Variable Bound Tightening and Valid Constraints for Multiperiod Blending

Yifu Chen,[a] Christos T. Maravelias[b,c,*]

a. Department of Chemical and Biological Engineering, University of Wisconsin-Madison

1415 Engineering Dr., Madison, WI 53706, USA

b. Department of Chemical and Biological Engineering, Princeton University

Princeton, NJ 08544, USA

c. Andlinger Center for Energy and the Environment, Princeton University

86 Olden Street, Princeton, NJ 08544

*Corresponding author

maravelias@princeton.edu

Abstract

Multiperiod blending has a number of important applications in a range of industrial sectors. It is typically formulated as a nonconvex Mixed Integer Nonlinear Program (MINLP), which involves binary variables and bilinear terms. In this study, we first propose a reformulation of the constraints involving bilinear terms using lifting. We introduce a method for calculating tight bounds on the lifted variables calculated by aggregating multiple constraints. We propose valid constraints derived from Reformulation-Linearization Technique (RLT) that utilize the bounds on the lifted variables to further tighten the formulation. Computational results indicate our method can substantially reduce the solution time and optimality gap.

Keywords: Preprocessing, Reformulation-Linearization Technique, Variable Lifting, Bilinear Terms

List of Symbols

Indices/Sets

$i \in \mathbf{I}$: Inputs (Streams)

$j \in \mathbf{J}$: Blenders

$k \in \mathbf{K}$: Products

$l \in \mathbf{L}$: Properties

$t \in \mathbf{T}$: Time points: $\{0,1, \dots, |\mathbf{T}|\}$/time periods: $\{1,2, \dots, |\mathbf{T}|\}$

Subset

$\mathbf{L}_{ik}$: Properties for product *k* whose specification is violated by stream i

$\mathbf{I}_{lk}$: Streams that violate the specification for property *l* for product *k*

Parameters: problem data

α_{jk}^{F}: Fixed cost for flow from blender *j* to product *k*

α_{i}^{V}: Variable cost for stream i

β_k: Price of product *k*

γ_j: Inventory capacity of blender *j*

δ_{jk}: Upper bound on flow between blender *j* and product *k*

ξ_i: Availability of stream i

π_{il} : Value of property l for stream i

π_{kl}^{U} : Upper bounding specification for property l for product k

ω_k : Maximum demand for product k

Parameters: calculated during preprocessing

$\hat{\gamma}_{ijkl}$: Tightened bound on inventory of stream i in blender j when it is feeding product k derived from property l

$\bar{\gamma}_{ijk}$: Tightened bound on inventory of stream i in blender *j* when it is feeding product *k*

μ_{ikl}: Violation of specification for property l for product *k* from stream i

μ_l^*: Value of property l of the "best" stream for property l

μ_l^+: Value of property l of the "second best" stream for property l

Variables: nonnegative continuous

F_{ijt}: Flow of stream i to blender *j* at time point t

I_{ijt}: Inventory of stream i in blender *j* during time period t

R_{jkt}: Split fraction for inventory in blender *j* to product *k* at time point *t*

$\hat{F}_{ijkt}$: Flow of stream i from blender *j* to product *k* at time point *t*

U_{ijkt}: Inventory of stream i in blender *j* during time period t when not feeding product *k*

V_{ijkt}: Inventory of stream i in blender *j* during time period t when feeding product *k*

Variables: binary

X_{jkt}: = 1 when blender *j* feeds product *k* at time point *t*

Y_{jt}: = 1 when blender *j* feeds products at time point *t*

1. Introduction

Planning and scheduling for blending processes over multiple time periods have received considerable attention. Since the introduction of the pooling problem (Haverly 1978), blending processes have been studied extensively, with the bilinearities required to model the blending process receiving considerable attention (Wicaksono and Karimi 2008; Gounaris, Misener, and Floudas 2009; Misener and Floudas 2012; Kolodziej, Castro, and Grossmann 2013; Gupte et al. 2017; Chen and Maravelias 2020). Previous works have been focused on both crude oil blending (Reddy, Karimi, and Srinivasan 2004; Li et al. 2007; Li, Misener, and Floudas 2012; Castro and Grossmann 2014; Castro 2016) as well as final product blending (Li and Karimi 2011; Kolodziej et al. 2013; Castillo and Mahalec 2014; Neiro, Murata, and Pinto 2014; Li, Xiao, and Floudas 2016). Multiperiod blending also arises in wastewater management (Bagajewicz 2000; Jeżowski 2010) and mining (Blom et al. 2014; Blom, Pearce, and Stuckey 2016; Boland et al. 2016). In the multiperiod setting, binary variables are introduced to enforce additional operating rules, which lead to a nonconvex Mixed Integer Nonlinear Program (MINLP). Such problems are important in terms of the potential economic benefits that can be achieved (DeWitt et al. 1989; Kelly and Mann 2003).

Global optimization of nonconvex MINLPs is performed using branch-and-bound algorithms which involve solving convex relaxations of the original problem. The tightness of the convex relaxation strongly depends on variable bounds. Various bounds tightening methods have been proposed (Belotti et al. 2009; Puranik and Sahinidis 2017), including, for example, methods based on reduced cost (Ryoo and Sahinidis 1996), which utilizes the optimal solution to the relaxed problem. Bounds tightening techniques that do not require such information have also been proposed. A well-known technique is Optimality Based Bound Tightening (OBBT) which typically relies on solving linear programs (LP) (Quesada and Grossmann 1995; Maranas and Floudas 1997; Shectman and Sahinidis 1998; Smith and Pantelides 1999). OBBT can be computationally expensive, and methods aim to improve its efficiency have been studied (Gleixner et al. 2017). Feasibility Based Bound Tightening (FBBT), which considers a single constraint at a time and utilizes interval arithmetic to infer variable bounds, has been employed in solving both mixed integer linear program (MILP) (Savelsbergh 1994; Achterberg et al. 2020) and MINLP (Achterberg 2007). FBBT has received considerable attention in both mathematical programming and artificial intelligence communities (Street 1989). Though computationally inexpensive, FBBT is known to be less effective compared to OBBT in terms of the tightness of the bounds found.

Tightening methods that utilize information from multiple constraints at a time have also been studied. For example, Achterberg et al. (Achterberg et al. 2020) studied presolve methods for MILP that consider multiple constraints simultaneously. Specifically, for variable bounds tightening purpose, their methods are based on special block structure in the problem matrix. Domes and

Neumaier (Domes and Neumaier 2016) proposed constraint aggregation method for rigorous global optimization that utilizes information from local solutions. Belotti (Belotti 2013) proposed a procedure that infers variable bounds using a pair of constraints. Aggregating multiple constraints can lead to tighter variable bounds compared to FBBT, while it is computationally inexpensive compared to OBBT. However, which constraints to be aggregated and their weights require further investigation.

In this paper, we propose a bound tightening method for multiperiod blending problem based on aggregating multiple constraints. The selection of constraints and weights assignment are based on the understanding of the physical system we model. Our method works on reformulations of the multiperiod blending problem that contain certain structure; however, such structure may be found in a range of models in different fields.

This paper is structured as follows. In section 2, we present background material, including the problem statement and two formulations. In section 3, we introduce reformulations of the two formulations and a preprocessing method for variable bounds tightening. In section 4, we propose valid constraints derived from Reformulation-Linearization Technique (RLT) utilizing the bounds obtained from preprocessing. We demonstrate the effectiveness of our method in section 5.

2. Background

We present the problem statement and two formulations for multiperiod blending problem. Throughout the paper we use Roman lowercase italic letters for indices, Roman uppercase bold letters for sets, Greek lowercase letters for parameters, and Roman uppercase italics for variables.

2.1. Problem statement

We use a discrete uniform time representation, where time point *t* is at the end of time period *t*.

The problem we study is defined in terms of the following sets:

$i \in \mathbf{I}$: Inputs (streams)

$j \in \mathbf{J}$: Blenders

$k \in \mathbf{K}$: Products

$l \in \mathbf{L}$: Properties

$t \in \mathbf{T}$: Time points: $\{0,1,\dots,|\mathbf{T}|\}$/time periods: $\{1,2,\dots,|\mathbf{T}|\}$

It can be stated as follows:

Given are:

α^{F}_{jk}: Fixed cost for flow from blender *j* to product *k*

α^{V}_{i}: Variable cost for stream *i*

β_k: Price of product *k*

γ_j: Inventory capacity of blender j

δ_{jk}: Upper bound on flow between blender j and product k

ξ_i: Availability of stream i

π_{il} : Value of property l for stream i

π_{kl}^{U} : Upper bounding specification for property l for product k

ω_k : Maximum demand for product k

The flow from blender to product must satisfy the corresponding specifications, and blender feeding and withdrawing cannot occur simultaneously. We aim to find a blend schedule that leads to the highest profit. We assume that all product properties are the average of the properties of the streams blended weighted by volume fraction. Finally, we assume no flow between blenders in this study.

2.2. Source-based formulation

We first present a source-based formulation inspired from the literature (Lotero et al. 2016). We define the following nonnegative continuous variables:

F_{ijt}: Flow of stream i to blender j at time point t

I_{ijt}: Inventory of stream i in blender j during time period t

R_{jkt}: Split fraction for inventory in blender j to product k at time point t

$\hat{F}_{ijkt}$: Flow of stream i from blender j to product k at time point t

We also define the following binary variable:

X_{jkt}: = 1 when blender j feeds product k at time point t

We illustrate the variables in the source-based formulation in Figure 1.

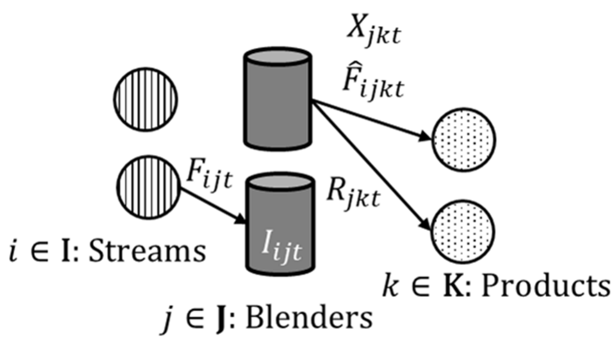

Figure 1. Illustrative graph for sets and variables in the source-based formulation.

We have the following constraints:

Stream availability:

$$\sum_{j \in \mathbf{J}} \sum_{t \in \mathbf{T}} F_{ijt} \leq \xi_i, \qquad i \in \mathbf{I} \tag{1}$$

Blender capacity:

$$\sum\nolimits_{i\in\mathbf{I}} I_{ijt} \le \gamma_j, \qquad j \in \mathbf{J}, t \in \mathbf{T} \tag{2}$$

Flow variable upper bound:

$$\sum\nolimits_{i\in\mathbf{I}} \hat{F}_{ijkt} \le \delta_{jk} X_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{3}$$

We note that $\sum_{i\in\mathbf{I}} \hat{F}_{ijkt}$ is also upper bounded by $\gamma_j X_{jkt}$; however, in practice we typically have $\delta_{jk} \le \gamma_j$.

Maximum product demand:

$$\sum\nolimits_{i\in\mathbf{I}} \sum\nolimits_{j\in\mathbf{J}} \sum\nolimits_{t\in\mathbf{T}} \hat{F}_{ijkt} \le \omega_k, \qquad k \in \mathbf{K} \tag{4}$$

Nonlinear constraint for inventory splitting:

$$\hat{F}_{ijkt} = I_{ijt} R_{jkt}, \quad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{5}$$

For split fraction we have:

$$\sum\nolimits_{k\in\mathbf{K}} R_{jkt} \le 1, \qquad j \in \mathbf{J}, t \in \mathbf{T} \tag{6}$$

Operating logic:

$$R_{jkt} \le X_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{7}$$

$$\sum\nolimits_{i\in\mathbf{I}} F_{ijt} \le \gamma_j \left(1 - X_{jkt}\right), \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{8}$$

Eqn. (3) and (7)– (8) enforce the operating rule that blender feeding and withdrawing cannot occur simultaneously.

The inventory balance is:

$$I_{ij,t+1} = I_{ijt} + F_{ijt} - \sum\nolimits_{k\in\mathbf{K}} \hat{F}_{ijkt}, \ \ i \in \mathbf{I}, j \in \mathbf{J}, t \in \mathbf{T} \tag{9}$$

The specification for flow from blenders to products:

$$\sum\nolimits_{i\in\mathbf{I}} \pi_{il} \hat{F}_{ijkt} \le \pi_{kl}^{\mathrm{U}} \sum\nolimits_{i\in\mathbf{I}} \hat{F}_{ijkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{10}$$

while the inventory specification is:

$$\sum\nolimits_{i\in\mathbf{I}} \pi_{il} I_{ijt} \le \pi_{kl}^{\mathrm{U}} \sum\nolimits_{i\in\mathbf{I}} I_{ijt} + \gamma_j \pi_{kl}^{\mathrm{U}} \left(1 - X_{jkt}\right), \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{11}$$

We assume, without loss of generality, that we have only upper bounding specifications.

Tightening constraint for product flow:

$$\sum_{i\in\mathbf{I}} \hat{F}_{ijkt} \le \gamma_j R_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K} \tag{12}$$

Finally, we have the objective function:

$$\max \sum_{i\in\mathbf{I}} \sum_{j\in\mathbf{J}} \sum_{t\in\mathbf{T}} \left[\sum_{k\in\mathbf{K}} (\beta_k \hat{F}_{ijkt} - \alpha_{jk}^{\mathrm{F}} X_{jkt}) - \alpha_i^{\mathrm{V}} F_{ijt} \right] \tag{13}$$

Eqn. (1) – (13) comprise the source-based formulation, henceforth referred to as M^{SB}.

2.3. Proportion-based formulation

The multiperiod blending problem can also be reformulated using a proportion-based formulation. We define the following nonnegative continuous variable:

P_{ijt}: Proportion of stream i in the inventory of blender j at time point t

We have:

$$\sum_{i\in\mathbf{I}} P_{ijt} = 1, \qquad j \in \mathbf{J}, t \in \mathbf{T} \tag{14}$$

We also have the following nonlinear constraints:

$$I_{ijt} = P_{ijt} \sum_{i'\in\mathbf{I}} I_{i'jt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, t \in \mathbf{T} \tag{15}$$

$$\hat{F}_{ijkt} = P_{ijt} \sum_{i'\in\mathbf{I}} \hat{F}_{i'jkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{16}$$

Finally, similar to Eqn. (11), the proportion specification can be expressed as:

$$\sum_{i\in\mathbf{I}} \pi_{il} P_{ijt} \le \pi_{kl}^{\mathrm{U}} + (\max_i\{\pi_{il}\} - \pi_{kl}^{\mathrm{U}})(1 - X_{jkt}), \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{17}$$

Eqn. (1) - (4), (8) - (11), and (13) - (17) comprise the proportion-based formulation, henceforth referred to as M^{PB}.

