Computers & Industrial Engineering 88 (2015) 395–409

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Optimization model for the detailed scheduling of multi-source pipelines

Vanina G. Cafaro, Diego C. Cafaro, Carlos A. Méndez *, Jaime Cerdá

INTEC (UNL-CONICET), Argentina

ARTICLE INFO

Article history:
Received 13 August 2013
Received in revised form 19 March 2015
Accepted 28 July 2015
Available online 4 August 2015

Keywords:
Detailed scheduling
Multiproduct pipeline
Multiple sources
Oil industry
Logistics
MILP approach

ABSTRACT

Pipeline networks are the shippers' first choice for carrying large volumes of refined petroleum products from oil refineries to distant distribution terminals. Optimization approaches for solving the pipeline scheduling problem proceed in two hierarchical stages: the aggregate and the detailed planning steps. The aggregate plan determines the batch sizes, the sequence of batch injections, and the allocation of batches to customers. The subsequent stage refines the aggregate plan to find the detailed schedule of batch input and output operations. This paper presents a mixed-integer linear programming (MILP) formulation for the detailed scheduling of multi-source pipelines that accounts for parallel batch injections and simultaneous product deliveries to multiple terminals. It overcomes a critical drawback of previous models that assume single source configurations. Modeling multi-source pipeline networks is a great challenge, requiring a completely revised approach. The new model finds cost-effective solutions with remarkable efficiency.

1. Introduction

The two most efficient ways to transport oil products in large volumes are ships and pipelines. Compared to water transportation, pipelines operate around the clock in all seasons under almost all weather conditions, at low operation costs. Oil pipeline routes link isolated crude oil production areas to refineries, while refined products pipelines connect these facilities to major populated regions, transporting large volumes of different products through the same line. In the United States, there are 409,000 miles of pipelines carrying 17% of all ton-miles of freight (Trench, 2001). Batches with homogeneous grades of the same petroleum product, even supplied by different refiners, may be merged and shipped as a common stream. Major lines, like the Colonial Pipeline in the U.S., have multiple entry and exit points with several tanks, gauges, pumps, and valves, requiring a high degree of automation to operate efficiently. From a central control room, pipeline operators manage the product flows, start and stop pumps, open and close valves, and follow the batches along the pipeline network (Trench, 2001). Such tasks should be effectively planned to lower the power consumption, the largest pipeline operation cost. Since different petroleum products are pumped back-to-back into the same pipeline rarely using separation devices, some mixing occurs. In fact, smaller batch sizes make interface losses proportionally more important, while some product sequences are directly forbidden. Planning pipeline operations involves several decisions such as the sequence of products to inject at the source nodes, the batch sizes, the start/end times of every injection, and the sequence of product deliveries, among others. According to Siswanto, Essam, and Sarker (2011), transportation scheduling problems can be divided into four sub-problems to be solved sequentially or simultaneously: route selection, batching, loading, and unloading activity procedures.

There are several tools for scheduling transport operations: mathematical programming, heuristics, and hybrid techniques, among others. But even today, the planning and scheduling of real-world multiproduct pipelines is often based on simple worksheets (Ball, Dickerson, & Hertel, 2011) that assume a fixed flow rate of oil in the pipeline, to easily follow the batch movements. These simplified methods involve multiple trial-and-error iterations and are therefore very time consuming (Reddy, Karimi, & Srinivasan, 2004). Moreover, the assumption of a fixed flow rate does not allow the optimal utilization of the transport capacity.

More rigorous scheduling approaches have been developed over the last decade. On the one hand, discrete and continuous mathematical formulations for the optimal scheduling of unidirectional pipelines with a single source and multiple delivery nodes. Discrete approaches divide the pipeline volume into a finite

* Corresponding author.
E-mail addresses: vcafaro@fiq.unl.edu.ar (V.G. Cafaro), dcafaro@fiq.unl.edu.ar (D.C. Cafaro), cmendez@intec.unl.edu.ar (C.A. Méndez), jcerda@intec.unl.edu.ar (J. Cerdá).

http://dx.doi.org/10.1016/j.cie.2015.07.022
0360-8352/

Nomenclature

(a) Sets

B	blocks of individual injections
R	individual injections
I	batches moving through the pipeline during the planning horizon
I_b	batches moving through the pipeline during the execution of block b
J	pipeline terminals $\{j_0, j_1, \ldots, J\}$
JS	pipeline segments $\{j_1, \ldots, J\}$
JS_r	string of pipeline segments between the active source node and the farthest terminal for injection r
$J_r^{\oplus}$	active receiving terminals during the injection r
$J_{i,r}^{\oplus}$	active terminals receiving material from batch i while performing the injection r
K	ordered set of detailed pumping operations
K_b	subset of detailed operations of block b
R_b	subset of individual injections in block b

(b) Parameters

$a_{i,r}$	denotes that batch i is partially or fully pumped by injection r, in case $a_{i,r} = 1$
ca	unit flow restart cost
cs	unit flow stoppage cost
$dd_{i,j}^{(r)}$	total amount of product delivered from batch i to terminal j during the injection r
$d_{\min}$	minimum delivery size for a single operation
fco	fixed cost for performing a detailed operation
$l_{\min}/l_{\max}$	minimum/maximum allowed length of a detailed operation
pv	total pipeline volume
qq_r	total volume pumped during injection r
$q_{\max}$	maximum size of a product injection
st_b/ft_b	starting/completion time of block b given by the aggregate plan
$vb_{\min}^{(r)}/vb_{\max}^{(r)}$	minimum/maximum injection rate at the active source of r
$vb_{\min}^{(j)}/vb_{\max}^{(j)}$	minimum/maximum flow rate in pipeline segment j
$vd_{\max}^{(j)}$	maximum delivery rate from the pipeline to the receiving terminal j
wo_i	initial volume of batch i
σ_j	volumetric coordinate of depot j from the origin of the pipeline network

(c) Variables

Continuous variables

$AV_{j,k}$	volume of segment j activated at the start of operation k
C_k/L_k	completion time/length of the detailed operation k
$D_{i,j,k}$	volume of batch i diverted to depot j while performing operation k
$F_{i,k}$	front coordinate of batch i at time C_k
$Q_{r,k}$	volume of injection r pumped into the pipeline during operation k
$SV_{j,k}$	volume of segment j stopped at the start of operation k
$W_{i,k}$	size of batch i at time C_k
$\omega_{j,k}$	denotes the state of the pipeline segment j during operation k (it is limited to the closed interval [0; 1])

Binary variables

u_k	denotes the existence of the detailed operation k
$x_{i,j,k}$	denotes the existence of a delivery from batch i to depot j while performing the detailed operation k

number of "packs", and the planning horizon into time intervals of fixed duration. Most of them generally use a uniform time and volume partitioning scheme (Hane & Ratliff, 1995; Herrán, de la Cruz, & de Andrés, 2010; Magatão, Arruda, & Neves, 2004; Rejowski & Pinto, 2003; Zyngier & Kelly, 2009). Instead, mathematical representations based on continuous time and volume domains lead to more efficient and rigorous formulations of the pipeline scheduling problem (Cafaro & Cerdá, 2009; Castro, 2010). On the other hand, the scheduling of more complex pipeline configurations generally relies on hierarchical decomposition strategies, making the most critical decisions based on heuristic search techniques (Sasikumar, Prakash, Patil, & Ramani, 1997). The sequence of batch injections at every source node and the allocation of batches to customers are two of the key operational issues heuristically determined (Boschetto et al., 2010; García-Sánchez, Arreche, & Ortega-Mier, 2008; Lopes, Ciré, de Souza, & Moura, 2010; Moura, de Souza, Cire, & Lopes, 2008; Neves et al., 2007). The next step is to find out the timing of input and output operations using discrete-event simulation (Cafaro, Cafaro, Méndez, & Cerdá, 2011; Gleizes, Herrero, Cafaro, Méndez, & Cerdá, 2012; Mori et al., 2007), constraint programming (Moura et al., 2008), or optimization models (Cafaro, Cafaro, Méndez, & Cerdá, 2012; Cafaro et al., 2011).

Cafaro et al. (2011) propose one of the most effective approaches to tackle this problem. It consists of two hierarchical steps, each one involving a mixed-integer linear programming (MILP) formulation. Both models are based on continuous representations of the volume and time domains. All operating decisions made at the first stage are hard constraints for the second. At the first step, the sequence of product injections, batch sizes, and mean pump rates are found. The issue of how to perform the planned product deliveries is left to the second model. The so-called detailed optimization model refines the aggregate plan to determine the scheduling of input and output operations, and the flow rate profile at every pipeline segment. But up to now, optimization approaches for the detailed scheduling of oil products pipelines have assumed single source configurations.

This paper presents the first MILP formulation based on continuous time and volume scales for the detailed scheduling of pipeline networks with multiple sources. It assumes that the aggregate transportation plan is already available. The new model can be regarded as an extension of the model recently proposed by Cafaro et al. (2012), but unlike that approach, the new formulation can effectively handle parallel injections at two or more source nodes. In addition, several product deliveries to multiple terminals can simultaneously occur. The problem goal is to minimize the operation costs. As shown in the following sections, the inclusion of multiple sources performing parallel injections lead to a major rethinking of the optimization model. Computational experiments prove the model efficiency and show significant cost reductions with regards to other approaches typically used in practice.

2. Literature review

The relevant literature related to this work falls into three major topics: (1) optimization models for the pipeline transportation planning (batch sizing, batch sequencing, and allocation of batches to customers); (2) simulation and optimization tools for the detailed scheduling of single-source pipelines; and (3) discrete decomposition approaches. Previous publications on those three topics are reviewed in Sections 2.1–2.3, respectively.

2.1. Optimization models for the pipeline transportation planning

Cafaro and Cerdá (2009) and Castro (2010) propose the first MILP formulations based on continuous time and volume domains for planning the operation of unidirectional pipelines with multiple sources and destinations. Such optimization approaches are able to determine the so-called aggregate transportation plan. Given the product requirements and delivery dates at distribution terminals, the models find the size, origin and destinations of each batch, the product injection sequence at every source, and the start/end times of every injection. Injections at intermediate locations can either insert a new lot or increase the size of a batch in transit. However, the models assume that a single injection can at most be performed at any time.

Afterwards, Cafaro and Cerdá (2010) extend their approach to allow simultaneous batch injections at two or more source nodes. Through a better use of the pipeline transport capacity, the overall time needed to meet specific demands at distribution terminals substantially diminishes. They also consider dual-purpose stations that can simultaneously perform injection and delivery tasks, and flow rate variations caused by changes in the pipeline diameter. The new model permits to determine: (a) the optimal sequence of batch injections at every source node, (b) the batch sizes, (c) the mean flow rates, (d) the start/end times of every injection, and (e) the set of stripping operations transferring products from batches to receiving terminals. The problem goal is to satisfy all product requirements at minimum total cost, including backorder expenses.

