An Operational Scheduling Model to Product Distribution through a Pipeline Network

ARTICLE *in* INDUSTRIAL & ENGINEERING CHEMISTRY RESEARCH · MAY 2010

Impact Factor: 2.59 · DOI: 10.1021/ie900685v

Leandro Magatão
Federal Technological University of Paraná…

Lúcia Valéria Ramos de Arruda
Federal Technological University of Paraná…

Ana Barbosa-Povoa
Technical University of Lisbon

Susana Relvas
University of Lisbon

An Operational Scheduling Model to Product Distribution through a Pipeline Network

Suelen N. Boschetto,[†] Leandro Magatão,[†] William M. Brondani,[†] Flávio Neves-Jr,*[,†] Lucia V. R. Arruda,[†] Ana P. F. D. Barbosa-Póvoa,[‡] and Susana Relvas[‡]

Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná, Av. Sete de Setembro, 3165, 80230-901 - Curitiba, PR, Brasil, and Centro de Estudos de Gestão, CEG-IST, Instituto Superior Técnico, Universidade Técnica de Lisboa, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

Pipelines have been proved to be an efficient and economic way to transport oil products. However, the determination of the scheduling of operational activities in pipeline networks is a difficult task, and efficient methods to solve such complex problem are required. In this contribution, a real-world pipeline network is studied, and an optimization model is proposed in order to address the network scheduling activities. A hierarchical approach is proposed on the basis of the integration of a mixed integer linear programming (MILP) model and a set of heuristic modules. This article exploits the MILP model, the main goal of which is to determine the exact time instants that products should be pumped into the pipelines and received in the operational areas. These time instants must satisfy the pipeline network management and operational constraints for a predefined planning period. Such operational constraints include pipeline stoppages, movement of batches through many areas/pipelines, use of preferential routes to avoid contamination losses, on-peak demand hours of pumping, local constraints, reversions of flow direction, and surge tank operations, while satisfying a series of production/consumption requirements. The developed continuous-time model is applied to a large real-world pipeline system, where more than 14 oil derivatives and ethanol are transported and distributed between supply and demand nodes.

1. Introduction

Companies and decision makers are frequently looking for a way to effectively organize their activities and to use efficiently their resources, searching for an increase in performance according to economical criteria. This is the case of the oil industry where any improvement in the execution of the associated tasks (e.g., production, storage, distributions, etc.) generates considerable profits.

The supply chain of the oil industry is one of the systems where such improvements are required and where companies are looking forward to optimize their activities and resources. Such optimization should be performed from the strategic to the operational level. Looking at the tactical/operational level the planning and scheduling activities are some of the most important that span along the chain including production, transportation, storage, among others.

The transportation of products in the oil industry, from the product origin until its final consumption point, is made through diverse forms, such as: trucks, trains, vessels (more usual for crude oil), and pipelines. Nowadays, great part of the oil derivatives transportation is carried through pipelines. This type of transport has been proved to be reliable and economic.[1] The increasing interest for the planning activities and production scheduling by the oil industry has motivated the development of tools to aid the decision making process, in special those that use optimization techniques. Beyond aiming at the use of the resources in a more efficient, lucrative, and trustable form, these problems, generally, offer great challenges considering their dimension and complexity. Although, being a fundamental activity, the scheduling of products distribution using pipelines has not yet reached a consolidated solution approach. Rejowski and Pinto[2] mentioned that this is a very complex problem that in many cases has a small number of feasible solutions. Therefore, the use of optimization tools is crucial since these can provide systematic procedures to obtain high-quality feasible solutions.

In order to obtain the scheduling of activities in a pipeline network, the analysis of a large number of variables is required. Thus, a solution that contemplates all the involved variables, determining a good policy of conduction of the pipeline network, is not a trivial task, not even for experienced specialists. Currently, in the majority of the oil companies, pipeline scheduling is made by specialists (schedulers) on basis of their experience and through the use of manual calculations. Consequently, many petroleum refiners develop in-house tools to deal with the scheduling of their operations.[3] Moreover, studies from the literature usually focus on more restrictive and smaller pipeline topologies, each with a reduced set of operational constraints that, despite making the problem more tractable, are far from representing real-world issues. This is a consequence of the large number of hard sequencing and timing constraints involved. This problem is aggravated when larger and more generic topologies are considered. To support their decision, schedulers usually seek trustable indicators that can have direct influence, for instance, in tankage policies and refinery production, or even in vessels load/discharge policies. Then, a motivation exists to develop a model that assist decisions of short-term scheduling activities through a pipeline network within the oil supply chain.