Note that the proportion-based formulation is similar to the PQ-formulation (Tawarmalani and Sahinidis 2002) and the TP-formulation (Alfaki and Haugland 2013) for the pooling problem; and the source-based formulation is similar to a model proposed by Boland et al. (Boland, Kalinowski, and Rigterink 2016). To the best of our knowledge, there are no theoretical results regarding the relative tightness of the aforementioned three formulations. No formulation is known to be tighter of clearly faster than the other two.

3. Reformulation and preprocessing method

We first introduce a reformulation of M^{SB} using lifting, and a preprocessing method to calculate tight bounds and then present the reformulation and preprocessing method for model M^{PB}.

3.1. Reformulation of bilinear terms

Eqn. (5), an equality constraint with a bilinear term, is of particular interest.

We lift I_{ijt}, and partition it into nonnegative continuous variables U_{ijkt} and V_{ijkt}:

$$I_{ijt} = U_{ijkt} + V_{ijkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{18}$$

$$\sum\nolimits_{i \in \mathbf{I}} U_{ijkt} \leq \gamma_j (1 - X_{jkt}), \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{19}$$

$$\sum\nolimits_{i \in \mathbf{I}} V_{ijkt} \leq \gamma_j X_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{20}$$

where U_{ijkt} represents the inventory of stream i in blender j during time period t when there is no flow from blender j to product k ($X_{jkt} = 0$), and V_{ijkt} represents such inventory when $X_{jkt} = 1$.

Eqn. (5) now becomes:

$$\hat{F}_{ijkt} = V_{ijkt} \sum\nolimits_{i' \in \mathbf{I}} \hat{F}_{i'jkt}, \quad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{21}$$

and Eqn. (11) can be re-written as:

$$\sum\nolimits_{i \in \mathbf{I}} \pi_{il} V_{ijkt} \leq \pi_{kl}^{\mathrm{U}} \sum\nolimits_{i \in \mathbf{I}} V_{ijkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{22}$$

The reformulated model, with variables U_{ijkt} and V_{ijkt}, henceforth referred to as M^{UV}, consists of Eqn. (1) – (4), (6) – (10), and (12) – (18). In M^{UV}, the variables involved in a bilinear term are V_{ijkt} and R_{jkt}. We aim to tighten bounds on V_{ijkt}.

3.2. Preprocessing method for variable bounds tightening

A relaxation of Eqn. (20) is:

$$\sum\nolimits_{i \in \mathbf{I}} V_{ijkt} \leq \gamma_j, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{23}$$

The right hand side (RHS) parameter γ_j can be tightened. We first rewrite Eqn. (22) as:

$$\sum\nolimits_{i \in \mathbf{I}} (\pi_{il} - \pi_{kl}^{\mathrm{U}}) V_{ijkt} \leq 0, \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{24}$$

We define a parameter μ_{ikl} to represent the margin by which stream i violates the specification for property l for product k: $\mu_{ikl} = \pi_{il} - \pi_{kl}^{\mathrm{U}}$ (note that μ_{ikl} can be positive or negative). Eqn. (24) can thus be written as:

$$\sum_{i\in\mathbf{I}} \mu_{ikl} V_{ijkt} \le 0, \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{25}$$

We aim to calculate a tighter upper bound on V_{ijkt} using Eqn. (23) and (25). For simplicity, we drop indices *j, k*, and *t* for now, thus μ_{ikl} becomes $\mu_{il} = \pi_{il} - \pi_l^{\mathrm{U}}$. We consider the following:

$$\sum_{i\in\mathbf{I}} V_i \le \gamma \tag{26}$$

$$\sum_{i\in\mathbf{I}} \mu_{il} V_i \le 0, \qquad l \in \mathbf{L} \tag{27}$$

Eqn. (26) - (27), which contain $(1 + |\mathbf{L}|)$ constraints, will be used to find tight variable bounds on V_i for model $\mathrm{M^{SB}}$.

To illustrate the bound tightening procedure, we first define a parameter $\mu_l^* = \min_i\{\mu_{il}\}$ and a set function $b(l) = \arg\min_i\{\mu_{il}\}$ that returns the "best" stream for property l. It is possible that, for a property l, there are multiple streams with $\mu_{il} = \mu_l^*$ (i.e., multiple "best" streams). In that case, we consider $b(l)$ being the stream with the smallest index among all such streams. We assume $\mu_l^* < 0$ because (1) if $\mu_l^* > 0$ then $\mu_{il} > 0, \forall i$ and since $V_i \ge 0$, Eqn. (23) can be satisfied only if $V_i = 0, \forall i$; and (2) if $\mu_l^* = 0$, then Eqn. (23) can be satisfied only if $V_i = 0, \forall i\colon \mu_{il} \ne 0$).

We also define subset $\mathbf{L}_i = \{l\colon \mu_{il} > 0\}$, that is, the set of properties with specification violated by stream i. Similarly, we define subset $\mathbf{I}_l = \{i\colon \mu_{il} > 0\}$, that contains streams that violate the specification for property l.

To illustrate, we consider an illustrative example with $\mathbf{I} = \{1,2,3\}, \mathbf{L} = \{\mathrm{L1}, \mathrm{L2}\}$. Parameters $\pi_{il}, \pi_l^{\mathrm{U}}$ and μ_{il} calculated from them are given in Figure 2.

3.2.1. Bounds tightening using a pair of constraints

From Eqn. (26) it is clear that γ is a valid upper bound on V_i. To tighten such upper bound, we combine Eqn. (26) with one constraint in Eqn. (27). For V_i with positive coefficient in at least one constraint in Eqn. (27) (i.e., streams that violates at least one specification), bounds derived from such pairs of constraints will be tighter than γ.

To calculate bounds using aforementioned pairs of constraints, we first multiply all inequalities in Eqn. (27) by $-\frac{1}{\mu_l^*}$ (recall that $\mu_l^* < 0$) to obtain:

$$\sum_{i\in\mathbf{I}} \left(-\frac{\mu_{il}}{\mu_l^*}\right) V_i \le 0, \qquad l \in \mathbf{L}$$

Properties	π_{il} $i=1$	$i=2$	$i=3$	π_l^{U}
$l=$ L1	1	4	3	2
$l=$ L2	5	1	6	4

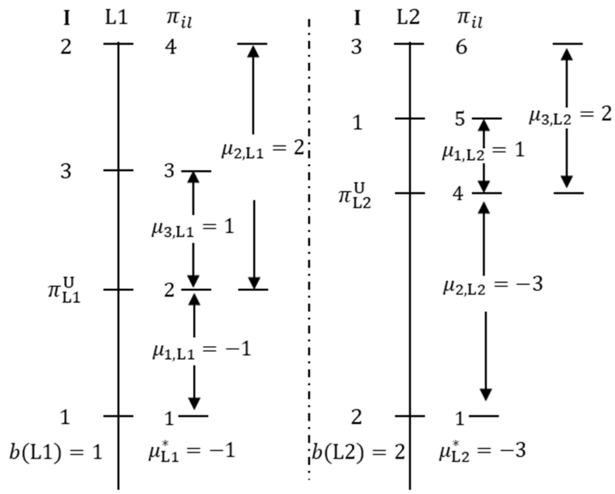

Figure 2. An illustrative example for parameters $\pi_{il}, \pi_l^{\mathrm{U}}, \mu_{il}$, and μ_l^*.

Next, we combine Eqn. (26) with a weight equal to 1, with each individual constraint above,

$$\sum\nolimits_{i\in\mathbf{I}}(1-\frac{\mu_{il}}{\mu_l^*})V_i \le \gamma, \qquad l\in\mathbf{L} \tag{28}$$

Each constraint in Eqn. (28) is obtained by combing a pair of constraints: Eqn. (26) and one constraint in Eqn. (27). Next, we derive bounds on V_i from Eqn. (28).

After using i' instead of i, we obtain:

$$\sum\nolimits_{i'\in\mathbf{I}}(1-\frac{\mu_{i'l}}{\mu_l^*})V_{i'} \le \gamma, \qquad l\in\mathbf{L}$$

For each $l\in\mathbf{L}$, we consider streams in the set $\mathbf{I}_l$, and isolate such streams, one at a time, from the summation in the left hand side (LHS):

$$(1-\frac{\mu_{il}}{\mu_l^*})V_i + \sum\nolimits_{i'\in\mathbf{I}\setminus\{i\}}(1-\frac{\mu_{i'l}}{\mu_l^*})V_{i'} \le \gamma, \qquad l\in\mathbf{L}, i\in\mathbf{I}_l$$

We examine the second term on the LHS of the above equation. By the definition of μ_l^* we have $\mu_{i'l}\ge\mu_l^*$. Thus, if $\mu_{i'l}<0$ then $\frac{\mu_{i'l}}{\mu_l^*}\in[0,1]$ and therefore $1-\frac{\mu_{i'l}}{\mu_l^*}\ge 0$; and if $\mu_{i'l}\ge 0$, then $-\frac{\mu_{i'l}}{\mu_l^*}\ge 0$ and therefore $1-\frac{\mu_{i'l}}{\mu_l^*}>1>0$. Given that $V_{i'}$ is nonnegative, we have $\sum_{i'\in\mathbf{I}\setminus\{i\}}(1-\frac{\mu_{i'l}}{\mu_l^*})V_{i'}\ge 0$. Thus, the following inequality, obtained by dropping the summation in the LHS of the above equation, is valid:

$$(1-\frac{\mu_{il}}{\mu_l^*})V_i \le \gamma, \qquad i\in\mathbf{I}, l\in\mathbf{L}_i \tag{29}$$

and since $1-\frac{\mu_{il}}{\mu_l^*}>0\ \forall i, l\in\mathbf{L}_i$ we have:

$$V_i \leq \gamma/(1-\frac{\mu_{il}}{\mu_l^*}), \qquad i \in \mathbf{I}, l \in \mathbf{L}_i$$

Or

$$V_i \leq \hat{\gamma}_{il} = -\frac{\mu_l^* \gamma}{\mu_{il}-\mu_l^*}, \qquad i \in \mathbf{I}, l \in \mathbf{L}_i \tag{30}$$

Note that $\hat{\gamma}_{il}$ is smaller than γ and serves as an upper bound on V_i derived from property l.

The physical interpretation of $\hat{\gamma}_{il}$ is as follows. Suppose we have to meet demand for volume γ for a product. Parameter $\hat{\gamma}_{il}$ represents the maximum volume of stream i that can be used towards volume γ based on property $l \in \mathbf{L}_i$. In other words, $\hat{\gamma}_{il}/\gamma$ is the maximum fraction of stream i that can be used for such product. This stream-specific volume, $\hat{\gamma}_{il}$, is derived by considering the binary mixture of streams i and $b(l)$ that satisfies the specification for property l exactly.

Once we calculate $\hat{\gamma}_{il}$ from Eqn. (30), the upper bound on V_i, denoted as $\bar{\gamma}_i$, is set to the smallest $\hat{\gamma}_{il}$, considering all properties that stream i violates (i.e., $\forall l \in \mathbf{L}_i$), $\bar{\gamma}_i = \min_{l\in\mathbf{L}_i}\{\hat{\gamma}_{il}\}$. For illustration purpose, we introduce a set function $m(i)$ that returns the property l from which $\bar{\gamma}_i$ is derived (i.e., $m(i) = \arg\min_{l\in\mathbf{L}_i}\{\hat{\gamma}_{il}\}$).

Consider the illustrative example shown in Figure 2 with $\gamma = 1$. Based on the calculated parameter μ_{il} shown in Figure 2, we have the following constraints for Eqn. (26) – (27):

$$V_1 + V_2 + V_3 \leq 1$$

$$-V_1 + 2V_2 + V_3 \leq 0$$

$$V_1 - 3V_2 + 2V_3 \leq 0$$

The calculations described above lead to bounds on V_i given in Table 1.

Table 1. Bounds calculated by aggregating pair of constraints

	$i = 1$	$i = 2$	$i = 3$
$\hat{\gamma}_{i,\text{L1}}$	–	1/3	1/2
$\hat{\gamma}_{i,\text{L2}}$	3/4	–	3/5
$\bar{\gamma}_i$	3/4	1/3	1/2

Note: "-" indicates the corresponding $\hat{\gamma}_{il}$ is not calculated since L1 $\notin \mathbf{L}_1$ and L2 $\notin \mathbf{L}_2$.

3.2.2. Bounds updating

In this subsection, we discuss how we can further tighten $\hat{\gamma}_{il}$. Recall that bounds on V_i are derived using pairs of constraints. For each such pair, we can derive bounds tighter than $\hat{\gamma}_{il}$ by considering one additional constraint in Eqn. (27) that is not included in such pair.

We elaborate the aforementioned idea in the context of blending. Recall that $\hat{\gamma}_{il}$ is based on the binary mixture of streams i and $b(l)$ with volume γ, which satisfies the specification for property l and contains $(\gamma - \hat{\gamma}_{il})$ volume of stream $b(l)$. It is possible that stream $b(l)$ violates specifications for other properties, and its maximum volume in γ volume of product is less than $(\gamma - \hat{\gamma}_{il})$. For all $(i, l \in \mathbf{L}_i, b(l))$ combinations, we check if the following holds: $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$.

If $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$, then there exists a property $m[b(l)]$ (the property from which $\bar{\gamma}_{b(l)}$ is derived, see section 3.2.1) whose specification is violated by the binary mixture of stream i and $b(l)$ that satisfies specification for property l exactly (i.e., $\mu_{i,m[b(l)]}\hat{\gamma}_{il} + \mu_{b(l),m[b(l)]}(\gamma - \hat{\gamma}_{il}) > 0$). Note that property $m[b(l)]$ is not considered when deriving $\hat{\gamma}_{il}$; when taking it into account, the binary mixture of stream i and $b(l)$ will not be able to satisfy the specifications for property l and property $m[b(l)]$ simultaneously. In such case, we include one additional stream to the binary mixture. Note that by including one additional stream, $\hat{\gamma}_{il}$ will be tightened since it is previously obtained from the binary mixture of streams i and $b(l)$ that satisfies specification for property l exactly.

