I&EC research
Industrial & Engineering Chemistry Research
ARTICLE
pubs.acs.org/IECR

A Rigorous Mathematical Formulation for the Scheduling of Tree-Structure Pipeline Networks

Diego C. Cafaro and Jaime Cerdá*

INTEC (UNL - CONICET), Güemes 3450, 3000 Santa Fe, Argentina

ABSTRACT: Pipeline networks are the dominant mode of conveying a wide variety of oil refined products from refineries to distribution terminals. Pipeline infrastructure usually comprises trunk lines serving high-volume, long-haul transportation requirements, and delivering lines moving smaller volumes over shorter distances. Lots of products are mostly sent through trunk lines to bulk terminals, while some portions are branched to delivering lines and supplied to nearby market areas. This work presents a novel continuous-time mixed-integer linear (MILP) formulation for the short-term operational planning of tree-structure pipeline systems. The problem goal is to find the optimal schedule of pumping and delivery operations to satisfy all terminal requirements at minimum operating cost. To this end, model constraints strictly monitor the branching of batches and the creation of new interfaces in delivering lines to avoid forbidden sequences and determine additional reprocessing costs. By allowing the transfer of multiple products to delivering lines during a batch injection, longer runs can be executed, and fewer ones are needed to find the optimal solution. The new approach has been successfully applied to three examples, one of them involving a real-world pipeline network. As compared to previous contributions, significant improvements in both solution quality and CPU time have been obtained.

1. INTRODUCTION

Large amounts of petroleum products are carried by pipelines from refineries to distribution terminals because it is safer and less expensive than other modes of transportation. Refined products pipelines can be classified into two major categories: trunk and delivering pipelines. Trunk pipelines serve high-volume, long-haul transportation requirements. An example of a trunk line is the U.S. Colonial Pipeline carrying a wide range of petroleum products from refineries sited at the Gulf Coast to major consumption areas in the East Coast. The average haul length on this type of pipelines is over 1000 km. On the contrary, delivering pipelines transport smaller volumes over shorter distances from bulk terminals to multiple, nearby market areas and feature an average haul length less than 250 km. For instance, a delivering pipeline network distributes oil refined products within the New York Harbor, and from this area to Pennsylvania and upstate New York. Therefore, the major difference between trunk and delivering pipelines is the scale in volume and distance. Delivering pipeline carriers operate lower-diameter ducts closer to the demand points and employ a higher number of smaller storage tanks than trunk carriers. As a result, delivering systems present less flexibility and higher chances for pipeline stoppages because of insufficient storage capacity to accept a shipment at some terminal.

Because different petroleum products are pumped back-to-back into the same pipeline with no physical barrier separating them, smaller batch sizes make interface losses proportionally more important. At intermediate depots along the line, “heart cuts” are normally made. In other words, the interface is allowed to go well past the depot location before the terminal starts receiving product. The product delivery ends before the trailing interface arrives. By taking the heart out of the batch, the quality integrity is ensured.[1] In some cases, when a batch of product is completely diverted to an intermediate delivery location, the associate interface volume may also be transferred to the same terminal. However, it is necessary to make sure that the related preceding and succeeding batches that will be put in contact after the delivery contain compatible products. To avoid mismatches, it is convenient that a little portion of the unloaded batch (the interface volume) remains in the pipeline to separate batches that may carry incompatible products. Moreover, the creation of interface material not only occurs in pipeline transit but also in local piping facilities directing products to storage tanks, and in the tanks themselves. The number of operating tasks performed in depots connected to delivering lines is much larger, causing interface losses to become even more crucial. Throughout this article, we also use the terms secondary, lateral, or split lines to refer to delivering lines, and the term mainline instead of trunk line.

Usually, a pipeline network comprising trunk and delivering lines presents a tree configuration with several secondary lines or branches emerging at different points of the trunk line (see Figure 1). Batches of petroleum products injected at the mainline origin are mostly sent through the trunk line to bulk terminals located at large consumption regions, while some portions of them are branched to delivering lines and transported to nearby large clients and market areas. Batches moving through the trunk line can be directly transferred (“tightlined”) to secondary lines or temporarily stored in storage tanks at branch points before shipping them through delivering pipelines. Planning the injection of new batches at the input station and the simultaneous branching flows to secondary lines, together with product

Special Issue: Puigjaner Issue

Received: July 8, 2010
Accepted: October 20, 2010
Revised: October 9, 2010
Published: December 28, 2010

dx.doi.org/10.1021/ie101462k | Ind. Eng. Chem. Res. 2011, 50, 5064–5085

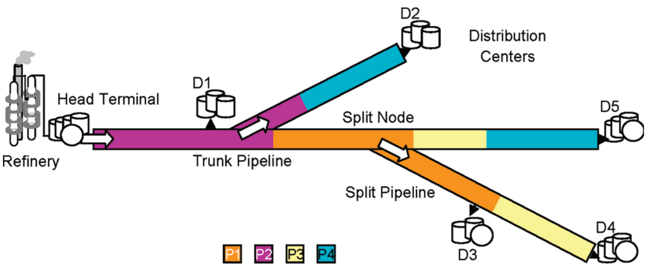

Figure 1. A multiproduct pipeline network with split pipelines.

deliveries to depots across the tree-structure pipeline network, is a complex logistic task that requires efficient supporting tools to do it.

Most contributions on scheduling of refined products pipelines consider a unidirectional trunk line connecting a single origin to multiple receiving terminals. They generally assume product demands at depots with a unique delivery due date. Different types of scheduling methodologies using knowledge-based heuristic techniques,[2] discrete-event simulation tools,[3,4] decomposition frameworks,[5] and rigorous optimization models were proposed. Optimization models can be either discrete or continuous depending on how volume and time domains are treated. Discrete formulations divide the pipeline volume into a significant number of single-product packs, and the planning horizon into time intervals of equal and fixed duration.[6−10] Moreover, Rejowski and Pinto[11] introduced a hybrid optimization approach based on a continuous-time MINLP formulation that still divides the pipeline content into single-product packs. On the other hand, a continuous MILP-formulation in both time and volume was first proposed by Cafaro and Cerdá[12] for the scheduling of a trunk line with a single input station and several distribution terminals. The same authors later extended the MILP model to deal with the operational planning of a similar pipeline system but over a multiperiod rolling horizon and considering multiple delivery due dates at period ends.[13] More recently, they further generalize their mathematical formulation to account for multiple input stations along the trunk line, that is, the multisource pipeline scheduling problem. In this case, pumping runs from different source points can be performed sequentially or simultaneously provided that they are noninteracting runs.[14,15] Dual-purpose terminals that can inject and receive material from the pipeline system were considered. To validate the first formulation of Cafaro and Cerdá,[12] Gleizes et al.[16] developed a discrete event simulation model on Arena platform to generate, through the information provided by the optimization approach, a detailed schedule of pumping and delivery operations. Another continuous MILP representation was presented by Relvas et al.[17] for the scheduling of a pipeline connecting a major refinery to a unique distribution center to meet daily product demands over a monthly horizon. This formulation was subsequently generalized to account for variable flow-rates, pipeline stoppages, and unexpected events.[18] More recently, MirHassani and Jahromi[19] presented a new continuous MILP formulation for scheduling the distribution of petroleum derivatives from a single oil refinery to a number of depots through a tree-structure pipeline network. The model provides both batch input and delivery schedules and explicitly considers the product contamination in trunk and secondary pipelines. However, it assumes that lots of a single product can at most be transferred to a delivering line during a pumping run. Diverting lots of different products to a split line while injecting a new batch at the origin is not a feasible option.

Besides, decomposition-based techniques for the scheduling of real-world pipeline networks with multiple sources, intermediate storage facilities, and final destinations have been developed.[20−23] They mostly rely on four major components: decomposition strategy, heuristic-based product sequencing, discrete event simulation, and optimization models to determine the exact times of batch injections and product deliveries. Interesting features of such pipeline systems are: (i) multiple input stations; (ii) parallel pipeline segments directly connecting a single source node to multiple depots (pipeline branching); (iii) parallel pipeline segments directly linking more than one source to a given depot; (iv) concurrent pumping runs at different input terminals; and (v) reversal flow in some pipelines. Nonetheless, simple principles are usually applied to develop the pipeline schedule as next explained. When a new lot is inserted at the inlet of a pipeline segment, another one with a similar volume at the other extreme of the same segment is pushed to a receiving terminal. Similarly, a pipeline segment can receive material from at most a single source at any time. Lately, Herrán et al.[24] proposed a new mathematical formulation for the short-term operational planning of multipipeline systems with a complex topology that includes most of the features (i)−(v) just described. The model is based on a discrete approach that divides both the planning horizon into time intervals of equal duration and the individual pipelines into packages of equal volume, each one containing a single product.

This work introduces a new MILP continuous model for the scheduling of tree-structure pipeline systems transporting a variety of oil refined products through trunk and delivering lines from a single source to multiple receiving terminals. The model broadens the scope of the formulation of Cafaro and Cerdá[12] by considering the possibility of branching product flows to delivering pipelines and diverting material from batches in trunk and secondary lines to accessible demanding depots during a pumping operation. As a result, it can simultaneously determine the pipeline input and output schedules. In contrast to a previous approach,[19] the transfer of multiple products to a delivering line while injecting a new batch is a feasible operation. Such an important model feature allows one to reduce the number of pumping runs required to discover the optimal pipeline schedule and substantially decreases the CPU time. Three examples, one of them involving a real-world case study, were successfully solved to optimality at low computational cost.

2. TREE-STRUCTURE PIPELINE SYSTEMS

A tree-structure pipeline system consists of a unidirectional trunk line ($l_o \in PL$) and a set of secondary pipelines $\{l_1, l_2, ...l_n\} \subset PL$. In a single-level tree-structure, all the branches emerge from the mainline. Batches of refined products injected at the origin of the trunk line (l_o) may be delivered to a set of distribution terminals ($j \in J_o \subseteq J$) that are accessible from l_o (i.e., the mainline terminals) and/or branched into delivering lines ($\{l_1, l_2, ...l_n\} \subset PL$). Every delivering line $l \neq l_o$ has its own starting or branching point along the mainline at volume coordinate ρ_l from the origin. At that node, lots of products can be directly transferred or "tightlined" from the trunk line to pipeline l without the need of breakout tanks. Coordinates of mainline terminals given by σ_j ($j \in J_o$) are referred to the system origin and represent the volume of the trunk line between the input station and depot j. In turn,

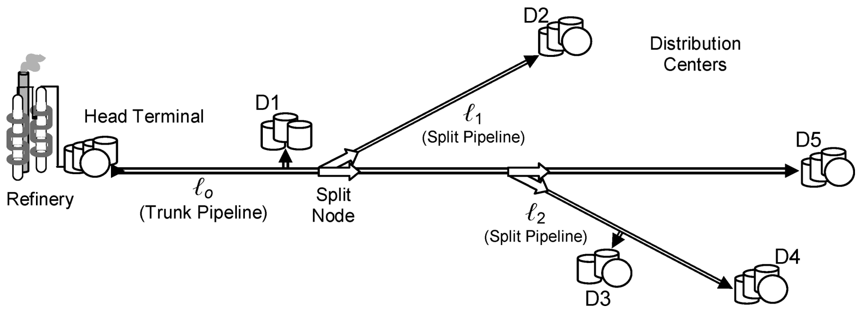

Figure 2. A single-level tree pipeline network.

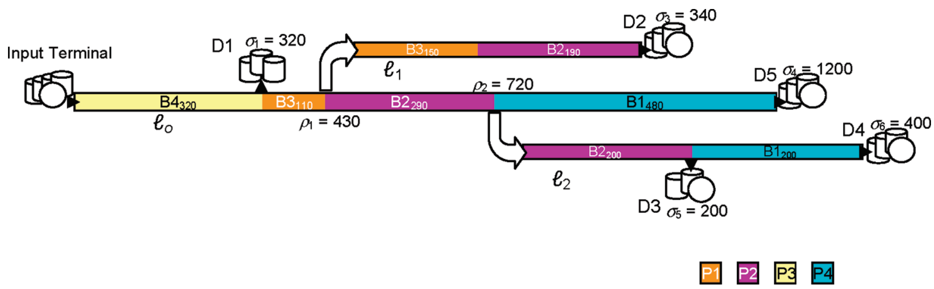

Figure 3. Describing the content of a tree-structure pipeline network.

coordinates for depots connected to a secondary line l ($j \in J_l$, $l \neq l_o$) are measured with regards to the origin of pipeline l and stand for the volume between the branching point and that terminal. Every depot $j \in J$ is connected to a single pipeline l, that is, $J_l \cap J_{l'} = \varnothing$ for $l' \neq l$. Moreover, distribution terminals are strict output nodes, and new product shipments can be just performed from the input station. The mathematical model will be focused on a single-level tree pipeline network where every branch l starts from the mainline and distribute products to receiving terminals $j \in J_l$. On a future publication, the proposed model will be generalized to deal with multilevel tree-structure pipeline networks, where first-level branches are connected to final depots and lower-level delivering lines.

Figure 2 presents an illustrative example of a single-level tree pipeline network. It consists of an input station located close to a major refinery at the origin of a trunk line, where refined product batches are injected. They are destined to five receiving terminals: J = {D1, D2, D3, D4, D5} demanding multiple refined products to meet accepted customer orders. Such terminals are connected to the input station through a set of three pipelines: $PL = \{l_o, l_1, l_2\}$. The trunk pipeline (l_o) is directly connected to depots D1 and D5, that is, J_o = {D1, D5}, while delivering pipelines l_1 and l_2 are branches of the trunk line supplying products to terminals D2 and D3−D4, respectively. Hence, J_1 = {D2} and J_2 = {D3, D4}.

In turn, Figure 3 presents the current content of every single pipeline in the network, the volumetric coordinates of every depot j (σ_j) and the branching point to a delivering line l (ρ_l). There are four batches in the trunk line l_o: $B4_{320}$, $B3_{110}$, $B2_{290}$, and $B1_{480}$, with the subscripts indicating their current volumes in hundreds of m^3. They all were previously injected at the head terminal. On the other hand, the split line l_1 contains two batches ($B3_{150}$, l_1) and ($B2_{190}$, l_1) that were originated by partially branching batches B3 and B2 from the mainline to l_1. Similarly, batches ($B2_{200}$, l_2) and ($B1_{200}$, l_2) moving along branch l_2 arise by partially rerouting B2 again and B1 from l_o to l_2.

3. PROBLEM STATEMENT

Given (a) a multiproduct pipeline network composed by a set of pipeline segments arranged in a tree configuration that connects a single input station to multiple distribution terminals, (b) a set of distribution terminals located along the mainline and delivering pipelines, (c) a planning horizon with a typical length of 1 month, (d) the storage facilities for refined products at every terminal, (e) the set of customer requests at every depot, each one involving a given volume of a certain refined product to be satisfied before the end of the planning horizon, (f) the sequence of "old" batches already inside the pipeline network as well as their contents and locations at the initial time, (g) the scheduled incoming flows from neighboring refineries to storage tanks at the input station, (h) the initial product inventories in storage tanks at the input station and distribution terminals, and (i) the minimum/maximum pipeline pumping rate and the maximum delivery rate from pipeline terminals to local markets, then the problem goal is to optimize the sequence and volumes of new product batches to be pumped into the trunk pipeline (the input schedule) and to determine how they should be split into pipeline branches and delivered to depots (the output schedule) to: (1) meet every product demand at distribution terminals located along trunk/delivering lines in a timely fashion; (2) maintain the inventory level in refinery and terminal tankage within the permissible range; (3) trace the size and location of the inputted batches in every segment of the pipeline network; and (4) minimize the sum of pumping, transition, down-time, back-order, and inventory carrying costs.

4. MODEL ASSUMPTIONS

Some of the model assumptions have already been considered for the scheduling of trunk pipelines. However, several ones just apply to tree-structure pipeline networks. The list of model assumptions is given below.

(1) A tree-structured pipeline network consisting of a unidirectional trunk line l_o and a set of delivering pipelines $\{l_1, l_2, ..., l_n\}$ directly branched from l_o, which transports oil refined products from a single input station to several downstream terminals is considered.

(2) Refined products are injected at the origin of the trunk line where it is located the input station (see Figure 2).

(3) Delivering lines receive flows of products from the mainline and supply them to output terminals. They are not connected to further branches. Future work will be focused on relaxing this assumption.

(4) Every distribution terminal can receive oil refined products from just a single trunk/delivering pipeline.

(5) At branching points, flows of products can be continuously diverted from the mainline to delivering pipelines without using breakout tankage (tightlining operation).

(6) The pumping rate at every pipeline may vary within a common permissible range.

(7) Every pipeline in the network remains completely full of liquid refined products at any time.

(8) Liquid refined products are incompressible fluids. The only way to get a certain volume of product out of the pipeline system is by injecting a similar volume at the origin.

(9) The pipeline system is operated in fungible mode. If individual batches of the same product meet common specifications, they can be consolidated and sent through the mainline as a single batch.

(10) A single lot can be simultaneously stripped out to one or more terminals and/or branched to multiple delivering lines. As a new product batch is injected at the origin, another one flowing through the trunk line can be diverted to a mainline terminal and/or transferred to a branch, while some others continue moving to more distant points.

