AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 6(2) (2025) 121-142
https://doi.org/10.22060/AJMC.2024.22240.1142

Original Article

A new relaxation technique based on fractional representation to solve bilinear models: Application to the long horizon crude oil scheduling problem

Mohammad Nasir Yarahmadi[a], Seyed Ali MirHassani[a], Farnaz Hooshmand*[a]

[a]*Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Iran*

ABSTRACT: This paper proposes a novel relaxation technique based on the fractional representation of bilinear terms. This technique is embedded into an iterative two-step MILP-NLP algorithm based on piecewise relaxation and domain reduction strategies. To evaluate the performance of the algorithm, it is compared to the recently addressed iterative MILP-NLP algorithm based on piecewise McCormick relaxation techniques over a variety of instances. Our method is also applied to the crude oil scheduling problem as an application. The results confirm the efficiency of the proposed algorithm from both solution quality and running time.

Review History:

Received:02 March 2023
Revised:18 January 2024
Accepted:07 February 2024
Available Online:01 March 2025

Keywords:

Bilinear programming
Two-step MILP-NLP
McCormick relaxation
Fractional representation
Crude oil scheduling

MSC (2020):

90C06; 90C11; 90C30

1. Introduction

Non-convex mixed integer nonlinear programming (MINLP) models are frequently employed in engineering problems such as energy storage [10], ethanol supply chains [9], water networks [22], gas networks [28], and pricing [15].

A special case of non-convex MINLPs is the category of bilinear mixed integer programming (BLMIP) problems in which nonlinear terms appear as the product of two variables. BLMIPs become intractable by increasing the problem size, and hence, their direct resolution by available solvers would be very time-consuming. Therefore, different techniques have been proposed in the literature to tackle these problems. McCormick [27] proposed a relaxation technique to find lower bounds on the optimal objective value of BLMIPs. Foulds et al. [18] presented an approach based on convex relaxation within a spatial branch-and-bound framework. Bergamini et al. [1] addressed a logic-based outer approximation method based on piecewise under-estimators and upper-estimators of bilinear terms. Liberti et al. [26] and D'Ambrosio et al. [13] addressed mechanisms to approximate BLMIPs by mixed integer linear programming (MILP) models. Hasan and Karimi [20] concentrated on piecewise linear

*Corresponding author.
E-mail addresses: nasiryar@aut.ac.ir, a_mirhassani@aut.ac.ir, f.hooshmand.khaligh@aut.ac.ir

approximation via partitioning of variables and considered various questions such as how many and which variables should be partitioned, which partitioning strategy is more effective, where the grid points have to be placed, etc. Faria and Bagajewicz [16] developed an algorithm based on variable partitioning, bound tightening, and branch and bound. Gupte et al. [19] addressed a reformulation technique for BLMIPs in which bilinear terms represent the product of integer variables. Teles et al. [32, 33], Castro [4] approximated the BLMIP model by an MILP so that the domain of variables in each bilinear term is discretized according to base-2 or base-10 numeric representation system. Dey et al. [12] proposed a new second-order cone representation (SOCP) relaxation and branching rule for bipartite bilinear programming. Fischetti et al. [17] provided a branch-and-cut algorithm based on a new family of intersection cuts to find an optimal solution. Kleinert et al. [24] proposed outer approximation based on a cutting-plane algorithm to bilevel optimization problems including BLMIP.

Two-step MILP-NLP methods are amongst the well-known methods addressed in the literature to solve BLMIPs where MILP and nonlinear programming (NLP) models are solved iteratively. In these methods, the BLMIP model is approximated by an MILP via some techniques including piecewise linear approximation [21], piecewise McCormick relaxation [3], and multi-parametric disaggregation [6]. Then, the discrete variables of the original BLMIP are fixed at the solution obtained by MILPs to get an NLP model. By iteratively solving MILP and NLP models, the domain of variables is condensed, and finally, the optimal (or near-optimal) solution is returned by the algorithm. In this regard, recently, Nagarajan et al. [30] presented a two-step method in which the optimal solution of the problem is found by iteratively solving an MILP and an NLP model where the MILP model is constructed based on the piecewise McCormick relaxation technique.

In this paper, we present a novel relaxation technique that is based on the fractional representation of bilinear terms. Then, we embed it into a two-step MILP-NLP based heuristic algorithm to efficiently solve general BLMIPs. Afterward, the crude oil scheduling problem (COSP) addressed by De Assis et al. [11] is considered an application to evaluate the performance of our algorithm. COSP deals with a crude oil terminal in which different types of crude oil (in terms of quality) are delivered from vessels to be stored in tanks and then transferred to the refinery via a pipeline connecting the terminal to the refinery. The scheduling decisions in the terminal should be made so that operational restrictions are observed and the total cost is minimized. De Assis et al. [11] formulated this problem as a BLMIP model and solved it by a two-step MILP-NLP algorithm based on piecewise McCormick envelopes. In this paper, we attempt to improve the method of De Assis et al. [11] by utilizing our novel fractional-based relaxation technique.

The main contributions of this paper are as follows. First, a new relaxation technique that is based on the fractional representation of bilinear terms is introduced. Then, it is embedded into a two-step MILP-NLP based heuristic algorithm to solve a general bilinear program. We show that the algorithm can provide an approximated model with fewer added binary variables compared to other approaches such as piecewise McCormick relaxation [3]. Also, it can be very useful in some real-world problems with a certain structure of non-linear constraints in the form of fractional equality like COSP. The results confirm the superiority of the new method over the one proposed by De Assis et al. [11] to solve COSP from both solution quality and running time.

The rest of this paper is organized as follows: Section 2 introduces our novel relaxation technique and reviews piecewise McCormick relaxation to provide two-step MILP-NLP based heuristic algorithms. Section 3 describes COSP in more detail, reviews its relevant literature, and formulates it as a BLMIP; further, we provide a short overview of the approximate algorithm proposed by De Assis et al. [11] to solve COSP. Moreover, we describe how our method can be adopted for COSP. Computational results are provided in Section 4 to evaluate the performance of the proposed approach. Section 5 concludes the paper and offers directions for future research.

2. Main idea

In this section, first, our novel relaxation technique which is based on the fractional representation of the bilinear term is introduced and then, it is utilized in a two-step MILP-NLP based heuristic algorithm.

Consider a non-convex BLMIP where each bilinear term is a product of two nonnegative continuous variables in the form of $x_i x_j$. Suppose the domain of variable x_i is given as $[L_i, U_i]$. Without loss of generality, suppose that $L_i \geq 1$; if this is not the case, one can use the substitution $x'_i = (x_i - L_i) + 1$ to satisfy this condition. Now, define the set $\mathbb{B}$ as follows:

$$\mathbb{B} = \{(i, j) : i < j \text{ and the model BLMIP contains the bilinear term } x_i x_j\}$$

The main idea behind our relaxation technique is to replace the bilinear term $x_i x_j$ (for every $(i, j) \in \mathbb{B}$) with a new nonnegative continuous variable $w_{i,j}$, rewrite the constraint $w_{i,j} = x_i x_j$ as $\frac{1}{x_i} = \frac{x_j}{w_{i,j}}$, and then approximate each nonlinear fraction linearly. To put it in more detail, note that the fractions on both sides of the equation $\frac{1}{x_i} = \frac{x_j}{w_{i,j}}$ take their values in the interval $\left[\frac{1}{U_i}, \frac{1}{L_i}\right]$. Therefore, we denote the interval associated with $\frac{1}{x_i}$ by $[L'_i, U'_i]$, and initialize it as $L_i' := \frac{1}{U_i}$, and $U_i' := \frac{1}{L_i}$. We partition $[L_i', U_i']$ into some sub-intervals with equal lengths, and denote the set of sub-intervals by $\mathbb{P} = \{1, \ldots, P\}$ (indexed by p), where the p^{th} sub-interval associated with $\frac{1}{x_i}$ is as follows:

$$\left[L_{i,p}', U_{i,p}'\right] = \left[L_i' + (p-1) \times \frac{(U_i' - L_i')}{P}, L_i' + p \times \frac{(U_i' - L_i')}{P}\right], \quad p \in \mathbb{P} \tag{1}$$

Therefore:

$$\bigvee_{p \in \mathbb{P}} \left[\left(L_{i,p}' \leq \frac{1}{x_i} \leq U_{i,p}'\right) \wedge \left(L_{i,p}' \leq \frac{x_j}{w_{i,j}} \leq U_{i,p}'\right)\right]$$

Thus, we provide relaxation of the original model by substituting the non-linear constraint $\frac{1}{x_i} = \frac{x_j}{w_{i,j}}$ by the following linear constraints:

$$-M_1(1 - \gamma_{i,p}) + L_{i,p}' x_i \leq 1 \qquad \forall p \in \mathbb{P} \tag{2}$$

$$U_{i,p}' x_i + M_2(1 - \gamma_{i,p}) \geq 1 \qquad \forall p \in \mathbb{P} \tag{3}$$

$$-M_3(1 - \gamma_{i,p}) + L_{i,p}' w_{i,j} \leq x_j \qquad \forall p \in \mathbb{P} \tag{4}$$

$$U_{i,p}' w_{i,j} + M_4(1 - \gamma_{i,p}) \geq x_j \qquad \forall p \in \mathbb{P} \tag{5}$$

$$\sum_{p \in \mathbb{P}} \gamma_{i,p} = 1 \tag{6}$$

$$\gamma_{i,p} \in \{0, 1\} \qquad \forall p \in \mathbb{P} \tag{7}$$

Where $\gamma_{i,p}$ is a binary variable that is 1 if the p^{th} sub-interval associated with $\frac{1}{x_i}$ is selected; 0 otherwise. Additionally, M_1, M_2, M_3 and M_4 are sufficiently large positive numbers that can be set as follows:

$$M_1 = \max\left\{1, L_{i,P}' U_i - 1\right\}, \qquad M_2 = \max\left\{1, 1 - U_{i,1}' L_i\right\}$$
$$M_3 = \max\left\{1, L_{i,P}' U_i U_j - L_j\right\}, \quad M_4 = \max\left\{1, U_j - U_{i,1}' L_i L_j\right\}$$

In the rest of the paper, to use our relaxation technique, we set $M = \max\{M_1, M_2, M_3, M_4\}$. We also refer to the model obtained by replacing the nonlinear constraint $\frac{1}{x_i} = \frac{x_j}{w_{i,j}}$ with linear constraints (2)-(7) as a piecewise partitioning model based on fractional relaxation (FRPPM for short). As the value of P (the number of sub-intervals) increases, the optimal solution obtained by FRPPM becomes closer to the optimal solution of original BLMIP, However, it is important to note that the solution time also increases due to the presence of a larger number of binary variables and the utilization of additional constraints.

Now, we utilize FRPPM in a two-step MILP-NLP algorithm in which the BLMIP model is approximated by the MILP model FRPPM. Then, the solution of FRPPM is used to fix the discrete variables of the original BLMIP to get an NLP model. By iteratively solving MILP and NLP models, the domain of variables is condensed, and finally, the optimal (or nearoptimal) solution is returned. We refer to this algorithm as a piecewise partitioning algorithm based on fractional relaxation (FRPPA for short). The general framework of FRPPA is as follows:

FRPPA

Step 0: Let k be a counter, $\varepsilon > 0$ be a given accuracy, and $Ctrl$ be a binary parameter that is 1 if the stopping criterion is observed, 0 otherwise. Let $[L_i', U_i']$ be the interval related to the fraction $\frac{1}{x_i}$ and initialize it as $\left[\frac{1}{U_i}, \frac{1}{L_i}\right]$. Determine the value of P as the number of sub-intervals associated with fractions, and initialize the sub-intervals based on (1). Assume that $z_{UB}^{(k)}$ denotes the objective value of the best feasible solution to BLMIP found until iteration k, and consider $z_{LB}^{(k)}$ as the objective function value of FRPPM solved in iteration k. Initialize $k := 1, Ctrl := 0$.

Step 1: While $Ctrl = 0$ do

Step 1-1: Solve the model FRPPM, and denote its optimal objective function value by $z_{LB}^{(k)}$.

Step 1-2: Fix the vector of binary variables of the model BLMIP to the optimal solution of FRPPM to get an NLP model. Denote the optimal objective function value of the NLP model by z_{NLP}^*, and set $z_{UB}^{(k)} = \min\left\{z_{NLP}^*, z_{UB}^{(k-1)}\right\}$.

Step 1-3: For $i = 1, \ldots, n_1$ do:

Step 1-3-1: Let p_0 be the index of the sub-interval containing the value of $\frac{1}{x_i}$ in the optimal solution to FRPPM. Additionally, let p_1 be the index of the sub-interval containing the value of $\frac{1}{x_i}$ in the optimal solution to the NLP model introduced in Step 1-2. Reduce the interval length associated with the fraction $\frac{1}{x_i}$ as follows:

$$L'_i = \min\left\{L'_{i,p_0}, L'_{i,p_1}\right\}, \quad U'_i = \max\left\{U'_{i,p_0}, U'_{i,p_1}\right\}$$

Step 1-4: If $z_{UB}^{(k)} - z_{LB}^{(k)} < \varepsilon$, set $Ctrl := 1$; else set $k := k+1$, and update the sub-intervals based on (1) and reduced bound.

Step 2: Return the solution associated with $z_{UB}^{(k)}$ as the best feasible solution found by the algorithm.