Specifically, we tighten $\hat{\gamma}_{il}$ by considering the "second best" stream for property l. We define $\mu_l^+ = \min_{i' \in \mathbf{I} \setminus \{b(l)\}}\{\mu_{i'l}\}$. Let $b^+(l)$ be a set function that returns the "second best" stream: $b^+(l) = \arg\min_{i' \in \mathbf{I} \setminus \{b(l)\}}\{\mu_{i'l}\}$, which implies $\mu_l^+ = \pi_{b^+(l),l} - \pi_{kl}^{\mathrm{U}}$. If there are multiple "second best" streams, we proceed as follows: for the specific $(i, l \in \mathbf{L}_i, b(l))$ combination being considered, if $\mu_{il} = \mu_l^+$ (i.e., stream i is one of the "second best" streams), then $b^+(l) = i$; else, $b^+(l)$ is the stream with the smallest index among all such streams.

To tighten $\hat{\gamma}_{il}$, we prove three propositions. In Proposition 1, we consider a special case where $b^+(l) = i$, while in Propositions 2 and Propositions 3 we consider the more general case where $b^+(l) \neq i$. For Propositions 2 and Propositions 3, we consider a mixture with volume γ that contains stream i, $b(l)$, and $b^+(l)$, and satisfies the specification for property l. Note that for volume γ of such mixture, the (current) upper bound on volume of stream $b(l)$ is $\bar{\gamma}_{b(l)}$. Assume we have volume $\hat{\gamma}_{il}$ for stream i and volume $(\gamma - \bar{\gamma}_{b(l)} - \hat{\gamma}_{il})$ for stream $b^+(l)$. Then, for property l we have:

$$\mu_l^* \bar{\gamma}_{b(l)} + \mu_{il}\hat{\gamma}_{il} + \mu_l^+(\gamma - \bar{\gamma}_{b(l)} - \hat{\gamma}_{il}) \le 0, \qquad i \in \mathbf{I}, l \in \mathbf{L}_i$$

which is equivalent to:

$$(\mu_{il} - \mu_l^+)\hat{\gamma}_{il} \le \mu_l^+(\bar{\gamma}_{b(l)} - \gamma) - \mu_l^* \bar{\gamma}_{b(l)}, \qquad i \in \mathbf{I}, l \in \mathbf{L}_i$$

For Propositions 2 and Propositions 3, since $b^+(l) \neq i$, by the definition of μ_l^+ it follows that $\mu_{il} - \mu_l^+ > 0$. Thus, we have:

$$\hat{\gamma}_{il} \le \frac{\mu_l^+(\bar{\gamma}_{b(l)} - \gamma) - \mu_l^* \bar{\gamma}_{b(l)}}{(\mu_{il} - \mu_l^+)}, \qquad i \in \mathbf{I}, l \in \mathbf{L}_i$$

Note that the RHS of the above equation can be nonpositive. Proposition 2 shows that in such case zero is a valid upper bound on V_i. If the RHS is positive, Proposition 3 shows that it is a valid upper bound on V_i.

Proposition 1 For $\left(i, l \in \mathbf{L}_i, b(l)\right)$ with $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$ and $b^+(l) = i$, if $\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} \leq 0$, then $\hat{\gamma}_{il} \leq 0$.

Proof (by contradiction).

Since $b^+(l) = i$ and $l \in \mathbf{L}_i$, it follows that the "second best" stream violates the specification for property l, thus $\mu_l^+ = \mu_{il} > 0$.

From Eqn. (30), we have $\hat{\gamma}_{il} = -\frac{\mu_l^* \gamma}{\mu_l^+ - \mu_l^*}$, which leads to $(\mu_l^+ - \mu_l^*)\hat{\gamma}_{il} = -\mu_l^* \gamma$. If we move all terms to the LHS, we have $\mu_l^+ \hat{\gamma}_{il} - \mu_l^* \hat{\gamma}_{il} + \mu_l^* \gamma = 0$, and thus, we have

$$\mu_l^+ \hat{\gamma}_{il} + \mu_l^*(\gamma - \hat{\gamma}_{il}) = 0 \tag{31}$$

To simplify the notation, we introduce $\varepsilon = \hat{\gamma}_{il}$, which means that Eqn. (31) can be written as:

$$\mu_l^+ \varepsilon + \mu_l^*(\gamma - \varepsilon) = 0 \tag{32}$$

Next, to prove the result using contradiction, we assume that $\hat{\gamma}_{il} = \varepsilon > 0$.

Recall that $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il} = \gamma - \varepsilon$, and thus, if we multiply both sides of the inequality with $\mu_l^* < 0$, we obtain $\mu_l^* \bar{\gamma}_{b(l)} > \mu_l^*(\gamma - \varepsilon)$. Therefore, from Eqn. (32), we have:

$$\mu_l^+ \varepsilon + \mu_l^* \bar{\gamma}_{b(l)} > 0 \tag{33}$$

We also have

$$\sum\nolimits_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} = \mu_{il} V_i + \sum\nolimits_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \tag{34}$$

If $V_i = \hat{\gamma}_{il} = \varepsilon$, then

$$\sum\nolimits_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} = \mu_l^+ \varepsilon + \sum\nolimits_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \tag{35}$$

Note that

$$\sum\nolimits_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} = \mu_l^* V_{b(l)} + \sum\nolimits_{i' \in \mathbf{I} \setminus \{b(l), i\}} \mu_{i'l} V_{i'} \tag{36}$$

with $\mu_l^* < 0$ and $V_{b(l)} \leq \bar{\gamma}_{b(l)}$.

Since the "second best" stream, in this case stream i, violates the specification for property l (i.e., $\mu_{il} > 0$), it follows that $\mu_{i'l} > 0, \forall i' \in \mathbf{I} \setminus \{b(l),\ i\}$, while $\mu_l^* < 0$. Since $V_{i'}$ is nonnegative, the RHS of Eqn. (36) decreases as the value of $V_{b(l)}$ increases. With $V_{b(l)}$ upper bounded by $\bar{\gamma}_{b(l)}$, we have:

$$\sum_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \geq \mu_l^* \bar{\gamma}_{b(l)} + \sum_{i' \in \mathbf{I} \setminus \{b(l), i\}} \mu_{i'l} V_{i'} \quad (37)$$

Combing Eqn. (35) and (37) we obtain:

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} = \mu_l^+ \varepsilon + \sum_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \geq \mu_l^+ \varepsilon + \mu_l^* \bar{\gamma}_{b(l)} + \sum_{i' \in \mathbf{I} \setminus \{b(l), i\}} \mu_{i'l} V_{i'} \quad (38)$$

with $\sum_{i' \in \mathbf{I} \setminus \{b(l), i\}} \mu_{i'l} V_{i'} \geq 0$ (since $\mu_{i'l} > 0, \forall i' \in \mathbf{I} \setminus \{b(l),\ i\}$ and $V_{i'}$ is nonnegative) and $\mu_l^+ \varepsilon + \mu_l^* \bar{\gamma}_{b(l)} > 0$ (see Eqn. (33)).

Thus, from Eqn. (38) it follows that $\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} > 0$, which leads to a contradiction. ■

Before presenting Proposition 2 and Proposition 3, we introduce some prerequisites. To derive a valid upper bound on V_i, for $\left(i, l \in \mathbf{L}_i, b(l)\right)$ with $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$ and $b^+(l) \neq i$, we again consider volume γ for a product (i.e., $\sum_{i' \in \mathbf{I}} V_{i'} = \gamma$) where we assume $V_i = \hat{\gamma}_{il}$. Such assumptions imply (1) $\sum_{i' \in \mathbf{I} \setminus \{i\}} V_{i'} = \gamma - \hat{\gamma}_{il}$ and (2)

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} = \mu_{il} \hat{\gamma}_{il} + \sum_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \quad (39)$$

The LHS of Eqn. (39) should be nonpositive (see Eqn. (27)). To prove Proposition 2 and Proposition 3 by contradiction, we show that under certain conditions, the RHS of Eqn. (39) is positive. We first investigate the second term in the RHS of Eqn. (39). We are interested in the lower bound on $\sum_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'}$ subject to $\sum_{i' \in \mathbf{I} \setminus \{i\}} V_{i'} = \gamma - \hat{\gamma}_{il}$ and $V_{b(l)} \leq \bar{\gamma}_{b(l)}$. In other words, we are interested in the solution of the following LP (LP1):

$$\begin{aligned} \min \quad & \sum_{i' \in \mathbf{I} \setminus \{i\}} \mu_{i'l} V_{i'} \\ \text{s.t.} \quad & \sum_{i' \in \mathbf{I} \setminus \{i\}} V_{i'} \leq \gamma - \hat{\gamma}_{il} \\ & -\sum_{i' \in \mathbf{I} \setminus \{i\}} V_{i'} \leq \hat{\gamma}_{il} - \gamma \\ & V_{b(l)} \leq \bar{\gamma}_{b(l)} \\ & V_{i'} \geq 0, i' \in \mathbf{I} \setminus \{i\} \end{aligned}$$

The optimal objective function value for LP1 provides a lower bound on $\sum_{i' \neq i} \mu_{i'l} V_{i'}$. LP1 contains $(|\mathbf{I}| - 1)$ variables and three inequality constraints. Here, we note that the optimal solution to LP1 is $V_{b^+(l)} = \gamma - \hat{\gamma}_{il} - \bar{\gamma}_{b(l)}$, $V_{b(l)} = \bar{\gamma}_{b(l)}$, with all other variables being zero. When $\mu_l^+ \leq 0$, the corresponding dual variables for the three inequality constraints are $0, \mu_l^+$, and $(\mu_l^* - \mu_l^+)$; when $\mu_l^+ > 0$, the corresponding dual variables for the three constraints are $-\mu_l^+, 0$, and $(\mu_l^* - \mu_l^+)$. One can verify the optimality of such solution with strong duality. We show the optimal tableau for LP1 in Appendix A.

The optimal solution mentioned above leads to an objective function value of $\mu_l^+(\hat{\gamma}_{il}-\gamma)+(\mu_l^*-\mu_l^+)\bar{\gamma}_{b(l)}$. Thus, from LP1 we have $\sum_{i'\in\mathbf{I}\setminus\{i\}}\mu_{i'l}V_{i'}\geq\mu_l^+(\hat{\gamma}_{il}-\gamma)+(\mu_l^*-\mu_l^+)\bar{\gamma}_{b(l)}$.

We now revisit Eqn. (39). From LP1, we have a lower bound on the second term of its RHS, so we have:

$$\sum_{i'\in\mathbf{I}}\mu_{i'l}V_{i'}\geq\mu_{il}\hat{\gamma}_{il}+\mu_l^+(\hat{\gamma}_{il}-\gamma)+(\mu_l^*-\mu_l^+)\bar{\gamma}_{b(l)} \quad (40)$$

if $\sum_{i'\in\mathbf{I}}V_{i'}=\gamma-\hat{\gamma}_{il}$ and $V_{b(l)}\leq\bar{\gamma}_{b(l)}$ hold.

We next present Proposition 2 and Proposition 3.

Proposition 2 For $(i,l\in\mathbf{L}_i,b(l))$ with $\bar{\gamma}_{b(l)}<\gamma-\hat{\gamma}_{il}$, $b^+(l)\neq i$, $\sum_{i'\in\mathbf{I}}V_{i'}=\gamma$, $V_{b(l)}\leq\bar{\gamma}_{b(l)}$, and $\frac{\mu_l^+(\bar{\gamma}_{b(l)}-\gamma)-\mu_l^*\bar{\gamma}_{b(l)}}{(\mu_{il}-\mu_l^+)}\leq 0$, if $\sum_{i'\in\mathbf{I}}\mu_{i'l}V_{i'}\leq 0$, then $\hat{\gamma}_{il}\leq 0$.

Proof (by contradiction)

Since $\sum_{i'\in\mathbf{I}}V_{i'}=\gamma$, if $V_i=\hat{\gamma}_{il}$, then $\sum_{i'\in\mathbf{I}\setminus\{i\}}V_{i'}=\gamma-\hat{\gamma}_{il}$. We also have $V_{b(l)}\leq\bar{\gamma}_{b(l)}$. Thus from Eqn. (40) we have:

$$\sum_{i'\in\mathbf{I}}\mu_{i'l}V_{i'}\geq\mu_{il}\,\hat{\gamma}_{il}+\mu_l^+(\hat{\gamma}_{il}-\gamma)+(\mu_l^*-\mu_l^+)\bar{\gamma}_{b(l)}=\mu_l^*\bar{\gamma}_{b(l)}+\mu_l^+\left(\gamma-\bar{\gamma}_{b(l)}\right)+(\mu_{il}-\mu_l^+)\hat{\gamma}_{il} \quad (41)$$

We examine the signs of $\mu_l^*\bar{\gamma}_{b(l)}+\mu_l^+\left(\gamma-\bar{\gamma}_{b(l)}\right)$ and $(\mu_{il}-\mu_l^+)\hat{\gamma}_{il}$ in the RHS of Eqn. (41) separately.

For $\mu_l^*\bar{\gamma}_{b(l)}+\mu_l^+\left(\gamma-\bar{\gamma}_{b(l)}\right)$: since $l\in\mathbf{L}_i$ and $b^+(l)\neq i$, it follows that $\mu_{il}>\mu_l^+$, and thus $\mu_{il}-\mu_l^+>0$. Since $\frac{\mu_l^+(\bar{\gamma}_{b(l)}-\gamma)-\mu_l^*\bar{\gamma}_{b(l)}}{(\mu_{il}-\mu_l^+)}\leq 0$ and the denominator is positive, it follows that the numerator $\mu_l^+\left(\bar{\gamma}_{b(l)}-\gamma\right)-\mu_l^*\bar{\gamma}_{b(l)}\leq 0$, which is equivalent to $\mu_l^*\bar{\gamma}_{b(l)}+\mu_l^+\left(\gamma-\bar{\gamma}_{b(l)}\right)\geq 0$.

For $(\mu_{il}-\mu_l^+)\hat{\gamma}_{il}$: we have $\mu_{il}-\mu_l^+>0$. To prove Proposition 2 using contradiction, we assume that $\hat{\gamma}_{il}>0$, so it follows that $(\mu_{il}-\mu_l^+)\hat{\gamma}_{il}>0$.