(11) Product requests at distribution terminals are due at the end of the planning horizon and can be satisfied by diverting material from more than one fungible batch.

(12) Product batches are sequentially pumped into every pipeline with no physical barrier separating them. The interface or contamination volume between any pair of refined products is a known constant, independent of the batch movements. For simplicity, it is assumed that the interface is kept into every pipeline until it reaches the farthest terminal where it is stored and reprocessed.

(13) The unit pumping cost is a known constant that changes with the product and the traveled distance but it is independent of the pumping rate.

(14) The maximum supply rate of refined products from the refinery to the input station tanks is always lesser than the lowest pumping rate of products into the mainline.

5. MODEL VARIABLES AND CONSTRAINTS

The mathematical formulation for the tree-structure pipeline network scheduling problem is defined in terms of four major sets: (a) the trunk and delivering pipelines in the tree-structure transport system ($l \in PL$); (b) the receiving pipeline terminals ($j \in J$); (c) the refined products ($p \in P$) transported by the pipeline network from the input station to depots; (d) the set of scheduled production runs at the refinery ($r \in R$); and (e) the old batches ($i \in I^{old}$) in pipeline transit at time $t = 0$, together with the new batches ($i \in I^{new}$) that can be injected at the origin of the pipeline system over the planning horizon. Any element $i \in I = I^{old} \cup I^{new}$ is regarded as a potential lot flowing through the pipelines by the problem model. The formulation assumes that the elements of set I are arranged in the same order that they were or will be injected at the origin of the mainline. Then, old batches $i \in I^{old}$ will arise first, and the insertion of a new batch i in the mainline should start after completing the injection of batch $(i - 1)$.

5.1. Model Variables. Likewise previous continuous approaches on operational scheduling of trunk pipelines,[12] the proposed mathematical model incorporates the following variables to characterize a new batch $i \in I^{new}$ and the related pumping operation: (a) the set of binary variables $y_{i,p}$ denoting the product assigned to batch i and its existence whenever one of the binaries is equal to 1; (b) the original batch size (Q_i), that is, the amount of product injected in the mainline with batch i; (c) the duration of the related pumping operation (L_i); (d) the completion time of the batch injection (C_i); and (e) the initial injection time ($C_i - L_i$). Because it is assumed that batch $(i - 1)$ precedes batch i in the trunk line, then the tracking of the interface between any pair of consecutive batches and the feasibility of the subsequence $(i - 1, i)$ in the mainline can easily be made. On the other hand, batch movements, batch size changes, and new interface volumes in trunk and lateral pipelines $l \in PL$ when accomplishing a new batch injection are traced through: (f) the upper volumetric coordinate of batch $i \in I$ in line $l \in PL$ after pumping a new batch $i' \in I^{new}$ ($i' \geq i$) at the input station [$F_{i,l}^{(i')}$]; (g) the size of batch $i \in I$ in line $l \in PL$ at the end of pumping run i' [$W_{i,l}^{(i')}$]; and (h) the volume of a new interface between batch i and the preceding lot in line $l \in PL$ [$WIF_{i,p,p',l}$], assuming that they contain products p' and p. Besides, the feasibility and the extent of product deliveries to depots $j \in J_l$ during the injection of batch $i' \in I^{new}$ are controlled by: (i) the binary variable $x_{i,j}^{(i')}$ denoting that depot $j \in J_l$ located along line $l \in PL$ is accessible from batch i during run $i' \geq i$, whenever $x_{i,j}^{(i')} = 1$; (j) the continuous variable $D_{i,j}^{(i')}$ representing the amount of material transferred from batch $i \in I$ in line $l \in PL$ to depot $j \in J_l$ during run $i' \in I^{new}$.

To handle tree-structure pipeline networks, additional variables are defined with respect to the formulation of Cafaro and Cerdá[12] for trunk lines. They are: (k) the binary variable $w_{i,l}$ denoting the existence of batch (i, l) moving along the lateral pipeline $l \neq l_o$; (l) the binary variable $wl_{i,l}^{(i')}$ indicating that line $l \neq l_o$ is accessible from batch (i, l_o) in the mainline during run $i' \geq i$; and (m) the continuous variable $T_{i,l}^{(i')}$ representing the amount of product diverted from batch (i, l_o) to lateral pipeline $l \neq l_o$. Variable $T_{i,l}^{(i')}$ can be greater than zero only if $wl_{i,l}^{(i')} = 1$.

Because the total number of new batches to inject in the pipeline network is not known beforehand, some elements of the set I^{new} may not be necessary at the optimum. Such entities never pumped into the mainline will be called "globally fictitious batches", featuring $y_{i,p} = 0$, $\forall\ p \in P$. On the other hand, the condition $wl_{i,l}^{(i')} = 1$ for some $i' \in I^{new}$ implies that $w_{i,l} = 1$, and consequently $y_{i,p} = 1$ for a certain product $p \in P$. However, some existing batch (i, l_o) traveling along the mainline may never be transferred to some lateral pipeline l. In that case, the predefined batch (i, l) for the lateral pipeline $l \neq l_o$ does not exist. It is said that batch (i, l) is a locally fictitious lot in pipeline l, and $w_{i,l} = 0$.

The other model variables monitoring product inflows/outflows and inventory levels in refinery and depot tanks to avoid overloadings or shortages [$SL_{i,r}$, $SU_{i,r}$, $IRF_p^{(i)}$, $IRS_p^{(i)}$], and tracking product deliveries from depots to consumer markets ($DM_{p,j}$), are similar to the ones already defined by Cafaro and Cerdá[12] for that purpose. All model variables are listed in the Nomenclature section of this article.

5.2. Model Constraints. Problem constraints can be grouped into three major categories: (1) Batch sequencing constraints that define the string of batch injections (product, volume, starting time, and duration) at the input station together with the interface size between any pair of consecutive lots in every line $l \in PL$, and prevent the execution of forbidden product sequences. Special constraints are included within this category for both determining the size of new interfaces arising in secondary lines and avoiding forbidden product sequences in pipelines $l \neq l_o$. (2) Batch tracking equations that define batch movements and batch size changes in trunk/secondary pipelines during each pumping run. They not only provide the location of every lot traveling through the pipeline network but also the batch size at the end of any injection. To do so, this block of equations should also determine the amount of material diverted to split lines and the product delivery flows from pipelines to depots during a pumping operation. More important, they monitor the feasibility of branching and stripping operations before they are planned. In contrast to previous approaches, the branching of batches containing different products to lateral pipelines during the same pumping run is a feasible operation. (3) Inventory management and demand constraints guarantee that inventory levels in refinery and depot tanks remain within the permissible range at the start/end of a batch injection, and customer demands placed at distribution terminals are fully satisfied. When backordered demands are allowed, these constraints provide their values at every terminal so that the objective function charges the corresponding backorder penalty cost. In addition, there is a small group of constraints defining the size and location of every old batch already in the pipeline network at $t = 0$. They are referred to as the initial conditions.

6. MATHEMATICAL FORMULATION

6.1. Batch Sequence Constraints. *6.1.1. Product Allocation.* Every new batch pumped into the trunk line l_o at the origin will contain at most a single refined petroleum product.

$$\sum_{p \in P} y_{i,p} \leq 1 \quad \forall\ i \in I^{new} \tag{1}$$

In fact, the variable $y_{i,p}$ denotes the existence of a new batch i in the pipeline system when $y_{i,p} = 1$. Globally fictitious batches never pumped into the mainline feature $y_{i,p} = 0$, $\forall\ p \in P$.

6.1.2. Batch Sequencing. The injection of a new batch $i \in I^{new}$ in the trunk line l_o should start after dispatching the previous one $(i - 1)$ and performing the subsequent changeover operation.

$$C_i - L_i \geq C_{i-1} + \tau_{p,p'}(y_{i-1,p} + y_{i,p'} - 1) \quad \forall\ i \in I^{new}; p, p' \in P \tag{2}$$

$$L_i \leq C_i \leq h_{max} \quad \forall\ i \in I^{new} \tag{3}$$

Variable C_i is the completion time for the pumping run of batch $i \in I^{new}$, L_i is the run duration, and h_{max} is the overall length of the scheduling horizon. Because continuous variables C_i and L_i are related to pumping runs always accomplished at the origin of the mainline, the subscript l_o has been omitted. Constraint 2 becomes active only if new batches $(i - 1)$ and i contain products p and p', respectively. For every pair of nonfictitious batches $(i - 1, i)$, only one of constraints 2 will become binding at every feasible solution.

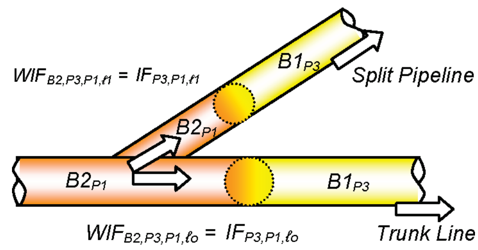

Figure 4. Interface material between batches B2 and B1 in both trunk and lateral pipelines.

6.1.3. Pumping Run Duration. If Q_i is the volume of the new batch i injected in the trunk line, the duration of the related run (L_i) should satisfy the following condition:

$$vb_{min}L_i \leq Q_i \leq vb_{max}L_i \quad \forall\ i \in I^{new} \tag{4}$$

where $[vb_{min}, vb_{max}]$ stands for the permissible pumping rate range. Because Q_i is the volume of batch i pumped into the pipeline system at the origin, the subscript l_o is again omitted. Besides, L_i must belong to the interval $[l_{min,p}, l_{max,p}]$ specified by the pipeline operator for injections of product p.

$$\sum_{p \in P} y_{i,p} l_{min,p} \leq L_i \leq \sum_{p \in P} y_{i,p} l_{max,p} \quad \forall\ i \in I^{new} \tag{5}$$

Constraint 5 just applies in case batch i is actually pumped into the mainline $(\sum_p y_{i,p} = 1)$. To avoid multiple equivalent solutions, fictitious batches $i \in I^{new}$ featuring $\sum_p y_{i,p} = 0$ and therefore $L_i = 0$ and $Q_i = 0$ are moved to the end of the batch sequence by restriction 6.

$$\sum_{p \in P} y_{i,p} \leq \sum_{p \in P} y_{i-1,p} \quad \forall\ i \in I^{new} \tag{6}$$

6.1.4. Interface Material between Consecutive Batches in the Trunk Line. Old and new batches $i \in I$ flowing through the trunk line l_o are arranged in the same order that they were injected. As a result, batch (i, l_o) directly follows lot $(i - 1, l_o)$ previously injected in the mainline. Because separation devices are rarely used, the volume of the interface loss between such consecutive batches in the trunk line l_o will never be lower than the parameter $if_{p,p',lo}$, if batches $(i - 1, l_o)$ and (i, l_o) contain products p and p', respectively. Similar to previous approaches,[12,13] the value of $if_{p,p',lo}$ for any ordered pair of products (p, p') is assumed to be known and independent of the pump rate and the number of stoppages. Transmix volumes are traced along the pipeline from the source point to the farthest destination, where they are removed and stored in separate tanks. The interface volume generated by every new lot i injected in the trunk line is given by constraint 7a.

$$WIF_{i,p,p',lo} \geq if_{p,p',lo}(y_{i-1,p} + y_{i,p'} - 1) \quad \forall\ i \in I^{new}, i > first(I^{new}), p, p' \in P \tag{7a}$$

Because the initial linefill is given, it is also known the product p_o last inserted in the mainline in the previous horizon. The interface volume generated by the first batch pumped in the trunk line during the present horizon will then be given by:

$$WIF_{i,po,p',lo} \geq if_{po,p',lo} y_{i,p'} \quad \forall\ i = first(I^{new}), p_o = P_{i-1}, p' \in P \tag{7b}$$

Moreover, interface volumes between batches already in the mainline at $t = 0$ are known data given by restriction 7c.

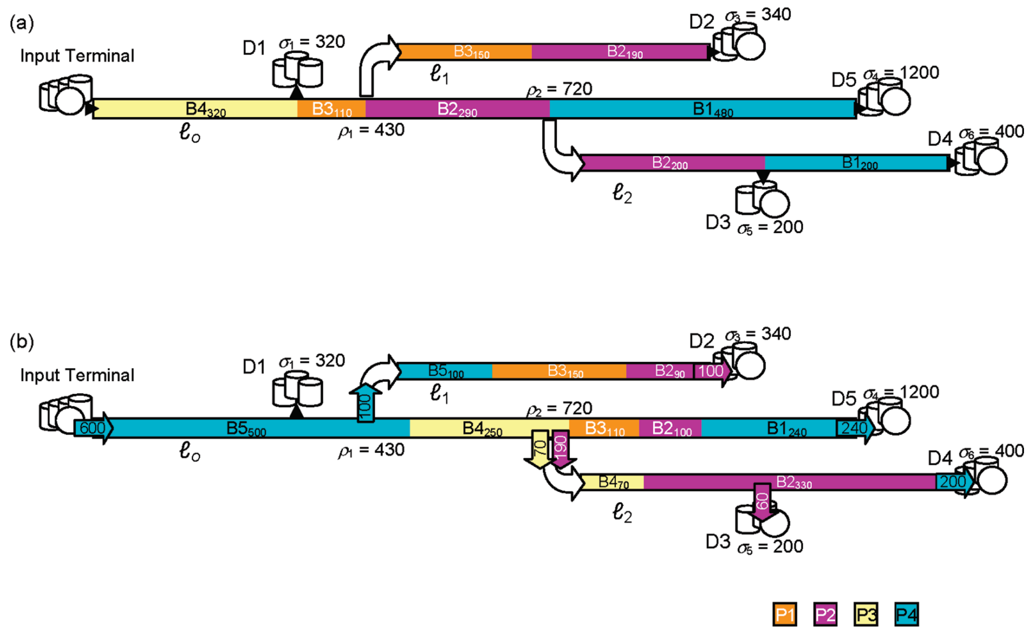

Figure 5. Pipeline network state before and after injecting lot B5.

$$WIF_{i,p,p',l_o} = if_{p,p',l_o} \quad \forall\ i \in I^{old}, i > 1, p = P_{i-1}, p' = P_i \tag{7c}$$

However, new batch interface losses can arise in lateral pipelines in addition to those occurring in the transmission line (see Figure 4). Section 6.1.6 is devoted to the estimation of interface volumes in lateral or delivering pipelines.

6.1.5. Forbidden Product Sequences in the Trunk Line. Because of product contamination, some sequences are strictly forbidden. If FS represents the set of forbidden product sequences and $(p, p') \in FS$, then batches containing products p and p' must never be consecutively pumped into the trunk line. To this end, constraints 8a and 8b have been included in the problem formulation. Condition 8b just applies to the new batch i first inserted in the pipeline system during the current horizon.

$$y_{i-1,p} + y_{i,p'} \le 1 \quad \forall\ i \in I^{new}, i > first(I^{new}), (p, p') \in FS \tag{8a}$$

$$y_{i,p} \le 0 \quad \forall\ i = first(I^{new}), p_o = P_{i-1}, (p_o, p) \in FS \tag{8b}$$

6.1.6. Interface Material between Consecutive Batches in Delivering Pipelines. One of the major difficulties arising in the operational planning of tree-structured pipeline systems is the tracking of product sequences and interfaces in secondary or delivering lines. In contrast to previous approaches,[19,24] the proposed formulation allows making a rigorous tracing of product batches in branches to know from which batch in the trunk line they were originated. A batch in line l diverted from batch i previously pumped at the origin of the trunk line is identified as batch i in branch l, that is, the batch (i, l). To denote the existence of batch (i, l), it is introduced a new binary variable $w_{i,l}$ that is equal to one only if the old/new lot i has been partially or completely transferred from the mainline l_o to branch l during the current horizon.

Every lot i ($>i'$) diverted to line l may generate a new interface with batch (i', l) previously pushed into that line. However, lot i' may not be the one injected right before lot i in the trunk line; that is, it is not necessarily $(i - 1)$. In other words, the product sequencing in branch l usually differs from the one adopted for the trunk line. Let us consider a simple example. Suppose that lot i is transferred to branch l directly after lot $(i - n)$, with $n > 1$. The sequence of lots $\{(i - n + 1), ..., (i - 1)\}$ traveling back-to-back along the trunk line will overpass the interconnection to branch l without diverting material at all to that line. Hence, the interfaces $[(i - n), (i - n + 1)]$, ..., $[(i - 1), i]$ are not regenerated in branch l. Instead, a new interface between lots $(i - n, l)$ and (i, l) arises in line l. It is important to note that not only new batches are redirected to delivering lines in the present horizon. Old lots $i \in I^{old}$ that are in the trunk line at $t = 0$ may also be branched into lateral pipelines while pumping new batches, thus generating further interfaces.