Although these models help to find cost-effective transportation plans, they present some practical limitations. The pipeline schedules they provide do not specify in which order to perform the prescribed set of deliveries. Moreover, there is sometimes no feasible way to fulfil a product demand by making a single delivery, and several non-consecutive batch "cuts" are needed. In other words, the detailed sequence of actions performed by the pipeline operator is not still available using the current methodologies. We require a refining step to decide on: (a) the number and size of detailed batch input/output operations to fulfil every product demand, and (b) the order and time of execution of such detailed operations.

Overall, as early stated by Hane and Ratliff (1995), the works previously referenced clearly show that the presence of multiple injection nodes along the pipeline network brings about new operational complexities that should be carefully studied.

2.2. Detailed scheduling of single-source pipelines

As already pointed out, the detailed scheduling problem has been tackled assuming single-source pipelines and using two alternative approaches: discrete-event simulation and optimization models. Simulation models can validate the pipeline transportation plan by generating a detailed schedule based on different empirical rules (Cafaro et al., 2011; Gleizes et al., 2012). The simulation model regards the pipeline as a multi-server queuing system, with a single server at the end extreme of every pipeline segment. In this way, each segment is a FIFO (First In, First Out) queue, and the server directs the movement of batch portions into either the distribution terminal or the next pipeline segment. An interesting feature of the simulation approach is that the detailed schedule is found at low computational cost, but its optimality cannot be guaranteed.

Cafaro et al. (2011) also solve the detailed scheduling problem for single-source pipelines through an MILP optimization model. At any time, a single depot can at most be receiving some amount of product from the line. Hence, the detailed schedule can be viewed as a sequence of batch cuts, each one diverting material from a single batch to a single receiving terminal. The refinement of the aggregate transportation plan seeks to minimize the flow restarts in idle segments of the pipeline and the number of operations over the planning horizon. The rationale behind this criterion is stated by Hane and Ratliff (1995): the pipeline operation cost is greatly influenced by the number of flow restarts made to deliver the batches to their destinations. The cost of a flow restart is related to how much work must be done to put the fluid in motion in idle pipeline segments. The more times the product flow is restarted, the more costly is the pipeline operation. Later, Cafaro et al. (2012) propose a new MILP optimization approach that provides an improved detailed schedule for single-source, multi-destination pipelines. The new model allows parallel cuts, i.e. simultaneous deliveries to different terminals at the same time. In this manner, more pipeline segments remain active and flow restarts significantly diminish. As recently proved by Cafaro, Cafaro, Méndez, and Cerdá (2015), parallel deliveries also yield substantial savings in the energy consumption. Such improvements arise because pipelines segments operate at much lower flow rates, reducing the head loss. Optimization models can also handle a specific flow rate range for each pipeline segment. By precisely determining the product amounts simultaneously sent to output terminals, the stream flow rates remain within the permissible ranges. However, none of these models is able to find a detailed operational schedule for pipeline networks having multiple sources.

2.3. Discrete decomposition approaches

Most approaches based on discrete representations can generate detailed pumping and delivery schedules in only one step. They deal with the scheduling of pipelines with multiple distribution terminals, but usually assuming a single source node. The earliest discrete formulations presented by Hane and Ratliff (1995) and Rejowski and Pinto (2003) assume equal-sized "packs" (batch portions) and time periods of constant length. The basic idea behind these representations is rather simple. As a new product pack is inputted in the pipeline origin, another pack of the same size is discharged into one of the receiving terminals. In these models, every pack is injected at the same rate given by the ratio between the pack size and the length of the time period. Rejowski and Pinto (2003) also propose another version of their discrete model assuming pipeline segments of different diameters. The model uses packs whose size depends on the segment diameter, i.e. smaller packs for smaller sections. When a product pack is transferred to a segment of smaller diameter, a portion of the pack is mandatorily diverted to the intermediate terminal, to compensate for the difference in their sizes. Such a splitting operation occurs even if the product contained in the pack is not demanded by the terminal. Despite this major drawback, the approach is the first to handle simultaneous product deliveries to multiple depots.

Later, Rejowski and Pinto (2008) propose a new discrete model for the scheduling of single-source pipelines that assumes a planning horizon composed of time periods of variable length to account for variations in the flow rate. Herrán et al. (2010) present the first MILP discrete model for the scheduling of multi-pipeline networks, including reversible (or bidirectional) lines. Like all discrete representations, every time a pack enters a pipeline another pack exits at the other extreme. Direct transfers from pipeline to pipeline and partial deliveries to intermediate depots are problem features not included in the model. As discrete models are approximate representations of the pipeline scheduling problem, feasible schedules require a fine discretization of the volume domain. Consequently, large-size formulations are solved even for rather short time horizons.

The detailed scheduling of more complex pipeline network configurations has raised the need of decomposition methods dividing the problem into several steps. Different types of techniques such as heuristics and local search algorithms (García-Sánchez et al., 2008), discrete-event simulation (Mori et al., 2007), global search meta-heuristics (Herrán, de la Cruz, & de Andrés, 2012), constraint programming (Moura et al., 2008), and optimization models are alternatively used at each step. These hybrid approaches are generally applied to mesh-structured pipeline networks with a particular feature: every pipeline is supplied by a single source node, and the stream flow is fully discharged into the output terminal at the other extreme.

3. Problem description

Fig. 1 depicts the aggregate transportation plan for a pipeline network involving a source node at the origin (N1), an intermediate dual-purpose terminal (N2), and two destination nodes (N3 and N4). Dual-purpose stations like N2 are intermediate pipeline terminals that can inject new lots and/or receive products from in-transit batches inserted at upstream sources. Then, they can simultaneously play the role of source and destination nodes. Of the four logistic nodes, N1 and N2 are source nodes, while N2, N3 and N4 can receive products from the line. The maximum flow rate in all pipeline segments (N1–N2, N2–N3 and N3–N4) is 200 units per hour. As shown in the second and third lines of Fig. 1, two sets (hereafter called blocks) of operations are planned to be performed during the time intervals [0:00–5:00 h] and [5:00–10:00 h], respectively. The first one consists of two parallel batch injections: (a) 800 units of product P1 pumped at the source node N1, and (b) 1000 units of P5 injected from node N2. At the same time, four product deliveries are accomplished. Node N2 receives 800 units of product P2, terminal N3 is supplied with 200 units of product P1, while 600 units of product P3 and 200 units of P1 are sent to node N4. The second block just comprises a single injection. A new batch with 1000 units of product P2 is pumped at N1 to deliver 400 units of product P1 and 600 units of P5 into the terminals N2 and N4, respectively.

At the detailed level of the planning process, both blocks should be disaggregated into a series of detailed operations. In the proposed problem representation, a single operation can involve more than a single batch injection, and is characterized by a particular set of on/off and open/closed states of pumps and valves throughout the pipeline network. In fact, the following conditions must hold along a detailed operation: (i) the inputted product and the pump rate at every active source should remain the same, and (ii) the diverted batch and the delivery rate to each terminal do not change.

As remarked in Section 2.2, previous works by Cafaro et al. (2011) and Gleizes et al. (2012) apply either discrete-event simulation or optimization approaches for disaggregating the transportation plan. In any case, the problem goal is to minimize the flow restarts in pipeline segments all along the planning horizon (Hane & Ratliff, 1995). To illustrate the way the disaggregation process is carried out, two alternative solutions to the aggregate plan shown in Fig. 1 are provided. The first one is based on the Nearest-First (NF) heuristic rule prioritizing the product delivery to the candidate terminal closest to the origin (N1) (Gleizes et al., 2012). The other one seeks to minimize the flow restarts in idle pipeline segments. Both detailed schedules allow the execution of parallel batch injections and simultaneous product deliveries to distribution terminals. In other words, two or more source nodes can concurrently inject new lots into the line while multiple depots receive some amounts of products from in-transit batches.

By prioritizing the candidate terminal closest to the origin, the first block of injections presented in Fig. 1 is disaggregated through the NF-rule into a sequence of six detailed operations (see Fig. 2a). The first operation is performed during the first hour. In that period, segment N1–N2 is activated by pumping 200 units of P1 from node N1, which pushes the same quantity of P2 out of the line into node N2 (highest priority node). Simultaneously, 200 units of P5 are injected at node N2 to displace the same volume of P3 into terminal N4. Though being a single detailed operation, two simultaneous injections have been performed. At time $t = 1$ h, the first batch of product P1 is properly located to be partially delivered to node N3. To do that, the flow through the segment N3–N4 is stopped, and the inlet valve of N3 is open during the second hour. From $t = 2$ h to $t = 4$ h, the flow in the segment N3–N4 is restarted to dispatch the remaining 400 units of P3 to node N4. Finally, during the fifth hour, segment N1–N2 is stopped while node N2 injects the remaining 200 units of product P5 to deliver 200 units of P1 to N4.

The first operation of the second block begins at $t = 5$ h. At that time, the states of valves and pumps are changed. Pumps at the source node N2 are turned off, and the inlet valves of terminal N4 are closed to supply node N2 with 400 units of P1 by injecting the same amount of product P2 at N1. Therefore, the only active segment over the first operation (from $t = 5$ h to $t = 7$ h) is N1–N2. During the second and last operation of this block, node N1 pumps 600 units of P2 into the pipeline to deliver a similar

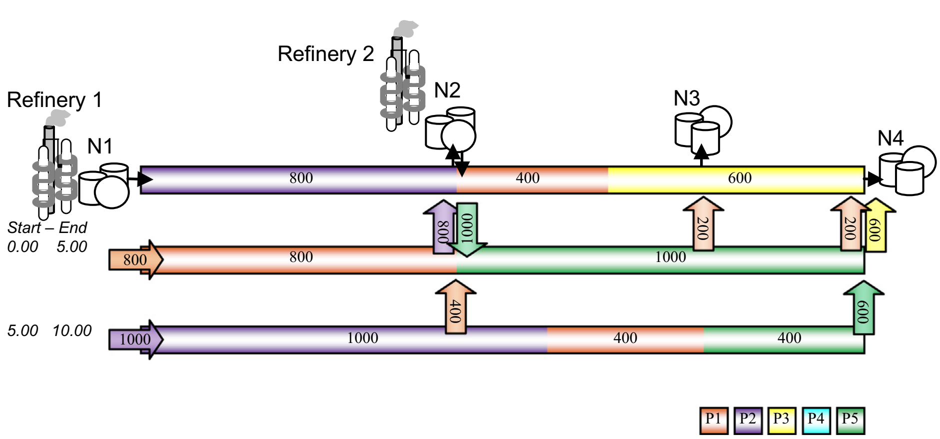

Fig. 1. The pipeline configuration and the blocks of operations to be disaggregated.