Thus, from Eqn. (41) we have $\sum_{i'}\mu_{i'l}V_{i'}>0$, which leads to a contradiction. ■

Proposition 3 For $(i,l\in\mathbf{L}_i,b(l))$ with $\bar{\gamma}_{b(l)}<\gamma-\hat{\gamma}_{il}$, $b^+(l)\neq i$, $\sum_{i'\in\mathbf{I}}V_{i'}=\gamma$, $V_{b(l)}\leq\bar{\gamma}_{b(l)}$, and $\frac{\mu_l^+(\bar{\gamma}_{b(l)}-\gamma)-\mu_l^*\bar{\gamma}_{b(l)}}{(\mu_{il}-\mu_l^+)}>0$, if $\sum_{i'\in\mathbf{I}}\mu_{i'l}V_{i'}\leq 0$, then $\hat{\gamma}_{il}\leq\frac{\mu_l^+(\bar{\gamma}_{b(l)}-\gamma)-\mu_l^*\bar{\gamma}_{b(l)}}{(\mu_{il}-\mu_l^+)}$.

Proof (by contradiction)

Since $\sum_{i'\in\mathbf{I}}V_{i'}=\gamma$, if $V_i=\hat{\gamma}_{il}$, then $\sum_{i'\neq i}V_{i'}=\gamma-\hat{\gamma}_{il}$. We also have $V_{b(l)}\leq\bar{\gamma}_{b(l)}$. Thus, from Eqn. (40) we have:

$$\sum_{i'\in\mathbf{I}}\mu_{i'l}V_{i'}\geq\mu_{il}\,\hat{\gamma}_{il}+\mu_l^+(\hat{\gamma}_{il}-\gamma)+(\mu_l^*-\mu_l^+)\bar{\gamma}_{b(l)}=\mu_l^*\bar{\gamma}_{b(l)}+\mu_l^+\left(\gamma-\bar{\gamma}_{b(l)}\right)+(\mu_{il}-\mu_l^+)\hat{\gamma}_{il} \quad (42)$$

To prove Proposition 3 using contradiction, we assume that $\hat{\gamma}_{il}=\frac{\mu_l^+(\bar{\gamma}_{b(l)}-\gamma)-\mu_l^*\bar{\gamma}_{b(l)}}{(\mu_{il}-\mu_l^+)}+\varepsilon$ with $\varepsilon>0$. From Eqn. (41) we have:

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} \geq \mu_l^* \bar{\gamma}_{b(l)} + \mu_l^+ \left(\gamma - \bar{\gamma}_{b(l)}\right) + (\mu_{il} - \mu_l^+)\left(\frac{\mu_l^+ \left(\bar{\gamma}_{b(l)} - \gamma\right) - \mu_l^* \bar{\gamma}_{b(l)}}{(\mu_{il} - \mu_l^+)} + \varepsilon\right)$$

or

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} \geq \mu_l^* \bar{\gamma}_{b(l)} + \mu_l^+ \left(\gamma - \bar{\gamma}_{b(l)}\right) + \mu_l^+ \left(\bar{\gamma}_{b(l)} - \gamma\right) - \mu_l^* \bar{\gamma}_{b(l)} + (\mu_{il} - \mu_l^+)\varepsilon$$

After rearranging terms, we obtain,

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} \geq \mu_l^* \bar{\gamma}_{b(l)} - \mu_l^* \bar{\gamma}_{b(l)} + \mu_l^+ \left(\gamma - \bar{\gamma}_{b(l)}\right) + \mu_l^+ \left(\bar{\gamma}_{b(l)} - \gamma\right) + (\mu_{il} - \mu_l^+)\varepsilon$$

which leads to

$$\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} \geq (\mu_{il} - \mu_l^+)\varepsilon$$

Since $l \in \mathbf{L}_i$ and $b^+(l) \neq i$, it follows that $\mu_{il} > \mu_l^+$, and thus $\mu_{il} - \mu_l^+ > 0$. With $\varepsilon > 0$ we have $\sum_{i' \in \mathbf{I}} \mu_{i'l} V_{i'} > 0$, which leads to a contradiction. ■

From Proposition 2 and Proposition 3, it follows that for $\left(i, l \in \mathbf{L}_i, b(l)\right)$ with $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$ and $b^+(l) \neq i$, we can calculate a valid upper bound on V_i $\hat{\gamma}_{il} = \max\left\{0, \frac{\mu_l^+ (\bar{\gamma}_{b(l)} - \gamma) - \mu_l^* \bar{\gamma}_{b(l)}}{(\mu_{il} - \mu_l^+)}\right\}$

Utilizing the above results, we update bounds as follows: for $i \in \mathbf{I}$ and $l \in \mathbf{L}_i$, we first check if $\bar{\gamma}_{b(l)} < \gamma - \hat{\gamma}_{il}$ ($\hat{\gamma}_{il}$ is calculated from Eqn. (30)); if that is the case, we have:

$$\hat{\gamma}_{il} = \begin{cases} 0, & \text{if } b^+(l) = i \\ \max\left\{0, \dfrac{\mu_l^+ \left(\bar{\gamma}_{b(l)} - \gamma\right) - \mu_l^* \bar{\gamma}_{b(l)}}{(\mu_{il} - \mu_l^+)}\right\}, & \text{otherwise} \end{cases}$$

To illustrate, we consider the example in Figure 2. Note that we have $b(\text{L2}) = 2, b^+(\text{L2}) = 1$, and from Table 1 we have $\bar{\gamma}_{b(\text{L2})} = \bar{\gamma}_2 = 1/3$, and $1/3 < \gamma - \hat{\gamma}_{3,\text{L2}} = 2/5$. Thus, we update $\hat{\gamma}_{3,\text{L2}}$. Since $b^+(\text{L2}) \neq 3$, and $[\mu_{\text{L2}}^+(\bar{\gamma}_2 - \gamma) - \mu_{\text{L2}}^* \bar{\gamma}_2]/\left(\mu_{3,\text{L2}} - \mu_{\text{L2}}^+\right) = 1/3 > 0$, we have $\hat{\gamma}_{3,\text{L2}} = 1/3$, and $\bar{\gamma}_3$ is updated to 1/3.

The bounds calculated by our method are given in Table 2. For comparison, we also show the bounds which would have been obtained by FBBT and OBBT for the same example. We note that for this example, bounds on all V_i obtained from our method are tighter than the bounds obtained from FBBT. For V_1 and V_2, the bounds obtained from our method are as tight as the bounds obtained from OBBT.

Table 2. Bounds calculated by different methods

$\bar{\gamma}_i$	$i = 1$	$i = 2$	$i = 3$
FBBT	1	1/2	3/4
OBBT	3/4	1/3	1/11
Our method	3/4	1/3	1/3

Note: Calculation performed by FBBT and OBBT can be found in Appendix B.

3.2.3. Complete procedure for bound tightening

The complete procedure, which combines the calculations described in the previous subsections, is summarized below. The pseudocode, where we bring back indices *j, k, t* and thus $\mathbf{L}_{ik} = \{l: \mu_{ikl} > 0\}$, is as follows:

Complete procedure for bound tightening

For $k \in \mathbf{K}$ **do**
 For $j \in \mathbf{J}$ **do**
  For $i \in \mathbf{I}$ **do**
   $\bar{\gamma}_{ijk} = \gamma_j$
   For $l \in \mathbf{L}_{ik}$ **do**
    $\hat{\gamma}_{ijkl} = -\dfrac{\mu_l^* \gamma_j}{\mu_{ikl} - \mu_l^*}$
    $\bar{\gamma}_{ijk} = \min\{\bar{\gamma}_{ijk}, \hat{\gamma}_{ijkl}\}$
   End
  End
  For $i \in \mathbf{I}$ **do**
   For $l \in \mathbf{L}_{ik}$ **do**
    If $\bar{\gamma}_{b(l),jk} < \gamma - \hat{\gamma}_{ijkl}$ **then**
     If $b^+(l) = i$ **then**
      $\hat{\gamma}_{ijkl} = 0$
     Else
      $\hat{\gamma}_{ijkl} = \max\left\{0,\ \frac{\mu_{kl}^+(\bar{\gamma}_{b(l),jk} - \gamma) - \mu_l^* \bar{\gamma}_{b(l),jk}}{(\mu_{ilk} - \mu_{lk}^+)}\right\}$
     End
     $\bar{\gamma}_{ijk} = \min\{\bar{\gamma}_{ijk}, \hat{\gamma}_{ijkl}\}$
    End
   End
  End
 End
End
Output: $\bar{\gamma}_{ijk}$

3.2.4. Reformulation and preprocessing for M^{PB}

We lift P_{ijt}, and partition it into nonnegative continuous variables U_{ijkt} and V_{ijkt}:

$$P_{ijt} = U_{ijkt} + V_{ijkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{43}$$

$$\sum_{i \in \mathbf{I}} U_{ijkt} \leq 1 - X_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{44}$$

$$\sum_{i \in \mathbf{I}} V_{ijkt} \leq X_{jkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{45}$$

where U_{ijkt} now represents the proportion of stream i in blender j during time period t when there is no flow from blender j to product k ($X_{jkt} = 0$), and V_{ijkt} represents such proportion when $X_{jkt} = 1$.

Eqn. (16) now becomes:

$$\hat{F}_{ijkt} = V_{ijkt} \sum_{i' \in \mathbf{I}} \hat{F}_{i'jkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{46}$$

and Eqn. (17) can be re-written as follow:

$$\sum_{i \in \mathbf{I}} \pi_{il} V_{ijkt} \leq \pi_{kl}^{\mathrm{U}} \sum_{i \in \mathbf{I}} V_{ijkt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{47}$$

Eqn. (1) - (4), (8) - (11), and (13) -(15), and (43) - (47) comprise a reformulation for the proportion-based formulation, henceforth referred to as $\mathrm{M}^{\mathrm{UV-P}}$.

From Eqn. (45) and (47) with previously defined parameter μ_{il}, we have:

$$\sum_{i \in \mathbf{I}} V_i \leq 1 \tag{48}$$

$$\sum_{i \in \mathbf{I}} \mu_{il} V_i \leq 0, \qquad l \in \mathbf{L} \tag{49}$$

Comparing Eqn. (48) and (49) with Eqn. (26) and (27) (from which bounds on V_i are calculated for model M^{SB}), the only difference is that instead of parameter γ in the RHS of Eqn. (26), here we have constant 1 in the RHS of Eqn. (48). It is therebefore straightforward to tighten bounds on V_i with the procedure shown in 3.2.3 by setting $\gamma_j = 1$, and thus the initial value for $\bar{\gamma}_{ijk}$ is 1.

4. Valid constraints

4.1. Model M^{SB}

We first present valid constraints for M^{SB}. Since $R_{jkt} \in [0,1]$, $(1 - R_{jkt}) \in [0,1]$, so multiplying $V_{ijkt} \leq \bar{\gamma}_{ijk}$ by $(1 - R_{jkt})$ yields:

$$(1 - R_{jkt}) V_{ijkt} \leq \left(1 - R_{jkt}\right) \bar{\gamma}_{ijk}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T}$$

and then:

$$V_{ijkt} - V_{ijkt} R_{jkt} \leq \left(1 - R_{jkt}\right) \bar{\gamma}_{ijk}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T}$$

Note that $\hat{F}_{ijkt} = V_{ijkt} R_{jkt}$, thus:

$$V_{ijkt} - \hat{F}_{ijkt} \leq \left(1 - R_{jkt}\right) \bar{\gamma}_{ijk}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{50}$$

If we reintroduce indices j, k, and t, Eqn. (28) can be written as:

$$\sum\nolimits_{i\in\mathbf{I}}(1-\frac{\mu_{ikl}}{\mu_l^*})V_{ijkt} \le \gamma_j, \qquad k\in\mathbf{K}, l\in\mathbf{L}, t\in\mathbf{T}$$

Multiplying both sides with $\left(1-R_{jkt}\right)$ leads to:

$$\sum\nolimits_{i\in\mathbf{I}}(1-\frac{\mu_{ikl}}{\mu_l^*})V_{ijkt}\left(1-R_{jkt}\right) \le \left(1-R_{jkt}\right)\gamma_j, \qquad j\in\mathbf{J}, k\in\mathbf{K}, l\in\mathbf{L}, t\in\mathbf{T}$$

or

$$\sum\nolimits_{i\in\mathbf{I}}(1-\frac{\mu_{ikl}}{\mu_l^*})\left(V_{ijkt}-V_{ijkt}R_{jkt}\right) \le \left(1-R_{jkt}\right)\gamma_j, \qquad j\in\mathbf{J}, k\in\mathbf{K}, l\in\mathbf{L}, t\in\mathbf{T} \tag{51}$$

Since $\hat{F}_{ijkt} = V_{ijkt}R_{jkt}$, Eqn. (51) can be written as follows:

$$\sum\nolimits_{i\in\mathbf{I}}(1-\frac{\mu_{ikl}}{\mu_l^*})(V_{ijkt}-\hat{F}_{ijkt}) \le \left(1-R_{jkt}\right)\gamma_j, \qquad j\in\mathbf{J}, k\in\mathbf{K}, l\in\mathbf{L}, t\in\mathbf{T} \tag{52}$$

Both Eqn. (50) and (52) are RLT constraints (Sherali and Adams 1999). Finally, we also have:

$$V_{ijkt} \le \bar{\gamma}_{ijk}, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{53}$$

Eqn. (53) enforces upper bounds on V_{ijkt} which may be tighter than the bounds obtained through general purpose bound tightening techniques such as FBBT.

Eqn. (50) and (52) - (53) are added to model M^{UV},resulting in model $\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$. We show an illustrative graph for our tightening methods using the example introduced in Figure 2 in Appendix B. We also introduce model $\mathrm{M}^{\mathrm{UV}}_{\mathrm{R}}$, which has the same constraints as $\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$, but without tightened bounds on V_{ijkt} (i.e., $\bar{\gamma}_{ijk} = \gamma_j$ in Eqn. (50), (52) - (53)). We summarize the models we consider in Table 3.

