找到案例12和案例14的可行解。GO-PMCR算法得到了比GO-NMDT算法更好的解, 而GO-NMDT求得了比ANTIGONE更好的解。注意, 添加边界条件会导致案例8的解的数量增加。由于即使考虑了最小化调合成本和最小化切换成本带来的约束, 8.1节中的解决方案依然可行, 这说明约束会对求解器产生影响。在用ANTIGONE求解案例12和案例14时也观察到了这样的情况。

关于最优上下界间隔, 结果与8.1节中的情况相似。对于案例4和案例8, 所有方法都得到了近似的最优上下界间隔。对于案例12和案例14, GO-PMCR比GO-NMDT和ANTIGONE求得了更小的最优上下界间隔。

商用求解器和本文提出的算法对案例4求解的结果都满足最小容差; BARON算法最慢。对于案例12和案例14, 采用GO-MPCR求得最优解的时间大于采用GO-NMDT, 但是GO-PMCR计算出的解比GO-NMDT更好。

总的来说, GO-PMCR再次取得了较好的效果。为了达到说明的目的, 通过PMCR算法求解得到的调合和交货调度结果分别在图10和图11中给出。

### 8.3. 与启发式模型的对比

本节将本文所提的算法与已发表的一些启发式模型[15,21]进行了对比。表7包含了根据这些启发式模型求得的最优解以及计算这些解所需的时间[15,21]。注意，

启发式模型不计算最优上下界间隔，因为它们旨在非常迅速地找到靠近最优解的次优解，并且它们不花时间去估计和精算全局最优解的值。这些启发式模型是针对本文工作中使用案例的特性制定的。这些方法通过将原始