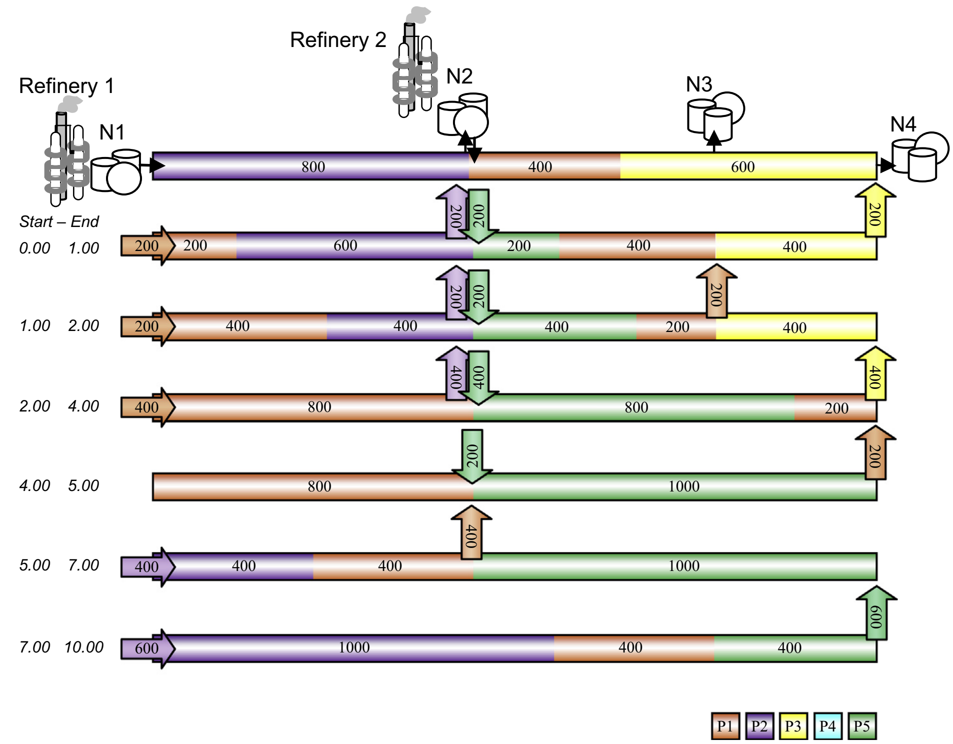

Fig. 2a. Detailed schedule obtained prioritizing closest-to-origin terminals.

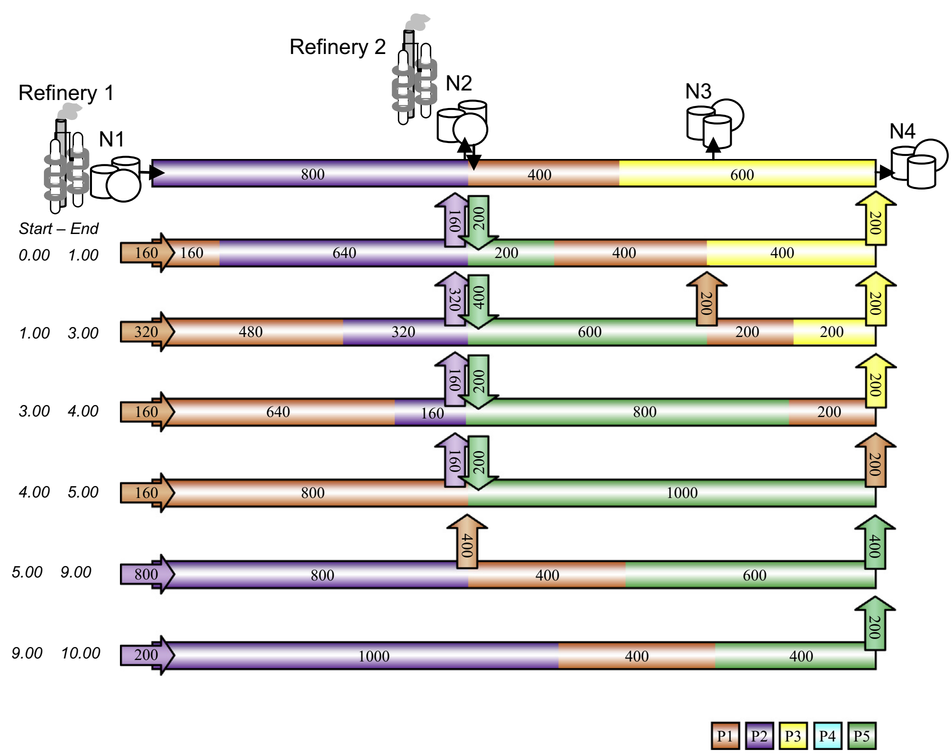

Fig. 2b. Detailed schedule obtained by minimizing flow restarts along the pipeline.

quantity of P5 into node N4. As a result, all the pipeline segments are activated and the flow rate in each one is set at the maximum value. Fig. 2a reveals that the flow is restarted twice in segment N3–N4 and only once in the pipeline sections N1–N2 and N2–N3, i.e. a total of 2200 volumetric units of products are put in motion during the planning horizon.

In turn, Fig. 2b shows a much better detailed schedule reducing the flow restarts in idle pipeline segments to zero. In this solution, six operations are also performed, but no flow stoppages and restarts occur. In other words, the pipeline operator pays no cost in terms of flow restarts. Its efficiency is based on the following actions:

(1) During the first five hours, the flow rate in segment N1–N2 is slowed down to 160 units per hour to synchronize the injections at nodes N1 and N2. By moderating the flow rate, we avoid unnecessary stoppages and restarts. From time $t = 1$ h to $t = 3$ h, the injection of 400 units of product P5 at node N2 forces simultaneous deliveries of products P1 and P3 to terminals N3 and N4, respectively.
(2) The inlet valve at node N4 remains open all the time to always have a finite flow in the last pipeline sections N2–N3 and N3–N4.
(3) When possible, deliveries to intermediate nodes are performed by partially diverting in-transit batches. For instance, from $t = 1$ h to $t = 3$ h, only half of the batch of product P1 is delivered to node N3, while the other portion continues moving to N4 at a rate of 100 units per hour.

A particularly complex situation arises at the starting time of the second block ($t = 5$ h), when one of the input stations is turned off. To still keep the fluid movement in all pipeline segments, we carefully plan a set of actions on pumps and valves: (a) the pump station at N2 is turned off; (b) half of the stream flow coming from segment N1–N2 is directed into segment N2–N3; and (c) the inlet valves at N2 and N4 are half-opened to receive products from the line. As a result, from time $t = 5$ h to $t = 9$ h, the injection of 800 units of P2 at N1 causes simultaneous deliveries of 400 units of P1 and 400 units of P5 to nodes N2 and N4, respectively.

The optimization model presented in the following section seeks to integrally solve all these complex and challenging issues arising in the operation of multi-source pipeline networks.

4. Mathematical formulation

In this section, we develop the optimization model for the detailed scheduling of multiproduct pipelines with multiple sources. The new approach is based on the mathematical model proposed by Cafaro et al. (2012) for the detailed scheduling of single-source pipelines. However, the new configuration requires a comprehensive reformulation. Model assumptions, major model elements (sets, parameters and variables) and their meanings are described in Sections 4.1–4.3. Finally, Section 4.4 presents the model formulation (constraints and objective function).

4.1. Problem assumptions

The optimization model for the detailed scheduling of multi-source pipelines is based on the following assumptions:

(a) We are dealing with a unidirectional pipeline with multiple sources and delivery nodes.
(b) The pipeline transports incompressible liquid products in ideal plug flow.
(c) The pipeline is full of products at any time.
(d) Product interfaces or "transmixes" have negligible sizes.
(e) An aggregate transportation plan is available.
(f) There are pipeline terminals having a dual (input/output) purpose. They can receive and inject products at the same time.
(g) Parallel injections taking place at different source nodes are allowed.
(h) Every product delivery from a batch in the pipeline is forced by only one batch injection.
(i) Each batch injection may produce concurrent product deliveries to an equal number of output terminals.
(j) Simultaneous product injections and deliveries start and finish at the same time, i.e. they cannot have a delayed start or be interrupted prematurely.
(k) The flow-rate in each pipeline segment keeps the same value all over a single operation.
(l) An admissible flow rate range for each pipeline segment is specified. Generally, it varies with the pipeline diameter and the power of pump stations.
(m) Operation and pump maintenance costs depend on both the cumulative volume of idle pipeline segments where the flow is restarted, and the number of on/off pump switchings.

4.2. Major problem sets

The mathematical model is formulated in terms of six major sets: (a) the pipeline terminals $j \in J = \{j_o\, j_1\, j_2 \ldots\}$, including input, output and dual-purpose nodes; (b) the pipeline segments $JS = J - \{j_o\} = \{j_1\ j_2 \ldots\}$ with the segment j connecting terminals $j-1$ and j; (c) the batches $i \in I$ moving through the pipeline over the planning horizon; (d) the chronologically ordered blocks of operations $b \in B$ specified by the aggregate plan; (e) the individual batch injections $r \in R$ planned at the aggregate level; and (f) the ordered set of detailed operations $k \in K$ performed by the pipeline operator to accomplish the aggregate plan. The number of detailed operations ($|K|$) is not known beforehand but can be estimated by a simple procedure, which is presented at the end of this section.

The problem data extracted from the aggregate plan also comprise the following subsets:

- The aggregate batch injections $r \in R_b \subset R$ concurrently performed within the block $b \in B$.
- The output terminals $j \in J_r^{\oplus} \subset J$ receiving products from the line through the batch injection r.
- The subset of depots $j \in J_{i,r}^{\oplus} \subset J_r^{\oplus}$ receiving a portion of batch i while executing the injection r.
- The batches $i \in I_b \subset I$ moving through the pipeline during the aggregate block of operations b.
- The string of pipeline segments $j \in JS_r \subseteq JS$ connecting the source node to the farthest receiving terminal supplied by the injection r.

We introduce the subset of detailed operations $k \in K_b$ in which the aggregate block b is to be decomposed. During an operation $k \in K_b$, simultaneous batch injections and multiple product deliveries may occur. For the sake of illustration, let us consider a refined products pipeline involving a pure-source node (j_o), a dual-purpose terminal (j_1) and a pure-receiving depot (j_2) as shown at the top of Fig. 3.