Table 3. Model description

Models	**Constraints and variable bounds**
M^{SB}	Eqn. (1) – (13)
M^{UV}	Eqn. (1) – (4), (6) – (10), (12) – (18)
$\mathrm{M}^{\mathrm{UV}}_{\mathrm{R}}$	M^{UV} + Eqn. (50), (52) - (53) $\bar{\gamma}_{ijk} = \gamma_j$
$\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$	M^{UV} + Eqn. (50), (52) - (53) $\bar{\gamma}_{ijk}$ obtained from our method

4.2. Model M^{PB}

For M^{PB} we have:

$$\sum\nolimits_{i'\in\mathbf{I}}\hat{F}_{i'jkt} \le \delta_{jk}, \qquad j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{54}$$

Multiplying both sides of $V_{ijkt} \leq \bar{\gamma}_{ijk}$ with $(\delta_{jk} - \sum_{i'} \hat{F}_{i'jkt})$ leads to:

$$\delta_{jk} V_{ijkt} - \hat{F}_{ijkt} \leq \bar{\gamma}_{ijk} \left(\delta_{jk} - \sum\nolimits_{i' \in \mathbf{I}} \hat{F}_{i'jkt} \right), \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{55}$$

Similarly, from Eqn. (28) we have:

$$\sum\nolimits_{i \in \mathbf{I}} (1 - \frac{\mu_{ikl}}{\mu_l^*}) V_{ijkt} \leq 1, \qquad k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T}$$

Multiplying both sides in the above equation with $(\delta_{jk} - \sum_{i \in \mathbf{I}} \hat{F}_{ijkt})$ leads to:

$$\sum\nolimits_{i \in \mathbf{I}} (1 - \frac{\mu_{ikl}}{\mu_l^*})(\delta_{jk} V_{ijkt} - \hat{F}_{ijkt}) \leq \left(\delta_{jk} - \sum\nolimits_{i \in \mathbf{I}} \hat{F}_{ijkt} \right), k \in \mathbf{K}, l \in \mathbf{L}, t \in \mathbf{T} \tag{56}$$

For the proportion-based formulation we test the following models: (1) the original model M^{PB}; (2) the reformulated model $\mathrm{M}^{\mathrm{UV-P}}$; (3) the model with RLT constraints (55) and (56) without bound tightening ($\bar{\gamma}_{ijk} = 1$), $\mathrm{M}_{\mathrm{R}}^{\mathrm{UV-P}}$; and (4) the model with RLT constraints (55) and (56) and tightened variable bounds, $\mathrm{M}_{\mathrm{R,T}}^{\mathrm{UV-P}}$.

5. Computational results

We test our methods on different models. Computational experiments are conducted on a Windows 10 machine with Intel Core i7 at 2.80 GHz and 8 GB of RAM. Models are coded in GAMS 28.2. We use BARON 19.7.13 with default options. Instances have five to eight streams, two to ten blenders, four products, four to six properties, and four to ten time points. Instances are modified from Adhya et al. (Adhya, Tawarmalani, and Sahinidis 1999), Ben-Tal et al. (Ben-Tal, Eiger, and Gershovitz 1994), and D'Ambrosio et al. (D'Ambrosio, Linderoth, and Luedtke 2011).

5.1. Case study

We first show the results for a case study. It has eight streams, three blenders, four products, six properties and five time periods. The parameters are given in Appendix C. An optimal schedule, with an objective function value of 3448.7, is shown in Figure 3 and the corresponding inventory profile in Figure 4. The model and solution statistics for different models are given in Table 4. After 300 seconds M^{SB} has an optimality gap of 2.43% while $\mathrm{M}_{\mathrm{R,T}}^{\mathrm{UV}}$ is solved to optimality in less than 50 seconds, indicating the effectiveness of the tighter bounds and RLT constraints.

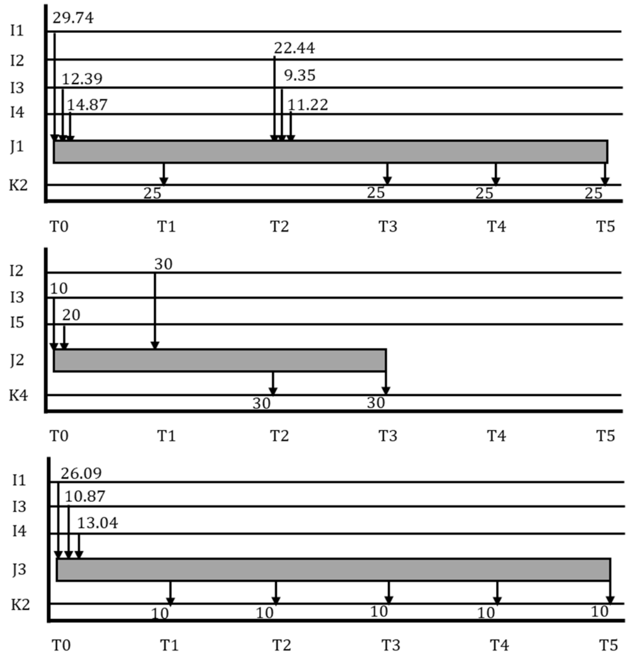

Figure 3. An optimal schedule for the case study.

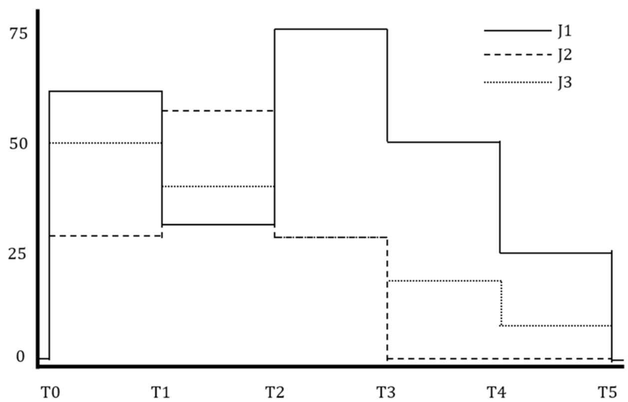

Figure 4. Inventory profile for the schedule shown in Figure 3.

Table 4. Model and solution statistics for the case study

	M^{SB}	M^{UV}	M_R^{UV}	$M_{R,T}^{UV}$
Con. Var.	1009	2161	2161	2161
Bin. Var.	72	72	72	72
Constraints	1921	2641	3649	2161
CPU Time (s)	>300	109.8	>300	47.7
Opt. Gap	2.4%	0	2.4%	0

5.2. MINLP models

We test our method on variants of the two MINLP models with 30 instances, and the show performance profiles in Figure 5. Each instance included in the performance profile satisfies the following two conditions: (1) it is solved by at least one of the models in 300 seconds, and (2) the slowest model for a given instance takes at least 15 seconds to solve it. Figure 5(a) is generated with 22 instances that satisfy the two conditions for variants for M^{SB}. Overall, we observe that $M^{UV}_{R,T}$ performs the best over the tested instances, with substantial improvement over the performance of M^{SB} for most instances. Further, we note that $M^{UV}_{R,T}$ performs better than both of M^{UV} and M^{UV}_{R}. These results indicate the effectiveness of the RLT constraints combined with the proposed bound tightening methods. We also test variants for M^{PB} on the same 30 instances, with 14 of them satisfying the two aforementioned conditions. The performance profile generated using those 14 instances is shown in Figure 5(b). Similar to M^{SB}, we observe that the model with RLT constraints and tightened bounds, $M^{UV-P}_{R,T}$, performs the best. The CPU times for all 30 instances can be found in Appendix D. Note that neither M^{SB} nor M^{PB} is consistently superior on those instances.

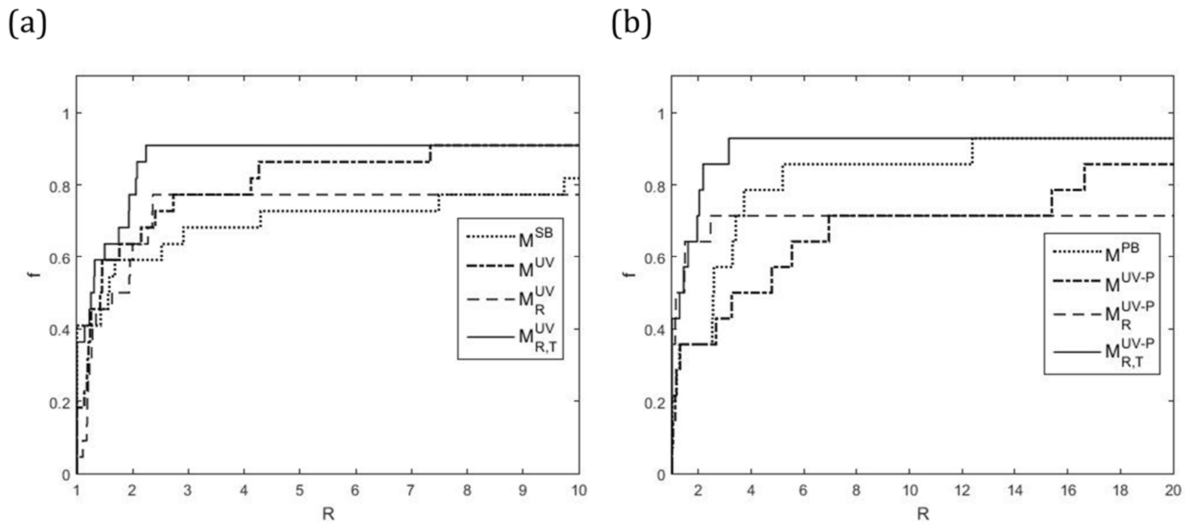

Figure 5. Performance profile for different variants of (a) M^{SB} and (b) M^{PB}

5.3. MILP Models

Mixed-integer linear models that approximate the MINLP models can be developed through discretization. In addition to providing approximate solutions, MILP models can also be used in solution methods (Kolodziej, Castro, and Grossmann 2013; Kolodziej et al. 2013; Gupte et al. 2017). Here, we allow the split fraction, R_{jkt}, to take values only from a discrete set $\mathbf{D}^{R}$, thereby linearizing nonlinear constraints Eqn. (5) and/or Eqn. (17). Specifically, we have $\mathbf{D}^{R} = \{0, \delta_1, \delta_2, \dots, \delta_n, 1\}$ with $\delta_1 = \delta_2 - \delta_1 = \cdots = \delta_n - \delta_{n-1} = 1 - \delta_n = \delta$. The MILP obtained from such discretization, referred to as ${}_{L1}M$, is guaranteed to return only feasible solutions to the original MINLP. A relaxation of ${}_{L1}M$, referred to as ${}_{L2}M$, is obtained by introducing additional continuous variables to allow R_{jkt} to take any values in [0,1]. The resulting bilinear terms with two continuous variables are then relaxed using linear constraints. A comprehensive list of the constraints of the two MILP models can be found in Appendix E. We test MILP models (both with and without our methods) over 20 instances.

Performance profiles for the MILP models are shown in Figure 6. The CPU times and objective function values for the MILP models can be found in Appendix F.

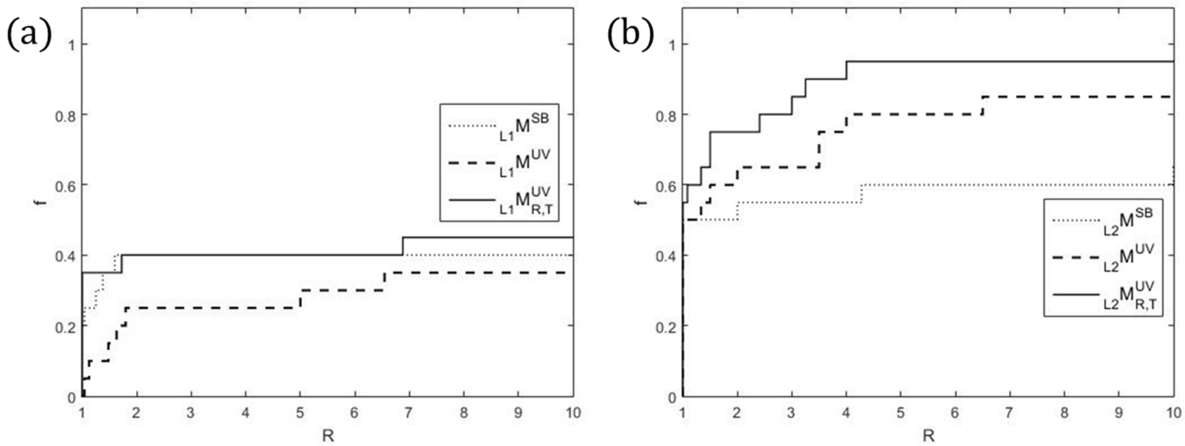

Figure 6. Performance profile for MILP models. (a) $_{\mathrm{L1}}\mathrm{M}$ with $\delta = 0.01$. (b) $_{\mathrm{L2}}\mathrm{M}$ with $\delta = 0.1$

Overall we observe that $_{\mathrm{L1}}\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$ and $_{\mathrm{L2}}\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$ perform best over the tested instances, indicating the effectiveness of our method. For $_{\mathrm{L2}}\mathrm{M}$, we see substantial improvement from the reformulation ($_{\mathrm{L2}}\mathrm{M}^{\mathrm{UV}}$) and bound tightening ($_{\mathrm{L2}}\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$) compared to the original model.

5.4. Decomposition method

We further test our methods on an MILP-MINLP decomposition method for multiperiod blending proposed by (Lotero et al. 2016). We briefly describe the method below: (1). A new binary variable Y_{jt} is introduced, which equals to 1 if blender j feeds products at time point t. (2). A relaxed problem (MILP) is solved in which Eqn. (17), the constraint that contains bilinear term, is replaced using McCormick envelopes with tightened bounds. (3). Binary Y_{jt} is fixed to the value obtained from the solution to the relaxed problem, and a reduced problem (MINLP) containing all constraints in $\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$ is solved ("reduced" in the sense that after fixing Y_{jt}, some X_{jkt} are also fixed, resulting in a reduced feasible space compared to $\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$). Solving one relaxed problem and one reduced problem completes one iteration, from which an upper bound and a lower bound (if the reduced problem is feasible) are obtained. A feasibility or optimality cut is added to the relaxed problem after solving the reduced problem in each iteration. We give the formulation of both the relaxed and reduced problem in Appendix G. More details about the decomposition method can be found in (Lotero et al. 2016).

We show computational results for five instances modified from D'Ambrosio et al. (D'Ambrosio, Linderoth, and Luedtke 2011) in Table 5. We set the maximum number of iterations to five, and time limits for the relaxed problem and reduced problem are set at 10 seconds and 30 seconds, respectively. We use CPLEX 12.9 to solve the relaxed problem.