Among the works referring to pipeline scheduling using MILP, various papers can be found addressing problems of oil transference using only a single multiproduct pipeline. Several mathematical programming approaches have appeared in the

* Author to whom correspondence may be sent. E-mail: neves@utfpr.edu.br. Telephone: +55 41 33104707.
[†] CPGEI - UTFPR.
[‡] CEG - IST.

10.1021/ie900685v
Published on Web 05/11/2010

literature. Rejowski and Pinto[4] considered two MILP models to represent the problem of derivatives transference through a pipeline that establishes connection among a refinery and five depots. The models use discrete time representation and the objective is to determine the pumping operations of the new products and the load and discharge operations of the tanks in the refinery and depots, minimizing the total cost (costs of storage, pumping procedure, and interface). The considered time horizon comprised three days, and the model found feasible solutions (considering a percentage of the global optimal) for the presented computational experiments. In a posterior work,[5] these authors used the same problem to include cut constraints in their model, aiming to improve the computational efficiency of the formulation rather than using preprocessing techniques that use reduction of variable and/or constraint relaxation/elimination. The optimal solution was found for three different instances and there was effectively computational time improvement in some situations.

Magatão et al.[6] addressed a transference problem of oil derivatives between a harbor and a refinery. The pipeline can operate with reverse flow (that is, a same pipeline transfers the products from the harbor to the refinery, and from the refinery to the harbor), and it is possible the usage of *plugs* to solve interface problems that occur when two products cannot be in contact during the pipeline transportation, but their sequencing improves the schedule. Also, this approach considered seasonal costs to deal with on-peak hours of electric energy consumption. Optimal solutions were found for the proposed MILP formulation that uses uniform discrete time intervals. Moreover, Magatão et al.[7] in a posterior work developed a hybrid technique combining MILP and Constraint Logic Programming (CLP) to approach the same problem presented in 2004. This method allowed a computational time reduction for the attainment of the final solution.

An alternative approach for pipeline scheduling with continuous time representation was addressed by Cafaro and Cerdá.[8] The problem was modeled using a continuous time MILP method, and the approach considered the possibility of inserting *plugs*. The objective function minimizes the interfaces, storage and pumping costs, being able to apply penalties in case of a product being pumped in the on-peak hours of electric energy consumption (seasonal constraints).

Relvas et al.[9] dealt with a problem of oil derivatives supply to a distribution center. The pipeline scheduling was made applying a MILP approach, based on a continuous representation of time and volume. The model was solved through a strategy of temporal decomposition, dividing the scheduling horizon in two parts. The final conditions of the first period are used as initial conditions for the second period, reducing the total time of resolution. In a posterior work, Relvas et al.[10] developed a novel rescheduling method, taking into account the variability of real operational changes. The successfully solved instances covered a horizon of 1 month, including initial plans and their revisions, with more than one change.

Neves-Jr et al.[11] addressed the operational scheduling of a real-world pipeline network. The network contains 15 pipelines that provide connections between 9 areas, involving 3 refineries, 4 depots, 1 harbor, and 1 final customer. The solution approach is based on the problem division in the key three elements of scheduling: assignment of resources, sequencing of activities, and timing determination for resources utilization by these activities.[12] Scheduling results were obtained with low CPU time, considering horizons of 7 and 30 days. Moura et al.[13] presented another work that uses a decomposition approach to solve the scheduling of a pipeline network. The problem constraints are similar to the ones presented by Neves-Jr et al.[11] The network involves 4 depots, including 5 bidirectional pipelines. They proposed a hybrid solution approach based on two iterative phases. The initial one comprises a heuristic strategy and the second a Constraint Programming (CP) model. The resulting method was tested with real-world instances, yielding feasible solutions.