When block b_1 comprising a single injection (r_1) is executed, a new batch $i4$ is injected at the source node j_o and a portion of batch $i3$ is diverted from the line into depot j_1. Hence, $J_{r1}^{\oplus} = J_{i3,r1}^{\oplus} = \{j_1\}$. In the next block b_2, the pipeline operates in a partitioned scheme because two batch injections r_2 and r_3 are simultaneously accomplished. Injection r_2 from node j_o increases the size of batch $i4$ and pushes another part of batch $i3$ into depot j_1. At the same time, injection r_3 inserts a new batch $i2$ from terminal j_1 and pushes a portion of batch $i1$ into the receiving terminal j_2. Hence, $J_{r2}^{\oplus} = J_{i3,r2}^{\oplus} = \{j_1\}$ and $J_{r3}^{\oplus} = J_{i1,r3}^{\oplus} = \{j_2\}$. Note that by assumption (h), $\cap_{r \in R_b} J_r^{\oplus} = \varnothing$ (see Section 4.1).

The cardinality of the subset K_b is not known beforehand, but a conservative estimation is given by the total number of aggregate product deliveries planned within the block b, i.e. $|K_b| = \sum_{r \in R_b} \sum_{i \in I_b} |J_{i,r}^{\oplus}|$. This estimation assumes that just a single product delivery takes place at any detailed operation. However, some operations may comprise more than one product delivery, especially if multiple batch injections are accomplished. Then, a better estimation of $|K_b|$ is given by the maximum number of product deliveries pushed by any of the injections in the block b, i.e. $|K_b| = \max_{r \in R_b} \left\{ \sum_{i \in I_b} |J_{i,r}^{\oplus}| \right\}$.

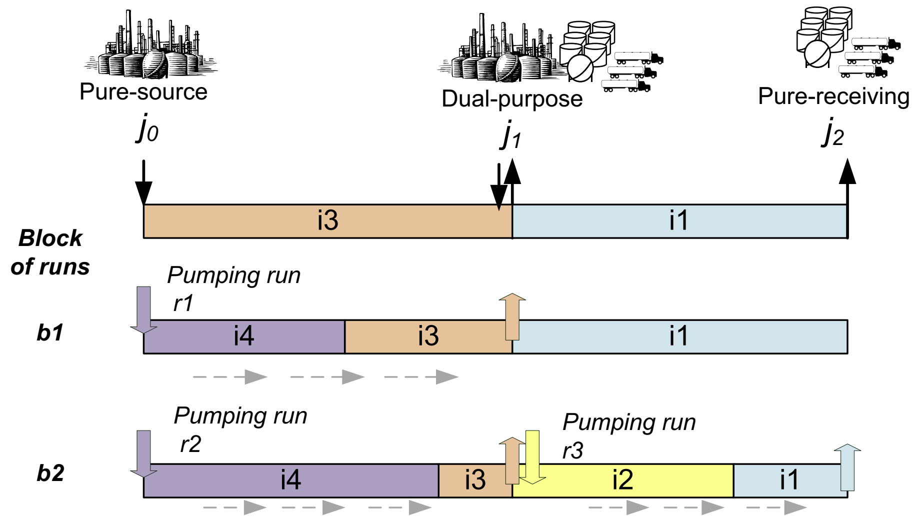

Fig. 3. An aggregate plan illustrating a composite input operation.

4.3. Major problem variables

The optimization model comprises discrete and continuous decision variables. On the one hand, the binary variable u_k stands for the existence of the detailed operation k; $x_{i,j,k}$ denotes the stripping of a portion of batch i to depot j during operation k; and the variable ω_{jk} indicates if the pipeline segment j connecting terminals $(j-1)$ and j is or is not active during operation k. On the other hand, the continuous variables taking part of the model are: (a) the volume of the injection r pumped during operation k ($Q_{r,k}$); (b) the length (L_k) and the completion time (C_k) of every detailed operation k; (c) the volume of batch i delivered to depot j during the execution of operation k, ($D_{i,j,k}$); (d) the volume of batch i at the completion time C_k of the detailed operation k ($W_{i,k}$); (e) the front coordinate of batch i at time C_k ($F_{i,k}$); (f) the activated volume ($AV_{j,k}$), if the stream flow in segment j is restarted when performing operation k; and (g) the stopped volume ($SV_{j,k}$), if the stream flow in segment j ceases during operation k.

4.4. Problem constraints

In the proposed formulation, constraints fall into five groups: (1) Model restrictions defining the volumes injected and delivered during a detailed operation k, as well as the timing and length of operation k, so that the aggregate transportation plan is accomplished and terminal demands are fulfilled. (2) Constraints monitoring the size and location of the batches moving through the pipeline. (3) Model equations controlling the physical feasibility and the size of product deliveries at every operation k. (4) Problem constraints identifying active and idle pipeline segments at every operation k. (5) Equations determining the volume stopped and/or restarted at every operation k.

4.4.1. Detailed batch injections and deliveries

Detailed batch injections. Every batch injection planned at the aggregate level is to be decomposed into detailed pumping operations. The total volume pumped into the pipeline at the active source of injection r should be equal to the specified amount qq_r fixed at the aggregate level.

$$\sum_{k\in K_b} Q_{r,k} = qq_r \quad \forall b\in B,\ r\in R_b \tag{1}$$

Fulfillment of depots' demands. The total volume of batch i supplied to terminal $j \in J_{i,r}^{\oplus}$ during the whole injection $r\in R_b$ of block b (i.e., through all the detailed operations $k\in K_b$) must be equal to the prescribed volume $dd_{i,j}^{(r)}$, aiming to fulfill customers' orders.

$$\sum_{k\in K_b} D_{i,j,k} = dd_{i,j}^{(r)} \quad \forall b\in B,\ i\in I_b,\ r\in R_b,\ j\in J_{i,r}^{\oplus} \tag{2}$$

Volume balance. Due to the product incompressibility assumption, an exact balance between input and output volumes at every operation k is satisfied. Parallel injections in block b ($r\in R_b$) produce batch movements and product deliveries that do not overlap because they involve different strings of pipeline segments ($\cap_{r\in R_b} JS_r = \varnothing$), different sources and output terminals ($\cap_{r\in R_b} J_r^{\oplus} = \varnothing$). For this reason, a balance equation between input and output volumes at every operation k can be written for every individual injection r.

$$\sum_{i\in I_b}\sum_{j\in J_{i,r}^{\oplus}} D_{i,j,k} = Q_{r,k} \quad \forall b\in B,\ r\in R_b,\ k\in K_b \tag{3}$$

Sequencing operations. A detailed operation $k\in K_b$ making part of block b, and possibly involving multiple batch injections, must never start before completing the previous operation $(k-1)\in K_b$.

$$C_k - L_k \geqslant C_{k-1} \quad \forall (k-1),\ k\in K_b,\ b\in B \tag{4}$$

Start and end times of the elements of the set K_b. The first and the last detailed operations in the block b must comply with the start and end times specified for that block in the aggregate transportation plan.

$$C_k - L_k = st_b \quad \forall b\in B,\ k = first(K_b) \tag{5}$$

$$C_k = ft_b \quad \forall b\in B,\ k = last(K_b) \tag{6}$$

Admissible range for the length of a detailed operation k. Assuming that the model parameters $l_{\min}$ and $l_{\max}$ represent the minimum and maximum allowed length of a detailed operation, then the length of every operation must belong to the range given by constraint (7).

$$l_{\min} u_k \leqslant L_k \leqslant l_{\max} u_k \quad \forall b\in B,\ k\in K_b \tag{7}$$

The binary variable u_k denotes the existence of operation k in case $u_k = 1$. If the element k is not used in the optimal solution, it will stand for a fictitious operation. In that case, $u_k = 0$ and constraint (7) drives the length L_k to zero.

Lower and upper bounds on the input rate. Let us assume that $vb_{\min}^{(r)}$ and $vb_{\max}^{(r)}$ stand for the minimum and maximum allowed rates for injection r. Hence, the volume $Q_{r,k}$ of injection r pumped into the pipeline during the operation k should be confined to the interval given by restriction (8).

$$vb^{(r)}_{\min}L_k \leqslant Q_{r,k} \leqslant vb^{(r)}_{\max}L_k \quad \forall b \in B,\ r \in R_b,\ k \in K_b \tag{8}$$

For fictitious operations, $L_k = 0$ and $Q_{r,k} = 0$ for all $r \in R_b$.

Fictitious elements in the set K_b. To avoid symmetric solutions, we include constraint (9). It makes fictitious operations k of block b ($k \in K_b$) to arise last in the set K_b. Moreover, they always satisfy $C_k - L_k = C_k = ft_b$ because of Eqs. (6) and (7).

$$u_k \leqslant u_{k-1} \quad \forall b \in B, (k-1),\ k \in K_b \tag{9}$$

4.4.2. Tracking batches along the pipeline

Location of two consecutive batches at the end of every detailed operation

Given that the pipeline is always full of products, the front coordinate of batch $(i+1)$ at the end of operation k, i.e. $F_{i+1,k}$, is equal to the back coordinate of the preceding batch $i \in I_b$ given by $(F_{i,k} - W_{i,k})$, where $W_{i,k}$ is the size of batch i at time C_k. In constraint (8), the set I_b includes batches travelling through the pipeline within the time interval $[st_b, ft_b]$.

$$F_{i+1,k} = F_{i,k} - W_{i,k} \quad \forall b \in B,\ i \in I_b,\ k \in K_b \tag{10}$$

Note that batch coordinates are all measured from the origin of the first pipeline segment. Absolute volume scales are distinctive features of continuous representations of pipeline networks.

Size of a batch at the end time of a detailed operation. The size of batch i at time C_k can be computed from its value at the end time of the previous operation $[W_{i,(k-1)}]$ by adding the volume of injection r merged with i, and subtracting the volume delivered from batch i to output terminals $j \in J^{\oplus}_{i,r}$. In constraint (11), the set $J^{\oplus}_{i,r}$ represents the active terminals receiving material from batch i, pushed by injection r. Moreover, the continuous variable $D_{i,j,k}$ denotes the volume of batch i delivered to terminal $j \in J^{\oplus}_{i,r}$ during operation k. The parameter $a_{i,r}$ provides a valuable information about the transportation plan to be disaggregated. If $a_{i,r} = 1$, then the batch i is totally or partially pumped through injection r. Otherwise, $a_{i,r} = 0$.

$$W_{i,k} = W_{i,(k-1)} + \sum_{r \in R_b}\left[a_{i,r}Q_{r,k} - \sum_{j \in J^{\oplus}_{i,r}} D_{i,j,k}\right] \quad \forall b \in B,\ i \in I_b,\ k \in K_b \tag{11}$$

For $k = 1$, $W_{i,(k-1)} = wo_i$, i.e. the initial volume of batch i.