Table 5. Computational results for decomposition method with different MINLP models

Inst.	M^{SB}			$M^{UV}_{R,T}$		
	# of Iter.	Opt. Gap	CPU Time (s)	# of Iter.	Opt. Gap	CPU Time (s)
1	5	0.3%	162.1	2	0	50.8
2	5	0.4%	155.7	2	0	42.6
3	5	0	171.6	5	0.1%	200
4	4	0	92.2	1	0	16.2
5	5	0.5%	194.3	1	0	18.6

Model M^{SB} and $M^{UV}_{R,T}$ do not solve the above five instances to global optimality in 300 seconds. The decomposition method using M^{SB} solves two instances to global optimality within five iterations, whereas the decomposition using $M^{UV}_{R,T}$ solves four instances to global optimality. We also observe that the decomposition method using $M^{UV}_{R,T}$ typically closes the optimality gap in fewer iterations.

6. Conclusion

We developed variable bound tightening methods, based on multiple constraints, for multiperiod blending. We first proposed a reformulation of the constraints involving bilinear terms using lifting. We introduced a preprocessing method to tighten the bounds on the lifted variables using multiple constraints. The reformulation and the selection of constraints to be considered for bound tightening are based on the understanding of the physical system. We proposed valid constraints derived from Reformulation-Linearization Technique (RLT) that utilize the bounds on the lifted variables to further tighten the formulation. We also discussed how the proposed methods can be coupled with other solution strategies for multiperiod blending problems. Computational results show the effectiveness of our methods in reducing computational requirements.

Acknowledgement

The authors acknowledge financial support from the National Science Foundation under grant CBET2028960.

Appendix A. Solving LP1

After introducing slack variables S_1, S_2, and S_3, LP1 is written as follows:

$$\min \quad \sum\nolimits_{i' \neq i} \mu_{i'l} V_{i'}$$

$$\text{s.t.} \quad \begin{aligned} \sum\nolimits_{i' \neq i} V_{i'} + S_1 &= \gamma - \hat{\gamma}_{il} \\ -\sum\nolimits_{i' \neq i} V_{i'} + S_2 &= \hat{\gamma}_{il} - \gamma \\ V_{b(l)} + S_3 &= \bar{\gamma}_{b(l)} \\ V_{i'} \geq 0, S_1, S_2, S_3 &\geq 0 \end{aligned}$$

By inspection, we have $V_{b^+(l)} = \gamma - \hat{\gamma}_{il}, V_{i'} = 0\ \forall i' \notin \{i, b^+(l)\}, S_1 = S_2 = 0$, and$S_3 = \bar{\gamma}_{b(l)}$ as initial feasible solution. Let $S_1, V_{b^+(l)}$, and S_3 be basic variables, we have the following tableau:

Basic var.	$V_{b(l)}$	$V_{b^+(l)}$	$[V_{i'}, \forall i' \notin \{i, b(l), b^+(l)\}]$	S_1	S_2	S_3	
S_1	0	0	$[0, \dots\dots, 0]$	1	1	0	0
$V_{b^+(l)}$	1	1	$[1, \dots\dots, 1]$	0	-1	0	$\gamma - \hat{\gamma}_{il}$
S_3	1	0	$[0, \dots\dots, 0]$	0	0	1	$\bar{\gamma}_{b(l)}$
z	$\mu_l^+ - \mu_l^*$	0	$[\mu_l^+ - \mu_{i'l}, \forall i' \notin \{i, b(l), b^+(l)\}]$	0	μ_l^+	0	$\mu_l^+(\hat{\gamma}_{il} - \gamma)$

where $[.]$ denotes a row vector of dimension $(|\mathbf{I}| - 3)$.

When $\mu_l^+ \leq 0$, we have the following optimal tableau:

Basic var.	$V_{b(l)}$	$V_{b^+(l)}$	$[V_{i'}, \forall i' \notin \{i, b(l), b^+(l)\}]$	S_1	S_2	S_3	
S_1	0	0	$[0, \dots\dots, 0]$	1	1	0	0
$V_{b^+(l)}$	0	1	$[1, \dots\dots, 1]$	0	-1	0	$\gamma - \hat{\gamma}_{il} - \bar{\gamma}_{b(l)}$
$V_{b(l)}$	1	0	$[0, \dots\dots, 0]$	0	0	1	$\bar{\gamma}_{b(l)}$
z	0	0	$[\mu_l^+ - \mu_{i'l}, \forall i' \notin \{i, b(l), b^+(l)\}]$	0	μ_l^+	$\mu_l^* - \mu_l^+$	$\mu_l^+(\hat{\gamma}_{il} - \gamma)$ $+ (\mu_l^* - \mu_l^+)\bar{\gamma}_{b(l)}$

When $\mu_l^+ > 0$, we have the following optimal tableau:

Basic var.	$V_{b(l)}$	$V_{b^+(l)}$	$[V_{i'}, \forall i' \notin \{i, b(l), b^+(l)\}]$	S_1	S_2	S_3	
S_1	0	0	$[0, \dots\dots, 0]$	1	1	0	0
$V_{b^+(l)}$	1	1	$[1, \dots\dots, 1]$	0	-1	0	$\gamma - \hat{\gamma}_{il} - \bar{\gamma}_{b(l)}$
$V_{b(l)}$	1	0	$[0, \dots\dots, 0]$	0	0	1	$\bar{\gamma}_{b(l)}$
z	0	0	$[\mu_l^+ - \mu_{i'l}, \forall i' \notin \{i, b(l), b^+(l)\}]$	$-\mu_l^+$	0	$\mu_l^* - \mu_l^+$	$\mu_l^+(\hat{\gamma}_{il} - \gamma)$ $+ (\mu_l^* - \mu_l^+)\bar{\gamma}_{b(l)}$

Appendix B. Illustrative example

B.1. Feasibility Based Bound Tightening

Recall that for the illustrative example we have:

$$V_1 + V_2 + V_3 \leq 1$$

$$-V_1 + 2V_2 + V_3 \leq 0$$

$$V_1 - 3V_2 + 2V_3 \leq 0$$

Assume we use 0 and 1 as the initial lower and upper bound, that is, $V_1, V_2, V_3 \in [0,1]$. FBBT uses the following inequality to find tighter upper bounds (note that 0 is the tightest lower bound on V_i):

$$V_i \leq \frac{1}{\alpha_{m^*,i}} \left[\beta_{m^*} - \sum_{i' \neq i} \min\left(a_{m^*,i'} \bar{\gamma}_{i'}, 0\right)\right] \quad a_{m^*,i} > 0 \tag{57}$$

where $\alpha_{m^*,i}$ is the coefficient of V_i for inequality m^*, β_{m^*} is the RHS of inequality m^*, and $\bar{\gamma}_i$ is the upper bound on V_i. In FBBT we choose an inequality with positive coefficient for V_i, to evaluate the RHS of Eqn. (57) to find its upper bound:

$$V_1 \leq \frac{1}{1}[1 - \min(1,0) - \min(1,0)] = 1$$

$$V_1 \leq \frac{1}{1}[0 - \min(-3,0) - \min(2,0)] = 3$$

$$V_2 \leq \frac{1}{1}[1 - \min(1,0) - \min(1,0)] = 1$$

$$V_2 \leq \frac{1}{2}[0 - \min(-1,0) - \min(1,0)] = 1/2$$

Note that we now have a tighter upper bound on V_2, so we update $\bar{\gamma}_2$: $\bar{\gamma}_2 = 1/2$.

$$V_3 \leq \frac{1}{1}[1 - \min(1,0) - \min(1/2,0)] = 1$$

$$V_3 \leq \frac{1}{1}[0 - \min(-1,0) - \min(1,0)] = 1$$

$$V_3 \leq \frac{1}{2}[0 - \min(1,0) - \min(-3/2,0)] = 3/4$$

Note that we now have a tighter upper bound on V_3, so we update $\bar{\gamma}_3$: $\bar{\gamma}_3 = 3/4$.

In FBBT we typically start another round of evaluation using the tightened bounds. For the illustrative example, no further improvement can be obtained. FBBT thus returns: $\bar{\gamma}_1 = 1, \bar{\gamma}_2 = 1/2, \bar{\gamma}_3 = 3/4$.

B.2. OBBT for the illustrative example

OBBT is based on the solution of the following LP:

$$\begin{aligned} \max \quad & V_i \ \ (i = 1,2,3) \\ \text{s.t} \quad & V_1 + V_2 + V_3 \leq 1 \\ & -V_1 + 2V_2 + V_3 \leq 0 \\ & V_1 - 3V_2 + 2V_3 \leq 0 \end{aligned}$$

The value of $\bar{\gamma}_i$ is equal to the objective function value of the i-th LP. After solving three LPs, OBBT returns: $\bar{\gamma}_1 = 3/4, \bar{\gamma}_2 = 1/3, \bar{\gamma}_3 = 1/11$.

B.3. Illustrative graph for our tightening methods

Consider the following nonlinear set:

$$\mathbf{S}_1 = \left\{ \left(\hat{F}_1, \hat{F}_2, \hat{F}_3, R, V_1, V_2, V_3\right) \in \mathbb{R}^+ : \begin{array}{c} V_1 + V_2 + V_3 \leq 1 \\ -V_1 + 2V_2 + V_3 \leq 0 \\ V_1 - 3V_2 + 2V_3 \leq 0 \\ \hat{F}_1 = V_1 R \\ \hat{F}_2 = V_2 R \\ \hat{F}_2 = V_3 R \end{array} \right\}$$

which contains three linear constraints that are identical to the constraints in the illustrative example in section 3, along with three nonlinear equality constraints to model the flows.

We introduce a hyperplane:

$$\mathbf{S}_2 = \left\{ (R, V_1, V_2, V_3) \in \mathbb{R}^+ : \begin{array}{c} R = 1/2 \\ V_1 = 2/3 \\ V_2 = 1/3 \\ V_3 = 0 \end{array} \right\}$$

The intersection of $\mathbf{S}_1$ and $\mathbf{S}_2$ is shown in Figure 7. It is point A on the $(\hat{F}_1, \hat{F}_2)$ plane.

We consider a linear relaxation of $\mathbf{S}_1$, denoted as $\mathbf{S}_1^{\mathrm{MC}}$, using McCormick envelopes without bound tightening. Since $V_1, V_2, V_3 \in [0,1]$, we have:

$$\hat{F}_i \leq R, \quad i = \{1,2,3\} \tag{58}$$

$$\hat{F}_i \leq V_i, \quad i = \{1,2,3\} \tag{59}$$

$$\hat{F}_i \geq R + V_i - 1, \quad i = \{1,2,3\} \tag{60}$$

$$\hat{F}_i \geq 0, \quad i = \{1,2,3\} \tag{61}$$

We also have the following RLT constraints:

$$\hat{F}_1 + \hat{F}_2 + \hat{F}_3 \leq R \tag{62}$$

$$-\hat{F}_1 + 2\hat{F}_2 + \hat{F}_3 \leq 0 \tag{63}$$

$$\hat{F}_1 - 3\hat{F}_2 + 2\hat{F}_3 \leq 0 \tag{64}$$

The set $\mathbf{S}_1^{\mathrm{MC}}$ is thus defined as:

$$\mathbf{S}_1^{\mathrm{MC}} = \left\{ \left(\hat{F}_1, \hat{F}_2, \hat{F}_3, R, V_1, V_2, V_3\right) \in \mathbb{R}^+ : \begin{array}{c} V_1 + V_2 + V_3 \leq 1 \\ -V_1 + 2V_2 + V_3 \leq 0 \\ V_1 - 3V_2 + 2V_3 \leq 0 \\ \text{Eqn. } (58) - (60) \\ \text{Eqn. } (62) - (64) \end{array} \right\}$$

The intersection of $\mathbf{S}_1^{\mathrm{MC}}$ and $\mathbf{S}_2$ is the quadrilateral $ABCD$.

We consider a linear relaxation of $\mathbf{S}_1$, denoted as $\mathbf{S}_1^{\mathrm{T}}$, using McCormick envelopes with tightened bounds. Our methods lead to: $V_1 \in [0, 3/4]$, $V_2 \in [0, 1/3]$, $V_3 \in [0, 1/3]$. McCormick envelopes constructed using such bounds are:

$$\hat{F}_1 \leq \frac{3}{4} R \tag{65}$$

$$\hat{F}_2 \leq \frac{1}{3} R \tag{66}$$

$$\hat{F}_3 \leq \frac{1}{3} R \tag{67}$$

$$\hat{F}_1 \geq \frac{3}{4} R + V_1 - \frac{3}{4} \tag{68}$$

$$\hat{F}_2 \geq \frac{1}{3} R + V_2 - \frac{1}{3} \tag{69}$$

$$\hat{F}_3 \geq \frac{1}{3} R + V_3 - \frac{1}{3} \tag{70}$$

together with Eqn. (59) and Eqn. (61). Note that Eqn. (68) – (70) are identical to Eqn. (50) for the illustrative example.

The set $\mathbf{S}_1^{\mathrm{T}}$ is thus defined as:

$$\mathbf{S}_1^{\mathrm{T}} = \left\{ \left(\hat{F}_1, \hat{F}_2, \hat{F}_3, R, V_1, V_2, V_3\right) \in \mathbb{R}^+ : \begin{array}{c} V_1 + V_2 + V_3 \leq 1 \\ -V_1 + 2V_2 + V_3 \leq 0 \\ V_1 - 3V_2 + 2V_3 \leq 0 \\ \text{Eqn. } (59), (61), (65) - (70) \\ \text{Eqn. } (62) - (64) \end{array} \right\}$$

The intersection of $\mathbf{S}_1^{\mathrm{T}}$ and $\mathbf{S}_2$ is also point A, which coincides with the intersection of the nonlinear set $\mathbf{S}_1$ and $\mathbf{S}_2$.