Within the pipeline scheduling literature, the majority of the works make use of mathematical programming, more specifically MILP, to deal with the related problems. The approaches based on this technique present, at least, a feasible solution for the problems. In some cases, "optimal" MILP solutions are found, as showed by Cafaro and Cerdá,[8] Magatão et al.,[6] and Neves-Jr et al.[11] In the majority of the cases, the success in obtaining solutions in a viable time is achieved by continuous time representations. Another important fact is that in order to compute solutions using low computational effort it is frequently used either heuristics and/or decomposition strategies. Some examples of works that adopt these approaches are, for instance, Magatão et al.,[7] Relvas et al.,[9] and Moura et al.[13] An important conclusion from the published works is that the use of hierarchical approaches or even other types of techniques (such as heuristics or metaheuristics) to overcome problem complexity is a relevant issue. With these techniques, good solutions can be found in a limited computational time.

The present work addresses the problem of developing an optimization structure to aid the operational decision-making of scheduling activities in a real-world pipeline network. The proposed method is based on a hierarchical approach to address complex problems with high computational burden.

Four main blocks form the decomposition approach: assignment, sequencing, simulation, and timing. Basically, the assignment block takes into account the production and consumption functions in order to determine volume of batches, a route for each batch, and to indicate an initial sequence of batches to be pumped. The sequencing block reorganizes the list of batches providing other sequences of pumping. Then, the simulation block makes an examination of obtained results from assignment/sequencing blocks. The main goal is to identify and to process the previously attained information, in order to establish parameters to the optimization block (MILP model). The MILP model determines the process timing, that is, the time points for sending and receiving activities, making an operational short-term scheduling. The developed structure was applied to the solution of a real-world problem that involves a multiproduct pipeline network, which connects different areas (nodes) that represent refineries, harbors, final clients, or distribution depots. Scheduling details must be given, including pumping sequence in each node, volume of batches, inventory constraints, and timing issues, while respecting a series of operational constraints. Moreover, the computational burden to determine a short-term scheduling within the considered real-world case is a challenging issue. Many insights have been derived from the obtained solutions, which are given in a reduced computational time for oil industrial-size problems.

This hierarchical approach presented here is based on the one proposed by Neves-Jr et al.[11] However, the present work proposes a novel and more detailed MILP model, which uses a simulation structure. In addition, the considered real-world case studies will be significantly expanded. Section 3 will further exploit these features.

The remainder of the paper is organized as follows. The next section describes the problem addressed in this work. First, the

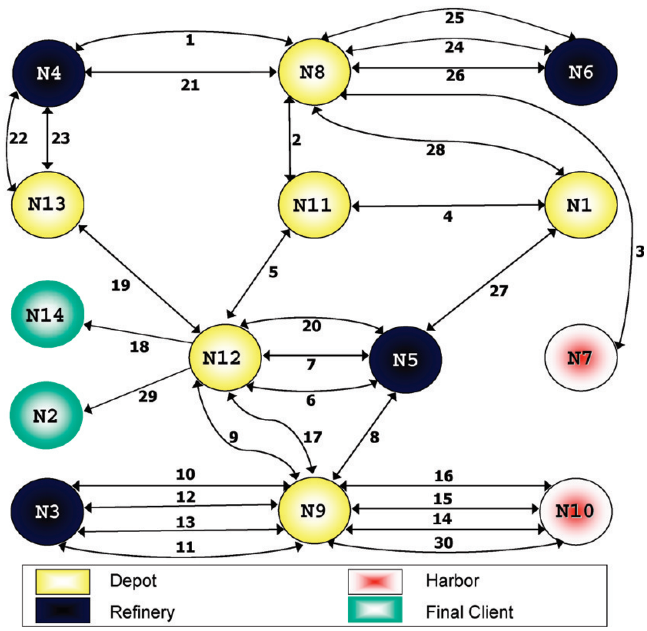


Figure 1. Pipeline network.

main problem constraints are presented according to general aspects and considerations. Subsequently, a structure to solve the problem, based on the key scheduling elements, is proposed. In section 3, the MILP model to solve the timing problem is presented, exploiting its mathematical features. In section 4, a real-world case study is examined, and results are discussed where scheduling details can be visualized. The proposed method is also compared with other method presented in the literature for another case study. Section 5 presents the final conclusions and future works.