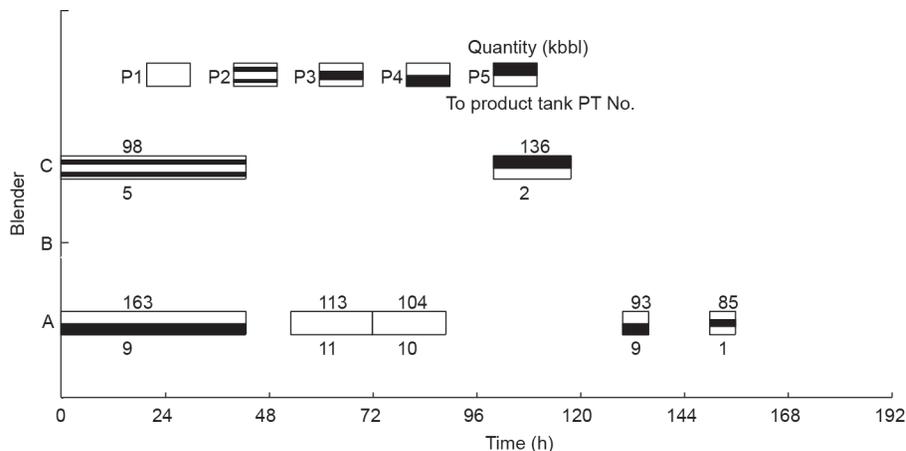


图10. 采用PMCR方法得到的案例14调合调度结果。1 kbbbl = 158.9873 m<sup>3</sup>。

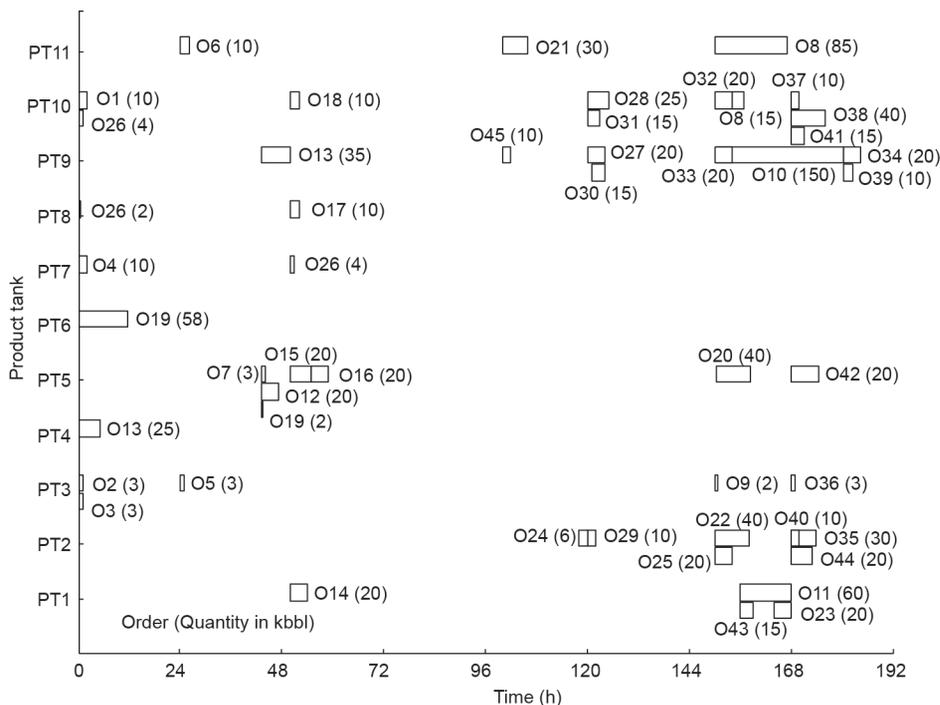


图11. 采用PMCR方法得到的案例14交货调度结果。

表7 与启发式方法比较

Example ID	Best solution found (1000 USD)					Time to best solution (s)			
	Castillo and Mahalec [15]	Cerdá et al. [21]	Cerdá et al. [21] adjusted values	GO-PMCR	GO-NMDT	Castillo and Mahalec [15]	Cerdá et al. [21]	GO-PMCR	GO-NMDT
4	4 633	4 613	4 633	4 633	4 633	3	0.4	30	14
8	8 203	8 163	8 223	8 206	8 204	6	7.5	103	140
12	15 403	15 342	15 442	15 384	15 403	17	31.0	2 674	742
14	21 263	21 181	21 301	21 270	21 360	24	21.0	2 574	2 845

问题分解为不同等级来重构最终的解，每个等级都有不同的准确率和复杂性。首先求解最简单级别的问题，然后在后续每一级中，将上一级问题的解赋给本级中最关键的变量，通过这种方法实现快速求解。

本文采用的调度模型目标函数与Castillo和Mahalec[15]提出的目标函数相同。这个目标函数会惩罚每一次独立调合批次，即使两次连续的调合批次调合的是同一种产品。另外，Cerdá等[21]的研究并没有添加对独立调合批次数目的惩罚项，而是仅对调合器中产品的切换进行惩罚。本文给出了对Cerdá等[21]的结果的修正值，即对独立调合批次进行惩罚。

每种方法对案例4都找到了相同的解。总体来说，对于其他几个案例，所有方法计算出的结果都很相似。案例8、12、14中，采用Cerdá等[21]的算法求出的结果成本最高，这是由于该方法没有对独立调合批次施加惩罚。如果Cerdá等[21]的算法中采用与本文中相同的目标函数，那么采用他们的方法应该会得到与其他方法近似的解。

由于启发式方法不像全局优化算法那样使用尽可能多的步数，因此启发式方法的求解时间显然要比本文提出的全局优化算法短。本文提出的全局优化算法与所选的两种启发式算法相比不能很快找到相同质量的解。在每一步迭代中为了找到可行解，本文提出的方法首先需要求解一个MILP问题(即PR模型)。这一步是最耗时的一步，因此会降低求解新的可行解的速度。另外，在算法开始阶段如果分区数较少则会使MILP松弛较差，产生的结果作为NLP问题的初值会大大偏离全局最优解。

这些结果表明，还需要改进相应的步骤来计算可行解，或者将本文提出的算法与启发式算法进行集成。

## 9. 结论

本文提出了一个可以解决带有双线性项和二次项形式的MINLP问题的算法。该算法采用PMCR或NMDT算法对原始问题进行松弛，然后对重构的MILP问题进行求解，计算出全局最优解的估计值。这两种方法通过引入额外的二进制变量和连续变量，将双线性项离散为几个段。为了改进全局最优解的估计值，在算法进行过程中分段数逐渐增加。