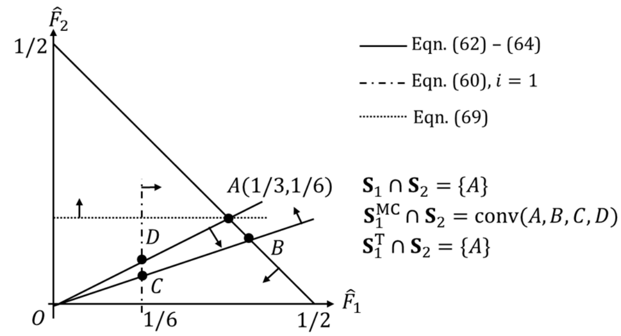

Figure 7. Illustrative graph for tightening constraints

Appendix C. Parameters for the Case Study

Table 6. Stream availability and cost, maximum product demand, and product price

	I1	I2	I3	I4	I5	I6	I7	I8	K1	K2	K3	K4
ξ_i	450	450	450	450	450	450	450	450	-	-	-	-
α_i^{V}	7	3	2	10	5	5	9	11	-	-	-	-
β_k	-	-	-	-	-	-	-	-	16	25	15	10
ω_k	-	-	-	-	-	-	-	-	60	150	180	60

Table 7. Stream properties and product specifications

	π_{il}								π_{kl}^{U}			
	I1	I2	I3	I4	I5	I6	I7	I8	K1	K2	K3	K4
L1	1	4	4	5	1	1.8	5	3	3	4	1.5	3
L2	6	1	5.5	3	2.7	2.7	1	3	3	2.5	5.5	4
L3	4	3	3	3	4	4	1.7	3	3.25	3.5	3.9	4
L4	0.5	2	0.9	1	1.6	3.5	2.9	1	0.75	1.5	0.8	1.8
L5	5	4	7	3	3	6.1	3.5	5	6	7	7	8
L6	9	4	10	4	7	3	2.9	2	5	6	6	6

Table 8. Flow upper bound

δ_{jk}	J1	J2	J3
K1	10	10	30
K2	25	10	10
K3	30	25	10
K4	10	30	25

All blenders have capacity of 75, and we do not consider fixed costs.

Appendix D. CPU time for MINLP models

Table 9. CPU time for variants of M^{SB}

Instance	CPU Time (in seconds)			
	M^{SB}	M^{UV}	M_R^{UV}	$M_{R,T}^{UV}$
6	28.615	6.666	8.519	14.918
7	200.41	>300	>300	>300
8	40.26	5.374	10.479	6.717
9	7.728	9.104	10.357	20.228
10	6.634	2.233	2.346	2.625
11	27.249	17.249	28.078	19.598
12	3.078	9.912	13.584	11.758
13	12.745	93.531	28.992	16.787
14	5.758	8.32	11.398	10.774
15	49.25	144.03	19.543	29.24
16	10.771	11.491	9.887	13.413
17	15.869	67.67	31.696	32.988
18	>300	37.919	35.161	31.799
19	287.808	267.84	>300	184.849
20	15.31	41.77	17.99	18.677
21	3.844	7.851	11.278	9.489
22	127.652	76.042	90.028	132.989
23	204.151	207.901	>300	143.268
24	39.983	56.587	77.448	77.129
25	32.329	56.916	76.232	62.605
26	>300	39.106	39.657	32.401
27	294.944	37.343	36.291	30.194
28	293.779	37.82	38.06	30.168
29	5.425	5.62	13.088	6.356
30	15.778	6.137	6.431	5.422
31	15.152	36.418	35.636	19.792
32	>300	296.73	224.94	72.01
33	135.662	291.935	>300	279.031
34	28.615	6.666	8.519	14.918
35	200.41	>300	>300	>300

Table 10. CPU time for variants of M^{PB}

Instance	**CPU Time (in seconds)**			
	M^{PB}	$\mathrm{M}^{\mathrm{UV-P}}$	$\mathrm{M}_{\mathrm{R}}^{\mathrm{UV-P}}$	$\mathrm{M}_{\mathrm{R,T}}^{\mathrm{UV-P}}$
6	3.835	9.23	5.509	5.751
7	>300	>300	>300	>300
8	4.744	4.261	9.277	6.155
9	>300	>300	>300	>300
10	>300	>300	>300	>300
11	0.975	5.151	9.871	2.113
12	116.591	41.741	8.729	19.012
13	0.965	1.946	8.383	7.959
14	69.524	27.557	67.814	44.317
15	73.422	196.148	82.157	105.66
16	1.862	9.251	8.838	5.027
17	49.268	24.718	19.033	37.207
18	>300	>300	>300	>300
19	63.105	202.299	>300	12.159
20	16.953	16.169	7.275	4.962
21	16.069	7.014	6.285	8.077
22	9.807	>300	12.442	8.356
23	141.06	231.052	>300	11.388
24	20.357	16.419	15.859	15.858
25	8.713	10.417	10.85	11.541
26	>300	>300	>300	>300
27	>300	>300	>300	>300
28	>300	>300	>300	>300
29	13.085	27.595	3.973	12.532
30	11.813	7.493	6.871	6.749
31	41.358	229.15	>300	83.994
32	3.082	3.564	3.52	50.9
33	48.323	199.432	259.681	12.952
34	3.835	9.23	5.509	5.751
35	>300	>300	>300	>300

Appendix E. MILP models

E.1. Discretized model

We introduce an index $m \in \mathbf{M}$ for the discrete values δ that R_{jkt} can assume, and define a set $\mathbf{D}^{\mathrm{R}}$ for these values: $R_{jkt} \in \mathbf{D}^{\mathrm{R}} = \{\delta_1, \dots, \delta_{|\mathbf{M}|}\}$ with $\delta_m < \delta_{m+1}$. We introduce variable Z_{jktm} to model the selection of the discrete value for R_{jkt}:

$$\sum_{m \in \mathbf{M}} Z_{jktm} = 1, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{71}$$

Split fraction R_{jkt} is calculated as:

$$R_{jkt} = \sum_{m \in \mathbf{M}} \delta_m Z_{jktm}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{72}$$

We introduce continuous variable I^{D}_{ijktm}:

$$I_{ijt} = \sum\nolimits_{m\in\mathbf{M}} I^{\mathrm{D}}_{ijktm}\,, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{73}$$

$$\sum\nolimits_{i\in\mathbf{I}} I^{\mathrm{D}}_{ijktm} \le \gamma_j Z_{jktm}, \qquad j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{74}$$

Flow from blender to product, $\hat{F}_{ijkt}$, now becomes:

$$\hat{F}_{ijkt} = \sum\nolimits_{m\in\mathbf{M}} \delta_m I^{\mathrm{D}}_{ijktm}\,, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{75}$$

Eqn. (1) – (4), (6) – (13) and (71)– (75) comprise ${}_{\mathrm{L1}}\mathrm{M}^{\mathrm{SB}}$.

For M^{UV}, we have:

$$V_{ijkt} = \sum\nolimits_{m\in\mathbf{M}} I^{\mathrm{D}}_{ijktm}\,, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{76}$$

$$I^{\mathrm{D}}_{ijktm} \le \bar{\gamma}_{ijk} Z_{jktm}, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, m\in\mathbf{M}, t\in\mathbf{T} \tag{77}$$

Eqn. (1) – (4), (6) – (13) and (71) – (72) and (75) – (77) comprise ${}_{\mathrm{L1}}\mathrm{M}^{\mathrm{UV}}$. ${}_{\mathrm{L1}}\mathrm{M}^{\mathrm{UV}}_{\mathrm{R,T}}$ contains Eqn. (50), (52) - (53), and all constraints in ${}_{\mathrm{L1}}\mathrm{M}^{\mathrm{UV}}$.

E.2. Discretized – relaxed model

We introduce two positive continuous variables: Z^{+}_{jktm} and Z^{-}_{jktm}. We have:

$$Z_{jktm} = Z^{+}_{jktm} + Z^{-}_{jktm}, \qquad j\in\mathbf{J}, k\in\mathbf{K}, m\in\mathbf{M}, t\in\mathbf{T} \tag{78}$$

while Z_{jktm} satisfies:

$$\sum\nolimits_{m\in\mathbf{M}\setminus\{|\mathbf{M}|\}} Z_{jktm} = 1, \qquad j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{79}$$

Split fraction R_{jkt} is now calculated as:

$$R_{jkt} = \sum\nolimits_{m\in\mathbf{M}\setminus\{|\mathbf{M}|\}} \delta_m Z_{jktm} + \sum\nolimits_{m\in\mathbf{M}\setminus\{|\mathbf{M}|\}} (\delta_{m+1}-\delta_m) Z^{+}_{jktm}\,, \qquad j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{80}$$

Similarly for I^{D}_{ijktm} we define positive continuous variables $I^{\mathrm{D}+}_{ijktm}$ and $I^{\mathrm{D}-}_{ijktm}$. We have:

$$I^{\mathrm{D}}_{ijktm} = I^{\mathrm{D}+}_{ijktm} + I^{\mathrm{D}-}_{ijktm}, \qquad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, m\in\mathbf{M}, t\in\mathbf{T} \tag{81}$$

For $\hat{F}_{ijkt}$ we now have:

$$\hat{F}_{ijkt} = \sum\nolimits_{m\in\mathbf{M}\setminus\{|\mathbf{M}|\}} \delta_m I^{\mathrm{D}}_{ijktm} + \sum\nolimits_{m\in\mathbf{M}\setminus\{|\mathbf{M}|\}} \bar{\gamma}_{ijk}(\delta_{m+1}-\delta_m) I^{\mathrm{D}+}_{ijkt|\mathbf{M}|}\,, \quad i\in\mathbf{I}, j\in\mathbf{J}, k\in\mathbf{K}, t\in\mathbf{T} \tag{82}$$

Eqn. (1) – (4), (6) – (13), (56) – (58), and (61) – (65) comprise ${}_{L2}M^{SB}$. Eqn. (1) – (4), (6) – (13), (75) – (82) comprise ${}_{L2}M^{UV}$. ${}_{L2}M^{UV}_{R,T}$ contains Eqn. (50), (52) - (53), and all constraints in ${}_{L2}M^{UV}$.

Appendix F. CPU time and objective function value for MILP models

Table 11. CPU time for variants of ${}_{L1}M$

Instance	CPU Time (in seconds)		
	${}_{L1}M^{SB}$	${}_{L1}M^{SB}$	${}_{L1}M^{SB}_{R,T}$
36	>300	>300	190
37	270	>300	196
38	>300	>300	>300
39	>300	>300	>300
40	117	71	48
41	4	20	55
42	33	216	227
43	10	9	8
44	26	26	25
45	110	124	69
46	>300	>300	>300
47	>300	>300	>300
48	>300	>300	>300
49	>300	>300	>300
50	54	88	93
51	>300	>300	>300
52	>300	>300	>300
53	>300	>300	>300
54	>300	>300	>300
55	>300	>300	>300

Table 12. CPU time for variants of ${}_{L2}M$

Instance	CPU Time (in seconds)		
	${}_{L2}M^{SB}$	${}_{L2}M^{SB}$	${}_{L2}M^{SB}_{R,T}$
36	4	4	6
37	30	15	45
38	>300	108	260
39	7	82	91
40	120	12	13
41	1	1	1
42	1	4	4
43	1	1	1
44	2	2	3
45	3	4	4
46	77	7	2
47	81	3	2
48	>300	4	2
49	>300	4	2

50	23	1	1
51	12	42	39
52	77	28	18
53	>300	104	16
54	5	5	7
55	>300	>300	108

Table 13. Objective function value of ${}_{L1}M$ and ${}_{L2}M$

Instance	Obj. function value	
	${}_{L1}M$	${}_{L2}M$
36	743.9	760.7
37	765.4	766.25
38	3332.0(3357.1)	3335
39	2202.8(2212.5)	2204.13
40	514.32	517.78
41	574.78	574.78
42	3347.5	3448.7
43	2057.1	2127.2
44	2127.36	2129.13
45	3180.1	3203.7
46	804.8(806.2)	806.2
47	3245.8(3331.5)	3329.4
48	804.8(806.2)	806.2
49	805.1(806.1)	806.2
50	805.1(806.3)	806.2
51	22799.5(22824.8)	22850.6
52	28619.4(28646.4)	28726.4
53	0(large)	28905.8
54	8813.9(8828.0)	8851
55	0(25766.7)	25766.7

Note: for ${}_{L1}M$, we report the best objective function value found in all three variants, and the best possible objective function value in the parentheses if all three variants cannot solve the instance in 300 seconds.

Appendix G. Decomposition method

We show the decomposition method proposed in (Lotero et al. 2016).

G.1. Relaxed problem

A binary variable Y_{jt} is introduced, which equals 1 if blender *j* feeds products at time point *t*. We have:

$$\sum_{i\in\mathbf{I}} F_{ijt} \leq (1 - Y_{jt})\gamma_j, \qquad j \in \mathbf{J}, t \in \mathbf{T} \tag{83}$$

$$X_{jkt} \leq Y_{jt}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{84}$$

$$\sum_{k\in\mathbf{K}} X_{jkt} \geq Y_{jt}, \qquad j \in \mathbf{J}, t \in \mathbf{T} \tag{85}$$

which are derived from the operating rule that blender feeding and withdrawing cannot occur simultaneously and the definition of Y_{jt}.

Recall that we have $\hat{F}_{ijkt} = V_{ijkt}R_{jkt}$, which is relaxed using McCormick envelopes:

$$\hat{F}_{ijkt} \geq V_{ijkt} + \bar{\gamma}_{ijk}R_{jkt} - \bar{\gamma}_{ijk}, \quad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{86}$$

$$\hat{F}_{ijkt} \leq V_{ijkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{87}$$

$$\hat{F}_{ijkt} \leq \bar{\gamma}_{ijk}R_{jkt}, \qquad i \in \mathbf{I}, j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T} \tag{88}$$

We also have the following optimality cut (Eqn. (89)) and feasibility cut (Eqn. (90)) to be added:

$$Z \leq -(UB - Z_n)\left(\sum_{(j,t):Y^*_{jtn}=1} Y_{jt} - \sum_{(j,t):Y^*_{jtn}=0} Y_{jt}\right) + (UB - Z_n)\left(\sum_{(j,t)} Y^*_{jtn} - 1\right) + UB, \quad n \in \widetilde{\mathbf{N}} \tag{89}$$

$$\sum_{(j,t):Y^*_{jtn}=1} (1 - Y_{jt}) + \sum_{(j,t):Y^*_{jtn}=0} Y_{jt} \geq 1, \qquad n \in \widehat{\mathbf{N}} \tag{90}$$

where n is the iteration index, $\widetilde{\mathbf{N}}$ denotes the set of iterations where the reduced problem (to be introduced later) is feasible, and $\widehat{\mathbf{N}}$ denotes the set of iterations where the reduced problem is infeasible. Z is the value of the objective function, UB is the global upper bound for the objective function, and Z_n (a parameter) is the best possible value of the objective function at iteration n after solving the relaxed problem.

Eqn. (1) – (4), (6) – (10), (12) – (16), (18), (83) – (90) comprise the relaxed problem, which is an MILP.