2. Problem Description

The considered multiproduct pipeline network is schematically presented in Figure 1, which illustrates a real case of the Brazilian pipeline network. This network contains different operational areas (refineries, harbors, final clients, depots) interconnected by bidirectional pipelines that transport a set of products between adjacent areas. The nodes in Figure 1 represent the areas and the edges represent the pipelines interconnecting these areas.

The pipeline network considered in Figure 1 involves 14 areas (nodes), and it is composed of 4 refineries (nodes N3, N4, N5, and N6), 2 harbors (N7 and N10), 2 final clients (N2 and N14) and 6 depots (N1, N8, N9, N11, N12, and N13), which receive or send products. In particular, area N1 does not have tanks to store products. The areas are interconnected by 30 bidirectional pipelines, each one with a particular volume.

A product can take several hours to reach its final destination as well as each batch can remain in a pipeline until another batch pushes it. Each product can be stored in specific tanks located in specific areas. More than 14 oil derivatives and ethanol can be transported. For instance, a "typical operation" involves pumping a batch from N6 to N14, passing by N8, N11, and N12. In this case, product is sent through pipelines 25, 2, 5, and 18.

The scheduling of the operational activities within the considered multiproduct pipeline network is made based on demand and production predictions for a period of one month (planning level). For instance, during one month the product 5 (P5) demand in node N14 is expected to be more than 220 000 volumetric units. However, during the scheduling horizon, the total demanded volume has to be split into smaller volumes (batches) that must arrive during an adequate time window. Volume of batches should be compatible with storage constraints and economical factors. Batches can be supplied by different sources, for example, nodes N4 and N6. The route from an origin area to a destination area can be therefore quite long. Since the pipelines are shared resources, the pumping of a specific batch in each pipeline must respect a series of pumping operations that are in course. Such operational conditions will be described later on.

2.1. Problem Considerations and Constraints. A series of factors are considered within the problem constraints. They are summarized as follows:

1. The transfer operations should occur, preferentially, in the predetermined time horizon.
2. In each period, batches are established considering typical transportation volumes, and are sent through preferential routes for each product.
3. Each route is composed of a sequence of areas intercalated by pipelines. Note that a route can have only one pipeline and two areas or n pipelines connecting $n+1$ areas. The first and the last areas in the route represent the origin and destination area, respectively. A route indicates the "movement path" for each batch, from the origin area until the destination area. An example of a route can be extracted from Figure 1, where a transfer operation can be made from N6 to N14. Thus, the route will contain the elements: {N6, 25, N8, 2, N11, 5, N12, 18, N14}.
4. One route can be used by different batches, containing different products, at different flow rates.
5. In each area, a batch received from a pipeline can be pumped to a specific tank or directed to another pipeline.
6. Each area has a tank farm that stores different products. However, there are upper and lower limits to the overall product inventory in each node.
7. The storage level can increase or decrease, according to the balance between the received batch volume, the local consumption (consumption within the area), and the market demand. The inventory levels are also affected by local production and consumption market of areas.
8. In addition, an operation called "surge tank" can occur when a batch is received in an area at a specific flow rate and has to be sent to another area at a different flow rate. In this case, intermediate storage has to be used.
9. Each area has proper operational features. For example, there exists a limited number of pumps and manifolds for each area. Thus, only a restricted number of batches is allowed to be sent/received from/to each area at the same time. This fact characterizes "local constraints", i.e., specific for each area.
10. Batches that are in the pipeline at the beginning of the scheduling horizon must be routed to their previously determined destination (*in transit* batches).
11. The pipelines operate/remain always fulfilled, and their utilization/occupation must be managed.
12. Network pipelines can have their flow direction reverted, according to operational convenience.
13. During transportation procedures, a contamination area (interface) between products is formed. Some interfaces are operationally undesired and must be avoided.
14. The electrical energy has different costs, according to season and period of day. Typically, from 17:30 to 20:30 (on-peak demand hours) the cost is higher than any other time period.