为了避免分段数变大导致模型规模快速扩大，引入了OBBT方法。如果分段数保持不变而变量的范围变小，

松弛的MILP问题将会更接近原始问题。本文采用并行框架用OBBT方法同时求解多个LP问题。

采用分支定界方法求解MILP问题时，CPLEX求解器的解库存储了多个不同可行解。这些可行解将作为初值赋给非线性求解器来求解原始问题。这一步也是以并行方式进行的。

本文提出的算法能用来解决考虑了分配问题和关键操作约束的汽油调合调度问题。这里采用连续时间MINLP调度模型[15]，其中，ethyl RT-70模型用来计算调合辛烷值。

本文提出的方法与两个商用求解器和两个启发式模型进行了对比。算法性能的评价指标为：得到的最优解、相应的最优上下界间隔、得到最优解所需的时间。本文提出的算法使用PMCR比NMDT能取得更好的效果。在大样本案例中，带有PMCR或NMDT的算法性能都好于全局求解器。结果表明，对算法的进一步研究将非常有前景。与全局算法相比，本文选取的两种启发式算法能在较短的求解时间内得到较好的结果。这表明可行解的求解步骤仍有待改进。

本文通过在两种情况下求解调度模型测试了算法的性能：①不考虑最小调合成本和切换成本的约束；②考虑上述约束。不考虑约束的情况较难求解，这种情况代表了一类模型，这类模型不需要尽量减少搜索空间。给调合成本的下界施加一个较紧的约束，同时给切换成本的下界加一个较紧的约束，能够使算法的搜索空间减小并使得算法性能提高。文中的结果表明对原始问题的松弛还不够紧。在未来的工作中，需要考虑推导和增加冗余约束和对称性缺失约束；对二次项、三次项和更高阶项进行不同的松弛方案测试(如外部近似)；改进边界收缩方式以提高求解速度。

## Acknowledgements

Support by Ontario Research Foundation, McMaster Advanced Control Consortium, and Fundação para a Ciência e Tecnologia (Investigador FCT 2013 program and project UID/MAT/04561/2013) is gratefully acknowledged.

## Compliance with ethics guidelines

Pedro A. Castillo Castillo, Pedro M. Castro, and Vladimir Mahalec declare that they have no conflict of interest or financial conflicts to disclose.

## Nomenclature

### Sets and subscripts

$\mathbf{BL} = \{bl\}$	Blenders
$\mathbf{E} = \{e\}$	Quality properties
$\mathbf{I} = \{i\}$	Blend components and corresponding storage tanks
$\mathbf{N1} = \{n\}$	Time slots
$\mathbf{QN} = \{(\theta, n)\}$	Time slot $n$ is associated with the period with quality profile $\theta$

### Parameters

$a_1, a_2, \dots, a_6$	Coefficients for the ethyl RT-70 model
$Q_{bc}(i, e, \theta)$	Value of quality property $e$ of blend component $i$ during quality profile $\theta$
$sens(i, \theta)$	Octane number sensitivity, i.e., octane difference RON – MON for blend component $i$ during quality profile $\theta$

### Continuous variables

$Ar_{avg}(bl, n)$	Volumetric average of the aromatics content of the processed material by blender $bl$ during slot $n$
$Ar_{avg}^{sq}(bl, n)$	Volumetric average of the squared value of the aromatics content of the processed material by blender $bl$ during slot $n$
$Ar2_{avg}(bl, n)$	Squared value of $Ar_{avg}(bl, n)$
$Ar2_{avg}^{sq}(bl, n)$	Squared value of $Ar_{avg}^{sq}(bl, n)$
$Ar3_{avg}(bl, n)$	Product of $Ar_{avg}^{sq}(bl, n)$ and $Ar2_{avg}(bl, n)$
$Ar4_{avg}(bl, n)$	Squared value of $Ar2_{avg}(bl, n)$
$Ol_{avg}(bl, n)$	Volumetric average of the olefins content of the processed material by blender $bl$ during slot $n$
$Ol_{avg}^{sq}(bl, n)$	Volumetric average of the squared value of the olefins content of the processed material by blender $bl$ during slot $n$
$Ol2_{avg}(bl, n)$	Squared value of $Ol_{avg}(bl, n)$
$Q_{pr}(bl, e, n)$	Value of quality property $e$ of the processed material by blender $bl$ during slot $n$
$r(i, bl, n)$	Volume fraction of blend component $i$ going into blender $bl$ during slot $n$
$r_{avg}^{MON}(bl, n)$	Volumetric average of the motor octane number of the processed material by blender $bl$ during slot $n$
$r_{avg}^{RON}(bl, n)$	Volumetric average of the research octane number of the processed material by blender $bl$ during slot $n$
$rs_{avg}^{MON}(bl, n)$	Product of $r_{avg}^{MON}(bl, n)$ and $sens_{avg}(bl, n)$
$rs_{avg}^{RON}(bl, n)$	Product of $r_{avg}^{RON}(bl, n)$ and $sens_{avg}(bl, n)$
$sens_{avg}(bl, n)$	Volumetric average of the octane number sensitivity of the processed material by blender $bl$ during slot $n$
$sens_{avg}^{MON}(bl, n)$	Volumetric average of the octane number sensitivity times the motor octane number
$sens_{avg}^{RON}(bl, n)$	Volumetric average of the octane number sen-