One of the well-known techniques to obtain an approximated model of MINLP is tightening piecewise McCormick relaxation [3] which was used by De Assis et al. [11]. He provides a two-step MILP-NLP algorithm (for short MRPPA) to solve COSP as a non-convex MINLP.

To explain MRPPA, consider BLMIP with bilinear terms in the form of $x_i x_j$ such that $(x_i, x_j) \in [L_i, U_i] \times [L_j, U_j]$. Firstly, $[L_i, U_i]$ and $[L_j, U_j]$ are partitioned into some sub-intervals with equal lengths which are denoted by $\mathbb{O} = \{1, \ldots, O\}$ (indexed by o) and $\mathbb{S} = \{1, \ldots, S\}$ (indexed by s) respectively as follows:

$$\left[\underline{SI}x_{i,o}, \overline{SI}x_{i,o}\right] = \left[L_i + (o-1) \times \frac{(U_i - L_i)}{O}, L_i + o \times \frac{(U_i - L_i)}{O}\right] \quad o \in \mathbb{O} \tag{8}$$

$$\left[\underline{SI}x_{j,s}, \overline{SI}x_{j,s}\right] = \left[L_j + (s-1) \times \frac{(U_j - L_j)}{S}, L_j + s \times \frac{(U_j - L_j)}{S}\right] \quad s \in \mathbb{S} \tag{9}$$

Therefore:

$$\bigvee_{o,s \in \mathbb{O} \times \mathbb{S}} \left[\left(\underline{SI}x_{i,o} \leq x_i \leq \overline{SI}x_{i,o}\right) \wedge \left(\underline{SI}x_{j,s} \leq x_j \leq \overline{SI}x_{j,s}\right)\right],$$

Therefore, piecewise McCormick relaxation of the original model by substituting the nonlinear constraint $w_{ij} = x_i x_j$ by the following linear constraints:

$$w_{ij} \geq \underline{SI}x_{i,o}x_j + \underline{\underline{SI}}x_{j,s}x_i - \underline{SI}x_{i,o}\underline{SI}x_{j,s} - M'\left(1 - \alpha_{i,j,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} \tag{10}$$

$$w_{ij} \geq \overline{SI}x_{i,o}x_j + \overline{\overline{SI}}x_{j,s}x_i - \overline{\overline{SI}}x_{i,o}\overline{\overline{SI}}x_{j,s} - M'\left(1 - \alpha_{i,j,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} \tag{11}$$

$$w_{ij} \leq \underline{SI}x_{i,o}x_j + \overline{SI}x_{j,s}x_i - \underline{SI}x_{i,o}\overline{SI}x_{j,s} + M'\left(1 - \alpha_{i,j,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} \tag{12}$$

$$w_{ij} \leq \overline{SI}x_{i,o}x_j + \underline{SI}x_{j,s}x_i - \overline{SI}x_{i,o}\underline{SI}x_{j,s} + M'\left(1 - \alpha_{i,j,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} \tag{13}$$

$$\sum_{o \in \mathbb{O}} \sum_{s \in \mathbb{S}} \alpha_{i,j,o,s} = 1 \tag{14}$$

$$\alpha_{i,j,o,s} \in \{0, 1\}, \quad \forall o \in \mathbb{O}, s \in \mathbb{S} \tag{15}$$

Where $\alpha_{i,j,o,s}$ is a binary variable that is 1 if the o^{th} sub-interval associated with x_i and s^{th} subinterval associated with x_j are selected; 0 otherwise. Also, M' are sufficiently large positive numbers that can be set $U_i U_j - L_i L_j$. (In the rest of the paper, M' is adjusted similarly) The MILP model obtained by using constraints (2)-(7) is referred to as a piecewise partitioning model based on McCormick relaxation (for short MRPPM). The general framework of MRPPA is as follows:

MRPPA

Step 0: Let k be a counter, $\varepsilon > 0$ be a given accuracy, and $Ctrl$ be a binary parameter that is 1 if the stopping criterion is observed; 0 otherwise. Let $\left[\widetilde{L}_i, \widetilde{U}_i\right]$ and $\left[\widetilde{L}_j, \widetilde{U}_j\right]$ be the intervals related to x_i and x_j, respectively. Initialize it as $\left[\tilde{L}_i, \widetilde{U}_i\right] = [L_i, U_i]$ and $\left[\tilde{L}_j, \widetilde{U}_j\right] = [L_j, U_j]$. Determine the values of O and S as the number of sub-intervals associated with x_i and x_j, respectively. Then initialize the sub-intervals based on (8) and (9). Assume that $z_{UB}^{(k)}$ denotes the objective value of the best feasible solution to BLMIP found until iteration k, and consider $z_{LB}^{(k)}$ as the objective function value of MRPPM solved in iteration k. Initialize $k := 1, Ctrl := 0$.

Step 1: While Ctrl = 0 do

Step 1-1: Solve the model MRPPM, and represent its optimal objective function value by $z_{LB}^{(k)}$.

Step 1-2: Fix the vector of binary variables of the model BLMIP to the optimal solution of MRPPM to get an NLP model. Denote the optimal objective function value of the NLP model by z_{NLP}^*, and set $z_{UB}^{(k)} = \min\left\{z_{NLP}^*, z_{UB}^{(k-1)}\right\}$.

Step 1-3: For $i = 1, \ldots, n_1$ do:

Step 1-3-1: Let o_0 and s_0 be the indices of the sub-intervals containing the value of x_i and x_j in the optimal solution to MRPPM solved in Step 1-1, respectively. Moreover, let o_1 and s_1 be the indices of the sub-intervals containing the value of x_i and x_j in the optimal solution to the NLP solved in Step 1-2, respectively. Reduce the length of the interval associated with variables x_i and x_j as follows:

$$\tilde{L}_i = \min\left\{\underline{SI}x_{i,o_0}, \underline{SI}x_{i,o_1}\right\}, \quad \widetilde{U}_i = \max\left\{\overline{SI}x_{i,o_0}, \overline{SI}x_{i,o_1}\right\}$$
$$\tilde{L}_j = \min\left\{\underline{SI}x_{j,s_0}, \underline{SI}x_{j,s_1}\right\}, \quad \widetilde{U}_j = \max\left\{\overline{SI}x_{j,s_0}, \overline{SI}x_{j,s_1}\right\}$$

Step 1-4: If $z_{UB}^{(k)} - z_{LB}^{(k)} < \varepsilon$, set $Ctrl := 1$; else set $k := k+1$, and update the sub-intervals based on (8) and (9).

Step 2: Return the solution associated with $z_{UB}^{(k)}$ as the best feasible solution found by the algorithm.

The following example illustrates FRPPA and MRPPA in a simple instance.

3. Illustrative example

Consider the following BLMIP:

$$\begin{aligned}
(M1): \min\ & z = 2\delta_1 + 3\delta_2 + 4x_1 + 3x_2 \\
\text{s.t. } & 3\delta_1 + 4\delta_2 + 2x_1x_2 + 2x_1 + 3x_2 \geq 14 \\
& \delta_1 + \delta_2 + x_1x_2 \geq 1 \\
& 1 \leq x_i \leq 2 \quad \forall i = 1, 2 && (16) \\
& \delta_i \in \{0, 1\},\ x_i \geq 0 \quad \forall i = 1, 2 && (17)
\end{aligned}$$

The optimal solution to M1 is as follows:

$$(\delta_1^*, \delta_2^*) = (1, 0), \quad (x_1^*, x_2^*) = (1.225, 1.633), \quad z^* = 11.798.$$

We substitute the bilinear term x_1x_2 by the nonnegative continuous variable $w_{1,2}$ and add the new constraint $w_{1,2} = x_1x_2$ to get the following equivalent model:

$$\begin{aligned}
(M2): \min\ & z = 2\delta_1 + 3\delta_2 + 4x_1 + 3x_2 \\
\text{s.t. } & (16) - (17) \\
& 3\delta_1 + 4\delta_2 + 2w_{1,2} + 2x_1 + 3x_2 \geq 14 && (18) \\
& \delta_1 + \delta_2 + w_{1,2} \geq 3 && (19) \\
& w_{1,2} = x_1x_2 && (20) \\
& w_{1,2} \geq 0 && (21)
\end{aligned}$$

In what follows, we describe how FRPPA is implemented on M2.
First, M2 is rewritten as the following equivalent model:

$$\begin{aligned}
(M3): \min\ & z = 2\delta_1 + 3\delta_2 + 4x_1 + 3x_2 && (22) \\
\text{s.t. } & (16) - (19), (21) && (23) \\
& \frac{1}{x_1} = \frac{x_2}{w_{1,2}} && (24)
\end{aligned}$$

The fraction $\frac{1}{x_1}$ takes its value in the interval $[0.5, 1]$, i.e. $L_1' = 0.5, U_1' = 1$. We set $P = 10$, and partition

$$[L_1', U_1']$$

into the following sub-intervals:

$$[L'_1, U'_1] = \bigcup_{p=1}^{10} [L'_{1,p}, U'_{1,p}] = [0.5, 0.55] \cup [0.55, 0.6] \cup \ldots \cup [0.95, 1]$$

Therefore, the following MILP model is constructed:

$$(M4): \min\ z = 2\delta_1 + 3\delta_2 + 4x_1 + 3x_2 \quad (25)$$

$$\text{s.t. } (16) - (19), (21) \quad (26)$$

$$-M(1-\gamma_{1,p}) + L'_{1,p}x_1 \le 1 \le U'_{1,p}x_1 + M(1-\gamma_{1,p}) \quad \forall p \in \mathbb{P} \quad (27)$$

$$-M(1-\gamma_{1,p}) + L'_{1,p}w_{1,2} \le x_2 \le U'_{1,p}w_{1,2} + M(1-\gamma_{1,p}) \quad \forall p \in \mathbb{P} \quad (28)$$

$$\sum_{p=1}^{10} \gamma_{1,p} = 1 \quad (29)$$

$$\gamma_{1,p} \in \{0,1\} \quad \forall p \in \mathbb{P} \quad (30)$$

The optimal solution to M4 is $\left(\hat{\delta}_1, \hat{\delta}_2\right) = (1, 0), \hat{x}_1 = 1.25, \hat{x}_2 = 1.5, \hat{z} = 11.5$ and hence, $z_{LB}^{(1)} = 11.5$. We have $\frac{1}{\hat{x}_1} = 1$; thus, $\frac{1}{\hat{x}_1}$ belongs to the seventh sub-interval, and we have $p_0 = 7$. If we solve the NLP model obtained by fixing (δ_1, δ_2) at $(1, 0)$, we get $\hat{x}_1 = 1.225$, $\hat{x}_2 = 1.633, \hat{z} = 11.798$, and hence, $z_{UB}^{(1)} = 11.798$. We have $\frac{1}{\hat{x}_1} = 0.816$; hence, $\frac{1}{\hat{x}_1}$ belongs to the seventh sub-interval, and we have $p_1 = 7$. Since the difference between $z_{LB}^{(1)}$ and $z_{UB}^{(1)}$ is not sufficiently small, the new interval associated with $\frac{1}{x_1}$ is determined as $L'_1 = \min\{0.8, 0.8\} = 0.8$ and $U'_1 = \max\{0.85, 0.85\} = 0.85$. So, the interval $[0.8, 0.85]$ is divided into ten subintervals with equal lengths as follows:

$$[L'_1, U'_1] = \cup_{p=1}^{10} [L'_{1,p}, U'_{1,p}] = [0.805, 0.81] \cup [0.81, 0.815] \cup \ldots \cup [0.845, 0.85]$$

The same process is repeated, and finally, the algorithm terminates after four iterations. Table 1 summarizes the results of each iteration.

Table 1: The results of implementing FRPPA on example M2

k	$[L'_1, U'_1]$	MILP (M4)					NLP			
		$\hat{x}_1$	$\hat{x}_2$	$\widehat{w}_{1,2}$	$[L'_{1,p_0}, U'_{1,p_0}]$	$z_{LB}^{(k)}$	$\hat{x}_1$	$\hat{x}_2$	$[L'_{1,p_1}, U'_{1,p_1}]$	$z_{UB}^{(k)}$
1	$[0.5, 1]$	1.250	1.500	2.0	$[0.80, 0.85]$	11.500	1.225	1.633	$[0.80, 0.85]$	11.798
2	$[0.80, 0.85]$	1.227	1.620	2.0	$[0.810, 0.815]$	11.768	1.225	1.633	$[0.815, 0.820]$	11.798
3	$[0.810, 0.820]$	1.225	1.630	2.0	$[0.816, 0.8165]$	11.792	1.225	1.633	$[0.8165, 0.8170]$	11.798
4	$[0.816, 0.817]$	1.225	1.633	2.0	$[0.8163, 0.8164]$	11.798	1.225	1.633	$[0.8163, 0.8164]$	11.798

Now, we describe how MRPPA is implemented on M2.