Therefore, a new interface between batches i and i' ($<i$) in the lateral line l is created only if the following two conditions hold: (a) such old/new batches are transferred to branch l during the current horizon ($w_{i,l} + w_{i',l} = 2$), and (b) the sequence of batches $(i' + 1)$ to $(i - 1)$ flowing between batches i' and i in the trunk line l_o is not diverted at all to branch l during the current horizon ($\sum_{i'<k<i} w_{k,l} = 0$). Condition (a) also holds if i' is the last batch in the initial linefill of pipeline l and the old/new batch i is the first lot transferred to line l during the current horizon. Assuming that batches i and i' ($<i$) satisfy the above two conditions and contain products p and p', respectively ($y_{i,p} + y_{i',p'} = 2$), the interface volume between them should not be lower than the characteristic interface size between lots of products p' and p in branch l, that is, the parameter $if_{p',p,l}$:

$$WIF_{i,p',p,l} \ge if_{p',p,l}\left(y_{i,p} + y_{i',p'} + w_{i,l} + w_{i',l} - \sum_{k=i'+1}^{i-1} w_{k,l} - 3\right) \forall i', i \in I, i' < i, (p', p) \notin FS, l \ne l_o \tag{9}$$

Restriction 9 becomes active just in case batch i is directly preceded by another lot i' ($<i$) in branch l ($\sum_{i'<k<i} w_{k,l} = 0$), and additionally they contain products p and p', respectively. Indeed, constraint 9 assumes that lots $[i', i]$ are both new batches partially

or completely diverted to branch l during the current horizon. When either lot i' or both $[i', i]$ are old batches rerouted to line l in the present horizon, the formulation of constraint 9 becomes simpler. Products contained in old batches are problem data, and the values of the associated binaries $y_{i,p}/y_{i',p'}$ are known in advance. If $P_{i'}$ is the product transported by the old batch i', then $y_{i',Pi'} = 1$ and $y_{i',p} = 0$ for any $p \neq P_{i'}$. Moreover, those instances of constraint 9 bounding the size of the new interface between batch (i, l) and the old lot (i', l) containing $P_{i'}$ $(WIF_{i,Pi',p,l})$ will just account for nonforbidden product sequences $(P_{i'}, p) \notin FS$. If instead $[i', i]$ are both old batches that reside in the trunk line l_o at $t = 0$ and, in addition, those batches transport products $P_{i'}$ and P_i, respectively, just the ordered pair of products $(P_{i'}, P_i) \notin FS$ is considered to define a lower bound on $WIF_{i,p',p,l}$. On the other hand, it may also occur that the old batch i' containing $P_{i'}$ is the last element in the initial linefill of line l. If so, the formulation of restriction 9 becomes much simpler, because $w_{i',l} = 1$. In this way, particular forms of constraint 9 when one or both lots i and i' branched to line l are old batches can be easily derived. Finally, interface losses between batches already into secondary pipelines at $t = 0$ are model parameters given by:

$$WIF_{i,p',p,l} = if_{p',p,l} \quad \forall\, i, i' \in I^{old}, p' = P_{i'}, p = P_i, l \neq l_o, (i', i) \in TS_l^o \tag{10}$$

where $TS^o{}_l$ stands for the set of adjacent batches located in branch l at $t = 0$.

Let us consider an example to illustrate the simpler formulation of constraint 9 when old batches are branched to secondary lines. Figure 5a presents the initial state and the product transported by every lot $i \in I^{old}$ = {B1(P4), B2(P2), B3(P1), B4(P3)} in the pipeline network comprising a mainline and two delivering pipelines. The initial batch sequence in the trunk line l_o is {B4(P3), B3(P1), B2(P2), B1(P4)}, while the starting linefills in branches l_1 and l_2 are {B3(P1), B2(P2)} and {B2(P2), B1(P4)}, respectively. Figure 5b illustrates the position and volume of all the batches in pipeline transit after injecting the new batch B5 with product P4 at the origin of the trunk line. Colored arrows indicate product deliveries from the mainline to secondary lines and terminal tankage during the pumping run of batch B5. Because batch B4 is not branched to line l_1, the interface P1−P3 traveling along the trunk line between lots B3 and B4 is not regenerated in line l_1. Instead, a new interface P1−P4 is created by diverting a portion of the new batch B5 into line l_1 immediately after lot (B3, l_1). Besides, a part of lot B4 is rerouted to branch l_2, thus generating a new interface P2−P3 that is not present in the mainline. In this simple example, B2 and B4 are both old batches containing products P2 ($y_{B2,P2} = 1$) and P3 ($y_{B4,P3} = 1$), whereas B2 is the last lot in the initial linefill of branch l_2 ($w_{B2,l2} = 1$). When lots $[i', i]$ are old batches containing products $[P_{i'}, P_i]$, and lot i' is initially the last one in branch l, the formulation of constraint 9 providing a lower bound on the size of the potential new interface $P_{i'} - P_i$ in line l reduces to:

$$WIF_{i,p',p,l} \geq if_{p',p,l}\left(w_{i,l} - \sum_{k=i'+1}^{i-1} w_{k,l}\right) \quad \forall\, i, i' \in I^{old}, i' < i, p = P_i, p' = P_{i'} \tag{9'}$$

6.1.7. Forbidden Product Sequences in Secondary Lines. By selectively transferring material from lots in transit along the trunk line to secondary pipelines, the original batch sequence no longer arise in branches without changes. Although restrictions 8a,b prevent from consecutively shipping forbidden product sequences $(p', p) \in FS$ through the trunk line, they do not avoid interfaces between noncompatible products in lateral pipelines. Consequently, an additional constraint must be imposed to not diverting unsuited products one after the other into any secondary line l.

$$y_{i,p} + y_{i',p'} + w_{i,l} + w_{i',l} - \sum_{k=i'+1}^{i-1} w_{k,l} - 3 \leq 0 \quad \forall i', i \in I^{new}, i' < i, (p', p) \in FS, l \neq l_o \tag{11}$$

If the interface $p'-p$ is noneligible, and new batches $[i', i]$ containing such products ($y_{i',p'} = y_{i,p} = 1$) are diverted to line l ($w_{i',l} = w_{i,l} = 1$), constraint 11 guarantees that one or more intermediate lots will be traveling between lots (i', l) and (i, l). Similarly to constraint 9, a simpler formulation of restriction 11 results when (a) lot i or both $[i', i]$ are old batches located in the trunk line at $t = 0$, and (b) lot i' is the last one diverted to line l in the previous horizon while old/new batch i is the first rerouted to line l during the current horizon.

6.2. Batch-Tracing Constraints. *6.2.1. Pipeline Coordinates of Batch $i \in I$ in Pipeline $l \in PL$ at Time Point $C_{i'}$.* Let $F_{i,l}^{(i')}$ denote the upper volumetric coordinate of batch $i \in I$ in pipeline l when completing the injection of batch $i' \in I^{new}$ $(i' \geq i)$ in the trunk line. Such a continuous variable represents the volume between the origin of pipeline l (i.e., the bifurcation point for a lateral pipeline) and the interface between batch (i, l) and its predecessor in the same line, at the completion time of run i' $(C_{i'})$. The value of $F_{i,l}^{(i')}$ is found by adding the upper coordinate of batch i_1 directly chasing lot i in line l $(F_{i1,l}^{(i')})$ plus the content of lot (i, l) given by $W_{i,l}^{(i')}$, both at time $C_{i'}$. However, the direct successor of lot (i, l) is not known beforehand. To overcome this problem, the batch-tracing constraint 12 is imposed to every pair of consecutive batches $(i, i+1)$ in the set I.

$$F_{i+1,l}^{(i')} + W_{i,l}^{(i')} = F_{i,l}^{(i')} \quad \forall\, l \in PL, i \in I, i' \in I^{new}, i' \geq i \tag{12}$$

If the model decides not to transfer lot $(i + 1)$ from the mainline to the secondary pipeline $l \neq l_o$ but divert a portion of batch $(i + 2)$ into l, then the batch chasing (i, l) will be $(i+2, l)$. In that case, $W_{i+1,l}^{(i')} = 0$ for any $i' \in I^{new}$, and constraints 12 for the pairs $[i+1, i+2]$ and $[i, i+1]$ reduce to $F_{i+2,l}^{(i')} = F_{i+1,l}^{(i')}$, and $F_{i+1,l}^{(i')} + W_{i,l} = F_{i,l}^{(i')}$. Hence, $F_{i+2,l}^{(i')} + W_{i,l} = F_{i,l}^{(i')}$ and the model is able to identify the lot chasing (i, l) in the lateral pipeline. Similarly, if lot i is directly followed by $(i + n)$ in pipeline l, eq 12 will lead to $F_{i+n,l}^{(i')} + W_{i,l}^{(i')} = F_{i,l}^{(i')}$ because lots $(i + 1)$ to $(i + n - 1)$ will be fictitious batches with null size in pipeline l. Fictitious batches in intermediate positions are strictly forbidden by conditions 4−6 in the trunk line, but they are allowed in lateral pipelines.

Revisiting the example illustrated in Figure 5, it can be mentioned that batches B4 and B3 become fictitious elements in pipelines l_1 and l_2, respectively. As a result, the pipeline coordinates of batches (B3, l_1) and (B2, l_2) provided by eq 12 are: (a) $F_{B5,l1}^{(B5)} + W_{B3,l1}^{(B5)} = F_{B3,l1}^{(B5)}$, because $W_{B4,l1}^{(B5)} = 0$, and (b) $F_{B4,l2}^{(B5)} + W_{B2,l2}^{(B5)} = F_{B2,l2}^{(B5)}$, because $W_{B3,l2}^{(B5)} = 0$.

6.2.2. Material Diverted from a New Batch $i \in I^{new}$ to Depots and/or Secondary Pipelines while Being Injected in the Mainline. Let $W_{i,lo}^{(i)}$ be the volume of batch $i \in I^{new}$ in the main pipeline at the completion time of its own pumping run (C_i). If Q_i is the total volume injected in the pipeline system through batch

i, then $[Q_i - W_{i,lo}{}^{(i)}]$ is the amount of material transferred from batch i to main depots ($j \in J_o$) and/or to secondary pipelines ($l \neq l_o$) while being pumped into the trunk line, that is, during the time interval $[C_i - L_i, C_i]$. Obviously, $Q_i \geq W_{i,lo}{}^{(i)}$ and the lower coordinate of batch i in the trunk line at time C_i is equal to zero.

$$Q_i = W_{i,lo}{}^{(i)} + \sum_{j \in Jo} D_{i,j}{}^{(i)} + \sum_{\substack{l \in PL \\ l \neq lo}} T_{i,l}{}^{(i)} \tag{13}$$

$$F_{i,lo}{}^{(i)} - W_{i,lo}{}^{(i)} = 0 \quad \forall\ i \in I^{new}$$

In eq 13, $T_{i,l}{}^{(i)}$ and $D_{i,j}{}^{(i)}$ represent the volume of product transferred from batch i to lateral pipeline l and to mainline terminal j during run i, respectively. In turn, PL is the set of lines in the pipeline network including the trunk line l_o.

6.2.3. Volume of Batch $i \in I$ Flowing through the Mainline after Pumping a Later Batch $i' \in I^{new}$. By definition, $C_{i'}$ is the time at which the injection of the new batch $i' \in I^{new}$ into the pipeline system has been completed. Let us assume that a finite portion of batch $i \in I$ ($i < i'$) is located in the mainline right before injecting i'. The volume of batch i in the trunk line at the completion time $C_{i'}$ is given by the difference between its size at time $C_{i'-1}$ and the total volume transferred from batch i to mainline depots ($j \in J_o$) and/or secondary pipelines ($l \neq l_o$) while injecting batch i'.

$$W_{i,lo}{}^{(i')} = W_{i,lo}{}^{(i'-1)} - \sum_{j \in Jo} D_{i,j}{}^{(i')} - \sum_{\substack{l \in PL \\ l \neq lo}} T_{i,l}{}^{(i')} \quad \forall\ i \in I, i' \in I^{new}, i' > i \tag{14}$$

6.2.4. Volume of Batch $i \in I$ Flowing through Secondary Pipelines at the Completion Time of a New Pumping Run. The volume of batch $i \in I$ traveling along a secondary pipeline ($l \neq l_o$) may change because of two reasons: (a) it receives a further amount of product from the mainline, and/or (b) it delivers some amount of material to depots $j \in J_l$ on branch l. Hence, the volume of batch i into the split pipeline l at time $C_{i'}$ is obtained by adding the further portion of batch i transferred from the mainline $[T_{i,l}{}^{(i')}]$, and subtracting the total quantity of product delivered from batch i to depots $j \in J_l$ given by $[\sum_{j \in Jl} D_{i,j}{}^{(i')}]$ while injecting a new batch i'.

$$W_{i,l}{}^{(i')} = W_{i,l}{}^{(i'-1)} + T_{i,l}{}^{(i')} - \sum_{j \in J_l} D_{i,j}{}^{(i')} \quad \forall\ l \in PL, l \neq l_o, i \in I, i' \in I^{new}, i' > i \tag{15}$$

In contrast to previous approaches,[19,24] the proposed formulation is capable of managing multiple product deliveries to lateral pipelines during a single pumping run. In other words, more than one product can be branched to split pipeline l during a new pumping run i', if they are correctly positioned in the mainline. This model feature significantly decreases the required number of new batch injections and the cardinality of set I, thus reducing the model size. Furthermore, a single batch can be branched into two or more (consecutive/nonconsecutive) split lines during the same pumping run. As shown in Figure 5, the injection of lot B5 pushes an additional volume of batch B2 into pipeline l_2 (190 units of P2), and during the same pumping operation, a portion of batch B4 (70 units of P3) is transferred to line l_2 when lot B4 reaches the bifurcation point.

6.2.5. Feasibility Conditions for Delivering Flows of Products to Mainline Depots. Diverting material from batch (i, l_o) to depot $j \in J_o$ is feasible only if the interconnection to depot j is accessible from batch (i, l_o) while pumping a later batch $i' \in I^{new}$ ($i' \geq i$). To fulfill such feasibility conditions, it is required that:

(a1) The upper coordinate of batch i at time $C_{i'}$ decreased by the volume of the interface material ($\sum_p \sum_{p'} WIF_{i,p'p,lo}$) should never be lower than the jth-terminal coordinate σ_j (except for the farthest depot of the mainline, where interface material is removed). The feasibility condition for the farthest depot $|J_o|$ is achieved when $F_{i,lo}{}^{(i')} = \sigma_{\|Jo\|}$.

(b1) The lower coordinate of batch i at time $C_{i'-1}$ must be less than the depot coordinate σ_j by at least a certain volume φ. The value of φ represents the maximum volume of product that can be diverted from batch i to mainline terminals up to depot j (including j) and to upstream lateral pipelines featuring $\rho_l \leq \sigma_j$, while pumping batch i'. Parameter ρ_l is the coordinate of the interconnection point between the trunk line and the upstream lateral pipeline l.

Let $x_{i,j}{}^{(i')}$ be a binary variable denoting that batch i is diverted to the jth-terminal tankage while injecting batch i' ($x_{i,j}{}^{(i')} = 1$). Otherwise, $x_{i,j}{}^{(i')} = 0$ and no material will be transferred from batch i to depot j. Note that $x_{i,j}{}^{(i')}$ can be driven to zero because of three reasons: (i) batch i does not still reach the interconnection to terminal $j \in J$, (ii) batch i has already overpassed the interconnection to terminal $j \in J$, or (iii) the model decides to not divert material to depot j despite it has an adequate location to do it. Therefore:

$$d_{min} x_{i,j}{}^{(i')} \leq D_{i,j}{}^{(i')} \leq d_{max} x_{i,j}{}^{(i')} \quad \forall\ i \in I, i' \in I^{new}, i' \geq i, j \in J \tag{16}$$

where d_{min}/d_{max} stand for the lower/upper bound on the amount of material that can be transferred from batch i to depot j. Moreover, constraints 17 and 18 represent the feasibility conditions a1 and b1, respectively.

$$F_{i,lo}{}^{(i')} - \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} WIF_{i,p',p,lo} \geq \sigma_j x_{i,j}{}^{(i')} \quad \forall i \in I, i' \in I^{new}, i' \geq i, j \in J_o, j < |J_o|$$
$$F_{i,lo}{}^{(i')} \geq \sigma_j x_{i,j}{}^{(i')} \quad \forall i \in I, i' \in I^{new}, i' \geq i, j \in J_o, j = |J_o| \tag{17}$$

$$F_{i,lo}{}^{(i'-1)} - W_{i,lo}{}^{(i'-1)} + \sum_{k=1}^{j} D_{i,k}{}^{(i')} + \sum_{\substack{l \neq lo \\ \rho_l \leq \sigma_j}} T_{i,l}{}^{(i')} \leq \sigma_j + (pv_{lo} - \sigma_j)(1 - x_{i,j}{}^{(i')}) \forall i \in I, i' \in I^{new}, i' > i, j \in J_o \tag{18}$$

Parameter pv_{lo} is the total volume of the trunk line.