$V_{blend}(bl, n)$	Volume being processed by blender $bl$ during slot $n$
$V_{comp}(i, bl, n)$	Volume of blend component $i$ transferred to blender $bl$ during slot $n$

### References

- [1] Harjunkoski I, Maravelias CT, Bongers P, Castro PM, Engell S, Grossmann IE, et al. Scope for industrial applications of production scheduling models and solution methods. *Comput Chem Eng* 2014;62:161–93.
- [2] Méndez CA, Grossmann IE, Harjunkoski I, Kaboré P. A simultaneous optimization approach for off-line blending and scheduling of oil-refinery operations. *Comput Chem Eng* 2006;30(4):614–34.
- [3] Li J, Karimi I. Scheduling gasoline blending operations from recipe determination to shipping using unit slots. *Ind Eng Chem Res* 2011;50(15):9156–74.
- [4] Li J, Xiao X, Floudas CA. Integrated gasoline blending and order delivery operations: Part I. Short-term scheduling and global optimization for single and multi-period operations. *AIChE J* 2016;62(6):2043–70.
- [5] Singh A, Forbes JF, Vermeer PJ, Woo SS. Model-based real-time optimization of automotive gasoline blending operations. *J Process Contr* 2000;10(1):43–58.
- [6] Joly M, Pinto JM. Mixed-integer programming techniques for the scheduling of fuel oil and asphalt production. *Chem Eng Res Des* 2003;81(4):427–47.
- [7] Floudas CA, Lin X. Continuous-time versus discrete-time approaches for scheduling of chemical processes: A review. *Comput Chem Eng* 2004;28(11):2109–29.
- [8] Sundaramoorthy A, Maravelias CT. Computational study of network-based mixed-integer programming approaches for chemical production scheduling. *Ind Eng Chem Res* 2011;50(9):5023–40.
- [9] Maravelias CT. General framework and modeling approach classification for chemical production scheduling. *AIChE J* 2012;58(6):1812–28.
- [10] Jia Z, Ierapetritou M. Mixed-integer linear programming model for gasoline blending and distribution scheduling. *Ind Eng Chem Res* 2003;42(4):825–35.
- [11] Jia Z, Ierapetritou M. Efficient short-term scheduling of refinery operations based on a continuous time formulation. *Comput Chem Eng* 2004;28(6–7):1001–19.
- [12] Glismann K, Gruhn G. Short-term scheduling and recipe optimization of blending processes. *Comput Chem Eng* 2001;25(4–6):627–34.
- [13] Li J, Karimi I, Srinivasan R. Recipe determination and scheduling of gasoline blending operations. *AIChE J* 2010;56(2):441–65.
- [14] Castillo PAC, Mahalec V. Inventory pinch based, multiscale models for integrated planning and scheduling—Part II: Gasoline blend scheduling. *AIChE J* 2014;60(7):2475–97.
- [15] Castillo PAC, Mahalec V. Inventory pinch gasoline blend scheduling algorithm combining discrete- and continuous-time models. *Comput Chem Eng* 2016;84:611–26.
- [16] Castillo PAC, Mahalec V. Improved continuous-time model for gasoline blend scheduling. *Comput Chem Eng* 2016;84:627–46.
- [17] Lotero I, Trespalacios F, Grossmann IE, Papageorgiou DJ, Cheon MS. An MILP-MINLP decomposition method for the global optimization of a source based model of the multiperiod blending problem. *Comput Chem Eng* 2016;87:13–35.
- [18] Castro PM. New MINLP formulation for the multiperiod pooling problem. *AIChE J* 2015;61(11):3728–38.
- [19] Kolodziej SP, Grossmann IE, Furman KC, Sawaya NW. A discretization-based approach for the optimization of the multiperiod blend scheduling problem. *Comput Chem Eng* 2013;53:122–42.
- [20] Cerdá J, Pautasso PC, Cafaro DC. A cost-effective model for the gasoline blend optimization problem. *AIChE J* 2016;62(9):3002–19.
- [21] Cerdá J, Pautasso PC, Cafaro DC. Optimizing gasoline recipes and blending operations using nonlinear blend models. *Ind Eng Chem Res* 2016;55(28):7782–800.
- [22] Tawarmalani M, Sahinidis NV. A polyhedral branch-and-cut approach to global optimization. *Math Program* 2005;103(2):225–49.
- [23] Misener R, Floudas CA. ANTIGONE: Algorithms for continuous/integer global optimization of nonlinear equations. *J Glob Optim* 2014;59(2):503–26.
- [24] Boland N, Kalinowski T, Rigtering F. New multi-commodity flow formulations for the pooling problem. *J Glob Optim* 2016;66(4):669–710.
- [25] Sherali HD, Alameddine A. A new reformulation-linearization technique for bilinear programming problems. *J Glob Optim* 1992;2(4):379–410.
- [26] Ryoo HS, Sahinidis NV. A branch-and-reduce approach for global optimization. *J Glob Optim* 1996;8(2):107–38.
- [27] Smith EMB, Pantelides CC. Global optimization of nonconvex MINLPs. *Comput Chem Eng* 1997;21(Suppl):S791–6.
- [28] Belotti P, Lee J, Liberti L, Margot F, Wächter A. Branching and bounds tightening techniques for non-convex MINLP. *Optim Methods Softw* 2009;24(4–5):597–634.
- [29] Achterberg T. SCIP: Solving constraint integer programs. *Math Program Comput* 2009;1(1):1–41.