Both the variables x_1 and x_2 take their values in the interval [1,2], i.e. $\tilde{L}_1 = \tilde{L}_2 = 1$ and $\widetilde{U}_1 = \widetilde{U}_1 = 2$. We set $O = S = 5$, and partition $\left[\widetilde{L}_1, \widetilde{U}_1\right]$ and $\left[\widetilde{L}_2, \widetilde{U}_2\right]$ into the following sub-intervals:

$$\left[\widetilde{L}_1, \widetilde{U}_1\right] = \bigcup_{o=1}^{5} \left[\underline{SI}x_{1,o}, \overline{SI}x_{1,o}\right] = [1, 1.2] \cup [1.2, 1.4] \cup \ldots \cup [1.8, 1]$$

$$\left[\widetilde{L}_2, \widetilde{U}_2\right] = \bigcup_{s=1}^{5} \left[\underline{SI}x_{2,s}, \overline{SI}x_{2,s}\right] = [1, 1.2] \cup [1.2, 1.4] \cup \ldots \cup [1.8, 1]$$

Therefore, the following MILP model is constructed:

$$
\begin{aligned}
(M5): &\min\ z = 2\delta_1 + 3\delta_2 + 4x_1 + 3x_2 && (31)\\
&\text{s.t. } (16)-(19), (21) && (32)\\
&w_{1,2} \geq \underline{SI}x_{1,0}x_2 + \underline{SI}x_{2,s}x_1 - \underline{SI}x_{1,o}\underline{SI}x_{2,s} - M'\left(1-\alpha_{1,2,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} && (33)\\
&w_{1,2} \geq \overline{SI}x_{1,o}x_2 + \overline{SI}x_{2,s}x_1 - \overline{SI}x_{1,o}\overline{SI}x_{2,s} - M'\left(1-\alpha_{1,2,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} && (34)\\
&w_{12} \leq \underline{SI}x_{1,o}x_2 + \overline{SI}x_{2,s}x_1 - \underline{SI}x_{1,o}\overline{SI}x_{2,s} + M'\left(1-\alpha_{1,2,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} && (35)\\
&w_{12} \leq \overline{SI}x_{1,o}x_2 + \underline{SI}x_{2,s}x_1 - \overline{SI}x_{1,o}\underline{SI}x_{2,s} + M'\left(1-\alpha_{1,2,o,s}\right) \quad \forall o \in \mathbb{O}, s \in \mathbb{S} && (36)\\
&\sum_{o\in\mathbb{O}}\sum_{s\in\mathbb{S}} \alpha_{1,2,o,s} = 1 && (37)\\
&\alpha_{1,2,o,s} \in \{0,1\} \quad \forall o \in \mathbb{O}, s \in \mathbb{S} && (38)
\end{aligned}
$$

The optimal solution to M5 is $\left(\hat{\delta}_1, \hat{\delta}_2\right) = (1,0), \hat{x}_1 = 1.227, \hat{x}_2 = 1.627, \hat{z} = 11.787$ and hence, $z_{LB}^{(1)} = 11.787$. So, $\hat{x}_1$ and $\hat{x}_2$ belong to the second and fourth sub-interval, respectively. Therefore, we have $o_0 = 2$ and $s_0 = 4$. If we solve the NLP model obtained by fixing (δ_1, δ_2) at $(1,0)$, we get $\hat{x}_1 = 1.225, \hat{x}_2 = 1.633, \hat{z} = 11.798$, and hence, $z_{UB}^{(1)} = 11.798$. So, $\hat{x}_1$ and $\hat{x}_2$ belong to the second and fourth sub-interval, respectively. Therefore, we have $o_1 = 2$ and $s_1 = 4$. Since the difference between $z_{LB}^{(1)}$ and $z_{UB}^{(1)}$ is not sufficiently small, the new intervals associated with x_1 and x_2 are determined as follows:

$$
\begin{aligned}
\tilde{L}_1 &= \min\{1.2, 1.2\} = 1.2, \quad \widetilde{U}_1 = \max\{1.4, 1.4\} = 1.4\\
\tilde{L}_2 &= \min\{1.6, 1.6\} = 1.6, \quad \widetilde{U}_2 = \max\{1.8, 1.8\} = 1.8
\end{aligned}
$$

So, each interval is divided into five sub-intervals with equal lengths as follows:

$$
\begin{aligned}
\left[\tilde{L}_1, \widetilde{U}_1\right] &= \cup_{o=1}^{5}\left[\underline{SI}x_{1,o}, \overline{SI}x_{1,o}\right] = [1.2, 1.24] \cup [1.24, 1.28] \cup \ldots \cup [1.36, 1.4]\\
\left[\tilde{L}_2, \widetilde{U}_2\right] &= \cup_{s=1}^{5}\left[\underline{SI}x_{2,s}, \overline{SI}x_{2,s}\right] = [1.6, 1.64] \cup [1.64, 1.68] \cup \ldots \cup [1.76, 1.8]
\end{aligned}
$$

The same process is repeated, and finally, the algorithm terminates after three iterations. Table 2 summarizes the results of each iteration.

Table 2: The results of implementing MRPPA on example M2

k	$\begin{pmatrix}[\tilde{L}_1, \tilde{U}_1]\\ [\tilde{L}_2, \tilde{U}_2]\end{pmatrix}$	MILP (M5)				NLP		
		$\begin{pmatrix}\hat{x}_1\\ \hat{x}_2\end{pmatrix}$	$\hat{w}_{1,2}$	$\begin{pmatrix}[\underline{SI}x_{1,o_0}, \overline{SI}x_{1,o_0}]\\ [\underline{SI}x_{2,s_0}, \overline{SI}x_{2,s_0}]\end{pmatrix}$	$z_{LB}^{(k)}$	$\begin{pmatrix}\hat{x}_1\\ \hat{x}_2\end{pmatrix}$	$\begin{pmatrix}[\underline{SI}x_{1,o_1}, \overline{SI}x_{1,o_1}]\\ [\underline{SI}x_{2,s_1}, \overline{SI}x_{2,s_1}]\end{pmatrix}$	$z_{UB}^{(k)}$
1	$\begin{pmatrix}[1,2]\\ [1,2]\end{pmatrix}$	$\begin{pmatrix}1.227\\ 1.627\end{pmatrix}$	2.0	$\begin{pmatrix}[1.2, 1.4]\\ [1.6, 1.8]\end{pmatrix}$	11.500	$\begin{pmatrix}1.225\\ 1.633\end{pmatrix}$	$\begin{pmatrix}[1.2, 1.4]\\ [1.6, 1.8]\end{pmatrix}$	11.798
2	$\begin{pmatrix}[1.2, 1.4]\\ [1.6, 1.8]\end{pmatrix}$	$\begin{pmatrix}1.228\\ 1.628\end{pmatrix}$	2.0	$\begin{pmatrix}[1.2, 1.24]\\ [1.6, 1.64]\end{pmatrix}$	11.768	$\begin{pmatrix}1.225\\ 1.633\end{pmatrix}$	$\begin{pmatrix}[1.2, 1.24]\\ [1.6, 1.64]\end{pmatrix}$	11.798
3	$\begin{pmatrix}[1.2, 1.4]\\ [1.6, 1.64]\end{pmatrix}$	$\begin{pmatrix}1.225\\ 1.633\end{pmatrix}$	2.0	$\begin{pmatrix}[1.224, 1.232]\\ [1.632, 1.640]\end{pmatrix}$	11.792	$\begin{pmatrix}1.225\\ 1.633\end{pmatrix}$	$\begin{pmatrix}[1.224, 1.232]\\ [1.632, 1.640]\end{pmatrix}$	11.798

As can be seen, both FRPPA and MRPPA could find the optimal solution of M2, but in general, it is not a guarantee to find the optimal solution. Another notable point is the number of added binary variables that FRPPA and MRPPA used which are 10 and 25 respectively. To increase the solution quality of the approximated model, the number of binary variables in MRPPA (be influenced by $\alpha_{i,j,o,s}$) is higher than FRPPA (be influenced by $\gamma_{i,p}$).

Another type of constraint in problems that can better show the efficiency of FRPPA compared to MRPPA is as follows:

$$
\frac{x_i}{\sum_{i=1}^{n_1} x_i} = \frac{y_i}{\sum_{i=1}^{n_1} y_i}, \quad \forall i = 1, \ldots, n_1 \tag{39}
$$

Where x_i and y_i (for each i) are nonnegative variables. We substitute $\sum_{i=1}^{n_1} x_i$ and $\sum_{i=1}^{n_1} y_i$ by nonnegative variables x^{To} and y^{To}, respectively and rewrite constraints (39) as follows:

$$\frac{x_i}{x^{To}} = \frac{y_i}{y^{To}} \quad \forall i, \quad x^{To} = \sum_{i=1}^{n_1} x_i, \quad y^{To} = \sum_{i=1}^{n_1} y_i, \quad x^{To}, y^{To} \geq 0 \tag{40}$$

It is clear that both fractions $\frac{x_i}{x^{To}}$ and $\frac{y_i}{y^{To}}$ belong to $[L'_i, U'_i] = [0, 1]$. Therefore, to use MRPPA, we first provide the following partitioning:

$$[L'_{i,p}, U'_{i,p}] = \left[L'_i + (p-1) \times \frac{(U'_i - L'_i)}{P}, L'_i + p \times \frac{(U'_i - L'_i)}{P}\right], \quad \forall i, p \in \mathbb{P}$$

Then the non-linear constraint $\frac{x_i}{x^{To}} = \frac{y_i}{y^{To}}$ is substituted by the following constraints:

$$-M(1-\gamma_{i,p}) + L'_{i,p} x^{\text{total}} \leq x_i \leq U'_{i,p} x^{\text{total}} + M(1-\gamma_{i,p}) \, \forall i, p \in \mathbb{P} \tag{41}$$

$$-M(1-\gamma_{i,p}) + L'_{i,p} y^{\text{total}} \leq y_i \leq U'_{i,p} y^{\text{total}} + M(1-\gamma_{i,p}) \, \forall i, p \in \mathbb{P} \tag{42}$$

Now, to use MRPPA, we first replace the non-linear constraint $\frac{x_i}{x^{To}} = \frac{y_i}{y^{To}}$ with $x_i y^{To} = y_i x^{To}$ that both bilinear terms are substituted by the linear constraints similar to (2)-(7). As a result, by considering p partitions for domain of x_i, y_i, x^{To} and y^{To}, we need to $2n_1p^2$ additional binary variables in MRPPA, while this number is n_1p in FRPPA. Therefore, the growth rate of the number of added binary variables in MRPPA is more than FRPPA and can have a significant impact on running time and solution quality.

One of the real-world problems that contain the structure of constraints (39) is COSP. So, we adopt FRPPA to this problem and compare its effect with MRPPA.

Adopting FRPPA to solve COSP

The crude oil scheduling problem (COSP) is a well-known optimization problem that is usually formulated as a BLMIP model and has received great attention from researchers. For example, Lee et al. [25] presented a BLMIP model for refinery short-term scheduling of crude oil unloading and approximated it as an MILP. Karuppiah et al. [23] presented a BLMIP model to schedule crude oil movement at the front end of a petroleum refinery and solved it via an outer approximation algorithm. Mouret et al. [29] integrated refinery planning and crude oil operations scheduling problems as a BLMIP model and developed a Lagrangian decomposition algorithm to solve them. Chen et al. [8] addressed the COSP for a refinery that imports various types of crude oil from two terminals via bidirectional pipelines and presented a hierarchical decomposition method to solve the problem. Evazabadian et al. [14] developed a fuzzy stochastic model for the short-term COSP under preventive maintenance for charging tanks. Stanzani et al. [31] addressed the delivery of different types of crude oil from various offshore platforms to coastal terminals. Castro and Grossmann [5], Cerda et al. [7], and De Assis et al. [11] formulated the COSP as a BLMIP model and presented a two-step MILPNLP method to solve it.

In this section, after providing a detailed description and assumptions of COSP, the BLMIP model proposed by De Assis et al. [11] is repeated. Then, the iterative two-step MILP-NLP heuristic algorithm based on the piecewise McCormick relaxation technique which is presented by De Assis et al. [11] is considered as a base to be compared by our algorithm. Finally, our FRPPA, addressed in Section 2, is adopted to solve COSP.

3.1. COSP description

Consider a crude oil terminal in which different types of crude oil are delivered from vessels to be stored in tanks and then transferred to the refinery via a pipeline connecting the terminal to the refinery. The scheduling decisions in the terminal should be made so that operational restrictions are observed and the total cost is minimized. The following assumptions are made:

A1. The initial status of the crude oil in the pipeline and tanks is given.

A2. The planning horizon is divided into discrete periods. We refer to the period before the planning horizon as period 0.

A3. For each period, the volume of different types of crude oil delivered from vessels to the terminal and the refinery's demand for each crude oil is known in advance.

A4. The total capacity of each tank is given.

A5. A penalty is imposed on any difference between the crude oils injected into the pipeline and the refinery's demand.

A6. For each period, the maximum number of crude oil types that can be stored in a tank, the lower bound on the volume delivered from a vessel to a tank, and the upper bound on the total volume injected from tanks to the pipeline are known in advance.
A7. For maintenance requirements, each tank may be set to be out of service, full, or empty in certain periods.
A8. The number of tanks that can simultaneously feed the pipeline is limited.
A9. Some pairs of crude oils can cause a mixture and are not allowed to be in touch. Thus, they should not be stored in the same tank at the same time.
A10. In transferring crude oils from tanks to the refinery, the oil quality should be kept. In other words, in each period the concentration of the crude oil in the tank and the concentration of the one injected into the pipeline should be the same.
A11. In each period at most one of the inlet and outlet operations can be implemented on a given tank.
A12. When a tank receives crude oil from a vessel, outlet operations must wait at least one period for brine separation.