Constraint 18 becomes active whenever $x_{i,j}{}^{(i')} = 1$ and a portion of batch (i, l_o) is diverted from the mainline to depot j during the injection of the new batch $i' > i$. In that case, constraint 18 reduces to the following expression:

$$F_{i,lo}{}^{(i'-1)} - W_{i,lo}{}^{(i'-1)} + \sum_{k=1}^{j} D_{i,k}{}^{(i')} + \sum_{\substack{l \neq lo \\ \rho_l \leq \sigma_j}} T_{i,l}{}^{(i')} \leq \sigma_j$$

dx.doi.org/10.1021/ie101462k |Ind. Eng. Chem. Res. 2011, 50, 5064–5085

or equivalently

$$D_{i,j}^{(i')} \leq \sigma_j - (F_{i,lo}^{(i'-1)} - W_{i,lo}^{(i'-1)}) - \sum_{k=1}^{j-1} D_{i,k}^{(i')} - \sum_{\substack{l \neq lo \\ \rho_l \leq \sigma_j}} T_{i,l}^{(i')} \tag{18'}$$

If a portion of batch (i, l_o) is delivered to depot j during run i' ($x_{i,j}^{(i')} = 1$), the volume transferred ($D_{i,j}^{(i')}$) is bounded by the RHS of constraint 18′. Because of the unidirectional flow condition, just the portion of batch (i, l_o) located upstream of the terminal site (σ_j) before the execution of pumping run i' can be diverted to depot j. The first two terms of the RHS account for that bound on the delivery size. However, such a portion of the batch could also be transferred to other depots or lateral pipelines located upstream of terminal j, that is, $k \in J_o$ with $k < j$, and $l \neq l_o$ featuring $\rho_l \leq \sigma_j$. The last two terms of inequality 18′ account for such potential deliveries to upstream depots and lateral pipelines that further reduce product deliveries to mainline terminals. Note that if $\sigma_j < (F_{i,lo}^{(i'-1)} - W_{i,lo}^{(i'-1)})$, then batch (i, l_o) has surpassed destination j before the execution of run i', and expression 18′ would make $D_{i,j}^{(i')} < 0$. Therefore, the only way to meet restriction 18 is by doing $x_{i,j}^{(i')} = 0$ so that it becomes a redundant constraint.

6.2.6. Feasibility Conditions for Delivering Flows of Products to Secondary Pipelines. The transfer of material from batch (i, l_o) to branch $l \neq l_o$ is feasible only if the bifurcation node to line l is accessible from that lot. Similarly to the previous section, such condition requires that: (a2) the upper coordinate of batch (i, l_o) at time $C_{i'}$ decreased by the volume of the interface material should never be lower than the coordinate of the bifurcation point, ρ_l; and (b2) the lower coordinate of batch (i, l_o) at time $C_{i'-1}$ must be less than the branch coordinate ρ_l by at least a certain volume φ. In this case, the value of φ represents the maximum volume of product that can be transferred from batch i to upstream branches $l' \leq l$ (including l) and upstream terminals featuring $\sigma_j \leq \rho_l$, during the injection of batch i'.

Let us consider the new binary variable $wl_{i,l}^{(i')}$ denoting that batch (i, l_o) has been partially or completely diverted to branch l while injecting batch $i' \geq i$ ($wl_{i,l}^{(i')} = 1$). Otherwise, $wl_{i,l}^{(i')} = 0$ and no material from batch (i, l_o) is rerouted to line l.

$$t_{\min} wl_{i,l}^{(i')} \leq T_{i,l}^{(i')} \leq t_{\max} wl_{i,l}^{(i')} \quad \forall\, i \in I, i' \in I^{\text{new}}, i' \geq i, l \neq l_o \tag{19}$$

$t_{\min}/t_{\max}$ are lower/upper bounds on the amount of material that can be transferred from a batch in the mainline to a split pipeline. Besides, constraints 20 and 21 stand for the feasibility conditions a2 and b2, respectively.

$$F_{i,lo}^{(i')} - \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} WIF_{i,p',p,lo} \geq \rho_l wl_{i,l}^{(i')} \quad \forall\, i \in I, i' \in I^{\text{new}}, i' \geq i, l \neq l_o \tag{20}$$

$$F_{i,lo}^{(i'-1)} - W_{i,lo}^{(i'-1)} + \sum_{k=1}^{l} T_{i,k}^{(i')} + \sum_{\substack{j \in J \\ \sigma_j \leq \rho_l}} D_{i,j}^{(i')} \leq \rho_l + (pv_{lo} - \rho_l)(1 - wl_{i,l}^{(i')}) \quad \forall i \in I, i' \in I^{\text{new}}, i' > i, l \neq l_o \tag{21}$$

The meaning of constraints 18 and 21 is rather similar, but in this case the restriction 21 provides an upper bound on the volume of product transferred from mainline to secondary pipelines. If a part of batch (i, l_o) is diverted to the lateral pipeline l during run $i' > i$ ($wl_{i,l}^{(i')} = 1$), restriction 21 takes the following form:

$$T_{i,l}^{(i')} \leq \rho_l - (F_{i,lo}^{(i'-1)} - W_{i,lo}^{(i'-1)}) - \sum_{k=1}^{l-1} T_{i,k}^{(i')} - \sum_{\substack{j \in J \\ \sigma_j \leq \rho_l}} D_{i,j}^{(i')} \tag{21'}$$

On the other hand, if a portion of batch i is diverted from the mainline to pipeline l during a new execution i' ($wl_{i,l}^{(i')} = 1$ for some $i' \in I^{\text{new}}$), the branching decision must be activated ($w_{i,l} = 1$).

$$wl_{i,l}^{(i')} \leq w_{i,l} \quad \forall\, i \in I, i' \in I^{\text{new}}, i' \geq i, l \neq l_o \tag{22}$$

Reciprocally, if batch i is transferred from the mainline to pipeline l ($w_{i,l} = 1$), at least one of the branching variables $wl_{i,l}^{(i')}$ should be equal to 1 ($\sum_{i' \in I\text{new}} wl_{k,l}^{(i')} \geq 1$).

$$w_{i,l} \leq \sum_{\substack{i' \in I\text{new} \\ i' \geq i}} wl_{i,l}^{(i')} \quad \forall\, i \in I, l \neq l_o \tag{23}$$

6.2.7. Feasibility Conditions for Diverting Material from Branches to Depots. The transfer of material from batch (i, l) to depot $j \in J_l$ is feasible only if the connection to depot j is accessible from batch (i, l) during run i'. Such a condition implies that: (a3) the upper coordinate of batch i in branch l at time $C_{i'}$ decreased by the volume of the interface material should never be lower than the jth terminal coordinate σ_j (except for the farthest depot of pipeline l, where interface material is removed); and (b3) the lower coordinate of batch (i, l) at time $C_{i'-1}$ must be less than the depot coordinate σ_j by at least a certain amount φ. For split pipelines, the maximum volume of product that can be diverted from batch (i, l) to distribution terminals over branch l up to depot j (including j) may be greater than φ because of the additional volume of batch i branched from the mainline to pipeline l [$T_{i,l}^{(i')}$] during the same pumping run i'. $T_{i,l}^{(i')}$ can take a nonzero value only if batch i is the last lot branched to lateral pipeline l during a previous run. Constraints 24 and 25 stand for the feasibility conditions a3 and b3, respectively.

$$F_{i,l}^{(i')} - \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} WIF_{i,p',p,l} \geq \sigma_j x_{i,j}^{(i')} \quad \forall i \in I, i' \in I^{\text{new}}, i' \geq i, l \neq l_o, j \in J_l, j < |J_l|$$

$$F_{i,l}^{(i')} \geq \sigma_j x_{i,j}^{(i')} \quad \forall\, i \in I, i' \in I^{\text{new}}, i' \geq i, l \neq l_o, j \in J_l, j = |J_l| \tag{24}$$

$$F_{i,l}^{(i'-1)} - W_{i,l}^{(i'-1)} + \sum_{\substack{k=1 \\ k \in J_l}}^{j} D_{i,k}^{(i')} - T_{i,l}^{(i')} \leq \sigma_j + (pv_l - \sigma_j)(1 - x_{i,j}^{(i')}) \quad \forall i \in I, i' \in I^{\text{new}}, i' > i, l \neq l_o, j \in J_l \tag{25}$$

Similar to constraint 18, if a portion of batch (i, l) is diverted to depot j during run i' ($x_{i,j}^{(i')} = 1$), the maximum value for $D_{i,j}^{(i')}$ is given by:

$$D_{i,j}^{(i')} \leq \sigma_j - (F_{i,l}^{(i'-1)} - W_{i,l}^{(i'-1)}) - \sum_{\substack{k=1 \\ k \in J_l}}^{j-1} D_{i,k}^{(i')} + T_{i,l}^{(i')} \quad (25')$$

In contrast to expression 18′, the term accounting for potential batch deliveries to upstream secondary lines is omitted, because there are no branches emerging from lateral pipelines. Moreover, product deliveries from batch (i, l) to depots $j \in J_l$ can be enlarged by a further branching of lot i from the mainline given by the last term of inequality 25′.

6.2.8. Bound on the Total Amount of Material Transferred from a Batch in the Mainline to Branches $l \neq l_o$ and Depots $j \in J_o$. The total volume transferred from batch (i, l_o) to branches $l \neq l_o$ and depots $j \in J_o$ while pumping a new batch $i' \in I^{new}$ $(i' > i)$ must never exceed the pure part of batch (i, l_o) at time point $C_{i'-1}$, that is, its pure content before pumping the new batch i'.

$$\sum_{\substack{j \in J_o \\ j < |J_o|}} D_{i,j}^{(i')} + \sum_{l \neq l_o} T_{i,l}^{(i')} \leq W_{i,l_o}^{(i'-1)} - \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} WIF_{i,p',p,l_o}$$

$$\forall i \in I, i' \in I^{new}, i' > i$$

$$\sum_{j \in J_o} D_{i,j}^{(i')} + \sum_{l \neq l_o} T_{i,l}^{(i')} \leq W_{i,l_o}^{(i'-1)}$$

$$\forall\ i \in I, i' \in I^{new}, i' > i \quad (26)$$

At the farthest mainline depot $(j = |J_o|)$, the interface volume is removed for reprocessing.

6.2.9. Bound on the Amount of Material Diverted from a Batch (i, l) along Branch l to Depots $j \in J_l$. The total volume diverted from batch (i, l) to depots $j \in J_l$ while pumping a later batch $i' \in I^{new}$ $(i' > i)$ should never be greater than the pure content of batch (i, l) at time $C_{i'-1}$, plus the further amount of product received by (i, l) from the mainline during injection i'. The latter contribution can be nonzero only if batch i is the last element diverted to line l during a previous run. Once again, interfaces generated into split pipelines are removed at the farthest extreme of every line.

$$\sum_{\substack{j \in J_l \\ j < |J_l|}} D_{i,j}^{(i')} \leq W_{i,l}^{(i'-1)} - \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} WIF_{i,p',p,l} + T_{i,l}^{(i')}$$

$$\forall i \in I, i' \in I^{new}, i' > i, l \neq l_o$$

$$\sum_{j \in J_l} D_{i,j}^{(i')} \leq W_{i,l}^{(i'-1)} + T_{i,l}^{(i')}$$

$$\forall\ i \in I, i' \in I^{new}, i' > i, l \neq l_o \quad (27)$$

6.2.10. Overall Balance around the Pipeline Network during the Injection of Batch $i' \in I^{new}$. Because of the liquid incompressibility property, the overall volume diverted from batches in transit along every pipeline of the network to depots $j \in J$ while pumping the new batch $i' \in I^{new}$ must be equal to the injected volume $Q_{i'}$, that is, the total volume of the new batch i'.

$$\sum_{\substack{i \in I \\ i \leq i'}} \sum_{j \in J} D_{i,j}^{(i')} = Q_{i'} \quad \forall\ i' \in I^{new} \quad (28)$$

6.2.11. Volume Balance around the Mainline during the Injection of Batch $i' \in I^{new}$. Focusing on the trunk pipeline, the overall volume transferred from in-transit batches (i, l_o) to branches $l \neq l_o$ and mainline depots $j \in J_o$ during run $i' \in I^{new}$ must be equal to the total amount injected through run i' $(Q_{i'})$.

$$\sum_{\substack{i \in I \\ i \leq i'}} \left(\sum_{j \in J_o} D_{i,j}^{(i')} + \sum_{l \neq l_o} T_{i,l}^{(i')} \right) = Q_{i'} \quad \forall\ i' \in I^{new} \quad (29)$$

6.2.12. Volume Balance around a Secondary Pipeline $l \neq l_o$ during the Injection of Batch $i' \in I^{new}$. The overall volume of products diverted from branch l to depots $j \in J_l$ throughout the pumping of the new lot $i' \in I^{new}$ must be equal to the total amount transferred from the mainline to pipeline l during the same interval.

$$\sum_{\substack{i \in I \\ i \leq i'}} \sum_{j \in J_l} D_{i,j}^{(i')} = \sum_{\substack{i \in I \\ i \leq i'}} T_{i,l}^{(i')} \quad \forall\ i' \in I^{new}, l \neq l_o \quad (30)$$

6.3. Inventory Management Constraints. *6.3.1. Monitoring Product Inventories in the Input Terminal.* Because the tree pipeline network structure studied in this work comprises a single input terminal where all refined products are injected, model constraints devoted to monitor product inventories at origin tanks are similar to those already proposed in previous continuous-time approaches for the scheduling of single-source trunk lines.[12,13] Such restrictions are presented in the Appendix.

6.3.2. Satisfying Market Demands and Monitoring Product Inventories in Receiving Terminals. To fully satisfy product demands and monitor inventory levels at distribution terminals, the set of constraints given below has been incorporated in the problem formulation.

6.3.2.1. Amount of Product p Delivered from Batch i to Depot j while Injecting Lot i′. Let $DP_{i,p,j}^{(i')}$ be the amount of product p contained in batch (i, l) delivered to depot $j \in J_l$ during the pumping of batch $i' \geq i$. The variable $DP_{i,p,j}^{(i')}$ will be equal to zero whenever $y_{i,p} = 0$ and batch i does not convey product p. In case $y_{i,p} = 1$, then $DP_{i,p,j}^{(i')} = D_{i,j}^{(i')}$. Hence, for new batches $i \in I^{new}$:

$$DP_{i,p,j}^{(i')} \leq d_{max} y_{i,p} \quad \forall\ i \in I, p \in P, j \in J, i' \in I^{new} \quad (31)$$

$$\sum_{p \in P} DP_{i,p,j}^{(i')} = D_{i,j}^{(i')} \quad \forall\ i \in I, j \in J, i' \in I^{new} \quad (32)$$

On the other hand, the value of $DP_{i,p,j}^{(i')}$ for an old batch $i \in I^{old}$ already in the pipeline system at $t = 0$ is given by:

$$DP_{i,p,j}^{(i')} = D_{i,j}^{(i')} \quad \forall\ i \in I^{old},$$
$$p = P_i, j \in J, i' \in I^{new} \quad (33)$$

because product P_i contained in an old batch $i \in I^{old}$ is a known problem datum.

6.3.2.2. Product Deliveries to Consumer Markets. Let us introduce the variable $DM_{p,j}^{(i')}$ to represent the amount of product p delivered from depot $j \in J$ to consumer markets during the time interval $[C_{i'-1}, C_{i'}]$. If $vm_{p,j}$ stands for the maximum supply rate of product p to neighboring markets from terminal j, an upper bound on the value of $DM_{p,j}^{(i')}$ is provided by constraint 34.

$$DM_{p,j}^{(i')} \leq (C_{i'} - C_{i'-1})vm_{p,j} \quad \forall\, p \in P, j \in J, i' \in I^{\text{new}} \tag{34}$$

If $dem_{p,j}$ stands for the total amount of product p to be supplied from terminal j to consumer markets before the end of the time horizon, the fulfillment of such product demands at terminal j is ensured by enforcing eq 35 in the problem formulation.

$$\sum_{i' \in \text{Inew}} DM_{p,j}^{(i')} + B_{p,j} = dem_{p,j} \quad \forall\, p \in P, j \in J \tag{35}$$

Nonetheless, eq 35 includes the term $B_{p,j}$ standing for backorders of product p at terminal j to consider the possibility of product shortages. The inclusion of variables $B_{p,j}$ not only avoids model solution failures because of depleted inventories at some terminals but also allows one to know where such product shortages arise and which product is not available.

6.3.2.3. Monitoring Product Inventories in Receiving Terminals. A key operational issue is the coordination among incoming flows from the pipeline network and outgoing flows to consumer markets at receiving terminals. The problem goal is to fulfill specified market demands before the end of the time horizon, while keeping product inventory levels at pipeline terminals within the feasible range. In other words, the main goal is to avoid pipeline stoppages due to tank overloading, and backorders because of product shortages.