G.2. Reduced problem

After solving the relaxed problem, variable Y_{jt} is fixed to Y^*_{jtn}. Eqn. (83) – (85) become:

$$\sum_{i\in\mathbf{I}} F_{ijt} \leq \left(1 - Y^*_{jtn}\right)\gamma_j, \qquad j \in \mathbf{J}, t \in \mathbf{T}, n = n^* \tag{91}$$

$$X_{jkt} \leq Y^*_{jtn}, \qquad j \in \mathbf{J}, k \in \mathbf{K}, t \in \mathbf{T}, n = n^* \tag{92}$$

$$\sum_{k\in\mathbf{K}} X_{jkt} \geq Y^*_{jtn}, \qquad j \in \mathbf{J}, t \in \mathbf{T}, n = n^* \tag{93}$$

where n^* denotes the current iteration.

Eqn. (1) – (4), (6) – (10), (12) – (18), (91) – (93) comprise the relaxed problem, which is an MINLP.

G.3. Workflow for the decomposition method

The flowchart for the decomposition method is shown in Figure 8 where Z^* denotes the optimal solution to the reduced problem at the current iteration. More details for the workflow can be found in (Lotero et al. 2016).

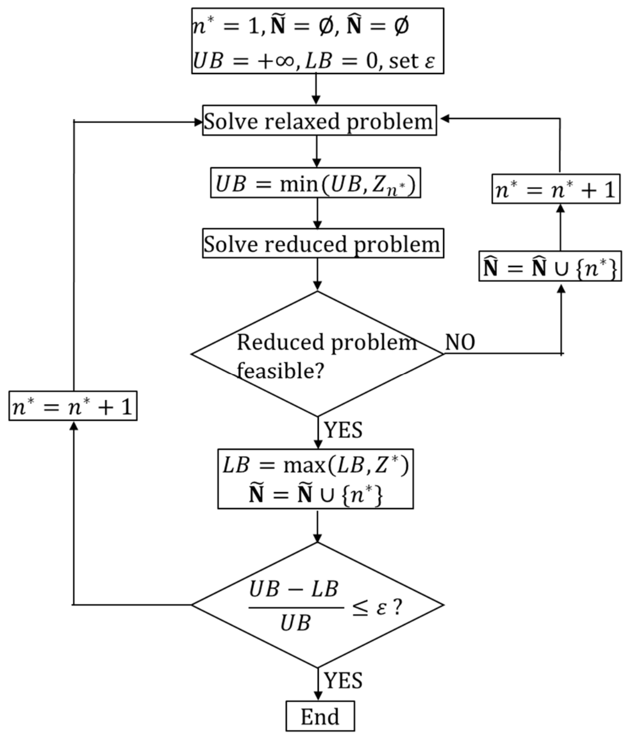

Figure 8. Flowchart for the decomposition method

References

Achterberg, Tobias. 2007. "Constraint Integer Programming." Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften. https://depositonce.tu-berlin.de/handle/11303/1931.

Achterberg, Tobias, Robert E. Bixby, Zonghao Gu, Edward Rothberg, and Dieter Weninger. 2020. "Presolve Reductions in Mixed Integer Programming." *INFORMS Journal on Computing* 32 (2): 473–506. https://doi.org/10.1287/ijoc.2018.0857.

Adhya, Nilanjan, Mohit Tawarmalani, and Nikolaos V. Sahinidis. 1999. "A Lagrangian Approach to the Pooling Problem." *Industrial and Engineering Chemistry Research* 38 (5): 1956–72. https://doi.org/10.1021/ie980666q.

Alfaki, Mohammed, and Dag Haugland. 2013. "Strong Formulations for the Pooling Problem." In *Journal of Global Optimization*, 56:897–916. Springer. https://doi.org/10.1007/s10898-012-9875-6.

Bagajewicz, Miguel. 2000. "A Review of Recent Design Procedures for Water Networks in Refineries and Process Plants." *Computers & Chemical Engineering* 24 (9–10): 2093–2113. https://doi.org/10.1016/S0098-1354(00)00579-2.

Belotti, Pietro. 2013. "Bound Reduction Using Pairs of Linear Inequalities." *Journal of Global Optimization* 56 (3): 787–819. https://doi.org/10.1007/s10898-012-9848-9.

Belotti, Pietro, Jon Lee, Leo Liberti, François Margot, and Andreas Wächter. 2009. "Branching and Bounds Tighteningtechniques for Non-Convex MINLP." *Optimization Methods and Software* 24 (4–5): 597–634. https://doi.org/10.1080/10556780903087124.

Ben-Tal, Aharon, Gideon Eiger, and Vladimir Gershovitz. 1994. "Global Minimization by Reducing the Duality Gap." *Mathematical Programming* 63 (1–3): 193–212. https://doi.org/10.1007/BF01582066.

Blom, Michelle L., Christina N. Burt, Adrian R. Pearce, and Peter J. Stuckey. 2014. "A Decomposition-Based Heuristic for Collaborative Scheduling in a Network of Open-Pit Mines." *INFORMS Journal on Computing* 26 (4): 658–76. https://doi.org/10.1287/ijoc.2013.0590.

Blom, Michelle L., Adrian R. Pearce, and Peter J. Stuckey. 2016. "A Decomposition-Based Algorithm for the Scheduling of Open-Pit Networks Over Multiple Time Periods." *Management Science* 62 (10): 3059–84. https://doi.org/10.1287/mnsc.2015.2284.

Boland, Natashia, Thomas Kalinowski, and Fabian Rigterink. 2016. "New Multi-Commodity Flow Formulations for the Pooling Problem." *Journal of Global Optimization* 66 (4): 669–710. https://doi.org/10.1007/s10898-016-0404-x.

Boland, Natashia, Thomas Kalinowski, Fabian Rigterink, and Martin Savelsbergh. 2016. "A Special Case of the Generalized Pooling Problem Arising in the Mining Industry." http://www.optimization-online.org/DB_FILE/2015/07/5025.pdf.

Castillo, Pedro A Castillo, and Vladimir Mahalec. 2014. "Inventory Pinch Based, Multiscale Models for Integrated Planning and Scheduling-Part II: Gasoline Blend Scheduling." *AIChE Journal* 60 (6): 2158–78. https://doi.org/10.1002/aic.14444.

Castro, Pedro M. 2016. "Source-Based Discrete and Continuous-Time Formulations for the Crude Oil Pooling Problem." *Computers and Chemical Engineering* 93 (4): 382–401. https://doi.org/10.1016/j.compchemeng.2016.06.016.

Castro, Pedro M., and Ignacio E. Grossmann. 2014. "Global Optimal Scheduling of Crude Oil Blending Operations with RTN Continuous-Time and Multiparametric Disaggregation." *Industrial and Engineering Chemistry Research*. https://doi.org/10.1021/ie503002k.

Chen, Yifu, and Christos T. Maravelias. 2020. "Preprocessing Algorithm and Tightening Constraints for Multiperiod Blend Scheduling: Cost Minimization." *Journal of Global Optimization* 77 (3): 603–25. https://doi.org/10.1007/s10898-020-00882-3.

D'Ambrosio, Claudia, Jeff Linderoth, and James Luedtke. 2011. "Valid Inequalities for the Pooling Problem with Binary Variables." In *Integer Programming and Combinatoral Optimization*, edited by Oktay Günlük and Gerhard J Woeginger, 117–29. Berlin, Heidelberg: Springer Berlin Heidelberg.

Domes, Ferenc, and Arnold Neumaier. 2016. "Constraint Aggregation for Rigorous Global Optimization." *Mathematical Programming* 155 (1–2): 375–401. https://doi.org/10.1007/s10107-014-0851-4.

Gleixner, Ambros M., Timo Berthold, Benjamin Müller, and Stefan Weltge. 2017. "Three Enhancements for Optimization-Based Bound Tightening." *Journal of Global Optimization* 67 (4): 731–57. https://doi.org/10.1007/s10898-016-0450-4.

Gounaris, Chrysanthos E., Ruth Misener, and Christodoulos A. Floudas. 2009. "Computational Comparison of Piecewise−Linear Relaxations for Pooling Problems." *Industrial & Engineering Chemistry Research* 48 (12): 5742–66. https://doi.org/10.1021/ie8016048.

Gupte, Akshay, Shabbir Ahmed, Santanu S. Dey, and Myun Seok Cheon. 2017. "Relaxations and Discretizations for the Pooling Problem." *Journal of Global Optimization* 67 (3): 631–69. https://doi.org/10.1007/s10898-016-0434-4.

Haverly, C. A. 1978. "Studies of the Behavior of Recursion for the Pooling Problem." *ACM SIGMAP Bulletin*, no. 25 (December): 19–28. https://doi.org/10.1145/1111237.1111238.

Jeżowski, Jacek. 2010. "Review of Water Network Design Methods with Literature Annotations." *Industrial & Engineering Chemistry Research* 49 (10): 4475–4516. https://doi.org/10.1021/ie901632w.

Kolodziej, Scott P., Pedro M. Castro, and Ignacio E. Grossmann. 2013. "Global Optimization of Bilinear Programs with a Multiparametric Disaggregation Technique." *Journal of Global Optimization* 57 (4): 1039–63. https://doi.org/10.1007/s10898-012-0022-1.

Kolodziej, Scott P., Ignacio E. Grossmann, Kevin C. Furman, and Nicolas W. Sawaya. 2013. "A Discretization-Based Approach for the Optimization of the Multiperiod Blend Scheduling Problem." *Computers and Chemical Engineering* 53: 122–42. https://doi.org/10.1016/j.compchemeng.2013.01.016.

Li, Jie, and I. A. Karimi. 2011. "Scheduling Gasoline Blending Operations from Recipe Determination to Shipping Using Unit Slots." *Industrial and Engineering Chemistry Research*. https://doi.org/10.1021/ie102321b.

Li, Jie, Wenkai Li, I. A. Karimi, and Rajagopalan Srinivasan. 2007. "Improving the Robustness and Efficiency of Crude Scheduling Algorithms." *AIChE Journal* 53 (10): 2659–80. https://doi.org/10.1002/aic.11280.

Li, Jie, Ruth Misener, and Christodoulos A. Floudas. 2012. "Continuous-Time Modeling and Global Optimization Approach for Scheduling of Crude Oil Operations." *AIChE Journal* 58 (1): 205–26. https://doi.org/10.1002/aic.12623.

Li, Jie, Xin Xiao, and Christodoulos A. Floudas. 2016. "Integrated Gasoline Blending and Order Delivery Operations: Part I. Short-Term Scheduling and Global Optimization for Single and Multi-Period Operations." *AIChE Journal* 62 (6): 2043–70.

Lotero, Irene, Francisco Trespalacios, Ignacio E. Grossmann, Dimitri J. Papageorgiou, and Myun Seok Cheon. 2016. "An MILP-MINLP Decomposition Method for the Global Optimization of a Source Based Model of the Multiperiod Blending Problem." *Computers and Chemical Engineering*. https://doi.org/10.1016/j.compchemeng.2015.12.017.

Maranas, Costas D., and Christodoulos A. Floudas. 1997. “Global Optimization in Generalized Geometric Programming.” *Computers & Chemical Engineering* 21 (4): 351–69. https://doi.org/10.1016/S0098-1354(96)00282-7.

Misener, Ruth, and Christodoulos A. Floudas. 2012. “Global Optimization of Mixed-Integer Quadratically-Constrained Quadratic Programs (MIQCQP) through Piecewise-Linear and Edge-Concave Relaxations.” *Mathematical Programming* 136 (1): 155–82. https://doi.org/10.1007/s10107-012-0555-6.

Neiro, Sérgio M. S., Valéria V. Murata, and José M. Pinto. 2014. “Hybrid Time Formulation for Diesel Blending and Distribution Scheduling.” *Industrial & Engineering Chemistry Research* 53 (44): 17124–34. https://doi.org/10.1021/ie5009103.

Puranik, Yash, and Nikolaos V. Sahinidis. 2017. “Domain Reduction Techniques for Global NLP and MINLP Optimization.” *Constraints* 22 (3): 338–76. https://doi.org/10.1007/s10601-016-9267-5.

Quesada, I., and I.E. Grossmann. 1995. “Global Optimization of Bilinear Process Networks with Multicomponent Flows.” *Computers & Chemical Engineering* 19 (12): 1219–42. https://doi.org/10.1016/0098-1354(94)00123-5.

Reddy, P. Chandra Prakash, I. A. Karimi, and R. Srinivasan. 2004. “Novel Solution Approach for Optimizing Crude Oil Operations.” *AIChE Journal*. https://doi.org/10.1002/aic.10112.

Ryoo, Hong S., and Nikolaos V. Sahinidis. 1996. “A Branch-and-Reduce Approach to Global Optimization.” *Journal of Global Optimization* 8 (2): 107–38. https://doi.org/10.1007/BF00138689.

Savelsbergh, M. W. P. 1994. “Preprocessing and Probing Techniques for Mixed Integer Programming Problems.” *ORSA Journal on Computing* 6 (4): 445–54. https://doi.org/10.1287/ijoc.6.4.445.

Shectman, J. Parker, and Nikolaos V. Sahinidis. 1998. “A Finite Algorithm for Global Minimization of Separable Concave Programs.” *Journal of Global Optimization* 12 (1): 1–36. https://doi.org/10.1023/A:1008241411395.

Sherali, Hanif D., and Warren P. Adams. 1999. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Vol. 31. Nonconvex Optimization and Its Applications. Boston, MA: Springer US. https://doi.org/10.1007/978-1-4757-4388-3.

Smith, E.M.B., and C.C. Pantelides. 1999. “A Symbolic Reformulation/Spatial Branch-and-Bound Algorithm for the Global Optimisation of Nonconvex MINLPs.” *Computers & Chemical Engineering* 23 (4–5): 457–78. https://doi.org/10.1016/S0098-1354(98)00286-5.

Street, Larimer. 1989. “Constraint Propagation, Relational Arithmetic in AI Systems and Mathematical Programs.” *Annals of Operations Research* 21: 143–48.

Tawarmalani, Mohit., and Nikolaos V. Sahinidis. 2002. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming : Theory, Algorithms, Software, and Applications*. Kluwer Academic Publishers.

Wicaksono, Danan Suryo, and I. A. Karimi. 2008. “Piecewise MILP Under- and Overestimators for Global Optimization of Bilinear Programs.” *AIChE Journal* 54 (4): 991–1008. https://doi.org/10.1002/aic.11425.