The main decisions include the determination of the amount of crude oil loaded into the tanks from the vessels during each period and the determination of the volume of oils injected from tanks to the pipeline to meet the refinery's demand. The objective is to minimize the total cost associated with the violation of the refinery's demand, the mixture of oils in the tanks, not satisfying the maintenance schedules, and not filling a tank after a vessel-tank uploading operation.

3.2. COSP formulation

The following notations are used:

Sets, indices, parameters and decisions variables

$\mathbb{I} = \{1, \ldots, n\}$	:	Set of tanks, indexed by i
$\mathbb{J} = \{1, \ldots, J\}$	:	Set of crude oil types, indexed by j, j'
$\mathbb{T} = \{1, \ldots, T\}$	:	Set of periods, indexed by t
$d_{j,t}$	:	The volume of crude oil j demanded by the pipeline in period t
$d'_{j,t}$	:	The percentage of crude oil j demanded by the pipeline in period t
d''_t	:	The total volume of crude oil demanded by the pipeline in period t, note that $d_{j,t} = \frac{d'_{j,t}}{100} \times d''_t$
$a_{j,t}$	:	The volume of crude oil j arrived at the terminal by vessels in period t
a'_t	:	The total volume of crude oil arrived at the terminal by vessels in period t (note that $a'_t = \sum_{j \in \mathbb{D}} a_{j,t}$)
CAP_i	:	The total capacity of tank i
$\widehat{w}_{i,j,0}$:	:	The initial volume of crude oil j in the tank, i at the beginning of period 0
$\hat{\eta}_{i,0}$	:	The binary parameter is 1 if crude oil has been discharged from a vessel to tank i in period 0; 0 otherwise
$\hat{x}_{i,j,0}$	:	The volume of crude oil j discharged from a vessel to tank i in period 0
$\hat{y}_{i,j,0}$	:	The volume of crude oil j injected from tank i into the pipeline in period 0
n_1	:	Maximum number of tanks that can deliver crude oil to the pipeline simultaneously
n_2	:	Maximum number of crude oil types that can be stored in a tank in each period
$\overline{VOL}$	:	Upper bound on the total volume injected from tanks to the pipeline in each period
$\underline{VOL}$	:	Lower bound on the volume delivered from vessels to a tank in each period
$\text{FAIL}_{i,t}$	:	The binary parameter is 1 if tank i is unable to deliver crude oil to the pipeline in period t (for example, due to fixing requirements);0 otherwise
$FULL_{i,t}$	:	The binary parameter is 1 if tank i must be at full capacity in period t (for example, due to fixing requirements); 0 otherwise
$\text{EMPTY}_{i,t}$	:	The binary parameter is 1 if tank i must be empty in period t (for example, due to fixing requirements); 0 otherwise
$MIX_{j,j'}$	:	The binary parameter is 1 if crude oils j and j' are prohibited to be stored in a tank at the same period; 0 otherwise
$c_{1,j}$	:	The cost of the difference between the volume of crude oil j demanded by the refinery and the one injected into the pipeline
c_2	:	The cost of the difference between the total volume of crude oil demanded by the refinery and the volume injected into the pipeline

c_3 : The cost of not satisfying the maintenance requirements for a tank
c_4 : The cost of mixing different qualities in a tank
c_5 : The cost of not filling a tank after a vessel-tank uploading operation

Decisions variables

$x_{i,j,t}$: Nonnegative continuous variable indicating the volume of crude oil j delivered from vessels to tank i in period t
$x'_{i,t}$: Nonnegative continuous variable indicating the total volume of crude oil delivered from vessels to tank i in period t
$y_{i,j,t}$: Nonnegative continuous variable indicating the volume of crude oil j injected from tank i into the pipeline in period t
$y'_{i,t}$: Nonnegative continuous variable indicating the total volume of crude oil injected from tank i into the pipeline in period t
$y''_{j,t}$: Nonnegative continuous variable indicating the total volume of crude oil j injected from all tanks into the pipeline in period t
y'''_t : Nonnegative continuous variable indicating the total volume of crude oil injected from all tanks into the pipeline in period t
$w_{i,j,t}$: Nonnegative continuous variable indicating the volume of crude oil j in the tank, i at the beginning of period t
$w'_{i,t}$: Nonnegative continuous variable indicating the total volume of crude oil in the tank i at the beginning of period t
$\Delta_{j,t}$: Nonnegative continuous variable indicating the difference between the volume of crude oil j demanded by the refinery and the one injected into the pipeline
Δ'_t : Nonnegative continuous variable indicating the difference between the total volume of crude oil demanded by the refinery and the one injected into the pipeline
$\delta_{i,t}$: The binary variable is 1 if tank i is full in period t; 0 otherwise
$\delta'_{i,t}$: The binary variable is 1 if tank i becomes full after a vessel delivers crude oil to tank i in period $t; 0$ otherwise
$\eta_{i,t}$: The binary variable is 1 if a vessel delivers crude oil to tank i in period t; 0 otherwise.
$\beta_{i,t}$: The binary variable is 1 if crude oil is injected from tank i to the pipeline in period $t; 0$ otherwise.
$\gamma_{i,j,t}$: The binary variable is 1 if tank i contains crude oil j in period $t; 0$ otherwise.

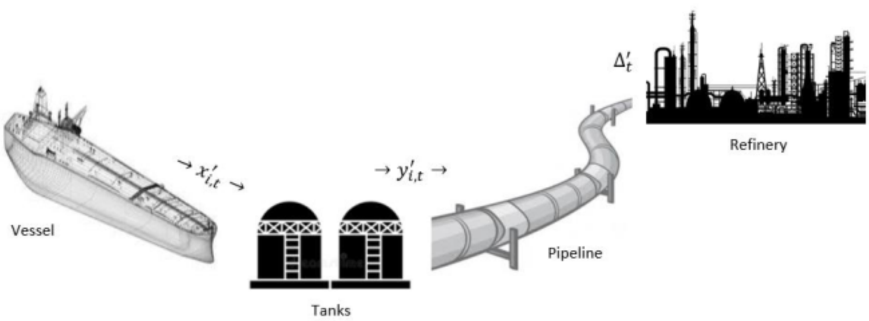

Figure 1: Illustration of COSP

Based on the above notations, COSP is formulated as the following BLMIP model which is adopted from De

Assis et al. [11].

(COSP)

$$\min z = \sum_{t\in\mathbb{T}}\sum_{j\in\mathbb{I}} c_{1,j}\Delta_{j,t} + c_2\sum_{t\in\mathbb{T}}\Delta'_t + c_3\left(\sum_{t\in\mathbb{T}}\sum_{i\in\mathbb{I}} EMPTY_{i,t}\sum_{j\in\mathbb{J}}\gamma_{i,j,t} + \sum_{t\in\mathbb{T}}\sum_{i\in\mathbb{I}} FULL_{i,t}\left(1-\delta_{i,t}\right)\right) \tag{43}$$

$$+ c_4\sum_{t\in\mathbb{T}}\sum_{i\in\mathbb{I}}\sum_{j\in\mathbb{J}}\gamma_{i,j,t} + c_5\left(\sum_{t\in\mathbb{T}:t\geq 2}\sum_{i\in\mathbb{I}}\left(\eta_{i,t-1}-\delta'_{i,t}\right) + \sum_{i\in\mathbb{I}}\left(\hat{\eta}_{i,0}-\eta_{i,1}\right)\right)$$

$$\text{s.t. } w_{i,j,1} = \widehat{w}_{i,j,0} + \hat{x}_{i,j,0} - \hat{y}_{i,j,0} \quad \forall i\in\mathbb{I}, j\in\mathbb{J} \tag{44}$$

$$w_{i,j,t} = w_{i,j,t-1} + x_{i,j,t-1} - y_{i,j,t-1} \quad \forall i\in\mathbb{I}, j\in\mathbb{J}, t\in\mathbb{T}: t\geq 2 \tag{45}$$

$$w'_{i,t} = \sum_{j\in\mathbb{J}} w_{i,j,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{46}$$

$$w'_{i,t} \leq CAP_i \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{47}$$

$$w'_{i,t} \geq CAP_i\delta_{i,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{48}$$

$$\sum_{j\in\mathbb{J}}\gamma_{i,j,t} \leq n_2 \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{49}$$

$$\gamma_{i,j,t} + \gamma_{i,j',t} \leq 1 \quad \forall j,j'\in\mathbb{J}: MIX_{j,j'} = 1, \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{50}$$

$$w_{i,j,t} \leq \text{CAP}_i\,\gamma_{i,j,t} \quad \forall i\in\mathbb{I}, j\in\mathbb{J}, t\in\mathbb{T} \tag{51}$$

$$\delta_{i,t} + \eta_{i,t-1} \leq 1 + \delta'_{i,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T}: t\geq 2 \tag{52}$$

$$\delta'_{i,t} \leq \delta_{i,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T}: t\geq 2 \tag{53}$$

$$\delta'_{i,t} \leq \eta_{i,t-1} \quad \forall i\in\mathbb{I}, t\in\mathbb{T}: t\geq 2 \tag{54}$$

$$\delta'_{i,1} = \delta_{i,1}\hat{\eta}_{i,0} \quad \forall i\in\mathbb{I} \tag{55}$$

$$a_{j,t} = \sum_{i\in\mathbb{I}} x_{i,j,t} \quad \forall j\in\mathbb{J}, t\in\mathbb{T} \tag{56}$$

$$x'_{i,t} = \sum_{j\in\mathbb{J}}^{i\in\mathbb{I}} x_{i,j,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{57}$$

$$\underline{VOL}\eta_{i,t} \leq x'_{i,t} \leq a'_t\eta_{i,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{58}$$

$$y'_{i,t} = \sum_{j\in\mathbb{J}} y_{i,j,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{59}$$

$$y''_{j,t} = \sum_{i\in\mathbb{I}}^{j\in\mathbb{J}} y_{i,j,t} \quad \forall j\in\mathbb{J}, t\in\mathbb{T} \tag{60}$$

$$y'''_t = \sum_{i\in\mathbb{I}} y'_{i,t} \quad \forall t\in\mathbb{T} \tag{61}$$

$$\sum_{i\in\mathbb{I}}\beta_{i,t} \leq n_1 \quad \forall t\in\mathbb{T} \tag{62}$$

$$y'_{i,t} \leq CAP_i\beta_{i,t} \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{63}$$

$$y'''_t \leq \overline{VOL} \quad \forall t\in\mathbb{T} \tag{64}$$

$$\beta_{i,t} = 0 \quad \forall i\in\mathbb{I}, t\in\mathbb{T}: Fail_{i,t} = 1 \tag{65}$$

$$\eta_{i,t} + \beta_{i,t} \leq 1 \quad \forall i\in\mathbb{I}, t\in\mathbb{T} \tag{66}$$

$$\eta_{i,t} + \beta_{i,t+1} \leq 1 \quad \forall i\in\mathbb{I}, t\in\mathbb{T}: t\leq T-1 \tag{67}$$

$$\Delta_{j,t} \geq y''_{j,t} - d_{j,t} \quad \forall j\in\mathbb{J}, t\in\mathbb{T} \tag{68}$$

$$\Delta_{j,t} \geq d_{j,t} - y''_{j,t} \quad \forall j\in\mathbb{J}, t\in\mathbb{T} \tag{69}$$

$$\Delta'_t \geq y'''_t - d''_t \quad \forall t\in\mathbb{T} \tag{70}$$

$$\tag{71}$$

$$\Delta'_t \geq d''_t - y'''_t \quad \forall t \in \mathbb{T} \tag{72}$$
$$y_{i,j,t} \times w'_{i,t} = y'_{i,t} \times w_{i,j,t} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{73}$$
$$x_{i,j,t}, y_{i,j,t}, w_{i,j,t} \geq 0, \gamma_{i,j,t} \in \{0,1\} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{74}$$
$$x'_{i,t}, y'_{i,t}, w'_{i,t} \geq 0, \delta_{i,t}, \delta'_{i,t}, \eta_{i,t}, \beta_{i,t} \in \{0,1\} \quad \forall i \in \mathbb{I}, t \in \mathbb{T} \tag{75}$$
$$y''_{j,t}, \Delta_{j,t} \geq 0 \quad \forall j \in \mathbb{J}, t \in \mathbb{T} \tag{76}$$
$$y'''_t, \Delta'_t \geq 0 \quad \forall t \in \mathbb{T} \tag{77}$$

The objective function (43) minimizes the total cost. Constraint sets (44) and (45) determine the volume of crude oil j in tank i at each period. Constraint set (46) specifies the relation between variables $w_{i,j,t}$ and $w'_{i,t}$. Constraint set (47) indicates the upper bound on the storage capacity. Constraint set (48) together with (47) implies that if $\delta_{i,t} = 1$ then $w'_{i,t} = CAP_i$. Constraint set (49) restricts the number of different types of crude oil in tank i in each period. Constraint set (50) prohibits tank i to contain both crude oils j, j' with $MIX_{j,j'} = 1$ at the same time. Constraint set (51) indicates that if $\gamma_{i,j,t} = 0$, then $w_{i,j,t} = 0$. Constraint sets (52)-(54) are linear representations of the following constraint ensuring that if tank i receives crude oil from vessels in period $t-1$ and it is in full status in time period t, then $\delta'_{i,t}$ takes 1.

$$\delta'_{i,t} = \delta_{i,t}\eta_{i,t-1} \quad \forall i \in \mathbb{I}, t \in \mathbb{T} : t \geq 2$$

Constraint set (55) ensures the same restriction for $t = 1$. Constraint set (56) guarantees that the total volume of crude oil j is delivered from the vessels to tanks in period t. Constraint set (57) calculates the total volume of crude oil delivered to tank i in period t. Constraint set (58) implies lower and upper bound on the total volume of crude oils delivered to tank i. Constraint set (59) calculates the total volume of crude oil injected from tank i into the pipeline in time period t. Constraint set (60) calculates the volume of crude oil j injected from all tanks into the pipeline in period t. Constraint set (61) determines the total volume of crude oils injected into the pipeline in period t. Constraint set (62) restricts the number of tanks that can deliver crude oil to the pipeline at any period. Constraint set (63) ensures that if $\beta_{i,t} = 0$, then $y'_{i,t} = 0$. Constraint set (64) limits the total volume of crude oil injected into the pipeline in period t. Constraint set (65) guarantees that if tank i is out of service, it cannot feed the pipeline.