The inventory level of product p in depot j at time event $C_{i'}$ is computed by adding the available volume at time $C_{i'-1}$ to the total amount supplied by in-transit batches i conveying product p, and simultaneously subtracting deliveries of product p from depot j to local markets.

$$ID_{p,j}^{(i')} = ID_{p,j}^{(i'-1)} + \sum_{\substack{i \in I \\ i \leq i'}} DP_{i,p,j}^{(i')} - DM_{p,j}^{(i')}$$

$$\forall\, p \in P, j \in J, i' \in I^{\text{new}} \tag{36}$$

Moreover, variable $ID_{p,j}^{(i')}$ should always remain within the feasible range given by the specified minimum and maximum inventory levels.

$$(id_{\min})_{p,j} \leq ID_{p,j}^{(i')} \leq (id_{\max})_{p,j} \quad \forall\, p \in P, j \in J, i' \in I^{\text{new}} \tag{37}$$

6.4. Initial Conditions. Every batch $i \in I^{\text{old}}$ already in the pipeline system may be included in the initial linefill of one or more pipelines $l \in PL$ at the start of the planning horizon. In other words, different portions of the same batch $i \in I^{\text{old}}$ can be initially located either in the mainline or in branches of the tree-structured pipeline network. Let $fo_{i,l}$ be the upper coordinate of the batch (i, l) in pipeline $l \in PL$, and $wo_{i,l}$ the associated volume at time $t = 0$. Therefore:

$$F_{i,l}^{(i'-1)} = fo_{i,l} \quad \forall\, i \in I^{\text{old}}, l \in PL, i' = first(I^{\text{new}}) \tag{38}$$

$$W_{i,l}^{(i'-1)} = wo_{i,l} \quad \forall\, i \in I^{\text{old}}, l \in PL, i' = first(I^{\text{new}}) \tag{39}$$

Because an old batch $(i - 1)$ has been pumped into the mainline right before batch i, it will be positioned farther from the origin of any pipeline l, that is, $fo_{i-1,l} \geq fo_{i,l}$. Besides, those batches $i \in I^{\text{old}}$ not initially in pipeline l will feature $wo_{i,l} = 0$, and therefore $fo_{i-1,l} = fo_{i,l}$.

6.5. Problem Objective Function. The problem objective function includes five major terms accounting for (i) the energy consumed for pumping products from the origin of the trunk line to receiving terminals on the mainline and lateral pipelines, (ii) the degrading/reprocessing cost of the interface material between consecutive batches in every pipeline, (iii) the cost of product backorders not delivered on time to their destinations, (iv) the cost of underutilizing pipeline network transport capacity, and (v) the approximate cost of holding product inventories in input and receiving terminal tankage throughout the planning horizon.

$$\begin{aligned}\min z = & \sum_{p \in P} \sum_{j \in J} (cp_{p,j} \sum_{i \in I} \sum_{i' \in \text{Inew}} DP_{i,p,j}^{(i')}) \\ & + \sum_{l \in PL} \sum_{p \in P} \sum_{\substack{p' \in P \\ p' \neq p}} \sum_{\substack{i \in I \\ i > 1}} cf_{p',p,l} WIF_{i,p',p,l} + \sum_{p \in P} \sum_{j \in J} cb_{p,j} B_{p,j} \\ & + cu(h_{\max} - \sum_{i \in \text{Inew}} L_i) + \sum_{p \in P} \left[cir_p \left(\frac{1}{|I^{\text{new}}|} \sum_{i' \in \text{Inew}} IRS_p^{(i')} \right) \right. \\ & \left. + \sum_{j \in J} cid_{p,j} \left(\frac{1}{|I^{\text{new}}|} \sum_{i' \in \text{Inew}} ID_{p,j}^{(i')} \right) \right]\end{aligned} \tag{40}$$

In the first term of eq 40, the coefficient $cp_{p,j}$ represents the cost of conveying a unit volume of product p from the input station to terminal j. If the final destination j is on a secondary pipeline $l \neq l_o$, the parameter $cp_{p,j}$ comprises the following two contributions: (a) the cost of pumping a unit volume of p along the mainline up to the interconnection to branch l, and (b) the cost of transporting it from the origin of branch l to depot j.

Besides, $cf_{p',p,l}$ stands for the cost of degrading and/or reprocessing a unit amount of interface $p'-p$ generated in pipeline l between a pair of adjacent batches containing products p' and p, respectively. Parameter $cb_{p,j}$ stands for the unit backorder cost paid for failing to fulfill product requirements before the end of the time horizon. Meanwhile, the unitary cost cu penalizes the underutilization of the pipeline system, measured by the number of hours it remains idle. This term summarizes three critical components of the pipeline operational cost: (a) the indirect cost to afford for using a costly infrastructure (depreciation expenses), (b) the inventory carrying cost of a huge volume of in-transit products that increases with pipeline stoppages, and (c) the income losses due to an inefficient use of the transport capacity that may lead clients to choose other transportation modes. In other words, the model will tend to fulfill all product demands in a lesser time and release the resource as soon as possible for further usage.

Similarly to previous approaches,[12,13,17,19] the last term accounts for an estimation of the inventory carrying costs at input and receiving terminals, based on the average inventory level of each product in depot tanks throughout the time horizon. Parameters $cir_p/cid_{p,j}$ represent the unit cost of holding a unit volume of p in the input/receiving terminal j. The average inventory level of product p in every node is approximated by considering the product stocks at both the start and the end times of every batch injection i'. The start times will be selected for estimating inventory levels at the input station; meanwhile, the end times of batch injections will be used for averaging product stocks at receiving terminals.

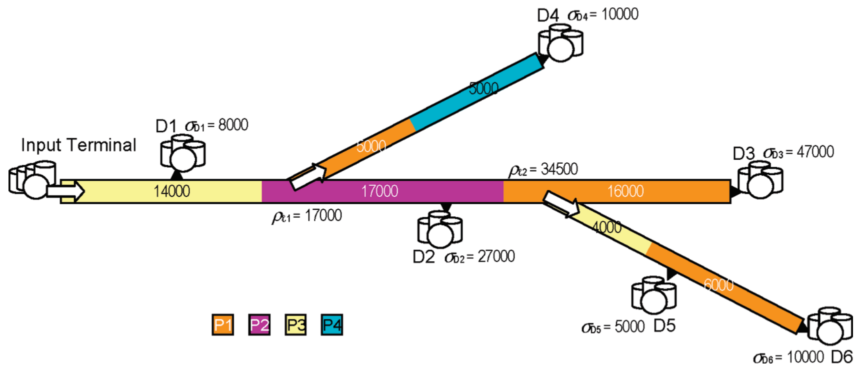

Figure 6. Initial state of the pipeline network for examples 1 and 2.

7. RESULTS AND DISCUSSION

Three examples have been tackled using the proposed problem formulation, with the first two (examples 1 and 2) representing different instances of the same case study. For every example, not only the input/output pipeline schedules but also the product inventory profiles in depot tanks are reported. In addition, the formulation is able to monitor interface movements into the trunk and secondary pipelines, and to track receiving/dispatching operations at refinery and distribution terminals over the planning horizon. In any case, the MILP model was solved to optimality on an Intel Quad Processor (2.80 GHz) with GAMS/CPLEX 11.2 as the MILP solver.[25] An optimality tolerance of 10^{-9} was adopted. Solving a variant of the same case study aims to show the advantages of the new formulation with regards to previous approaches in both solution quality and computational cost. The other case study (example 3) considers the scheduling of a real-world pipeline system with delivering lines.

7.1. Example 1. Example 1 first introduced by MirHassani and Jahromi[19] involves a typical tree pipeline network comprising a single trunk line (l_o) and two branches (l_1, l_2) for the distribution of four oil derivatives (P1−P4) from a single refinery to six receiving terminals (D1−D6) (see Figure 6). Depots D1, D2, and D3 are mainline terminals, while depots D4 and D5−D6 receive products from delivering lines l_1 and l_2, respectively. The trunk line capacity amounts to 47 000 m^3, whereas the total volume of refined products required to fill each branch is equal to 10 000 m^3. Besides, the volumetric coordinates of branching points to lines l_1 and l_2 are ρ_{l1} = 17 000 m^3 and ρ_{l2} = 34 500 m^3, both referred to the mainline origin. In every pipeline, product batches move at a transport rate ranging between 800 and 1000 m^3/h, and products are delivered from receiving terminals to consumer markets at a maximum rate of 400 m^3/h.

The problem goal is to find the optimal pipeline schedule to exactly satisfy all products requirements over the next 4 days (90 h) at minimum total cost. Product demands at every depot to be subsequently dispatched to nearby markets before the horizon end are given in Table 1.

Backordered demands are not allowed. For operational reasons, the duration of any batch injection should neither be shorter than 10 h nor longer than 30 h. Moreover, the minimum volume that can be transferred from a batch in the mainline to a lateral pipeline during a pumping run is restricted to 2000 m^3. Additional data related to the refinery and receiving terminals about location, initial inventories, min/max permissible levels, and unit pumping costs are all presented in Table 2. Furthermore, Table 3 provides the interface reprocessing cost for any

Table 1. Product Demands at Receiving Terminals ($dem_{p,j}$) for Examples 1 and 2

	product requirements (in 10^2 m^3)					
	D1	D2	D3	D4	D5	D6
P1	60	10	42	70	120	140
P2	115	37.5	60	30	100	140
P3	70	30	90	50	45	10
P4	125	40	150	30	10	50

pipeline, together with unit product inventory carrying costs at input and receiving terminals. Forbidden product sequences are denoted with an "X".

Finally, Table 4 reports the scheduled production runs at the refinery over the planning horizon whose outputs will be loaded in the input station tankage.

The initial state of the pipeline network is shown in Figure 6. As already mentioned, a major difference between the proposed formulation and previous approaches on tree pipeline scheduling is the rigorous tracking of batches and interfaces through the trunk and delivering lines, including the knowledge of the original batches from which lots moving through secondary lines were diverted. Such a type of information is also available for those lots available in the initial linefill of every branch. From Figure 6, it follows that there are three lots of product P1 in the pipeline system at t = 0: two of them are located in lines l_1 and l_2, and transport 5000 and 6000 m^3, respectively. The remaining one is positioned in the mainline with a size of 16 000 m^3. Assuming that both batches in l_1 and l_2 come from the same original batch, the flow of P1 in the mainline corresponds to another batch injection. This statement is quite easy to justify. In Figure 6, it is also observed that 4000 m^3 of product P3 were last inserted in line l_2 in a prior horizon, thus separating the two batches of P1 traveling along l_o and l_2. Surely P1(l_2) and P1(l_o) do not belong to the same original batch, and the probable sequence of batch injections producing the initial linefill is P1−P3−P1. The lot of P1 in line l_1 may come from either the older batch of P1(l_2) or the one initially into l_o. It is a valid assumption that lots P1(l_1) and P1(l_2) were diverted from the same original batch. Moreover, a batch of product P4 moves ahead of P1(l_1) in branch l_1 at t = 0, and the last two shipments previously pumped at the input station containing products P2 and P3 are still located in the mainline. Obviously, lot P3(l_2) was not diverted from the batch of P3 last injected in line l_o at a prior horizon. Therefore, the material initially contained in the pipeline

Table 2. Depot Locations, Product Inventories, and Pumping Costs for Examples 1 and 2

prod.	level	refinery [10^2 m^3]	receiving depots [10^2 m^3] D1	D2	D3	D4	D5	D6	prod.		receiving depots D1	D2	D3	D4	D5	D6
P1	min	50	90	20	40	10	90	110	P1	pump cost [\$/m^3]	1.5	2.5	3.5	1.2	5.3	6.3
	max	850	1000	1000	1000	80	200	300								
	initial	450	150	30	70	30	190	230								
P2	min	150	120	100	10	110	80	70	P2	pump cost [\$/m^3]	3.7	4.7	5.7	5.9	2.5	3.5
	max	950	1000	1000	1000	200	240	400								
	initial	550	220	120	50	140	180	210								
P3	min	100	40	90	110	120	20	50	P3	pump cost [\$/m^3]	4.1	5.1	6.1	3.4	1.8	2.8
	max	900	1000	1000	1000	230	90	130								
	initial	500	80	120	200	170	50	60								
P4	min	200	80	50	140	50	110	90	P4	pump cost [\$/m^3]	2.4	3.4	4.4	6.8	4.2	5.2
	max	1000	1000	1000	1000	180	210	370								
	initial	600	190	90	280	60	120	140								
volume coordinate from the origin [10^2 m^3]			80	270	470	100	50	100	connected to pipeline		l_o	l_o	l_o	l_1	l_2	l_2

Table 3. Inventory Carrying and Interface Costs

	interface cost [10^2 \$] P1	P2	P3	P4	inventory costs [\$/m^3] ref	D1	D2	D3	D4	D5	D6
P1		184	340	235	0.58	0.56	0.56	0.56	0.56	0.56	0.56
P2	184		250	413	0.27	0.12	0.12	0.12	0.12	0.12	0.12
P3	340	250		X	0.79	0.78	0.78	0.78	0.78	0.78	0.78
P4	235	413	X		0.31	0.34	0.34	0.34	0.34	0.34	0.34

Table 4. Scheduled Production Runs at the Oil Refinery for Examples 1 and 2

production run	product	volume [m^3]	production rate [m^3/h]	time interval [h]
R1	P1	18 700	550	1−35
R2	P2	22 750	650	10−45
R3	P3	21 000	600	5−40
R4	P4	24 500	700	15−50

network comes from, at least, six original batches: $B1_{P4(l1)}$, $B2_{P1(l1)\mathrm{and}(l2)}$, $B3_{P3(l2)}$, $B4_{P1(lo)}$, $B5_{P2(lo)}$, $B6_{P3(lo)}$. Subscripts indicate the product that each batch contains and the line where it is initially located.

The optimal tree pipeline schedule for example 1 is shown in Figure 7. It includes a sequence of three batch injections (B7, B8, B9) featuring the following products and sizes: $P2^{8000}/P4^{19500}/P4^{30000}$, where the superscripts stand for the batch volumes in m^3. Such batches are inserted in the mainline not only to push the initial pipeline content to the destinations but also to fulfill requirements of P2 at terminal D1, and P4 at terminals D1 and D3.

Batch B7 initially featuring a volume of 8000 m^3 is pumped from time 0.00 to 10.00 h. During the injection of B7, four product deliveries from old batches [B1, B2, B3, B6] to depots are accomplished: (i) a volume of 3000 m^3 of product P3 from batch B6 initially containing 14 000 m^3 into the mainline ($W_{B6,lo}^{(B6)}$ = 14 000) is transferred to depot D1 ($D_{B6,D1}^{(B7)}$ = 3000); (ii) a volume of 2000 m^3 of product P4 from batch B1 initially conveying 5000 m^3 into line l_1 is diverted to depot D4; (iii) 2000 m^3 of product P3 from batch B3 located in line l_2 are delivered to depot D5; and (iv) 1000 m^3 of P1 from B2 (positioned in pipeline l_2) are supplied to depot D5. During the same batch injection, two lots in the mainline are partially rerouted to branches l_1 and l_2: 2000 m^3 of P3 from B6 to l_1, and 3000 m^3 of P1 from B4 to l_2.

The last injection of product P4 is made to push forward the pipeline content so as to keep inventories of products P1 and (P2, P4) within permissible levels in tanks of depots D6 and D3, respectively. From Figure 7, it is observed that the new batch of P4 previously pumped at the origin moves forward until reaching the mainline end to supply 1000 m^3 of product P4 to D3. Overall, five new interfaces were created, two in the mainline (P3−P2 and P2−P4), one in line l_1 (P1−P3), and two more in line l_2 (P3−P1 and P1−P2).

Model size and computational requirements for example 1 are summarized in Table 5. The best pipeline operations schedule for example 1 is similar to the one reported by the previous approach,[19] but it was found in a much lower CPU time. The proposed mathematical model was solved to optimality in 2.46 s after exploring a total of 1236 nodes. Comparison with the solution time of 60 s required by the previous model[19] leads to conclude that the new formulation reduces the computational cost by 1 order of magnitude, although the sizes of both mathematical representations are rather similar.

Variations of product inventories in depot tanks over the planning horizon are depicted in Figure 8. With the exception of $P1^{D3}$ (i.e., product P1 in depot D3), $P2^{D5}$, $P3^{D4}$, $P3^{D5}$, and $P4^{D4}$, the final product inventories are all at their minimum levels. Note that product P2 in batch B7 (8000 m^3) is pumped in excess, because only 1500 m^3 are required at depot D1. The minimum permissible run duration is 10 h, and the pumping rate is lower bounded by 800 m^3/h. Interestingly, the exceeding material is delivered to the destination with the lowest pumping cost for P2, that is, depot D5 (see Table 2).