4.4.3. Product delivery constraints

Conditions to divert a portion of a batch to a depot. The delivery of a portion of batch i to terminal j during operation k can occur only if: (a) batch i has reached the location of terminal j at the end of operation $(k-1)$, and (b) batch i has not surpassed terminal j at the end of operation k. Such feasibility conditions are forced through constraints (12) and (13). The parameter pv denotes the total volume of the pipeline network.

$$F_{i,(k-1)} \geqslant \sigma_j x_{i,j,k} \quad \forall b \in B,\ i \in I_b,\ r \in R_b,\ j \in J^{\oplus}_{i,r},\ k \in K_b \tag{12}$$

$$F_{i,k} - W_{i,k} \leqslant \sigma_j + (pv - \sigma_j)(1 - x_{i,j,k}) \quad \forall b \in B,\ i \in I_b,\ r \in R_b,\ j \in J^{\oplus}_{i,r},\ k \in K_b \tag{13}$$

No deliveries in fictitious operations. No product deliveries occur during detailed operations that are fictitious, i.e. operations k featuring $u_k = 0$. Moreover, at most one batch can be delivered to each terminal during a single operation.

$$\sum_{i \in I_b} x_{i,j,k} \leqslant u_k \quad \forall b \in B,\ r \in R_b,\ j \in J^{\oplus}_r,\ k \in K_b \tag{14}$$

Maximum output rate. If $vd^{(j)}_{\max}$ denotes the maximum admissible rate for diverting batches from the pipeline into terminal j, the delivery size is related to the operation length as in constraint (15).

$$D_{i,j,k} \leqslant vd^{(j)}_{\max}L_k \quad \forall b \in B,\ i \in I_b,\ r \in R_b,\ j \in J^{\oplus}_{i,r},\ k \in K_b \tag{15}$$

Delivery size. The delivery of batch i to terminal j during operation k has a finite size whenever $x_{i,j,k} = 1$. Then,

$$d_{\min}x_{i,j,k} \leqslant D_{i,j,k} \leqslant dd^{(r)}_{i,j}x_{i,j,k} \quad \forall b \in B,\ i \in I_b,\ r \in R_b,\ j \in J^{\oplus}_{i,r},\ k \in K_b \tag{16}$$

Parameter $d_{\min}$ normally takes a small positive value, while $dd^{(r)}_{i,j}$ stands for the overall volume to be delivered from batch i to terminal j during injection r.

4.4.4. Active and idle pipeline segments during a detailed operation

Let us define the continuous variable $\omega_{j,k}$ to represent the state of the pipeline segment $j \in JS$ during the operation k. Its value is confined to the closed interval [0, 1]. Segment j is active during operation k if there is a fluid movement through it, and consequently $\omega_{j,k} = 1$. Otherwise, segment j is idle and $\omega_{jk} = 0$. To characterize the state of a pipeline segment during a non-fictitious operation the proposed formulation incorporates constraints (17)

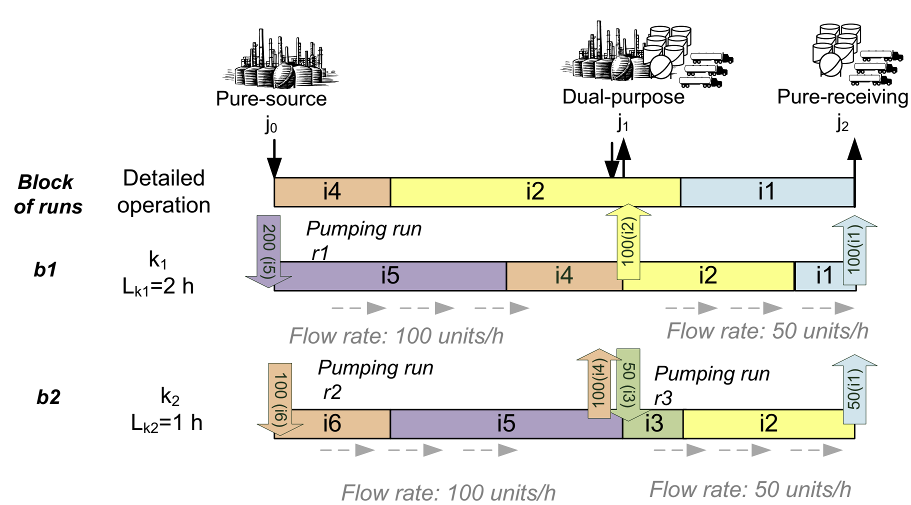

Fig. 4. Monitoring stream flow-rates in pipeline segments.

and (18) for every injection r of every block b. On the one hand, constraint (17) states that the segment $j \in JS_r$ is active during operation k only if there is a product delivery to any terminal $j' \geqslant j$, with $j' \in J_r^{\oplus}$ (i.e., terminals being supplied by injection r). On the other hand, restriction (18) states that segment $j \in JS_r$ is idle if no product delivery to any terminal $j' \geqslant j$ (with $j' \in J_r^{\oplus}$) takes place during the non-fictitious operation k.

$$\omega_{j,k} \geqslant \sum_{i \in I_b} x_{i,j',k} \quad \forall b \in B,\ r \in R_b,\ j \in JS_r,\ j' \in J_r^{\oplus}(j' \geqslant j),\ k \in K_b \tag{17}$$

$$\omega_{j,k} \leqslant \left[\sum_{i \in I_b} \sum_{\substack{j' \in J_r^{\oplus} \\ j' \geqslant j}} x_{i,j',k} \right] - u_k + 1 \quad \forall b \in B,\ r \in R_b,\ j \in JS_r,\ k \in K_b \tag{18}$$

Finally, pipeline segments not belonging to the set $JS_b = \cup_{r \in R_b} JS_r$ will also be idle at every operation of block b, as stated by Eq. (19).

$$\omega_{j,k} = 0 \quad \forall b \in B,\ j \notin JS_b,\ k \in K_b \tag{19}$$

For fictitious operations featuring $u_k = 0$, constraints (17) and (18) are redundant. It is important to point out that the value of $\omega_{j,k}$ for fictitious operations resembles the state of segment j at the last non-fictitious operation. This is achieved by including new constraints (20) and (21) that are both redundant if $u_k = 1$. In other words, $\omega_{j,k} = \omega_{j,(k-1)}$ for any fictitious operation k.

$$\omega_{j,k} \geqslant \omega_{j,(k-1)} - u_k \quad \forall b \in B, (k-1), k \in K_b,\ j \in JS \tag{20}$$

$$\omega_{j,k} \leqslant \omega_{j,(k-1)} + u_k \quad \forall b \in B, (k-1), k \in K_b,\ j \in JS \tag{21}$$

Controlling the flow rate in pipeline segments. The flow rate in every active segment j during operation k ($\omega_{j,k} = 1$) should belong to the admissible range given by $[vb_{\min}^{(j)}; vb_{\max}^{(j)}]$. The total volume

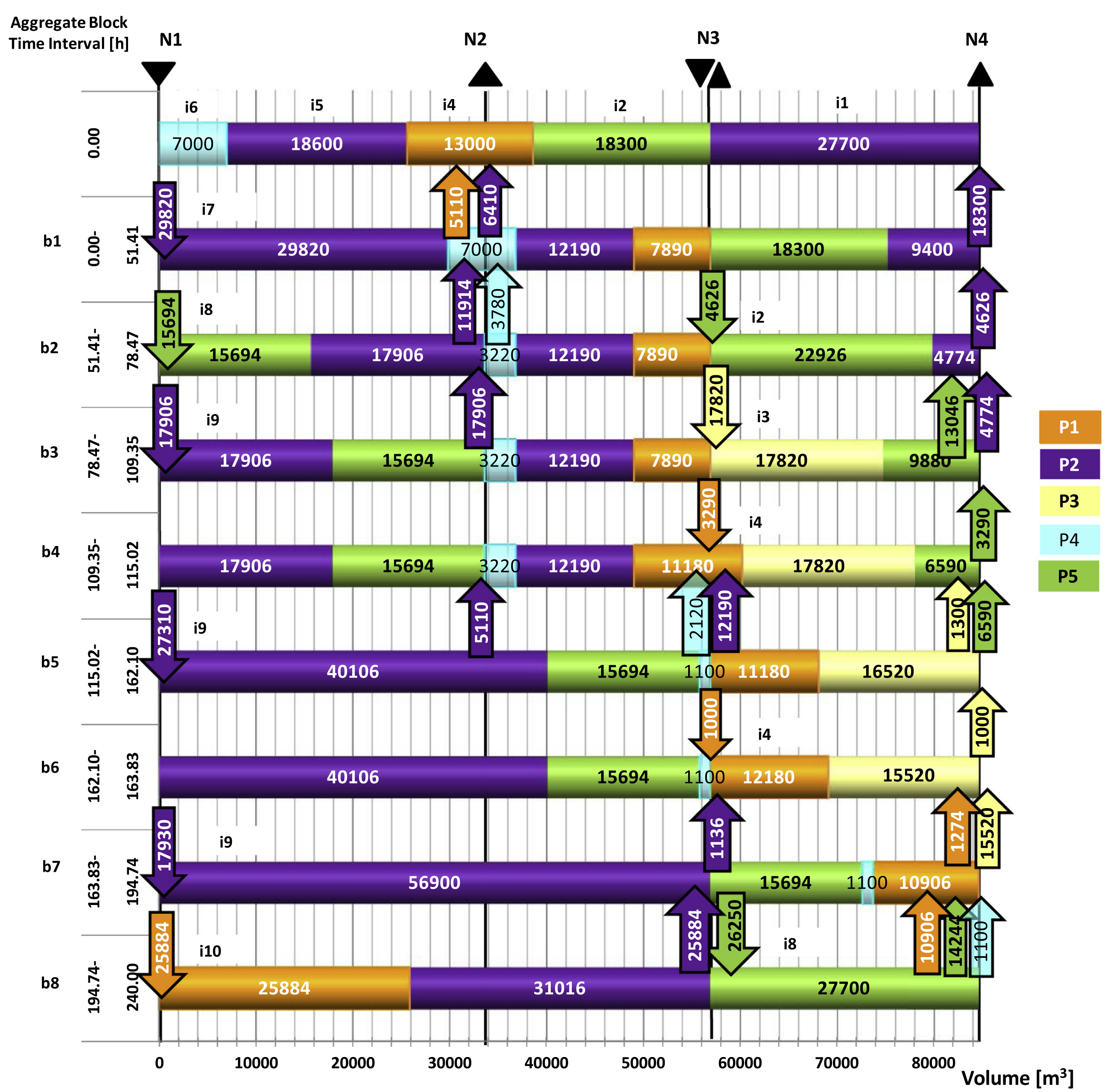

Fig. 5. Aggregate pipeline schedule reported by Cafaro and Cerdá (2010).

flowing through an active segment j during operation k is computed by summing all the products deliveries $D_{i,j,k}$ to downstream terminals $j' \in J_r^{\oplus}$ (with $j' \geqslant j$).

$$vb^{(j)}_{\min} L_k - q_{\max}(1 - \omega_{j,k}) \leqslant \sum_{i \in I_b} \sum_{\substack{j' \in J^{\oplus}_{i,r} \\ j' \geqslant j}} D_{i,j',k} \leqslant vb^{(j)}_{\max} L_k$$
$$\forall b \in B,\ r \in R_b,\ j \in JS_r,\ k \in K_b \tag{22}$$

$q_{\max}$ denotes the maximum volume that can be injected through a single operation.