Constraint set (66) indicates that inlet and outlet operations in tank i cannot be implemented at the same period. Constraint set (67) implies that when a tank gets crude oil from vessels, outlet operations must be postponed for at least one period. Constraint sets (68) and (69) together with minimizing $\Delta_{j,t}$ in the objective function ensure that $\Delta_{j,t}$ takes $\left|y''_{j,t} - d_{j,t}\right|$. Constraint sets (70) and (72) together with minimizing Δ'_t in the objective function ensure that Δ'_t takes $|y'''_t - d''_t|$. Constraint set (73) is a bilinear restatement of the following constraint indicating that the concentration of crude oil j in a batch injected from tank i into the pipeline is equal to the concentration of crude oil j inside the tank.

$$\frac{w_{i,j,t}}{w'_{i,t}} = \frac{y_{i,j,t}}{y'_{i,t}} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}$$

Constraint sets (74)-(77) define the type of variables.

The challenging parts of the model COSP are bilinear terms that appear in the constraint set (73) and cause the problem to be non-convex and difficult to solve. Thus, MINLP solvers would be inefficient to solve moderate and large-sized instances of this problem in a reasonable amount of time. Therefore, as a solution method, De Assis et al. [11] utilized the piecewise McCormick relaxation technique, addressed by Castro [3], and the domain reduction strategy to develop an iterative two-step MILP-NLP heuristic algorithm to solve COSP. We refer to their algorithm as a piecewise portioning algorithm based on the McCormick relaxation technique (for short MRPPA′ which is MRPPA investigated in Section 2 to solve COSP). In the rest of this section, after a short review on the MRPPA′, we explain how our new algorithm FRPPA is adopted to solve COSP as an improvement of MRPPA′.

3.3. McCormick relaxation-based method

MRPPA′, proposed by De Assis et al. [11], is a two-step MILP-NLP algorithm based on piecewise McCormick envelopes. In this algorithm, first, the bilinear terms are relaxed by applying piecewise McCormick envelopes to get an MILP relaxation model considering the current iteration bound reduction operations. Then, the binary variables of COSP are fixed at the optimal solution obtained by the MILP to get an NLP model. Based on the optimal solutions to MILP and NLP models, the domain of variables contained in the bilinear terms is updated and reduced for the next iteration. This process is repeated and the algorithm terminates when the difference between the objective function value of the recent MILP and the best upper bound found so far for the original problem falls into a given tolerance.

Let $\underline{y}_{i,j,t}, \underline{y}'_{i,t}, \underline{w}_{i,j,t}$, and $\underline{w}'_{i,t}$ be lower bounds of variables $y_{i,j,t}, y'_{i,t}, w_{i,j,t}$, and $w'_{i,t}$, respectively. Additionally, consider $\bar{y}_{i,j,t}, \bar{y}'_{i,t}$, $\bar{w}_{i,j,t}$, and $\bar{w}'_{i,t}$ as upper bounds of these variables, correspondingly. The bounds are initialized as follows:

$$\left[\underline{y}_{i,j,t}, \bar{y}_{i,j,t}\right] = [0, \overline{VOL}], \quad \left[\underline{y}'_{i,t}, \bar{y}'_{i,t}\right] = [0, \overline{VOL}], \quad \left[\underline{w}_{i,j,t}, \bar{w}_{i,j,t}\right] = [0, CAP_i], \quad \left[\underline{w}'_{i,t}, \bar{w}'_{i,t}\right] = [0, CAP_i] \tag{78}$$

The above intervals are partitioned into some sub-intervals with equal lengths and the set of sub-intervals associated with $y_{i,j,t}, y'_{i,t}, w_{i,j,t}$, and $w'_{i,t}$ are denoted by $\mathbb{L} = \{1, \ldots, L\}$ (indexed by ℓ), $\mathbb{O} = \{1, \ldots, O\}$ (indexed by o), $\mathbb{R} = \{1, \ldots, R\}$ (indexed by r) and $\mathbb{S} = \{1, \ldots, S\}$ (indexed by s), respectively, where the $\ell^{\text{th}}, o^{\text{th}}, r^{\text{th,}}$ and s^{th} sub-intervals associated with variables $y_{i,j,t}$, $\left[\underline{SIw'}_{i,t,s}, \overline{SIw'}_{i,t,s}\right]$, correspondingly, and we have:

$$\left[\underline{SIy}_{i,j,t,\ell}, \overline{\text{SIy}}_{i,j,t,\ell}\right] = \left[\underline{y}_{i,j,t} + (\ell - 1) \times \frac{\bar{y}_{i,j,t} - \underline{y}_{i,j,t}}{L}, \underline{y}_{i,j,t} + \ell \times \frac{\bar{y}_{i,j,t} - \underline{y}_{i,j,t}}{L}\right] \quad \ell \in \mathbb{L} \tag{79}$$

$$\left[\underline{SIy'}_{i,t,o}, \overline{SIy}'_{i,t,o}\right] = \left[\underline{y}'_{i,t} + (o - 1) \times \frac{\bar{y}'_{i,t} - \underline{y}'_{i,t}}{o}, \underline{y}'_{i,t} + o \times \frac{\bar{y}'_{i,t} - \underline{y}'_{i,t}}{o}\right] \quad o \in \mathbb{O} \tag{80}$$

$$\left[\underline{SI}w_{i,j,t,r}, \overline{\text{SIw}}_{i,j,t,r}\right] = \left[\underline{w}_{i,j,t} + (r - 1) \times \frac{\bar{w}_{i,j,t} - \underline{w}_{i,j,t}}{R}, \underline{w}_{i,j,t} + r \times \frac{\bar{w}_{i,j,t} - \underline{w}_{i,j,t}}{R}\right] \quad r \in \mathbb{R} \tag{81}$$

$$\left[\underline{SIw'_{i,t,s}}, \overline{SIw'}_{i,t,s}\right] = \left[\underline{w}'_{i,t} + (s - 1) \times \frac{\bar{w}'_{i,t} - \underline{w}'_{i,t}}{s}, \underline{w}'_{i,t} + s \times \frac{\bar{w}'_{i,t} - \underline{w}'_{i,t}}{s}\right] \quad s \in \mathbb{S} \tag{82}$$

Then, the bilinear terms in the constraint set (73) (i.e., $y_{i,j,t} \times w'_{i,t}$ and $y'_{i,t} \times w_{i,j,t}$) are replaced by new nonnegative continuous variables $v^{LHS}_{i,j,t}$ and $v^{RHS}_{i,j,t}$, respectively, and the following MILP is constructed by using the piecewise McCormick envelopes. We refer to this model as a piecewise partitioning model based on McCormick relaxation (MRPPM′ for short), in which $\alpha_{i,j,t,\ell,s}$ is a binary variable that is 1 if the McCormick envelope is associated with the ℓ^{th} sub-interval of $y_{i,j,t}$ and the s^{th} sub-interval of $w'_{i,t}$ is selected. Similarly, $\theta_{i,j,t,o,r}$ is a binary variable that is 1 if the McCormick envelope is associated with the o^{th} sub-interval of $y'_{i,t}$ and the r^{th} sub-interval of $w_{i,j,t}$ is selected. Additionally, $y_{i,j,t,\ell,s}$ is a nonnegative continuous variable that is equal to $y_{i,j,t}$ if $\alpha_{i,j,t,\ell,s} = 1; 0$ otherwise. Similarly, $w'_{i,j,t,\ell,s}$ is a nonnegative continuous variable that is equal to $w'_{i,t}$ if $\alpha_{i,j,t,\ell,s} = 1; 0$ otherwise. Furthermore, $y'_{i,j,t,o,r}$ is a nonnegative continuous variable that is equal to $y'_{i,t}$ if $\theta_{i,j,t,o,r} = 1; 0$ otherwise. In the same way, $w_{i,j,t,o,r}$, is a nonnegative continuous variable that is equal to $w_{i,j,t}$ if $\theta_{i,j,t,o,r} = 1; 0$ otherwise.

MRPPM′.

$$\begin{aligned}\min z = &\sum_{t \in \mathbb{T}} \sum_{j \in \mathbb{J}} c_{1,j} \Delta_{j,t} + c_2 \sum_{t \in \mathbb{T}} \Delta'_t \\ &+ c_3 \left(\sum_{t \in \mathbb{T}} \sum_{i \in \mathbb{I}} EMPTY_{i,t} \sum_{j \in \mathbb{J}} \gamma_{i,j,t} + \sum_{t \in \mathbb{T}} \sum_{i \in \mathbb{I}} FULL_{i,t} \left(1 - \delta_{i,t}\right) \right) \\ &+ c_4 \sum_{t \in \mathbb{T}} \sum_{i \in \mathbb{I}} \sum_{j \in \mathbb{I}} \gamma_{i,j,t} + c_5 \left(\sum_{t \in \mathbb{T}: t \geq 2} \sum_{i \in \mathbb{I}} \left(\eta_{i,t-1} - \delta'_{i,t}\right) + \sum_{i \in \mathbb{I}} \left(\hat{\eta}_{i,0} - \eta_{i,1}\right) \right)\end{aligned}$$

$$v^{LHS}_{i,j,t} = v^{RHS}_{i,j,t} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{83}$$

$$v^{LHS}_{i,j,t} \geq \sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} \left(\underline{SIw'}_{i,t,s} y_{i,j,t,\ell,s} + \underline{SIy}_{i,j,t,\ell} w'_{i,j,t,\ell,s} - \underline{SIw'}_{i,t,s}\underline{SIy}_{i,j,t,\ell}\alpha_{i,j,t,\ell,s}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{84}$$

$$v^{LHS}_{i,j,t} \geq \sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} \left(\overline{SIw'}_{i,t,s} y_{i,j,t,\ell,s} + \overline{SIy}_{i,j,t,\ell} w'_{i,j,t,\ell,s} - \overline{SIw'}_{i,t,s}\overline{SIy}_{i,j,t,\ell}\alpha_{i,j,t,\ell,s}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{85}$$

$$v^{LHS}_{i,j,t} \leq \sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} \left(\underline{SIw}'_{i,t,s_{i,j,t,\ell,s}} + \overline{SIy}_{i,j,t,\ell} w'_{i,j,t,\ell,s} - \underline{SIw}'_{i,t,s}\overline{SIy}_{i,j,t,\ell}\alpha_{i,j,t,\ell,s}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{86}$$

$$v^{LHS}_{i,j,t} \leq \sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} \left(\overline{SIw'}_{i,t,s} y_{i,j,t,\ell,s} + \underline{SIy}_{i,j,t,\ell} w'_{i,j,t,\ell,s} - \overline{SIw'}_{i,t,s}\underline{SIy}_{i,j,t,\ell}\alpha_{i,j,t,\ell,s}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{87}$$

$$v^{RHS}_{i,j,t} \geq \sum_{o\in\mathbb{O}}\sum_{r\in\mathbb{R}} \left(\underline{SIw}_{i,j,t,r} y'_{i,j,t,o,r} + \underline{SIy'}_{i,t,o} w_{i,j,t,o,r} - \underline{SIw}_{i,j,t,r}\underline{SIy'}_{i,t,o}\theta_{i,j,t,o,r}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{88}$$

$$v^{RHS}_{i,j,t} \geq \sum_{o\in\mathbb{O}}\sum_{r\in\mathbb{R}} \left(\overline{SIw}_{i,j,t,r} y'_{i,j,t,o,r} + \overline{SIy'}_{i,t,o} w_{i,j,t,o,r} - \overline{SIw}_{i,j,t,r}\overline{SIy'}_{i,t,o}\theta_{i,j,t,o,r}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{89}$$

$$v^{RHS}_{i,j,t} \leq \sum_{o\in\mathbb{C}}\sum_{r\in\mathbb{R}} \left(\underline{SIw}_{i,j,t,r}{}'_{i,j,t,o,r} + \overline{SIy'}_{i,t,o} w_{i,j,t,o,r} - \underline{SIw}_{i,j,t,r}\mathrm{SIy}'_{i,t,o}\theta_{i,j,t,o,r}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{90}$$

$$v^{RHS}_{i,j,t} \leq \sum_{o\in\mathbb{C}}\sum_{r\in\mathbb{R}} \left(\overline{SIw}_{i,j,t,r}{}'_{i,j,t,o,r} + \underline{SIy'}_{i,t,o} w_{i,j,t,o,r} - \overline{SIw}_{i,j,t,r}\underline{\mathrm{SIy}'}_{i,t,o}\theta_{i,j,t,o,r}\right) \\ \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{91}$$