7.2. Example 2: A Variant of Example 1. Let us introduce a pair of changes in the data set of example 1. Late customer orders placed at depot D5 have increased the demands of products P3 and P4. As a result, the values of $dem_{P3,D5}$ and $dem_{P4,D5}$ rise to 8000 and 2000 m^3, respectively, and the overall terminal demands show an increase of 4500 m^3. The optimal solution

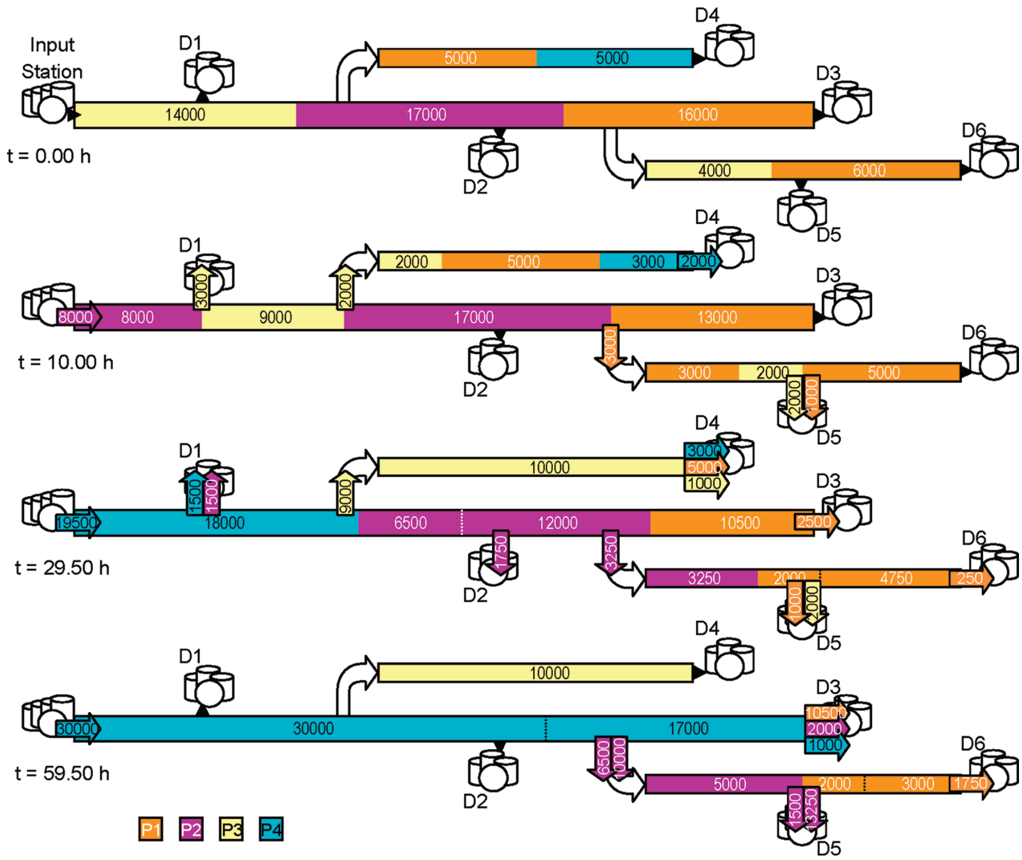

Figure 7. Optimal schedule of pump and delivery operations for example 1.

Table 5. Model Sizes, Computational Requirements, and Results for Examples 1−3

case	eqs.	cont. variables	binary variables	CPU time (s)	nodes explored	optimal solution (10^2 \$)	interface cost (10^2 \$)	pumping cost (10^2 \$)	inventory cost (10^2 \$)	makespan
example 1	2863	1387	239	2.46	1236	5538.1	1527.0	2016.5	1994.6	59.5 h
example 2	2863	1387	239	1.98	875	6199.4	1563.0	2629.5	2006.9	69.0 h
example 3	5123	2654	515	65.08	2804	70.00	70.00			13.1 d

for this variant of example 1, called example 2, is shown in Figure 9.

Similarly to example 1, the best solution involves the injection of three new lots and the creation of five new interfaces (three in the mainline and two in line l_2). However, the product sequence inserted in the mainline changes from $P2^{8000}$−$P4^{19500}$−$P4^{30000}$ to $P1^{12000}$−$P4^{27000}$−$P2^{30000}$, and the injected batches have an overall volume of 69 000 m^3. As compared to example 1, it is required the shipment of an additional 11 500 m^3 of petroleum products to fulfill further terminal demands as large as 4500 m^3. As a result, the final volume stored in depots tanks increases by 7000 m^3. Moreover, the makespan rises from 59.5 to 69.0 h. From Figure 9, it is observed that batches containing three different products are delivered to depot D5 during the last batch injection. In this manner, product demands at distribution terminals can be fully satisfied through three pumping runs. The class of pipelines schedules with several products transferred to a split-line during a single run is ignored by MirHassani and Jahromi's approach.[19] In the optimal schedule of example 2, batch $B6^{P3}$ is no longer diverted to branch l_1 during the first pumping run. Instead, it is transferred to the secondary line l_2 to meet the extra demand of P3 at depot D5. Moreover, a portion of the new lot $B7^{P1}$ is tightlined to branch l_1 during the second batch injection to satisfy product requirements at depot D4 without generating a new interface. The remaining part of $B7^{P1}$ moves forward to fulfill the additional demand of P1 at depot D5.

To conjecture the outcome of MirHassani and Jahromi's model for example 2,[19] further constraints forbidding the transfer of batches with different products to a lateral line during a single pumping run were included in our formulation. Such a model change aims to mimic the behavior of the other approach.[19] As expected, a nonoptimal schedule is found if at most three pumping runs can be performed (see Figure 10).

The product sequence injected at the origin is $P1^{25500}$−$P4^{30000}$−$P2^{27000}$, and the interface reprocessing costs for both approaches are similar. In contrast, the makespan is longer (82.5 h), and the total volume pumped into the mainline rises to 82 500 m^3.

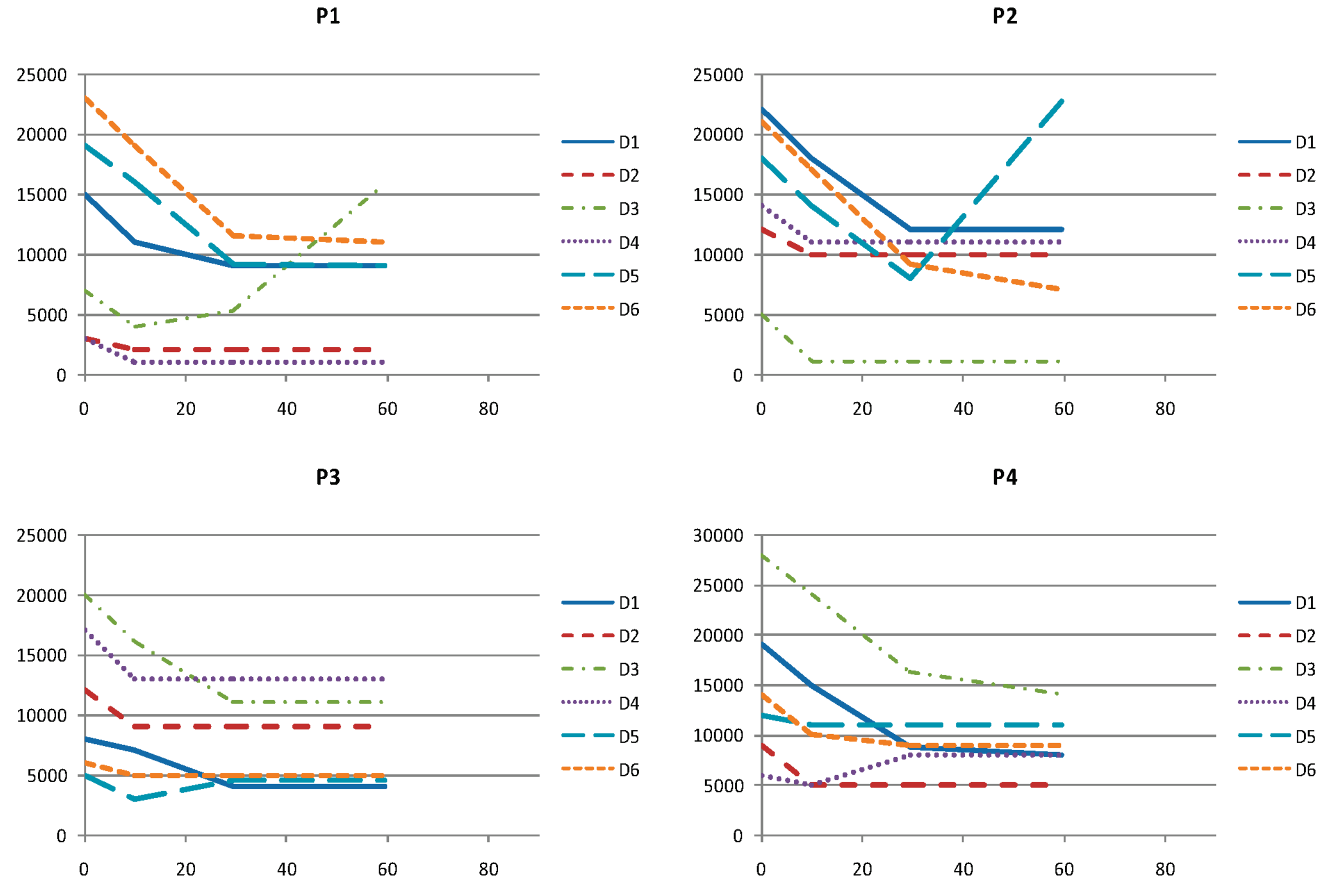

Figure 8. Product inventories in depot tanks over the planning horizon (example 1).

Consequently, the combined pumping and inventory cost rises by \$63947, that is, 10.32% larger. Such an extra cost can be reduced by increasing the cardinality of the set I^{new}. By executing five pumping runs rather than three, a much better pipeline schedule is provided by the modified formulation mimicking the previous approach.[19] It presents an extra cost of only \$114, but the model size is much larger and the solution time increases by a factor of 500. Therefore, the best solution to example 2 cannot be discovered by the other model[19] because batches of different products are to be diverted to a secondary line during the same run. Such a model limitation generally leads to increase the value of $|I^{new}|$ and perform a higher number of pumping runs to discover the true optimal pipeline schedule. As a result, the related computational cost substantially grows.

7.3. Example 3. Example 3 deals with a real-world case study first presented by MirHassani and Jahromi.[19] The pipeline network comprises a 22 in. trunk pipeline with a length of 840 km that links a major oil refinery to six depots (D1−D6) as shown in Figure 11. Besides, it supplies refined products to a pair of 8 in. delivering pipelines l_1 and l_2 that are directly connected to depots D7 and D8, respectively. The mainline diameter decreases to 20" after the first branching point located 400 km far from the origin. Three petroleum products (gas oil, P1; kerosene, P2; and gasoline, P3) are transported by the pipeline system to meet specified product demands at distribution terminals. The problem goal consists on supplying the required amounts of oil distillates from trunk/delivering pipelines to depots to meet their demands in a timely fashion over a monthly planning horizon, at minimum interface reprocessing cost. Backordered demands are not allowed. The mainline capacity is 185 960 m^3, and the lengths and volumes of branches l_1−l_2 are 151 km (4894 m^3) and 93 km (3014 m^3), respectively. The connection between the mainline l_o and the first branch l_1 is located at coordinate ρ_{l1} = 92 166 m^3 from the origin, while the second delivering line l_2 starts at ρ_{l2} = 176 034 m^3. The injection rate is fixed at 25 200 m^3/day, whereas shipments from depots to consumer markets take place at a maximum rate of 1 000 000 m^3/day. Pumping and inventory carrying costs are neglected, and product sequences P1−P3 and P3−P1 have been forbidden. Besides, it is assumed that every product is continuously produced in the refinery at a uniform rate of 2000 m^3/day through the whole planning horizon. The length of product injections at the origin should range from 1 to 8 days, and any part of a batch transferred to a lateral pipeline during a pumping run should have at least a minimum volume of 1000 m^3, and the size of product deliveries to receiving terminals should never be lower than 100 m^3.

Additional data for example 3 are presented in Tables 6 and 7. In contrast to example 1, low product inventories are available in depot tanks at t = 0. Most of terminal demands should then be covered by diverting products from trunk/delivering pipelines. Moreover, new batch injections are not only required to push old batches in the initial linefill to the assigned destinations, but also they are needed to match further depot requirements. Additional amounts of products P1 (at least 13 108 m^3) and especially P3 (not less than

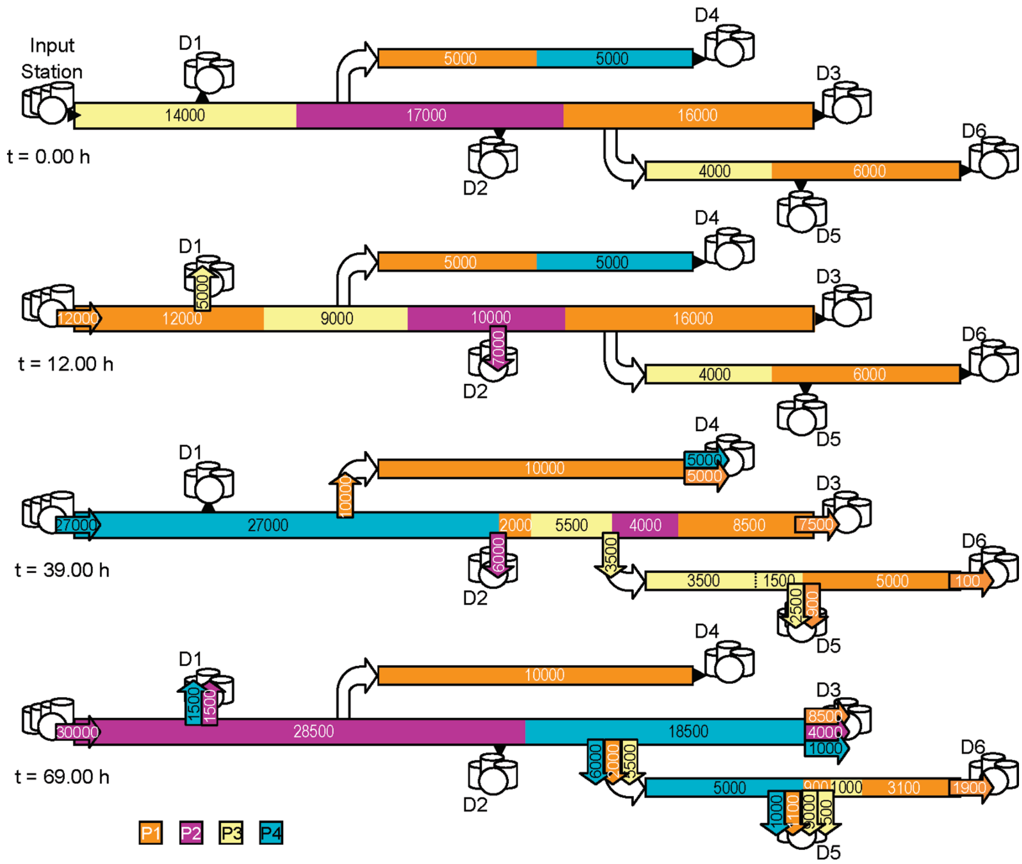

Figure 9. Optimal schedule of pipeline operations for example 2.

100 881 m^3) should be pumped into the mainline to meet demands of such products at distribution terminals.

At time t = 0, the pipeline network contains eleven lots of petroleum products coming from nine original batches (B1−B9) injected at the origin in previous horizons (see Figure 11). The oldest ones (B1−B2) are in transit along pipeline l_2 transporting products P3^{1369} and P2^{1645}, respectively.

Similarly to example 1, the cardinality of set I^{new} is initially equal to $|P|$ = 3. By assuming $|I^{new}|$ = 3, however, terminal demands cannot be satisfied, and the problem becomes infeasible. The value of $|I^{new}|$ is then increased to 4, and the model is solved again. In this way, the optimal pipeline schedule involving the injection of four batches (B10, B11, B12, B13) is found (see Figure 12). No improvement in the interface cost is achieved when $|I^{new}|$ = 5. The new batches pumped into the mainline contain the following products and volumes (given as superscripts in m^3): P1^{25200}/P2^{25200}/P3^{201600}/P2^{78329}. The optimal product sequence is given by P1−P2−P3−P2, with P2 separating the noncompatible products P1−P3. As expected, the injected volumes of the three derivatives from refinery tanks (25 200, 103 529, 201 600 m^3) do not exceed product availabilities.

Pumping runs are 1.000, 1.000, 8.000, and 3.108 days long, and 36 product deliveries to depots are accomplished over the planning horizon. The new batch injections generate three additional interfaces in the mainline: P1−P2, P2−P3, and P3−P2. Moreover, two lots [B8(P2)^{5012} and B9(P1)^{23112}] are rerouted to branch l_1 while pumping batch B12(P3) at the origin, thus generating a new interface P2−P1 in l_1. In turn, branch l_2 receives three new lots [B4(P2)^{17497}, B7(P3)^{22275}, and B12-(P3)^{19327}] during the injection of batch B12, adding a new interface P2−P3 in l_2. Interestingly, the pipeline network remains in operation only 13.108 of the 30 days to meet all product demands at distribution terminals. A total volume of 330 329 m^3 of refined products is transferred to depots to cover net terminal demands as large as 272 314 m^3. Such a difference between product needs and product deliveries is simple to justify. At t = 0, the pipeline content comprises a total of 80 305 m^3 of P2 much larger than the net total demand of P2 (44 762 m^3) at distribution terminals. As a result, almost 40 000 m^3 of P2 are swept out of the pipeline and loaded into storage tanks of almost every terminal to allow batches of P1 and P3 to reach the farthest destinations and fulfill their product demands. Variations of product inventories at depots with time are shown in Figure 13. Because the maximum delivery rate is significantly higher than the pumping rate (1 000 000 vs 25 200 m^3/day), product inventories at some depots remain roughly unchanged even when a new product lot is received from the pipeline network. Each receiving lot, if demanded by the market, is almost instantaneously discharged from the terminal tank and delivered to neighboring customers during the execution of the same pumping run. As a result, tank overloading never occurs. However, this matter may become more critical if much lower delivery rates are adopted.