Fig. 4 presents a simple example illustrating the flow rate calculation. It consists of a pipeline with a source node j_o, a dual-purpose terminal j_1, and a receiving terminal j_2. The first line in Fig. 4 shows the initial state of the pipeline. Three batches {$i1$, $i2$, $i4$} are in transit at $t = 0$. The next line presents the state of the pipeline after executing the block b_1, just comprising a single injection, i.e. $R_{b1} = \{r_1\}$. The active source node for r_1 is j_0, and the sets of active receiving terminals and pipeline segments are: $J_{r1}^{\oplus} = \{j_1,\ j_2\}$, $JS_{r1} = \{j_1,\ j_2\}$, $J^{\oplus}{}_{i2,r1} = \{j_1\}$, $J^{\oplus}{}_{i1,r1} = \{j_2\}$. Block b_1 is executed in only one detailed operation (k_1) whose length is $L_{k1} = 2$ h. As a result, the flow rate in segment j_1 is estimated by: $[D_{i2,j1,k1} + D_{i1,j2,k1}]/L_{k1} = (100 + 100)/2$ h = 200/2 = 100 units/h. In segment j_2, the estimated flow rate is 50 units/h because $D_{i1,j2,k1} = 100$.

The bottom of Fig. 4 depicts the final state of the pipeline after executing block b_2. This block involves two injections ($R_{b2} = \{r_2, r_3\}$) that are accomplished through a single operation k_2, whose length is $L_{k2} = 1$ h. The active sources for injections r_2 and r_3 are the nodes j_0 and j_1, pumping batches $i6$ and $i3$, respectively. Moreover, the sets of active receiving terminals and pipeline segments for the individual injections r_2 and r_3 are: $J_{r2}^{\oplus} = \{j_1\}$, $J_{r3}^{\oplus} = \{j_2\}$, $JS_{r2} = \{j_1\}$, $JS_{r3} = \{j_2\}$, $J^{\oplus}{}_{i4,r2} = \{j_1\}$, and $J^{\oplus}{}_{i1,r3} = \{j_2\}$. For pipeline segments j_1 and j_2 the flow-rates are given by: $D_{i4,j1,k2}/L_{k2} = 100/1 = 100$ units/h, and $D_{i1,j2,k2}/L_{k2} = 50/1 = 50$ units/h. Note that flow rates remain unchanged with regards to the previous operation, but in this case two simultaneous injections are performed instead of a single one.

4.4.5. Start and stop volumes during a detailed operation

To determine the operating costs at each detailed operation it is necessary to identify the pipeline segments where the stream flow is stopped or restarted. This is achieved by comparing the state of each pipeline segment $j \in JS$ in two successive operations. The volume of the pipeline segment j connecting terminals $(j - 1)$ and j is the difference between the volumetric coordinates σ_j and

Table 1
Size, active terminals and active pipeline segments for each batch injection.

Block	Constituent injections	Injected batch	Injected batch size (qq_r) in m^3	Active receiving terminals ($J_r^{\oplus}$)	Candidate for active segments (JS_r)
b1	r1	i7	29,820	N2, N4	N1–N2, N2–N3, N3–N4
b2	r2	i8	15,694	N2	N1–N2
	r3	i2	4626	N4	N3–N4
b3	r4	i9	17,906	N2	N1–N2
	r5	i3	17,820	N4	N3–N4
b4	r6	i4	3290	N4	N3–N4
b5	r7	i9	27,310	N2, N3, N4	N1–N2, N2–N3, N3–N4
b6	r8	i4	1000	N4	N3–N4
b7	r9	i9	17,930	N3, N4	N1–N2, N2–N3, N3–N4
b8	r10	i10	25,884	N3	N1–N2, N2–N3
	r11	i8	26,250	N4	N3–N4

Table 2
Source node and aggregate product deliveries for every injection.

Block	Starting time (h)	Completion time (h)	Constituent injections	Source node	Aggregate product deliveries		
					Batch	Terminal	Size
b1	0.00	51.41	r1	N1	i4	N2	5110
					i5	N2	6410
					i1	N4	18,300
b2	51.41	78.47	r2	N1	i6	N2	3780
					i7	N2	11,914
			r3	N3	i1	N4	4626
b3	78.47	109.34	r4	N1	i7	N2	17,906
			r5	N3	i2	N4	13,046
					i1	N4	4774
b4	109.34	115.02	r6	N3	i2	N4	3290
b5	115.02	162.10	r7	N1	i9	N2	5110
					i5	N3	12,190
					i6	N3	2120
					i2	N4	6590
					i3	N4	1300
b6	162.10	163.83	r8	N3	i3	N4	1000
b7	163.83	194.74	r9	N1	i9	N3	1136
					i3	N4	15,520
					i4	N4	1274
b8	194.74	240.00	r10	N1	i9	N3	25,884
			r11	N3	i4	N4	10,906
					i6	N4	1100
					i8	N4	14,244

σ_{j-1}; with $\sigma_o = 0$ representing the coordinate of the source node at the pipeline origin. Hence, the activated and stopped volumes during operation k are determined by the new constraints (23) and (24), respectively. If segment j is idle at operation $(k-1)$ and the flow is restarted during the next operation k, then $\omega_{j,k} - \omega_{j,(k-1)} = 1$ and the activated volume is $AV_{j,k} = \sigma_j - \sigma_{j-1}$, i.e. the volume of segment j. In the opposite case, $\omega_{j,(k-1)} - \omega_{j,k} = 1$ and the stream flow in segment j is stopped during operation k. Besides, the stopped volume is given by $SV_{j,k} = \sigma_j - \sigma_{j-1}$.

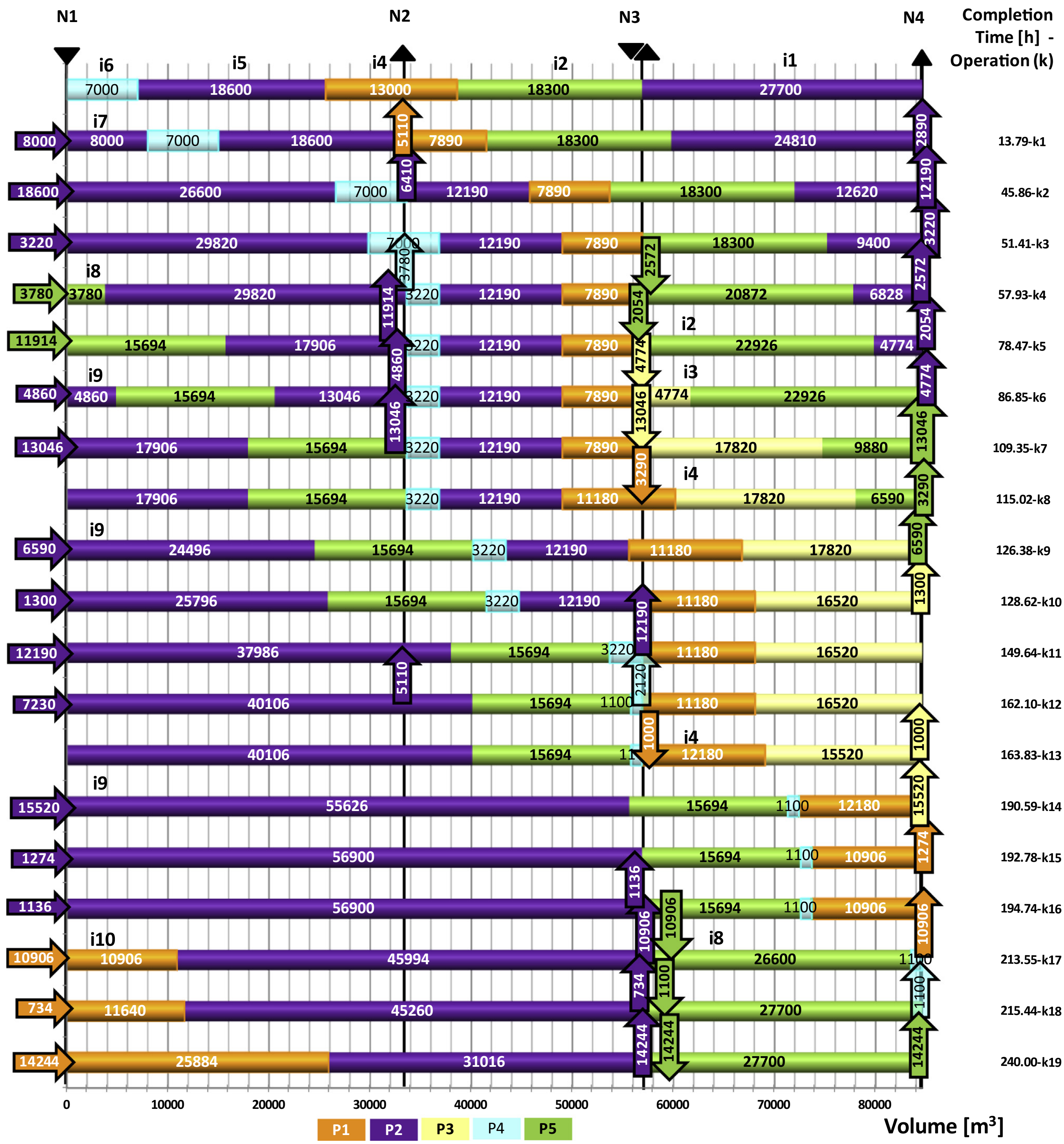

Fig. 6. Optimal detailed schedule of pumping and delivery operations.

Table 3
Model sizes, computational requirements and results.

Aggregate blocks $\lvert B\rvert$	Injections $\lvert R\rvert$	Detailed operations $\lvert K\rvert$	Activated volume (10^2 m^3)	Stopped volume (10^2 m^3)	CPU time (s)	Cont. vars.	Binary vars.	Eqs.
8	11	19	2538	1692	0.107	1199	116	2190

$$SV_{j,k} \geqslant (\sigma_j - \sigma_{j-1})(\omega_{j,(k-1)} - \omega_{j,k}) \quad \forall j \in JS,\ k \in K \tag{23}$$

$$AV_{j,k} \geqslant (\sigma_j - \sigma_{j-1})(\omega_{j,k} - \omega_{j,(k-1)}) \quad \forall j \in JS,\ k \in K \tag{24}$$

For $k = first\ (K)$, $\omega_{j,(k-1)}$ indicates the state of segment j at the start of the planning horizon. If it is equal to one, then segment j is active at time $t = 0$, and zero otherwise.