$$y_{i,j,t} = \sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} y_{i,j,t,\ell,s} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{92}$$

$$y'_{i,t} = \sum_{o\in\mathbb{C}}\sum_{r\in\mathbb{R}} y'_{i,j,t,o,r} \quad \forall i \in \mathbb{I}, j \in \mathbb{I}, t \in \mathbb{T} \tag{93}$$

$$w_{i,j,t} = \sum_{o\in\mathbb{O}}\sum_{r\in\mathbb{R}} w_{i,j,t,o,r} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{94}$$

$$w'_{i,t} = \sum_{\ell\in\mathbb{L}}^{o\in\mathbb{Q}}\sum_{s\in\mathbb{S}}^{\mathbb{R}\in\mathbb{R}} w'_{i,j,t,\ell,s} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{95}$$

$$\underline{SIy}_{i,j,t,\ell_{i,j,t,\ell,s}} \leq y_{i,j,t,\ell,s} \leq \overline{SIy}_{i,j,t,\ell}\alpha_{i,j,t,\ell,s} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, \ell \in \mathbb{L}, s \in \mathbb{S} \tag{96}$$

$$\underline{SIy}'_{i,t,o}\theta_{i,j,t,o,r} \leq y'_{i,j,t,o,r} \leq \overline{SIy'}_{i,t,o}\theta_{i,j,t,o,r} \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, o \in \mathbb{O}, r \in \mathbb{R} \tag{97}$$

$$\underline{SIw}_{i,j,t,r_{i,j,t,o,r}} \leq w_{i,j,t,o,r} \leq \overline{SIw}_{i,j,t,r}\theta_{i,j,t,o,r} \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, o \in \mathbb{O}, r \in \mathbb{R} \tag{98}$$

$$\underline{SIw'}_{i,j,t,r_{i,j,t,o,r}} \leq w_{i,j,t,o,r} \leq \overline{SIw'}_{i,j,t,r}\theta_{i,j,t,o,r} \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, o \in \mathbb{O}, r \in \mathbb{R} \tag{99}$$

$$\sum_{\ell\in\mathbb{L}}\sum_{s\in\mathbb{S}} \alpha_{i,j,t,\ell,s} = 1 \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{100}$$

$$\sum_{o\in\mathbb{C}}\sum_{r\in\mathbb{R}} \theta_{i,j,t,o,r} = 1 \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{101}$$

$$\alpha_{i,j,t,\ell,s} \in \{0,1\}, y_{i,j,t,\ell,s} \geq 0, w'_{i,j,t,\ell,s} \geq 0 \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, \ell \in \mathbb{L}, s \in \mathbb{S} \tag{102}$$

$$\theta_{i,j,t,o,r} \in \{0,1\}, w_{i,j,t,o,r} \geq 0, y_{i,j,t,o,r} \geq 0 \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T}, o \in \mathbb{O}, r \in \mathbb{R} \tag{103}$$

$$v^{LHS}_{i,j,t}, v^{RHS}_{i,j,t} \geq 0 \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{104}$$

The general framework of MRPPA′ is as follows:

MRPPA′

Step 0: Let k be a counter, $\varepsilon > 0$ be a given accuracy, and $Ctrl$ be a binary parameter that is 1 if the stopping criterion is observed, 0 otherwise. Suppose that $\underline{y}_{i,j,t}, \underline{y}'_{i,t}, \underline{w}_{i,j,t}$, and $\underline{w}'_{i,t}$ are lower bounds of variables $y_{i,j,t}, y'_{i,t}, w_{i,j,t}$, and $w'_{i,t}$, respectively. Additionally, consider $\bar{y}_{i,j,t}$, $\bar{y}'_{i,t}, \bar{w}_{i,j,t}$, and $\bar{w}'_{i,t}$ as upper bounds of these variables, correspondingly. Initialize the bound based on (78). Furthermore, determine the value of p arameters L, O, R, and S as the number of sub-intervals associated with variables $y_{i,j,t}, y'_{i,t}$, $w_{i,j,t}$, and $w'_{i,t}$, correspondingly, and construct the sub-intervals based on (79)-(82). Assume that $z_{UB}^{(k)}$ denotes the objective value of the best solution found until iteration k, and consider $z_{LB}^{(k)}$ as the objective function value of MRPPM′ solved in iteration k. Initialize $k := 1$, and $Ctrl := 0$.

Step 1: While $Ctrl = 0$ do

Step 1-1: Solve the model MRPPM', and denote its optimal objective function value by $z_{LB}^{(k)}$.

Step 1-2: Fix the binary variables $\gamma_{i,j,t}, \delta_{i,t}, \delta'_{i,t}, \eta_{i,t}$, and $\beta_{i,t}$ of the model COSP to the optimal solution of MRPPM′ obtained in Step 1-1 to get an NLP model. Denote the optimal objective function value of the NLP model by z_{NLP}^{*}, and set $z_{UB}^{(k)} = \min\left\{z_{NLP}^{*}, z_{UB}^{(k-1)}\right\}$.

Step 1-3: For $i \in \mathbb{I}, j \in \mathbb{J}$, and $t \in \mathbb{T}$ do

Step 1-3-1: Let ℓ_0 and r_0 be the indices of the sub-intervals containing the value of $y_{i,j,t}$ and $w_{i,j,t}$ in the optimal solution to MRPPM′ solved in Step 1-1. Additionally, let ℓ_1 and r_1 be the indices of the sub-intervals containing the value of $y_{i,j,t}$ and $w_{i,j,t}$ in the optimal solution to the NLP model introduced in Step 1-2. Reduce the length of the interval associated with variables $y_{i,j,t}$ and $w_{i,j,t}$ as follows:

$$\underline{y}_{i,j,t} = \min\left\{\underline{SIy}_{i,j,t,\ell_0}, \underline{SIy}_{i,j,t,\ell_1}\right\}, \bar{y}_{i,j,t} = \max\left\{\overline{SIy}_{i,j,t,\ell_0}, \overline{SIy}_{i,j,t,\ell_1}\right\}$$

$$\underline{w}_{i,j,t} = \min\left\{\underline{SIw}_{i,j,t,r_0}, \underline{SIw}_{i,j,t,r_1}\right\}, \bar{w}_{i,j,t} = \max\left\{\overline{SIw}_{i,j,t,r_0}, \overline{SIw}_{i,j,t,r_1}\right\}$$

Step 1-4: For $i \in \mathbb{I}$, and $t \in \mathbb{T}$ do

Step 1-4-1: Let o_0 and s_0 be the indices of the sub-intervals containing the value of $y'_{i,t}$ and $w'_{i,t}$ in the optimal solution to MRPPM′ solved in Step 1-1. Additionally, let o_1 and s_1 be the indices of the sub-intervals containing the value of $y'_{i,t}$ and $w'_{i,t}$ in the optimal solution to the NLP model introduced in Step 1-2. Reduce the length of the interval associated with variables $y'_{i,t}$ and $w'_{i,t}$ as follows:

$$\left[\underline{y}'_{i,t}, \bar{y}'_{i,t}\right] = \left[\min\left\{\underline{SIy}'_{i,t,o_0}, \underline{SIy}'_{i,t,o_1}\right\}, \max\left\{\overline{SIy}_{i,t,o'_0}, \overline{SIy}'_{i,t,o_1}\right\}\right]$$

$$\left[\underline{w}'_{i,t}, \bar{w}'_{i,t}\right] = \left[\min\left\{\underline{SIw}_{i,t,s'_0}, \underline{SIw}_{i,t,s'_1}\right\}, \max\left\{\overline{SIw}_{i,t,s'_0}, \overline{SIw}'_{i,t,s_1}\right\}\right]$$

Step 1-5: If $z_{UB}^{(k)} - z_{LB}^{(k)} < \varepsilon$, set $Ctrl := 1$, else, set $k := k + 1$, and update the subintervals based on (79)-(82) and reduced bounds.

Step 2: Return the solution associated with $z_{UB}^{(k)}$ as the best solution found by the algorithm.

3.4. Adopting FRPPA for COSP

To adopt FRPPA for COSP, instead of working with constraint (73), we use its fractional representation as follows, and approximate each nonlinear fraction linearly.

$$\frac{w_{i,j,t}}{w'_{i,t}} = \frac{y_{i,j,t}}{y'_{i,t}} \quad \forall i \in \mathbb{I}, j \in \mathbb{J}, t \in \mathbb{T} \tag{105}$$

The above constraint is stated for every triple (i, j, t), and both sides take value in the same interval which is denoted by $\left[L'_{i,j,t}, U'_{i,j,t}\right]$. This interval is initialized at $[0, 1]$, and refined gradually.

For every triple (i, j, t), we partition $\left[L'_{i,j,t}, U'_{i,j,t}\right]$ into some sub-intervals with equal lengths, and denote the set of sub-intervals by $\mathbb{P} = \{1, \ldots, P\}$ (indexed by p), where the p^{th} sub-interval is associated with triple (i, j, t) is as follows:

$$\left[L'_{i,j,t,p}, U'_{i,j,t,p}\right] = \left[L'_{i,j,t} + (p-1) \times \frac{(U'_{i,j,t} - L'_{i,j,t})}{P}, L'_{i,j,t} + p \times \frac{(U'_{i,j,t} - L'_{i,j,t})}{P}\right], \quad p \in \mathbb{P} \tag{106}$$

Therefore, we have:

$$\bigvee_{p\in\mathbb{P}}\left[\left(L'_{i,j,t,p}\leq\frac{w_{i,j,t}}{w'_{i,t}}\leq U'_{i,j,t,p}\right)\wedge\left(L'_{i,j,t,p}\leq\frac{y_{i,j,t}}{y'_{i,t}}\leq U'_{i,j,t,p}\right)\right]$$

Thus, we provide relaxation of the original model by substituting the non-linear constraint (73) by the following linear constraints:

$$-M\left(1-\gamma_{i,j,t,p}\right)+L'_{i,j,t,p}w'_{i,t}\leq w_{i,j,t}\leq U'_{i,j,t,p}w'_{i,t}+M\left(1-\gamma_{i,j,t,p}\right)\quad\forall i\in\mathbb{I},j\in\mathbb{J},t\in\mathbb{T},p\in\mathbb{P}\tag{107}$$

$$-M\left(1-\gamma_{i,j,t,p}\right)+L'_{i,j,t,p}y'_{i,t}\leq y_{i,j,t}\leq U'_{i,j,t,p}y'_{i,t}+M\left(1-\gamma_{i,j,t,p}\right)\quad\forall i\in\mathbb{I},j\in\mathbb{J},t\in\mathbb{T},p\in\mathbb{P}\tag{108}$$

$$\sum_{p\in\mathbb{P}}\gamma_{i,j,t,p}=1\quad\forall i\in\mathbb{I},j\in\mathbb{J},t\in\mathbb{T}\tag{109}$$

$$\gamma_{i,j,t,p}\in\{0,1\}\quad\forall i\in\mathbb{I},j\in\mathbb{J},t\in\mathbb{T},p\in\mathbb{P}\tag{110}$$

Where $\gamma_{i,j,t,p}$ is a binary variable that is 1 if the p^{th} sub-interval is associated with triple (i,j,t) is selected; 0 otherwise. Additionally, M is a sufficiently large positive number. We refer to the model obtained by replacing nonlinear constraints (105) with linear constraints (107)-(110) as FRPPM$'$.

Based on the above description, the general framework of the FRPPA adopted for COSP (which we refer to as FRPPA$'$) is as follows:

FRPPA

Step 0: Let k be a counter, $\varepsilon>0$ be a given accuracy, and $Ctrl$ be a binary parameter that is 1 if the stopping criterion is observed, 0 otherwise. For every triple (i,j,t), let $\left[L'_{i,j,t},U'_{i,j,t}\right]$ be the interval associated with the fractions in (105), and initialize it at $[0,1]$. Determine the value of parameter P as the number of sub-intervals associated with fractions, and construct the subintervals based on (106). Assume that $z_{UB}^{(k)}$ denotes the objective value of the best feasible solution to COSP found until iteration k, and consider $z_{LB}^{(k)}$ as the objective function value of FRPPM $'$ solved in iteration k. Initialize $k:=1$, and $Ctrl:=0$.

Step 1: While $Ctrl=0$ do

Step 1-1: Solve the model FRPPM$'$, and denote its optimal objective function value by $z_{LB}^{(k)}$.

Step 1-2: Fix the vector of binary variables of the model COSP to the optimal solution of FRPPM$'$ to get an NLP model. Denote the optimal objective function value of the NLP model by z^*_{NLP}, and set $z_{UB}^{(k)}=\min\left\{z^*_{NLP},z_{UB}^{(k-1)}\right\}$.

Step 1-3: For $i\in\mathbb{I},j\in\mathbb{J},t\in\mathbb{T}$ do

Step 1-3-1: Let p_0 be the index of the sub-interval containing the value of fractions $\frac{w_{i,j,t}}{w'_{i,t}}$ and $\frac{y_{i,j,t}}{y'_{i,t}}$ based on the optimal solution to FRPPM'. Additionally, let p_1 be the index of the sub-interval containing the value of fractions $\frac{w_{i,j,t}}{w'_{i,t}}$ and $\frac{y_{i,j,t}}{y'_{i,t}}$ in the optimal solution to the NLP model introduced in Step 1-2. Reduce the interval length associated with the triple (i,j,t) as follows:

$$L'_{i,j,t}=\min\left\{L'_{i,j,t,p_0},L'_{i,j,t,p_1}\right\},U'_{i,j,t}=\max\left\{U'_{i,j,t,p_0},U'_{i,j,t,p_1}\right\}$$

Step 1-4: If $z_{UB}^{(k)}-z_{LB}^{(k)}<\varepsilon$, set $Ctrl:=1$; else, set $k:=k+1$, and update the subintervals based on (2) and reduced bound.