Model size and computational requirements for example 3 are also presented in Table 5. The optimal pipeline schedule was found in 65.08 s after exploring 2804 nodes, and the interface cost amounts to \$7000. In their work, MirHassani and Jahromi[19] did not fully report the best solution. Instead, such authors only

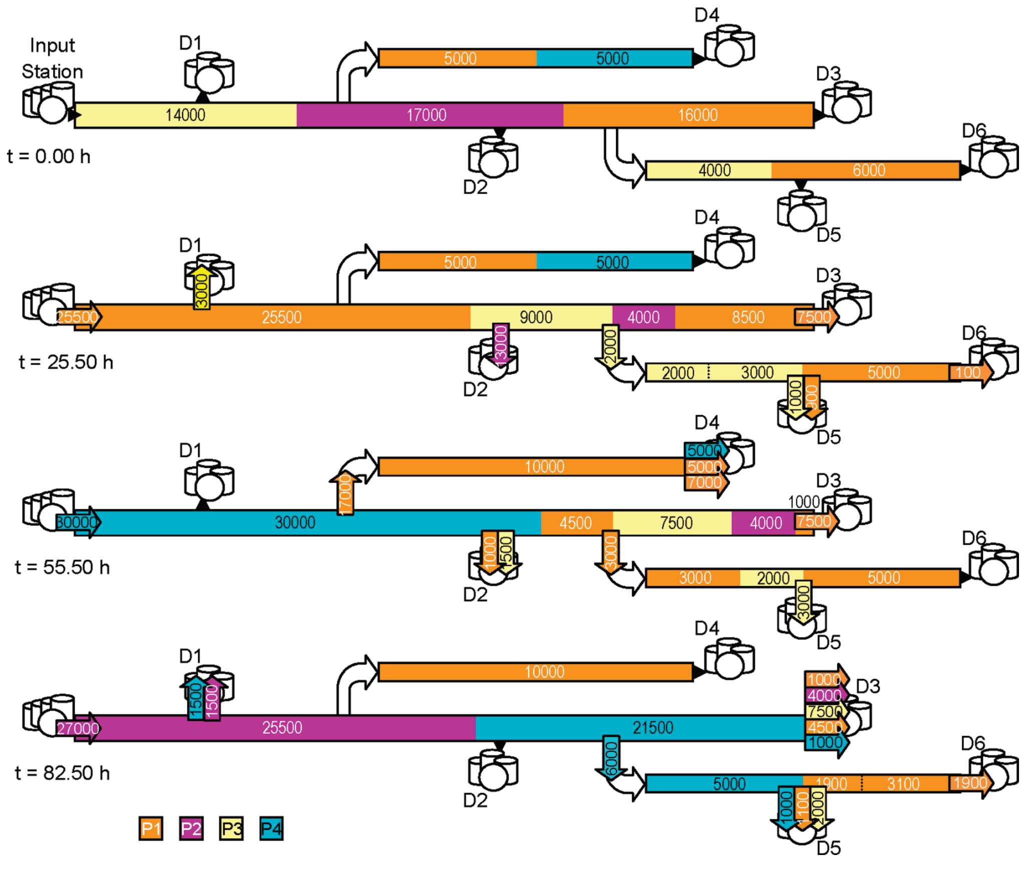

Figure 10. Nonoptimal schedule for example 2 considering single-product branching constraints.

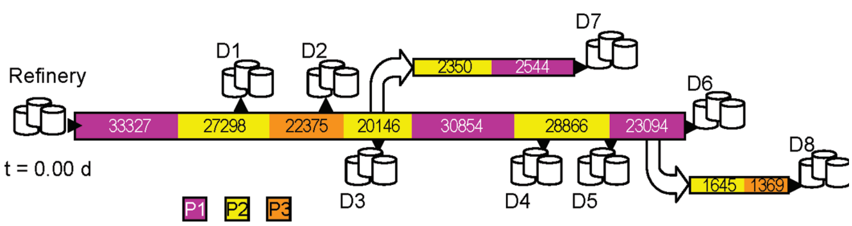

Figure 11. Initial content of the pipeline network for example 3.

Table 6. Product Demands and Interface Costs for Example 3

	product requirements (in m^3)								interface cost [$]		
	D1	D2	D3	D4	D5	D6	D7	D8	P1	P2	P3
P1	8215	4185	8525	32 550	15 810	17 205	21 855	0		2000	X
P2	465	620	775	14 570	2790	0	7750	20 150	2000		1000
P3	10 850	8215	18 135	27 280	24 645	0	0	42 060	X	1000	

stated that the optimal schedule involves the injection of four large batches containing P1−P2−P3−P3 in the mainline and the transfer of 57 batches to depots over a planning horizon of 30 days. They also revealed that their mathematical formulation was solved to optimality in 219 s by using CPLEX 10.0 solver and the modeling software AIMMS 3.7 running on a Pentium IV 2.4 GHz processor with 1 GB RAM. Therefore, the solution time is reduced by a factor of 3.4, and the number of stripping operations drops from 57 to 36 when our formulation is applied. In addition, the best product sequence P1−P2−P3−P2 differs from the one favored by the previous approach[19] (P1−P2−P3−P3). The number of variables and constraints in both formulations are of the same magnitude, despite that the proposed problem representation rigorously traces all the batches from the input station to every destination.

To evaluate the economical advantage of injecting the product sequence P1−P2−P3−P3 favored by the previous approach,[19] our mathematical model has been solved again. This time, it is not only assumed that four batches containing P1−P2−P3−P3 are injected at the input station, but we also include additional

Table 7. Depot Locations and Product Inventories for Example 3

prod.	level	refinery [m^3]	depots [m^3] D1	D2	D3	D4	D5	D6	D7	D8
P1	min	0	0	0	0	0	0	0	0	0
	max	1 000 000	13 204	5520	42 369	21 772	5259	72 594	23 085	5000
	initial	200 000	411	209	426	1628	791	860	1093	527
P2	min	0	0	0	0	0	0	0	0	0
	max	1 000 000	1774	12 220	35 835	26 312	9480	116 695	19 874	5000
	initial	200 000	23	31	39	729	140	0	388	1008
P3	min	0	0	0	0	0	0	0	0	0
	max	1 000 000	13 325	6000	55 675	20 875	15 052	71 775	23 963	5000
	initial	200 000	543	411	907	1364	1232	0	1395	2103
location from pipeline origin [m^3]			50 250	75 988	92 166	142 203	162 866	185 960	4894	3014
connected to pipeline			l_o	l_o	l_o	l_o	l_o	l_o	l_1	l_2

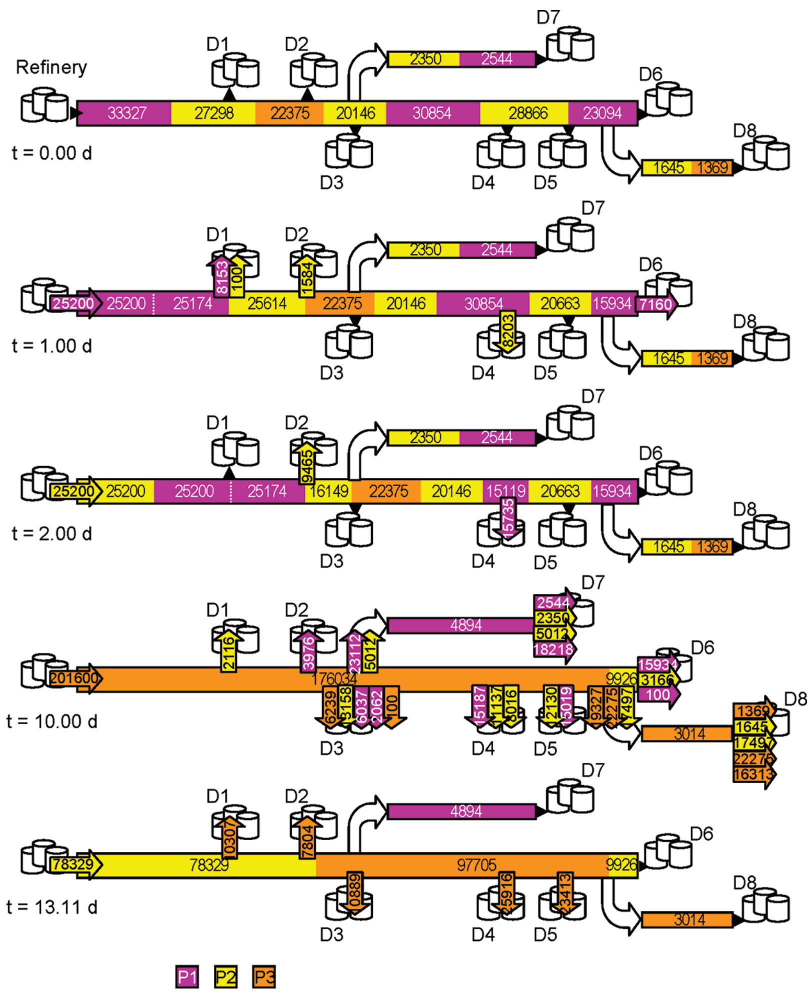

Figure 12. Optimal schedule of pipeline operations for example 3.

constraints to forbid the branching of batches with different products to a lateral line during the same pumping run. Surprisingly, the resulting model has no feasible solution because P1—P2—P3—P3 is a nonviable product sequence. To meet all product demands at receiving terminals during the planning horizon, the last two runs should inject 279 929 m^3 of product P3. Such amount is obtained by summing the additional volume of P3 to be pumped in the network due to depots demands (100 881 m^3) plus the volume of product needed to sweep previous batches to the farthest destination demanding P3, that is, D5 (176 034 m^3 on mainline +3014 m^3 on branch l_2). However, the initial inventory (200 000 m^3) plus the overall refinery throughput of P3 in 30 days (30 days × 200 m^3/day = 60 000 m^3) makes a total of 260 000 m^3. Therefore, there is a deficit of 19 929 m^3, and, consequently, terminal demands cannot be satisfied within the planning horizon and the problem

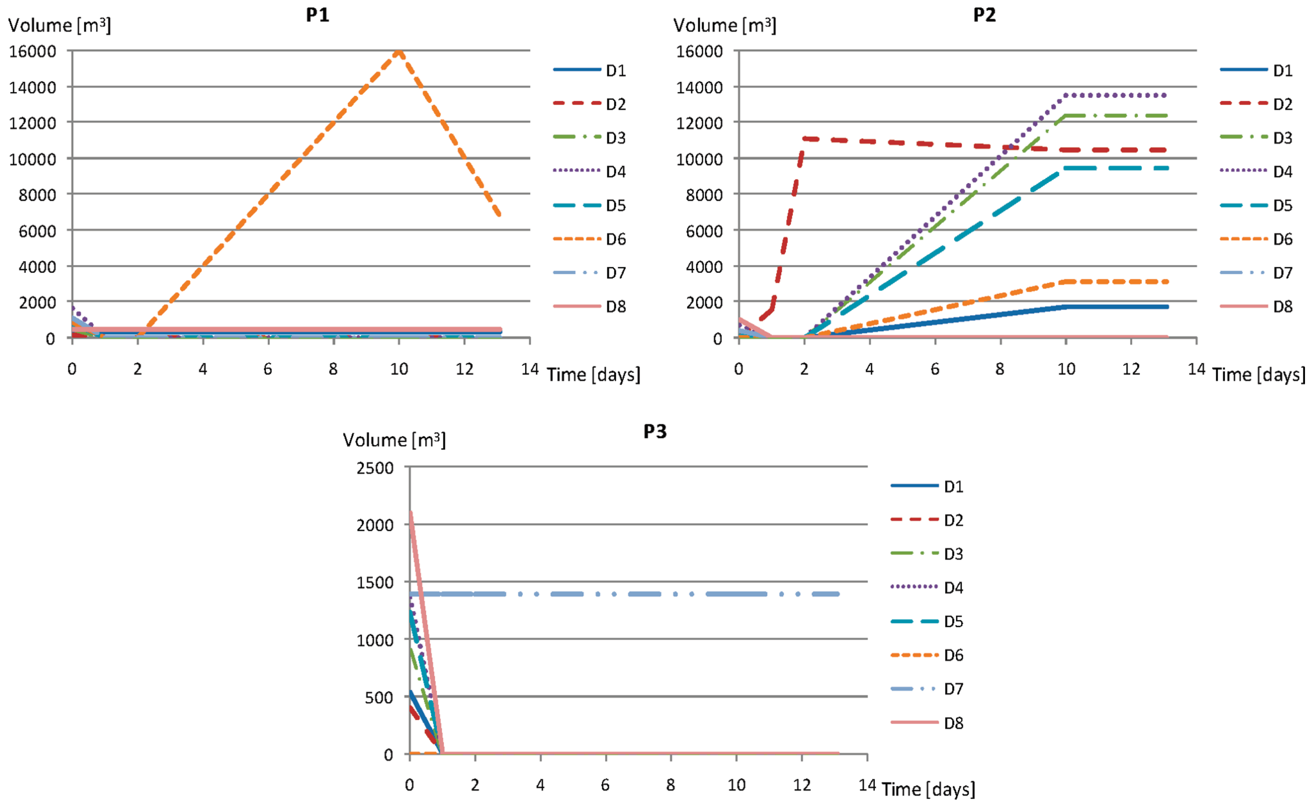

Figure 13. Product inventories in depot tanks over the planning horizon (example 3).

becomes infeasible. Perhaps, the previous approach[19] provides a feasible pipeline schedule by increasing $|I^{new}|$ and allowing the model to execute a higher number of pumping runs with a modified product sequence.

8. CONCLUSIONS

A new continuous MILP mathematical representation for the tree-structure pipeline network scheduling problem has been presented. It was obtained from the formulation of Cafaro and Cerdá[12] by considering the possibility of branching lots of refined products to lateral pipelines and delivering material from batches in trunk and secondary lines to accessible demanding depots during a pumping operation. A major difference between the proposed formulation and previous approaches on tree-structure pipeline scheduling is the rigorous tracking of batches and interfaces through trunk and delivering lines, including the knowledge of the original batches from which lots moving through secondary lines were diverted. Monitoring the arrangement of batches and the creation of new interfaces in lateral pipelines while transferring batches from the mainline permits one to avoid forbidden product strings and estimate the additional reprocessing costs at delivering pipelines. The problem goal is to optimally schedule receiving/dispatching operations at input and distribution terminals over the planning horizon at minimum total cost. Among the operational expenses, we considered pumping, interface, backordered demand, idle transport capacity, and inventory carrying costs. In contrast to a previous approach,[19] the transfer of multiple products to a delivering line while injecting a new batch is a feasible operation. Such a model feature allows one to extend the length of batch injections and decrease the number of pumping runs required to find the optimal pipeline schedule. As a result, a lower model size is obtained by reducing $|I^{new}|$, and consequently the solution time is substantially shortened. Three examples involving up to two single-level branches and eight depots were successfully solved to optimality at low computational cost. The CPU times for examples 1 and 3 previously solved by MirHassani and Jahromi[19] were decreased by 20 and 3.4 times, respectively. Moreover, better solutions have been found for examples 2 and 3. On a future publication, the proposed model will be generalized to deal with multilevel tree-structure pipeline networks, where mainline branches are connected to final depots and lower-level delivering lines.

■ APPENDIX

Monitoring Product Inventories in the Input Terminal. We introduce the set $R_p \subset R$ standing for the sequence of refinery production runs supplying product p to the input terminal tankage over the current horizon. Given the aggregate storage capacity assigned to each product based on the number of allocated tanks and their volumes, it is necessary to guarantee that: (i) enough product p will be available in the input station at the time of injecting a new batch of that product in the pipeline network; and (ii) the maximum inventory level of product p in the assigned tanks is never exceeded. A major model assumption is that the supply rate of any product is always lower than the mainline injection rate. Therefore, there are two critical time events at which the inventory level must be controlled: the starting and completion times of every pumping run. The worst condition for stock-outs of product p occurs at the completion time of every new injection $i \in I^{new}$ involving product p. Hence, the pumping of a new batch i containing product p can be executed only if at least a specified minimum inventory $(ir_{min})_p$ is still available in input terminal tanks at the end time (C_i) of the input operation. Otherwise, it should be delayed until the arrival of a new product replenishment. On the other hand, the worst condition for overloading product p occurs at the start time of every new injection inputting that product into the mainline. Product p will never spill from the assigned tanks in the input terminal if the upper bound $(ir_{max})_p$ is not exceeded before starting the injection of a new batch i, that is, at time $(C_i - L_i)$..