4.5. Objective function

The problem goal is to develop a detailed pipeline schedule that fulfills the aggregate demands at minimum flow restart and on/off pump switching costs, through the least number of operations. Eq. (25) accounts for the total flow restart and stoppage costs by adding the activated and stopped volumes over all the detailed operations, and multiplying the result by the corresponding unit cost coefficients ca and cs. In turn, the last term in the RHS of Eq. (25) seeks to minimize the total number of non-fictitious operations.

$$Min\ z = \sum_{k \in K} \sum_{j \in JS} (cs\, SV_{j,k} + ca\, AV_{j,k}) + \sum_{k \in K} fco\, u_k \tag{25}$$

5. Computational experiment

In this section, we apply the proposed detailed scheduling approach to a real-world example. It involves the transportation of five different oil refined products (P1, P2, P3, P4, P5) from two source nodes to three distribution terminals.

This example, first introduced by Cafaro and Cerdá (2010), deals with a pipeline network consisting of a series of four logistic nodes (N1, N2, N3, N4), with two of them (N1 and N3) acting as source nodes. Simultaneous batch injections from nodes N1 and N3 into the pipeline are permitted. A sketch of the pipeline system is shown at the top of Fig. 5. Node N3 is indeed a dual-purpose terminal that can inject a new batch into the pipeline and simultaneously receive an amount of product from a batch coming from an upstream source. The injection rate at both sources should be kept between 310 and 580 m^3/h. In turn, depots N2, N3 and N4 are the destinations for the batches transported by the pipeline. The output node N2 demands specific amounts of products P1, P2 and P4; the dual terminal N3 is another destination for lots of products P2 and P4; while terminal N4 demands all transported products. The length of the planning horizon is 240 h (ten days). Product demands at depots and initial product inventories at source nodes can be found in Cafaro and Cerdá (2010).

From Fig. 5, it follows that the pipeline network is composed of three pipeline segments, each one connecting a pair of consecutive nodes. The overall length of the pipeline network, from node N1 (at the origin) to terminal N4 (at the pipeline end), is over 1000 km, and the volumes of the three pipeline segments (N1–N2, N2–N3 and N3–N4) are 336, 233, and 277 [10^2 m^3], respectively. To avoid batch contamination, product sequences P2–P3, P3–P2, P3–P4 and P4–P3 are all forbidden.

The first line of Fig. 5 shows the initial pipeline state containing a sequence of five batches: $i1^{P2}$–$i2^{P5}$–$i4^{P1}$–$i5^{P2}$–$i6^{P4}$, with superscripts indicating the product of each batch. At the aggregate level, the problem goal is aimed at finding the pipeline transportation plan for the next ten days that satisfies all depot demands at minimum total (pumping, transition and backorder) cost. Backorder costs may arise because of unsatisfied demands at the end of the time horizon. The maximum flow rate in every pipeline segment is 580 m^3/h, while the minimum flow rate in active segments is 100 m^3/h.

The optimal aggregate plan for this example is shown in Fig. 5. The CPU time needed to find this solution is 687.4 s, using an Intel Xeon Due-Processor (2.67 GHz) with GAMS/GUROBI 4.5 (Brooke, Kendrick, Meeraus, & Raman, 2006). It comprises a total of 11 batch injections (6 from node N1 and 5 from N3) that are grouped into 8 blocks. Three blocks (b2, b3, b8) consist of a pair of parallel injections, and the other five include only one. Since some injections just add further amounts of products to existing batches, a total of 10 lots are transported by the pipeline over the ten-day horizon. The aggregate transportation plan includes 23 product deliveries to receiving terminals. The left side of Fig. 5 shows the starting and completion times of every block. From that figure, all the problem data (sets and parameters) used by the detailed scheduling model can be derived. They include the sets of potentially active pipeline segments, aggregate product deliveries, injections and source nodes. All of them are listed in Tables 1 and 2. We assume that the pipeline network is initially idle. Furthermore, cost coefficients used in the objective function are: ca = 0.10 [\$/m^3], cs = 0.00 [\$/m^3], and fco = 1000 [\$/operation].

As already explained, two alternative criteria can be applied to choose the cardinality of every set K_b. They are: (a) the total number of aggregate deliveries performed while executing the block of batch injections given by: $\sum_{r \in R_b} \sum_{i \in I_b} |J^{\oplus}_{i,r}|$, and ($b$) the maximum number of product deliveries prescribed by the aggregate plan for a certain injection of the block, i.e. $\max_{r \in R_b} \left\{ \sum_{i \in I_b} |J^{\oplus}_{i,r}| \right\}$. Using the first criterion, the value of $|K_b|$ for the 8 aggregate blocks are: [3, 3, 3, 1, 5, 1, 3, 4], and $|K|$ = 23. The other criterion leads to the following choices for $|K_b|$: [3, 2, 2, 1, 5, 1, 3, 3], and therefore $|K|$ = 20. Criterion (b) was selected for this experiment.

Based on all the previous data, the proposed MILP mathematical model is solved to optimality on an Intel Xeon Due-Processor (2.67 GHz) with GAMS/GUROBI 4.5 solver, using four parallel threads. The optimality tolerance is set at 10^{-9}. The optimal detailed schedule is found in 0.107 s of CPU time, and is presented in Fig. 6. No change in the optimal solution was observed if the criterion (a) is used to choose the cardinality of every set K_b. Computational results, including the model size, the number of cut operations and the best objective value are all reported in Table 3. The optimal schedule includes a sequence of 19 detailed operations to comply with the aggregate plan.

To carry out the first block $b1$ within the time interval [0.00 h; 51.41 h], a series of three detailed pumping operations (k_1, k_2, k_3) is performed. From time 0.00 h to 13.79 h (operation k_1) the source node N1 starts pumping batch $i7^{P2}$ into the pipeline, while terminal

Table 4
Variation of the flow rate in every pipeline segment.

Blocks	Detailed operation k	Flow rates (m^3/h)			Active source nodes	Active output nodes
		Segment N1–N2	Segment N2–N3	Segment N3–N4		
	k1	580	210	210	N1	N2, N4
b1	k2	580	380	380	N1	N2, N4
	k3	580	580	580	N1	N4
b2	k4	580	0	395	N1	N2, N4
	k5	580	0	100	N1, N3	N2, N4
b3	k6	580	0	570	N1, N3	N2, N4
	k7	580	0	580	N1, N3	N2, N4
b4	k8	0	0	580	N3	N4
	k9	580	580	580	N1	N4
b5	k10	580	580	580	N1	N4
	k11	580	580	0	N1	N3
	k12	580	170	0	N1	N2, N3
b6	k13	0	0	580	N3	N4
	k14	580	580	580	N1	N4
b7	k15	580	580	580	N1	N4
	k16	580	580	0	N1	N3
	k17	560	580	580	N1, N3	N3, N4
b8	k18	580	387	580	N1, N3	N3, N4
	k19	580	580	580	N1, N3	N3, N4

N2 receives 5110 m^3 of product P1, and terminal N4 2890 m^3 of product P2. Therefore, there is a finite flow all along the pipeline network. The second operation (k_2) going from $t = 13.79$ h to $t = 45.86$ h continues pumping lot $i7^{P2}$ from node N1 to supply depots N2 and N4 with 6410 m^3 and 12,190 m^3 of P2, respectively. As a result, all the segments still remain active. Finally, the last detailed operation of block $b1$, running from $t = 45.86$ h to $t = 51.41$ h, delivers the remaining demand of product P2 (3220 m^3) to node N4 by completing the injection of batch $i7^{P2}$.

A detailed tracking of the flow rate in every pipeline segment is shown in Table 4 and Fig. 7. During operations k_1, k_2 and k_3 the maximum allowed flow rate of 580 m^3/h is reached in segment N1–N2. In contrast, pipeline sections N2–N3 and N3–N4 present lower flow rates (210 and 380 m^3/h) during operations k_1 and k_2 because of the partial product deliveries to node N2 at rates of 370 and 200 m^3/h, respectively. Afterwards, both segments work at the maximum flow rate of 580 m^3/h throughout operation k_3 (see Table 4).

The next aggregate block $b2$ is implemented through two detailed operations (k_4, k_5). Operation k_4 begins the injection of lot $i8^{P5}$ into the pipeline from node N1 producing a delivery of 3780 m^3 of product P4 to terminal N2. At the same time, a batch of product P5 is also injected from node N3 to push 2572 m^3 of product P2 out of the line to terminal N4. As a result, the flow is stopped in segment N2–N3. In the succeeding operation k_5, both batch injections are continued, but at different pump rates. On

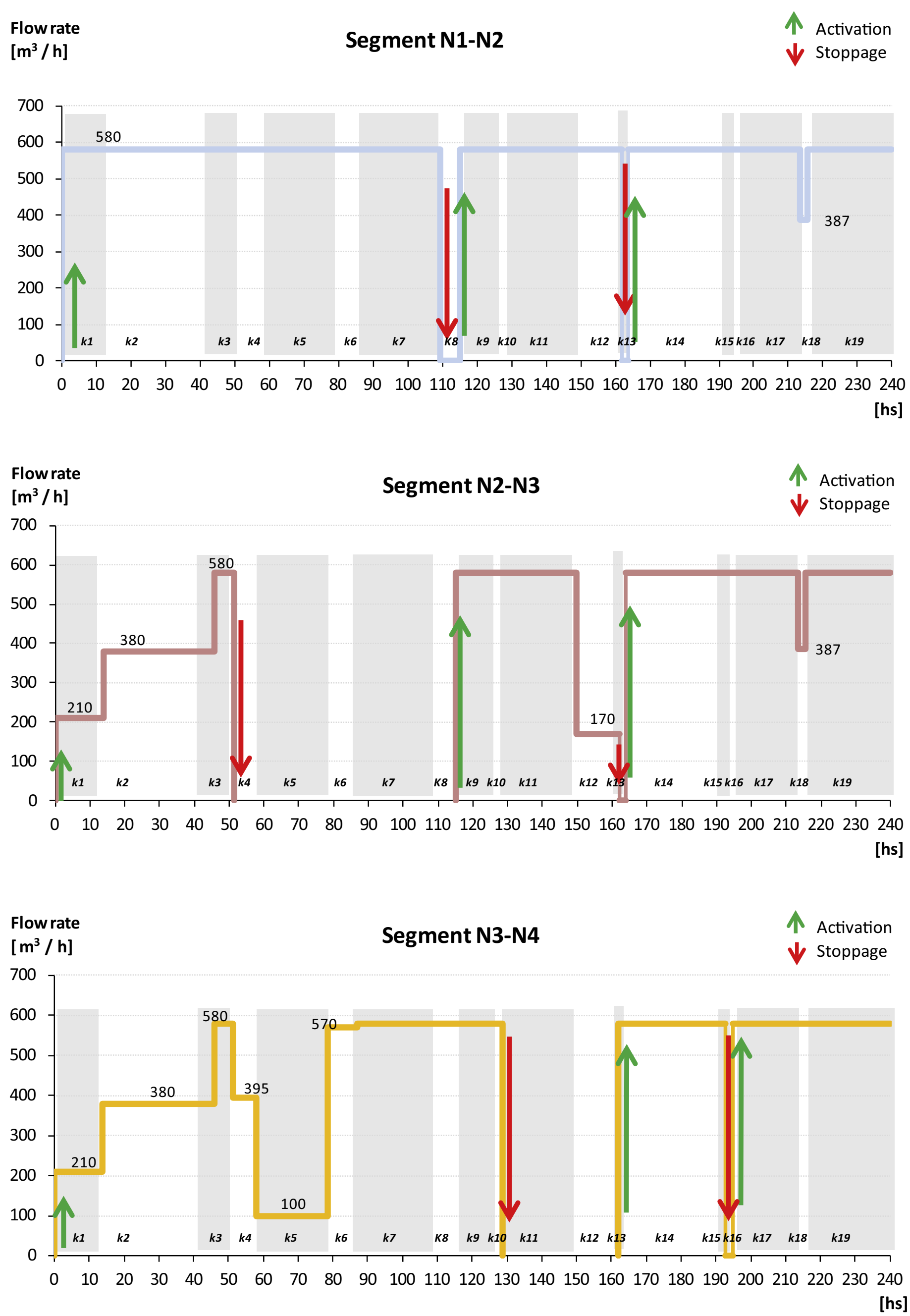

Fig. 7. Tracking the flow rate in every pipeline segment over the planning horizon.