Step 2: Return the solution associated with $z_{UB}^{(k)}$ as the best feasible solution found by the algorithm.

Remark 3.1. *Consider Step 1-3-1 of the FRPPA$'$, and let (i_0,j_0,t_0) be a given triple. If in the optimal solution to FRPPM$'$ or in the optimal solution to the NLP model solved in Step 1-3-1, we have $w'_{i_0,t_0}=0$ or $y'_{i_0,t_0}=0$, then it is better to update $\left[L'_{i_0,j_0,t_0},U'_{i_0,j_0,t_0}\right]$ to $[0,1]$. To justify the reason, assume that in the optimal solution to FRPPM $'$ or in the optimal solution to the NLP model solved in Step 1-3-1, we have $w'_{i_0,t_0}=0$; then based on the constraint (46), we have $w_{i_0,j_0,t_0}=0$. Thus, for these values, constraint (73) is satisfied no matter the value of the variables y'_{i_0,t_0} and y_{i_0,j_0,t_0} is. In this situation, updating $\left[L'_{i_0,j_0,t_0},U'_{i_0,j_0,t_0}\right]$ to $\left[\min\left\{L'_{i_0,j_0,t_0,p_0},L'_{i_0,j_0,t_0,p_1}\right\},\max\left\{U'_{i_0,j_0,t_0,p_0},U'_{i_0,j_0,t_0,p_1}\right\}\right]$ may cause unnecessary restriction on the value of variables y'_{i_0,t_0} and y_{i_0,j_0,t_0}, and losing some good solutions. Therefore, in this case, it is better to update $\left[L'_{i_0,j_0,t_0},U'_{i_0,j_0,t_0}\right]$ to $[0,1]$. Our preliminary experiments indicate that this issue may improve the quality of the solutions obtained by the FRPPA$'$.*

4. Computational results

In this section, we investigate the efficiency of the proposed FRPPA′ in solving COSP and comparing the results with MRPPA′ and the direct resolution (by BARON solver) in terms of solution quality and time. Experiments are carried out on a PC running Windows 7 operating system with a Core™ i3, 2.1 GHz processor, and 4.0 GB RAM. Algorithms are coded in the AIMMS mathematical modeling language [2], and all MINLP, MILP, and NLP models are solved using BARON, CPLEX, and CONOPT solvers, included in the AIMMS software, respectively. All solvers are used in their default settings with the exception that constraints (100), (101), and (109) are treated as a special ordered set of type 1 (SOS1).

Test instances are generated based on De Assis et al. [11], and categorized based on the cost level (i.e., low and high) and the number of periods in the planning horizon ($T = 18, 22, 26, 30$). Therefore, eight instances are generated and the name associated with each instance gives brief information about its characteristics. Indeed, the letter ‘ T ’ and its accompanying number indicate the number of periods in the planning horizon, and the last letter which is either ‘ H ’ or ‘ L ’, indicates the cost level (i.e., high or low). For example, the name ‘T18H’ implies an instance with 18 periods and a high-cost level. We define the following notations which are used to describe the columns of tables:

z_{BARON}	:	Objective function value of the best solution obtained by direct resolution of COSP by BARON in a given time limit
$z_{\text{MRPPA}'}$	:	Objective function value of the best solution obtained by MRPPA′
$Z_{\text{FRPPA}'}$	:	Objective function value of the best solution obtained by FRPPA′
RT_{BARON}	:	Running time of direct resolution of COSP by BARON (in second)
$RT_{\text{MRPPA}'}$	:	Running time of MRPPA′ (in second)
$RT_{\text{FRPPA}'}$	:	Running time of FRPPA′ (in second)
Iter	:	Iteration counter for MRPPA′ and FRPPA′
$G_{\text{MRPPA}'}$	:	Relative gain of MRPPA′ over BARON, calculated as $\frac{z_{BARON}-z_{\text{MRPPA}'}}{z_{\text{MRPPA}'}} \times 100$
$G_{\text{FRPPA}'}$	:	Relative gain of FRPPA′ over BARON, calculated as $\frac{z_{BARON}-z_{\text{FRPPA}'}}{z_{\text{FRPPA}'}} \times 100$

For instance, T18L, T18H, T22L, T22H, T26L, and T26H, we set the BARON to solve COSP within a time limit of 3600 seconds; however, for instances T30L, and T30H, this time limit was set at 14000 seconds. Additionally, for instances T18L, T18H, T22L, T22H, T26L, and T26H, we set MRPPA′ and FRPPA′ to run within a time limit of 3600 seconds; however, for instance, T30L, and T30H, a time limit of 2000 seconds is set on each iteration of MRPPA′ and FRPPA′.

Table 3 summarizes the results of the direct resolution of COSP by BARON in a time limit of 3600 seconds for instances T18L, T18H, T22L, T22H, T26L, and T26H. The columns labeled by “Total Var” and “Total Const” represent the total number of variables and constraints in COSP, respectively, and the columns labeled by “BVar” and “NLConst” refer to the number of binary variables and constraints of COSP, correspondingly. Additionally, in this table, the column “GAP” indicates the relative difference, between the objective function value of the best solution found within the given time limit and the lower bound provided by BARON.

Table 3: The BARON’s solution for instances T18, T22, and T26 in a time limit of 3600s

	Size of COSP				Cost level = Low			Cost level = High		
T	Total Var	Total Const	BVar	NLConst	z_{BARON}	GAP	RT_{BARON}	$\mathbf{z}_{\text{BARON}}$	GAP	RT_{BARON}
18	3618	7374	1134	630	40.057	0%	42	70.066	0%	333
22	4422	9014	1386	770	47.711	19%	> 3600	85.314	22%	> 3600
26	5226	10654	1638	910	55.831	35%	> 3600	158.171	126%	> 3600

The results of algorithms MRPPA′ and FRPPA′ for instance T18L, T18H, T22L, T22H, T26L, and T26H are provided in Table 4 and Table 5, respectively. Each row of these tables refers to a specific setting for the number of sub-intervals, and the term “NG” inserted in some rows of columns $G_{\text{MRPPA}'}$ or $G_{\text{FRPPA}'}$ indicates that the objective function value of the solution obtained by the corresponding algorithm is worse than the best solution found by BARON, and hence, no gain is achieved. The average of gains is calculated for successful instances in the last row of tables and “NG” is not considered. Also, the term “NA” inserted in some rows of columns $Z_{\text{MRPPA}'}$ indicates that MRPPA′ could not find a feasible solution.

Table 4: The MRPPA′ solutions for instances T18, T22, and T26

T	No. sub-intervals	Cost level = Low				Cost level = High			
	L-O-R-S	$z_{\text{MRPPA}'}$	$\text{RT}_{\text{MRPPA}'}$	Iter	$\text{G}_{\text{MRPPA}'}$	$z_{\text{MRPPA}'}$	$\text{RT}_{\text{MRPPA}'}$	Iter	$\text{G}_{\text{MRPPA}'}$
18	$2-1-2-1$	40.059	40	4	NG	160.068	81	23	NG
	$1-2-1-2$	48.945	199	24	NG	158.954	108	9	NG
	$2-2-2-2$	40.059	145	2	NG	158.954	226	21	NG
	$3-2-3-2$	40.057	1960	1	Optimal	70.066	602	2	Optimal
	$2-3-2-3$	40.057	955	1	Optimal	70.066	988	1	Optimal
	3-3-3-3	40.057	2354	1	Optimal	70.066	1636	2	Optimal
	Ave.	41.539	942	5.5	**0**	114.696	607	9.7	**0**
22	$2-1-2-1$	45.712	102	6	4%	83.310	155	4	2%
	$1-2-1-2$	45.712	353	14	4%	78.302	277	8	9%
	$2-2-2-2$	45.712	1241	3	4%	132.700	683	5	NG
	$3-2-3-2$	46.172	>3600	1	3%	481.371	>3600	2	NG
	$2-3-2-3$	46.172	1956	3	3%	249.572	2358	5	NG
	3-3-3-3	65.132	>3600	1	NG	960.086	>3600	1	NG
	Ave.	49.102	>1809	4.7	3.6%	330.890	>1779	4.2	5.5%
26	$2-1-2-1$	NA	663	1	NG	162.150	296	8	NG
	$1-2-1-2$	NA	759	1	NG	83.328	2192	10	90%
	$2-2-2-2$	81.960	2865	21	NG	85.383	3257	3	85%
	$3-2-3-2$	78.168	>3600	1	NG	108.935	>3600	1	45%
	$2-3-2-3$	85.601	>3600	1	NG	223.515	>3600	1	NG
	3-3-3-3	96.110	>3600	1	NG	NA	>3600	1	NG
	Ave.	85.460	>3416	4.3	-	132.662	>2758	4	73.3%

Table 5: The FRPPA′ solutions for instances T18, T22, and T26

T	No. sub-intervals	Cost level = Low				Cost level = High			
	P	$z_{\text{FRPPA}}{}'$	$\text{RT}_{\text{FRPPA}'}$	Iter	$\text{G}_{\text{FRPPA}'}$	z_{FRPPA}	$\text{RT}_{\text{FRPPA}'}$	Iter	$\text{G}_{\text{FRPPA}'}$
18	2	40.057	47	5	Optimal	70.066	42	5	Optimal
	3	40.057	43	3	Optimal	70.066	65	3	Optimal
	4	40.057	179	3	Optimal	70.066	56	3	Optimal
	5	40.057	81	2	Optimal	70.066	64	2	Optimal
	6	40.057	57	1	Optimal	70.066	97	1	Optimal
	Ave.	40.057	81	2.8	**0**	70.066	65	2.8	**0**
22	2	45.712	173	8	4%	83.312	403	9	2%
	3	46.172	299	4	4%	75.087	345	2	14%
	4	45.712	319	4	4%	131.332	1338	4	NG
	5	45.712	685	2	4%	130.395	813	4	NG
	6	45.712	2162	3	4%	72.586	2442	2	18%
	Ave.	45.804	728	4.2	4%	98.542	1068	4.2	11.3%
26	2	61.143	732	12	NG	137.388	1097	5	15%
	3	48.226	2227	4	16%	80.935	>3600	2	95%
	4	48.226	1565	4	16%	80.935	>3600	2	95%
	5	48.226	2195	3	16%	75.104	3383	3	111%
	6	48.226	2439	3	16%	75.104	>3600	2	111%
	Ave.	50.809	1832	5.2	16%	89.893	>3056	2.8	85.4%

By comparing the column $G_{\mathrm{MRPPA}'}$ of Table 4 and the column $G_{\mathrm{FRPPA}'}$ of Table 5, it is observed that MRPPA′ has a gain over BARON in 28% of instances whereas FRPPA′ has a gain over BARON in 47% of instances. Moreover, the average gains obtained by MRPPA′ and FRPPA ′ are 15.6% and 20.2%, respectively. These results confirm the superiority of the FRPPA ′ over MRPPA′ in terms of solution quality. Additionally, by comparing the column $\mathrm{RT}_{\mathrm{MRPPA}'}$ of Table 4 and the column R $\mathrm{T}_{\mathrm{FRPPA}'}$ of Table, it can be concluded that the running time of FRPPA′ is less than that of MRPPA′.

As can be observed in Tables 4 and 5, the execution time of the algorithm significantly increases with an increasing number of sub-intervals. Due to the imposed time limit of 3600 seconds for execution time of each algorithm, the number of iterations is reduced, resulting in a failure to obtain a high-quality feasible solution. If this execution time is not restricted, it is expected to obtain a better-quality solution.

Table 6 summarizes the results of the direct resolution of COSP by BARON in a time limit of 14000 seconds for instances T30L and T30H. The columns of this table have the same definition as in Table 3. The results of algorithms MRPPA′ and FRPPA′ for instances T30L and T30H are provided in Table 7 and Table 8, respectively. As can be seen in these tables, MRPPA′ and FRPPA′ have a gain over BARON in 50% and 100% of implementations. Additionally, the average gains achieved by MRPPA′ and FRPPA′ are 14.5% and 47.7% on T30L, and 6% and 41% on T30H, respectively. Thus, by comparing the columns $G_{\mathrm{MRPPA}'}$ and $G_{\mathrm{FRPPA}'}$ as well as the columns $\mathrm{RT}_{\mathrm{MRPPA}'}$ and $RT_{\mathrm{FRPPA}'}$, it can be concluded that FRPPA′ outperforms MRPPA′ in terms of solution quality and time.