Definition of Binary Variables $zl_{i,r}$ and $zu_{i,r}$. To determine the inventory level of product p in the input terminal at the end of every new injection $i \in I^{new}$, all the refinery production runs $r \in R_p$ partially and/or completely loaded in the terminal tanks before time C_i must be considered. Let $zl_{i,r}$ be the binary variable denoting that injection $i \in I^{new}$ has been completed before $(zl_{i,r} = 0)$ or after $(zl_{i,r} = 1)$ the start time of run r, that is, time a_r. Hence, every run $r \in R_p$ with $zl_{i,r} = 0$ must be ignored when computing the inventory level of product p in the input terminal at time C_i. In case $zl_{i,r} = 1$, run r has been at least partially executed, and the product amount loaded in the assigned tanks up to time C_i should be determined. Therefore:

$$a_r zl_{i,r} \leq C_i \leq a_r + h_{max} zl_{i,r} \quad \forall\ i \in I^{new}, r \in R \tag{A1}$$

Similarly, we introduce the binary variable $zu_{i,r}$ to denote that the injection of batch $i \in I^{new}$ starts before $(zu_{i,r} = 0)$ or after $(zu_{i,r} = 1)$ the completion time of run $r \in R_p$, that is, time b_r. A refinery production run r featuring $zu_{i,r} = 0$ can be either partially executed or not executed at all at time $(C_i - L_i)$. Instead, production runs featuring $zu_{i,r} = 1$ have already been completed and loaded in the assigned tanks at time $(C_i - L_i)$.

$$b_r zu_{i,r} \leq C_i - L_i \leq b_r + h_{max} zu_{i,r} \quad \forall\ i \in I^{new}, r \in R \tag{A2}$$

Amount of Product p from Run $r \in R_p$ Already Loaded in Input Terminal Tanks at Time C_i. The aim is to determine a fair upper bound on the amount of product p coming from run $r \in R_p$ already loaded in the tanks of the input station at time C_i $(SL_{i,r})$. Assuming that production runs have been previously scheduled by refiners, then parameters $[a_r,b_r]$ standing for the starting and end times of run r, s_r denoting the overall product volume of run $r \in R_p$, and vp_r representing the associated loading rate are all given data. Hence:

$$SL_{i,r} \leq s_r zl_{i,r} \tag{A3}$$

$$SL_{i,r} \leq vp_r(C_i - a_r zl_{i,r}) \quad \forall\ i \in I^{new}, r \in R \tag{A4}$$

If $C_i \geq b_r$, run r has been completely loaded in the assigned tanks at time C_i. In that case, binary $zl_{i,r} = 1$ and $SL_{i,r} = s_r$. On the other hand, if $C_i \leq a_r$, production run r has not yet begun at time C_i, then $zl_{i,r} = 0$ and $SL_{i,r} = 0$. Both conditions are modeled through constraint A3. In turn, constraint A4 becomes active whenever $a_r < C_i < b_r$. For such intermediate case, it holds that $zl_{i,r} = 1$ and $SL_{i,r} = \nu p_r (C_i - a_r)$. In other words, just a portion of the production run has already been loaded in tanks during the time interval $[a_r,C_i]$.

Product Supplies from Run $r \in R_p$ Already Loaded in the Input Station at Time $(C_i - L_i)$. Let $SU_{i,r}$ denote the smaller amount of product p coming from run $r \in R_p$ that has been surely loaded in the input station at time $(C_i - L_i)$. If $(C_i - L_i) \geq b_r$, binary $zu_{i,r} = 1$, and the whole production run r is already stored in the tanks at time $(C_i - L_i)$. On the contrary, if $(C_i - L_i) < b_r$, then $zu_{i,r} = 0$, and a portion of the production run r could be loaded in the assigned tanks. Therefore:

$$SU_{i,r} \geq s_r zu_{i,r} \tag{A5}$$

$$SU_{i,r} \geq vp_r[(C_i - L_i) - a_r - h_{max} zu_{i,r}] \quad \forall\ i \in I^{new}, r \in R \tag{A6}$$

Because the worst condition for overloading in the input station occurs at the start of a new injection, the model will tend to make $SU_{i,r}$ as small as possible. If $zu_{i,r} = 1$, run r has been fully loaded in the tanks at time $(C_i - L_i)$, then eq 35 makes $SU_{i,r} = s_r$. Otherwise, constraint A5 becomes redundant. In turn, restriction A6 will drive $SU_{i,r} = vp_r [(C_i - L_i) - a_r]$ just in case $zu_{i,r} = 0$ and $a_r < C_i - L_i < b_r$.

Amount of Product p Injected in the Mainline from Input Terminal Tanks. If the new batch i pumped into the mainline does not contain product p, the associated volume of p in batch i $(QP_{i,p})$ is zero. Otherwise, $QP_{i,p}$ is equal to the initial size of batch i (Q_i). Therefore:

$$q_{min,p} y_{i,p} \leq QP_{i,p} \leq q_{max,p} y_{i,p} \quad \forall\ i \in I^{new}, p \in P \tag{A7}$$

$$\sum_{p \in P} QP_{i,p} = Q_i \quad \forall\ i \in I^{new} \tag{A8}$$

where $q_{min,p}/q_{max,p}$ stand for the minimum/maximum permissible sizes for new injections of product p in the trunk line.

Avoiding Product Shortages and Tank Overloads in the Input Terminal. The inventory of product p in the input terminal is to be monitored so that it never drops below the minimum level $(ir_{min})_p$ at the end of every injection i. Moreover, it should be guaranteed that the maximum permissible level $(ir_{max})_p$ is never exceeded at the start time of every pumping run i. Let the variables $IRS_p^{(i)}$ and $IRF_p^{(i)}$ represent the inventory of product p at the initial and final times of pumping run i, respectively. As a result, safeguards for avoiding product shortages and tank overloads are provided by eqs A9 and A10.

$$IRF_p^{(i)} = iro_p + \sum_{r \in Rp} SL_{i,r} - \sum_{\substack{i' \in I^{new} \\ i' \leq i}} QP_{i',p} \geq (ir_{min})_p \quad \forall\ i \in I^{new}, p \in P \tag{A9}$$

$$IRS_p^{(i)} = iro_p + \sum_{r \in Rp} SU_{i,r} - \sum_{\substack{i' \in I^{new} \\ i' < i}} QP_{i',p} \leq (ir_{max})_p \quad \forall\ i \in I^{new}, p \in P \tag{A10}$$

Parameter iro^p stands for the initial inventory of product p in the head terminal.

AUTHOR INFORMATION

Corresponding Author

*Tel.: +54 342 4559175. Fax: +54 342 4550944. E-mail: jcerda@intec.unl.edu.ar.

ACKNOWLEDGMENT

Financial support received from FONCYT-ANPCyT under Grant PICT 01837, from CONICET under Grant PIP-2221, and from Universidad Nacional del Litoral under CAI+D is fully appreciated.

NOMENCLATURE

Sets

FS = ordered pairs of incompatible products
I = chronologically arranged batches ($I^{old} \cup I^{new}$)
I^{new} = new batches to be injected at the origin of the mainline
I^{old} = old batches inside the pipeline network at the start of the time horizon
J = distribution terminals
J_l = distribution terminals supplied by pipeline l
P = refined petroleum products
PL = set of pipelines in the tree-structure network
R = scheduled production runs at the oil refinery
R_p = scheduled production runs supplying product p
TS_l^{o} = ordered pairs of adjacent batches located in branch l at the initial time

Parameters

a_r/b_r = starting/finishing time of the refinery production run r
$cb_{p,j}$ = unit backorder penalty cost to tardily meet a requirement of product p at depot j
$cf_{p,p',l}$ = unit reprocessing cost of interface material involving products p and p' into line l
$cid_{p,j}$ = unit inventory holding cost for product p at depot j
cir_p = unit inventory holding cost for product p in refinery tanks
$cp_{p,j}$ = unit pumping cost to transport product p from the refinery to depot j
cu = unit idle transport capacity cost
$dem_{p,j,t}$ = overall demand of product p to be satisfied at depot j before the horizon end
d_{min}/d_{max} = minimum/maximum delivery size from a batch to a distribution terminal
$fo_{i,l}$ = current upper coordinate of old batch i in pipeline l
h_{max} = horizon length
$(id_{max})_{p,j}$ = maximum allowed inventory level for product p at depot j
$(id_{min})_{p,j}$ = minimum allowed inventory level for product p at depot j
$if_{p,p',l}$ = volume of interface between batches containing products p and p' into line l
iro_p = initial inventory of product p in refinery tanks
$(ir_{max})_p$ = maximum allowed refinery inventory level for product p
$(ir_{min})_p$ = minimum allowed refinery inventory level for product p
$l_{min,p}/l_{max,p}$ = minimum/maximum length of a new batch injection of product p
pv_l = total volume of pipeline l
$q_{min,p}/q_{max,p}$ = minimum/maximum injection size for product p
s_r = size of the refinery production run r
t_{min}/t_{max} = minimum/maximum branching size from a batch to a lateral pipeline
vb_{min}/vb_{max} = minimum/maximum pumping rates
$vm_{p,j}$ = maximum supply rate of product p to the local market from depot j
vp_r = production rate of run r
$wo_{i,l}$ = current volume of old batch i in pipeline l
ρ_l = volumetric coordinate of the split point to branch l from the origin of the mainline
σ_j = volumetric coordinate of depot j from the origin of each pipeline
$\tau_{p,p'}$ = changeover time between injections of products p and p'

Continuous Variables

$B_{p,j}$ = backorder of product p for depot j
C_i/L_i = completion time/length of pumping run i
$D_{i,j}^{(i')}$ = volume of batch i diverted to depot j while injecting batch i' into the mainline
$DM_{p,j}^{(i')}$ = amount of product p sent to local market j during the time interval $[C_{i'-1}, C_{i'}]$
$DP_{i,p,j}^{(i')}$ = amount of product p supplied by batch i to depot j during $[C_{i'} - L_{i'}, C_{i'}]$
$F_{i,l}^{(i')}$ = upper coordinate of batch i from the origin of pipeline l at time $C_{i'}$
$ID_{p,j}^{(i')}$ = inventory of product p in depot j at the end of pumping run i'
$IRF_p^{(i')}$ = inventory of product p in refinery at the end of pumping run i'
$IRS_p^{(i')}$ = inventory of product p in refinery at the start of pumping run i'
pv_l = total volume of pipeline l
Q_i = initial size of the new batch i
$QP_{i,p}$ = volume of product p injected in the mainline while pumping batch i
$SL_{i,r}$ = production output from run r available in refinery tanks at time C_i
$SU_{i,r}$ = production output from run r available in refinery tanks at time $(C_i - L_i)$
$T_{i,l}^{(i')}$ = volume of batch i branched to line l while injecting batch i' into the mainline
$WIF_{i,p,p',l}$ = interface volume between batch i and its predecessor containing products p' and p in pipeline l
$W_{i,l}^{(i')}$ = size of batch i in pipeline l at time $C_{i'}$

Binary Variables

$w_{i,l}$ = denotes that a portion of batch i is branched to line l
$wl_{i,l}^{(i')}$ = denotes that a portion of batch i is branched to line l while injecting i' into the mainline
$x_{i,j}^{(i')}$ = denotes that a portion of batch i is transferred to depot j while injecting i' into the mainline
$y_{i,p}$ = denotes that batch i contains product p
$zl_{i,r}$ = denotes that injection i ends after the refinery production run r has started
$zu_{i,r}$ = denotes that injection i begins after the refinery production run r has ended

REFERENCES

(1) Miesner, T. O.; Leffler, W. L. *Oil & Gas Pipelines in Nontechnical Language*; Pennwell: Tulsa, OK, 2006.

(2) Sasikumar, M.; Prakash, P. R.; Patil, S. M.; Ramani, S. PIPES: A Heuristic Search Model for Pipeline Schedule Generation. *Knowl.-Based Syst.* **1997**, *10*, 169–175.

(3) Maruyama Mori, F.; Lüders, R.; Arruda, L. V. R.; Yamamoto, L.; Bonacin, M. V.; Polli, H. L.; Aires, M. C.; Bernardo, L. F. J. Simulating the Operational Scheduling of a Realworld Pipeline Network. *Comput.-Aided Chem. Eng.* **2007**, *24*, 691–696.

(4) García-Sánchez, A.; Arreche, L. M.; Ortega-Mier, M. Combining Simulation and Tabu Search for Oil-Derivatives Pipeline Scheduling. *Stud. Comput. Intell.* **2008**, *128*, 301–325.

(5) Hane, C. A.; Ratliff, H. D. Sequencing Inputs to Multi-commodity Pipelines. *Ann. Operations Res.* **1995**, *57*, 73–101.

(6) Rejowski, R.; Pinto, J. M. Scheduling of a Multiproduct Pipeline System. *Comput. Chem. Eng.* **2003**, *27*, 1229–1246.

(7) Rejowski, R.; Pinto, J. M. Efficient MILP Formulations and Valid Cuts for Multiproduct Pipeline Scheduling. *Comput. Chem. Eng.* **2004**, *28*, 1511–1528.

(8) Magatão, L.; Arruda, L. V. R.; Neves, F., Jr. A Mixed Integer Programming Approach for Scheduling Commodities in a Pipeline. *Comput. Chem. Eng.* **2004**, *28*, 171–185.

(9) Zyngier, D.; Kelly, J. D. Multi-Product Inventory Logistics Modeling in the Process Industries. *Optim. Logistics Challenges Enterprise* **2009**, *30*, 61–95.

(10) MirHassani, S. A.; Ghorbanalizadeh, M. The Multi-Product Pipeline Scheduling System. *Comput. Math. Appl.* **2008**, *56*, 891–897.

(11) Rejowski, R.; Pinto, J. M. A Novel Continuous Time Representation for the Scheduling of Pipeline Systems with Pumping Yield Rate Constraints. *Comput. Chem. Eng.* **2008**, *32*, 1042–1066.

(12) Cafaro, D. C.; Cerdá, J. Optimal Scheduling of Multiproduct Pipeline Systems Using a Non-Discrete MILP Formulation. *Comput. Chem. Eng.* **2004**, *28*, 2053–2068.

(13) Cafaro, D. C.; Cerdá, J. Dynamic Scheduling of Multiproduct Pipelines with Multiple Delivery Due Dates. *Comput. Chem. Eng.* **2008**, *32*, 728–753.

(14) Cafaro, D. C.; Cerdá, J. Optimal Scheduling of Refined Products Pipelines with Multiple Sources. *Ind. Eng. Chem. Res.* **2009**, *48*, 6675–6689.

(15) Cafaro, D. C.; Cerdá, J. Operational Scheduling of Refined Products Pipeline Networks with Simultaneous Batch Injections. *Comput. Chem. Eng.* **2010**, *34*, 1687–1704.

(16) Gleizes, M. F.; Herrero, G.; Cafaro, D. C.; Méndez, C. A.; Cerdá, J. A Discrete-Event Simulation Tool for Efficient Operation of Refined Products Pipelines. *Comput.-Aided Chem. Eng.* **2010**, *28*, 1697–1702.

(17) Relvas, S.; Matos, H. A.; Barbosa-Póvoa, A. P. F. D.; Fialho, J.; Pinheiro, A. S. Pipeline Scheduling and Inventory Management of a Multiproduct Distribution Oil System. *Ind. Eng. Chem. Res.* **2006**, *45*, 7841–7855.

(18) Relvas, S.; Matos, H. A.; Barbosa-Póvoa, A. P. F. D.; Fialho, J. Reactive Scheduling Framework for a Multiproduct Pipeline with Inventory Management. *Ind. Eng. Chem. Res.* **2007**, *46*, 5659–5672.

(19) MirHassani, S. A.; Jahromi, H. F. Scheduling Multi-Product Tree-Structure Pipelines. *Comput Chem. Eng.* **2010**; doi: 10.1016/j.compchemeng.2010.03.018.

(20) Neves-Jr, F.; Magatão, L; Stebel, S. L.; Boschetto, S. N.; Felizari, L. C.; Czaikowski, D. I.; Rocha, R.; Ribas, P. C. An Efficient Approach to the Operational Scheduling of a Real-World Pipeline Network. *Comput-Aided Chem. Eng.* **2007**, *24*, 697–702.

(21) Moura, A. V.; de Souza, C. C.; Cire, A. A.; Lopes, T. M. Planning and Scheduling the Operation of a Very Large Oil Pipeline Network. *Lect. Notes Comput. Sci.* **2008**, *5202*, 36–51.

(22) Boschetto, S. N.; Felizari, L. C.; Yamamoto, L.; Magatão, L.; Stebel, S. L.; Neves-Jr, F.; Arruda, L. V. R.; Lüders, R.; Ribas, P. C.; Bernardo, L. F. J. An Integrated Framework for Operational Scheduling of a Real-World Pipeline Network. *Comput.-Aided Chem. Eng.* **2008**, *25*, 259–264.

(23) Boschetto, S. N.; Magatão, L.; Brondani, W. M.; Neves-Jr, F.; Arruda, L. V. R.; Barbosa-Póvoa, A. P. F. D.; Relvas, S. An Operational Scheduling Model to Product Distribution through a Pipeline Network. *Ind. Eng. Chem. Res.* **2010**, *49*, 5661–5682.

(24) Herrán, A.; de la Cruz, J. M.; de Andrés, B. A Mathematical Model for Planning Transportation of Multiple Petroleum Products in a Multi-Pipeline System. *Comput. Chem. Eng.* **2010**, *34*, 401–413.

(25) Brooke, A.; Kendrick, D.; Meeraus, A.; Raman, R. *GAMS - A user's guide*; GAMS Development Corp.: Washington, DC, 2006.