the one hand, the second injection of lot $i8^{P5}$ from node N1 pushes 11,914 m^3 of product P2 into terminal N2 at full rate. On the other hand, an extra amount of P5 is pumped into batch $i2^{P5}$ from N3 to divert another portion of $i1^{P2}$ to the receiving depot N4 at the lowest admissible rate (100 m^3/h).

Table 4 reports the list of detailed operations together with the flow rate in every pipeline segment. There are: (i) three restarts of segment N1–N2 during operations k_1, k_9, k_{14}, (ii) three restarts of segment N2–N3 during the same operations k_1, k_9, k_{14}, and (iii) three restarts of segment N3–N4 during operations k_1, k_{13}, and k_{17}. Besides, there are six flow stoppages, two at each segment. Overall, there are nine flow restarts (green[1] up-arrows in Fig. 7) and six flow stoppages (red down-arrows in Fig. 7) over the ten-day planning horizon. Three of the nine flow restarts taking place in segments N1–N2 and segment N2–N3 are already prescribed by the aggregate transportation plan. In addition, other three restarts occur because the pipeline is initially idle.

From the flow rate profiles shown in Fig. 7, it is clear that the optimal solution tends to keep a finite stream flow in every pipeline segment to avoid unnecessary flow restarts. A particular situation is observed in segment N3–N4 during the detailed operations k_4 and k_5. According to the aggregate plan, segment N2–N3 is idle during block *b*2. Simultaneous injections from nodes N1 and N3 push products toward depots N2 and N4, respectively. During such parallel injections, the first pipeline section (N1–N2) operates at the maximum flow rate (580 m^3/h), while the last segment (N3–N4) works at lower rates (395 and 100 m^3/h). This is so because the overall volume injected at N3 is less than the size of the batch concurrently pumped into the pipeline from node N1.

When compared to the solution provided by the heuristic procedure prioritizing the closest terminals to the active sources, the total number of pipeline segment activations is reduced by 28.57 % (from 7 to 5), the restarted volume by 15.51% (from 3004 to 2538 [10^2 m^3]), and the stopped volume by 30.51% (from 2435 to 1692 [10^2 m^3]).

More computational results are presented as Supplementary Data.

6. Conclusions

The detailed scheduling of multi-source pipeline networks can be effectively solved using the MILP optimization model described in this paper. Based on continuous time and volume scales, the new model successfully overcomes the challenge of scheduling detailed operations in more complex configurations. Transportation plans are decomposed into a sequence of actions to be taken by the pipeline operator. Two or more batch injections from different sources can take place at the same time. Through a precise coordination of product injections and deliveries, unnecessary pipeline stoppages and subsequent flow restarts are avoided. Partial deliveries permit to siphoning products out of the pipeline as the batches continue moving forward to farther destinations. These model capabilities bring about two advantages. On the one hand, the flow rate in every pipeline segment can be adjusted to comply with the specified flow rate range. On the other hand, the model yields substantial savings with regards to heuristic methods by keeping a finite flow in more pipeline segments for a longer period of time.

Compared to discrete decomposition approaches, the model size is much smaller because individual operations synthesize multiple product injections and deliveries. This fact has a major impact on the solving time. A real-world case study has been tackled using the proposed approach. It involves the transportation of five oil refined products from two source nodes to three distribution terminals. The optimal detailed schedule is found at very low CPU time. In fact, the computational effort is mostly determined by the aggregate planning stage. Based on these findings, we expect that the proposed methodology can be extended to even more complex configurations, like pipeline networks with a tree structure.

[1] For interpretation of color in Fig. 7, the reader is referred to the web version of this article.

Acknowledgments

Financial support received from CONICET under Grant PIP-2221 and from Universidad Nacional del Litoral under CAI+D 66-335 and 81-481 is fully appreciated.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.cie.2015.07.022.

References

Ball, D., Dickerson, P., & Hertel, J. (2011). Configurable tools for the pipeline scheduler. In *PSIG annual meeting*. Pipeline Simulation Interest Group. January.

Boschetto, S. N., Magatão, L., Brondani, W. M., Neves, F., Jr., Arruda, L. V. R., Barbosa-Póvoa, A. P. F. D., et al. (2010). An operational scheduling model to product distribution through a pipeline network. *Industrial and Engineering Chemistry Research, 49*, 5661–5682.

Brooke, A., Kendrick, D., Meeraus, A., & Raman, R. (2006). *GAMS – a user's guide*. Washington, DC: GAMS Development Corporation.

Cafaro, V. G., Cafaro, D. C., Méndez, C. A., & Cerdá, J. (2011). Detailed scheduling of operations in single-source refined products pipelines. *Industrial and Engineering Chemistry Research, 50*, 6240–6259.

Cafaro, V. G., Cafaro, D. C., Méndez, C. A., & Cerdá, J. (2012). Detailed scheduling of single-source pipelines with simultaneous deliveries to multiple offtake stations. *Industrial and Engineering Chemistry Research, 51*, 6145–6165.

Cafaro, V. G., Cafaro, D. C., Méndez, C. A., & Cerdá, J. (2015). MINLP model for the detailed scheduling of refined products pipelines with flow rate dependent pumping costs. *Computers and Chemical Engineering, 72*, 210–221.

Cafaro, D. C., & Cerdá, J. (2009). optimal scheduling of refined products pipelines with multiple sources. *Industrial and Engineering Chemistry Research, 48*, 6675–6689.

Cafaro, D. C., & Cerdá, J. (2010). Operational scheduling of refined products pipeline networks with simultaneous batch injections. *Computers and Chemical Engineering, 34*, 1687–1704.

Castro, P. M. (2010). Optimal scheduling of pipeline systems with a resource-task network continuous-time formulation. *Industrial and Engineering Chemistry Research, 49*, 11491–11505.

García-Sánchez, A., Arreche, L. M., & Ortega-Mier, M. (2008). Combining simulation and tabu search for oil-derivatives pipeline scheduling. *Studies in Computational Intelligence, 128*, 301–325.

Gleizes, M. F., Herrero, G., Cafaro, D. C., Méndez, C. A., & Cerdá, J. (2012). Managing distribution in refined products pipelines using discrete-event simulation. *International Journal of Information Systems and Supply Chain Management, 5*, 58–79.

Hane, C. A., & Ratliff, H. D. (1995). Sequencing inputs to multi-commodity pipelines. *Annals of Operations Research, 57*, 73–101.

Herrán, A., de la Cruz, J. M., & de Andrés, B. (2010). A mathematical model for planning transportation of multiple petroleum products in a multi-pipeline system. *Computers and Chemical Engineering, 34*, 401–413.

Herrán, A., de la Cruz, J. M., & de Andrés, B. (2012). Global search metaheuristics for planning transportation of multiple petroleum products in a multi-pipeline system. *Computers and Chemical Engineering, 37*, 248–261.

Lopes, T. M. T., Ciré, A. A., de Souza, C. C., & Moura, A. V. (2010). A hybrid model for a multiproduct pipeline planning and scheduling problem. *Constraints, 15*, 151–189.

Magatão, L., Arruda, L. V. R., & Neves, F. Jr., (2004). A mixed integer programming approach for scheduling commodities in a pipeline. *Computers and Chemical Engineering, 28*, 171–185.

Mori, F. M., Lüders, R., Arruda, L. V. R., Yamamoto, L., Bonacin, M. V., Polli, H. L., et al. (2007). Simulating the operational scheduling of a real-world pipeline network. *Computer Aided Chemical Engineering, 24*, 691–696.

Moura, A. V., de Souza, C. C., Cire, A. A., & Lopes, T. M. (2008). Planning and scheduling the operation of a very large oil pipeline network. *Lecture Notes in Computer Science, 5202*, 36–51.

Neves, F., Jr., Magatão, L., Stebel, S. L., Boschetto, S. N., Felizari, L. C., Czaikowski, D. I., et al. (2007). An efficient approach to the operational scheduling of a real-world pipeline network. *Computer Aided Chemical Engineering, 24*, 697–702.

Reddy, P. C. P., Karimi, I. A., & Srinivasan, R. (2004). A new continuous-time formulation for scheduling crude oil operations. *Chemical Engineering Science, 59*, 1325–1341.

Rejowski, R., & Pinto, J. M. (2003). Scheduling of a multiproduct pipeline system. *Computers and Chemical Engineering, 27*, 1229–1246.

Rejowski, R., & Pinto, J. M. (2008). A novel continuous time representation for the scheduling of pipeline systems with pumping yield rate constraints. *Computers and Chemical Engineering, 32*, 1042–1066.

Sasikumar, M., Prakash, P. R., Patil, S. M., & Ramani, S. (1997). PIPES: A heuristic search model for pipeline schedule generation. *Knowledge-Based Systems, 10*, 169–175.

Siswanto, N., Essam, D., & Sarker, R. (2011). Solving the ship inventory routing and scheduling problem with undedicated compartments. *Computers and Industrial Engineering, 61*(2), 289–299.

Trench, C. J. (2001). How pipelines make the oil market work – Their networks, operation and regulation. Technical report for the association of oil pipelines and the American Petroleum Institute's Pipeline Committee (December 2001).

Zyngier, D., & Kelly, J. D. (2009). Multi-product inventory logistics modeling in the process industries. *Optimization and Logistics Challenges in the Enterprise, 30*, 61–95.