Table 6: The BARON's solutions in a time limit of 14000s

	Size of COSP				Cost level = Low			Cost level = High		
T	Total Var	Total Const	BVar	NLConst	z_{BARON}	GAP	$\mathrm{RT}_{\mathrm{BARON}}$	z_{BARON}	GAP	$\mathrm{RT}_{\mathrm{BARON}}$
30	6030	12294	1890	1050	113.461	71%	14000	425.895	415%	14000

Table 7: The MRPPA′ for instance T30

No. sub-intervals	Cost level = Low				Cost level = High			
L-O-R-S	z_{MRPPA}	$\mathrm{RT}_{\mathrm{MRPPA}}$	Iter	$G_{\mathrm{MRPPA}'}$	$z_{\mathrm{MRPPA}'}$	$\mathrm{RT}_{\mathrm{MRPPA}}$	It′	$G_{\mathrm{MRPPA}'}$
$2-1-2-1$	NA	4000	2	NG	573.615	14000	7	NG
$1-2-1-2$	108.615	12000	6	4%	401.513	14000	7	6%
$2-2-2-2$	90.908	14000	7	25%	487.616	14000	7	NG
Ave.	-	**10000**	**5**	**14.5%**	-	**14000**	**7**	**6%**

Table 8: The FRPPA′ solution for instances T30

No. sub-intervals	Cost level = Low				Cost level = High			
P	$z_{\mathrm{FRPPA}'}$	$\mathrm{RT}_{\mathrm{FRPPA}'}$	Iter	$G_{\mathrm{FRPPA}'}$	$z_{\mathrm{FRPPA}'}$	$\mathrm{RT}_{\mathrm{FRPPA}}$	It′	$G_{\mathrm{FRPPA}'}$
2	66.625	10000	5	70%	324.839	10000	5	31%
3	67.713	8000	4	68%	274.645	12000	6	55%
4	108.034	4000	2	5%	310.113	14000	7	37%
Ave.	-	**7333**	**4**	**47.7%**	-	**12000**	**6**	**41%**

Fig. 2 and Fig. 3 indicate how the best objective function value is improved during the iterations of algorithms MRPPA′ and FRPPA′, respectively for instance T30L. Fig. 4 and Fig. 5 represent the same concept for instance T30H. As can be seen in Fig. 2, MRPPA′ achieves a gain over BARON for two settings of partitions in the first 6000 seconds of the running time; however, as can be seen in Fig. 3, FRPPA′ achieves a gain over BARON for three settings of partitions in the first 4000 seconds of the running time. Moreover, the best solution of FRPPA′ over different settings is found in at most 10000 seconds and then, no further improvement is observed. The same results can be inferred from Fig. 4 and Fig. 5.

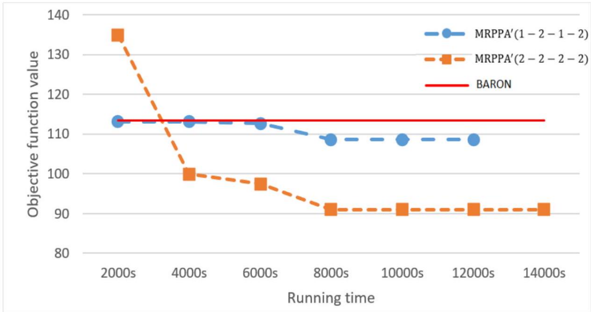

Figure 2: Improvement of the objective value of MRPPA′ for T30L

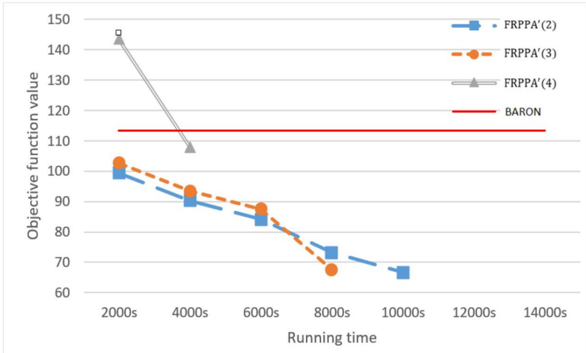

Figure 3: Improvement of the objective value of FRPPA′ for T30L

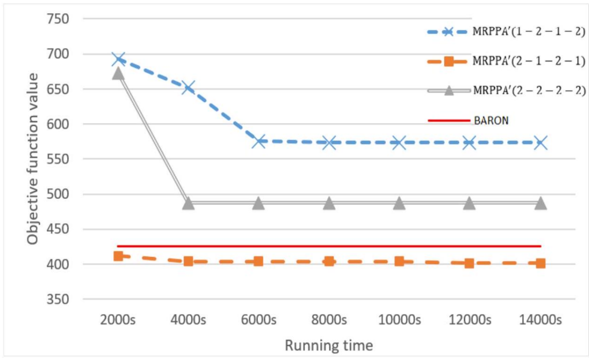

Figure 4: Improvement of the objective value of MRPP$MRPP$.

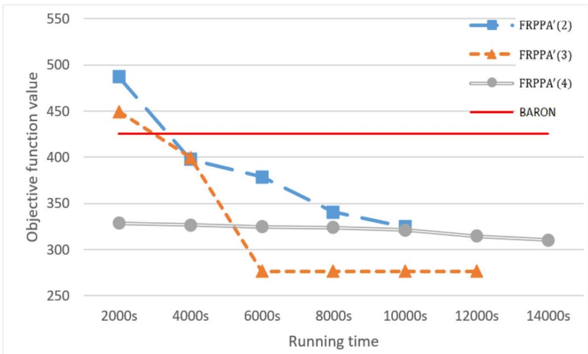

Figure 5: Improvement of the objective function value of FRPPA′ for T30H

5. Conclusion

In this paper, FRPPA′, as a novel two-step MILP-NLP algorithm based on piecewise partitioning and fractional relaxation technique was presented. The proposed algorithm was tested on different instances of COSP taken from the literature. The results confirmed the efficiency of the FRPPA′ over MRPPA′ and direct resolution (by BARON in a given time limit) from both solution quality and running time. In the investigated case study, MRPPA′ had an improvement over BARON in 31% of instances and the average gain was about 15%, whereas FRPPA′ had an improvement over BARON in 64% of instances and the average gain was about 25%. Additionally, the running time of the proposed algorithm is not sensitive to the problem size and shows a linear behavior.

Utilizing FRPPA′ to solve other nonlinear optimization problems, especially those containing fractional terms is suggested as future work. Furthermore, the extension of the COSP to deal with uncertainty in refinery demand or maintenance requirements may lead to a realistic but more complex model, and solving it by an efficient method would be a valuable research direction.

References

[1] M. L. Bergamini, P. Aguirre, and I. Grossmann, *Logic-based outer approximation for globally optimal synthesis of process networks*, Computers & Chemical Engineering, 29 (2005), pp. 1914–1933.

[2] J. Bisschop, *AIMMS - Optimization Modeling*, 2012.

[3] P. M. Castro, *Tightening piecewise mccormick relaxations for bilinear problems*, Computers & Chemical Engineering, 72 (2015), pp. 300–311.

[4] P. M. Castro, *Normalized multiparametric disaggregation: an efficient relaxation for mixed-integer bilinear problems*, Journal of Global Optimization, 64 (2016), pp. 765–784.

[5] P. M. Castro and I. E. Grossmann, *Global optimal scheduling of crude oil blending operations with RTN continuous-time and multiparametric disaggregation*, Industrial & Engineering Chemistry Research, 53 (2014), pp. 15127–15145.

[6] P. M. Castro, I. E. Grossmann, and Q. Zhang, *Expanding scope and computational challenges in process scheduling*, Computers & Chemical Engineering, 114 (2018), pp. 14–42.

[7] J. Cerdá, P. C. Pautasso, and D. C. Cafaro, *Efficient approach for scheduling crude oil operations in marine-access refineries*, Industrial & Engineering Chemistry Research, 54 (2015), pp. 8219–8238.

[8] X. Chen, S. Huang, D. Chen, Z. Zhang, L. Zheng, I. Grossmann, and S. Chen, *Hierarchical decomposition approach for crude oil scheduling: A sinopec case*, Interfaces, 44 (2014), pp. 269–285.

[9] G. Corsano, A. R. Vecchietti, and J. M. Montagna, *Optimal design for sustainable bioethanol supply chain considering detailed plant performance model*, Computers & Chemical Engineering, 35 (2011), pp. 1384–1398. Energy & Sustainability.

[10] R. Dai, H. Charkhgard, and F. Rigterink, *A robust biobjective optimization approach for operating a shared energy storage under price uncertainty*, International Transactions in Operational Research, 29 (2022), pp. 1627–1658.

[11] L. S. de Assis, E. Camponogara, B. Zimberg, E. Ferreira, and I. E. Grossmann, *A piecewise McCormick relaxation-based strategy for scheduling operations in a crude oil terminal*, Computers & Chemical Engineering, 106 (2017), pp. 309–321. ESCAPE-26.

[12] S. S. Dey, A. Santana, and Y. Wang, *New SOCP relaxation and branching rule for bipartite bilinear programs*, Optimization and Engineering, 20 (2019), pp. 307–336.

[13] C. D'Ambrosio, A. Lodi, and S. Martello, *Piecewise linear approximation of functions of two variables in milp models*, Operations Research Letters, 38 (2010), pp. 39–46.

[14] F. Evazabadian, M. Arvan, and R. Ghodsi, *Short-term crude oil scheduling with preventive maintenance operations: a fuzzy stochastic programming approach*, International Transactions in Operational Research, 26 (2019), pp. 2450–2475.

[15] M. Fampa and W. Pimentel, *Linear programing relaxations for a strategic pricing problem in electricity markets*, International Transactions in Operational Research, 24 (2017), pp. 159–172.

[16] D. C. Faria and M. J. Bagajewicz, *A new approach for global optimization of a class of minlp problems with applications to water management and pooling problems*, AIChE Journal, 58 (2012), pp. 2320–2335.

[17] M. Fischetti and M. Monaci, *A branch-and-cut algorithm for mixed-integer bilinear programming*, European Journal of Operational Research, 282 (2020), pp. 506–514.

[18] L. R. Foulds, D. Haugland, and K. Jörnsten, *A bilinear approach to the pooling problem†*, Optimization, 24 (1992), pp. 165–180.

[19] A. Gupte, S. Ahmed, M. S. Cheon, and S. Dey, *Solving mixed integer bilinear problems using MILP formulations*, SIAM Journal on Optimization, 23 (2013), pp. 721–744.

[20] M. M. F. Hasan and I. Karimi, *Piecewise linear relaxation of bilinear programs using bivariate partitioning*, AIChE Journal, 56 (2010), pp. 1880–1893.

[21] D. W. Holland and J. L. Baritelle, *School consolidation in sparsely populated rural areas: A separable programming approach*, American Journal of Agricultural Economics, 57 (1975), pp. 567–575.

[22] F. Hooshmand, M. Jamalian, and S. A. MirHassani, *Efficient two-phase algorithm to solve nonconvex minlp model of pump scheduling problem*, Journal of Water Resources Planning and Management, 147 (2021), p. 04021047.

[23] R. Karuppiah, K. C. Furman, and I. E. Grossmann, *Global optimization for scheduling refinery crude oil operations*, Computers & Chemical Engineering, 32 (2008), pp. 2745–2766. Enterprise-Wide Optimization.

[24] T. Kleinert, V. Grimm, and M. Schmidt, *Outer approximation for global optimization of mixed-integer quadratic bilevel problems*, Mathematical Programming, 188 (2021), pp. 461–521.

[25] H. Lee, J. M. Pinto, I. E. Grossmann, and S. Park, *Mixed-integer linear programming model for refinery short-term scheduling of crude oil unloading with inventory management*, Industrial & Engineering Chemistry Research, 35 (1996), pp. 1630–1641.

[26] L. Liberti, S. Cafieri, and F. Tarissan, *Reformulations in Mathematical Programming: A Computational Approach*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 153–234.

[27] G. P. McCormick, *Computability of global solutions to factorable nonconvex programs: Part i — convex underestimating problems*, Mathematical Programming, 10 (1976), pp. 147–175.

[28] M. Mikolajková, H. Saxén, and F. Pettersson, *Linearization of an minlp model and its application to gas distribution optimization*, Energy, 146 (2018), pp. 156–168. Process Integration for Energy Saving and Pollution Reduction – PRES 2016.

[29] S. Mouret, I. E. Grossmann, and P. Pestiaux, *A new lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling*, Computers & Chemical Engineering, 35 (2011), pp. 2750–2766.

[30] H. Nagarajan, M. Lu, S. Wang, R. Bent, and K. Sundar, *An adaptive, multivariate partitioning algorithm for global optimization of nonconvex programs*, Journal of Global Optimization, 74 (2019), pp. 639–675.

[31] A. d. L. Stanzani, V. Pureza, R. Morabito, B. J. V. d. Silva, D. Yamashita, and P. C. Ribas, *Optimizing multiship routing and scheduling with constraints on inventory levels in a brazilian oil company*, International Transactions in Operational Research, 25 (2018), pp. 1163–1198.

[32] J. P. Teles, P. M. Castro, and H. A. Matos, *Multi-parametric disaggregation technique for global optimization of polynomial programming problems*, Journal of Global Optimization, 55 (2013), pp. 227–251.

[33] ——, *Univariate parameterization for global optimization of mixed-integer polynomial problems*, European Journal of Operational Research, 229 (2013), pp. 613–625.

Please cite this article using:

Mohammad Nasir Yarahmadi, Seyed Ali MirHassani, Farnaz Hooshmand, A new relaxation technique based on fractional representation to solve bilinear models: Application to the long horizon crude oil scheduling problem, AUT J. Math. Comput., 6(2) (2025) 121-142
https://doi.org/10.22060/AJMC.2024.22240.1142