- [30] Castro PM. Spatial branch-and-bound algorithm for MIQCPs featuring multiparametric disaggregation. *Optim Methods Softw*. Epub 2016 Dec 13.
- [31] Castillo PC, Castro PM, Mahalec V. Global optimization algorithm for large-scale refinery planning models with bilinear terms. *Ind Eng Chem Res* 2017;56(2):530–48.
- [32] McCormick GP. Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems. *Math Program* 1976;10(1):147–75.
- [33] Karuppiah R, Grossmann IE. Global optimization for the synthesis of integrated water systems in chemical processes. *Comput Chem Eng* 2006;30(4):650–73.
- [34] Castro PM. Tightening piecewise McCormick relaxations for bilinear problems. *Comput Chem Eng* 2015;72:300–11.
- [35] Misener R, Thompson JP, Floudas CA. APOGEE: Global optimization of standard, generalized, and extended pooling problems via linear and logarithmic partitioning schemes. *Comput Chem Eng* 2011;35(5):876–92.
- [36] Kolodziej S, Castro PM, Grossmann IE. Global optimization of bilinear programs with a multiparametric disaggregation technique. *J Glob Optim* 2013;57(4): 1039–63.
- [37] Castro PM. Normalized multiparametric disaggregation: An efficient relaxation for mixed-integer bilinear problems. *J Glob Optim* 2016;64(4):765–84.
- [38] Castro PM, Grossmann IE. Global optimal scheduling of crude oil blending operations with RTN continuous-time and multiparametric disaggregation. *Ind Eng Chem Res* 2014;53(39):15127–45.
- [39] Castro PM. Source-based discrete and continuous-time formulations for the crude oil pooling problem. *Comput Chem Eng* 2016;93:382–401.
- [40] Castillo PAC, Mahalec V, Kelly JD. Inventory pinch algorithm for gasoline blend planning. *AIChE J* 2013;59(10):3748–66.
- [41] Healy WC, Maassen CW, Peterson RT. A new approach to blending octanes. In: *Proceedings of the 24th Midyear Meeting of American Petroleum Institute's Division of Refining*; 1959 May 27; New York, US; 1959. p. 132–136.
- [42] Castro PM, Grossmann IE. Optimality-based bound contraction with multiparametric disaggregation for the global optimization of mixed-integer bilinear problems. *J Glob Optim* 2014;59(2):277–306.
- [43] Kallrath J. Planning and scheduling in the process industry. *OR Spectrum* 2002;24(1):219–250.