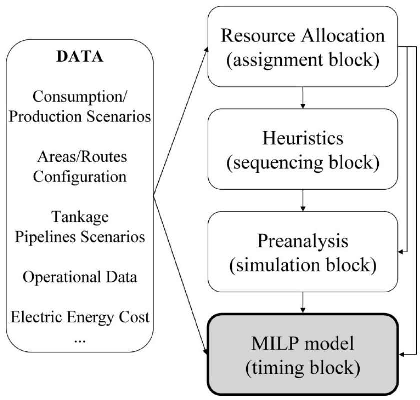


Figure 2. Hierarchical approach.

Therefore, at this interval, the pumping should be periodically interrupted in some operational areas during the scheduling horizon (e.g., N6, N7, and N8).

2.2. Problem Hierarchical Decomposition. A hierarchical approach is developed to address the scheduling problem (Figure 2). As proposed by Neves-Jr et al.,[11] the decomposition is based on the three main elements of scheduling: assignment of resources, sequencing of activities, and timing determination for resources utilization by these activities.[12] However a new block, named preanalysis, is introduced to simulate and compute parameters to the timing block.

The input data, consumption/production scenarios, areas/routes configuration, inventory management, electrical energy cost, and other operational data, characterizes a problem instance to be solved. The **assignment block** receives the data and uses a heuristic procedure to compute batch volumes and to indicate a route for each batch. The assignment model takes into account production and consumption information received from the upper level planning, inventory management information, and available tank capacity for each area. Then, based on the input data, the assignment model calculates time-windows to send/receive each batch in each origin/destination area, respectively.

The time windows define the release and deadline dates for sending and receiving operations. Figure 3 illustrates the determination of time windows for a producer (origin area) and for a consumer (destination area). Within this figure, we indicate the minimum/initial/maximum inventory levels and the storage capacity. Curves of production/consumption are also indicated in Figure 3. Such curves may vary significantly depending on the considered node/product, and are input data to the assignment block. The lower time bound to pump the batch b in the origin area (TED) indicates the releasing time of a batch b. Before this time, there is no sufficient volume of product b in the origin area to compose the minimum operational volume of b. The minimum volumes are indicated per node/product, and are input parameters based on operational practice. The upper time bound to pump the batch b in the origin area (TEC) indicates the deadline for sending a batch b. After that time point, managing tankage issues at the producer can be a difficult task. The lower time bound to receive the batch b in the destination area (TRD) indicates the minimum time at the consumption node at which there is sufficient tankage to receive b. The upper time bound to receive the batch b in the destination area (TRC) indicates the maximum due date for receiving b. After that time, minimum inventory issues are violated.

After defining the set of batches to be pumped, and the associated time windows, the assignment block indicates an initial pumping order of these batches. This order is defined per area, product, and pipeline. The batches are "ranked" on the basis of the values of the time windows of the origin area (TED_b; TEC_b) and the destination area (TRD_b; TRC_b). For instance, a batch b with small TEC_b has high priority of pumping. This procedure reflects the logical sending from a producer node. Based in a series of operational procedures, a heuristic function, e.g. eq 1, is used to determine an initial order of batches. The batches with lower Cd deserve priority of pumping. Thus, the batches are labeled in Cd crescent order, and this order must be observed during the scheduling horizon. According to eq 1, batches with small values of TED_b and TRD_b and with a small range for sending ($\approx TEC_b - TED_b$) and receiving ($\approx TRC_b - TRD_b$) time windows tend to have a high priority of pumping.

$$Cd_b = \left[TED_b + (TEC_b + 1 - TED_b)\cdot\frac{TED_b + 1}{TEC_b + 1}\right] + \left[TRD_b + (TRC_b + 1 - TRD_b)\cdot\frac{TRD_b + 1}{TRC_b + 1}\right] \quad (1)$$

The **heuristic block** can reorganize the list of batches initially sequenced by the Cd order. At this task, it uses a Genetic Algorithm to optimize the sequencing of batches to be pumped. This heuristics takes as inputs the previously calculated time windows, network start-up conditions, and inventory levels given by the assignment block. The main goal is to provide one or more sequences of ordered batches that satisfy the consumption/production requirements for each area. Further details of assignment/sequencing blocks can be attained in Yamamoto et al.[14] Thus, within the assignment/sequencing blocks a list of batches, including the origin/destination areas, route, volume,

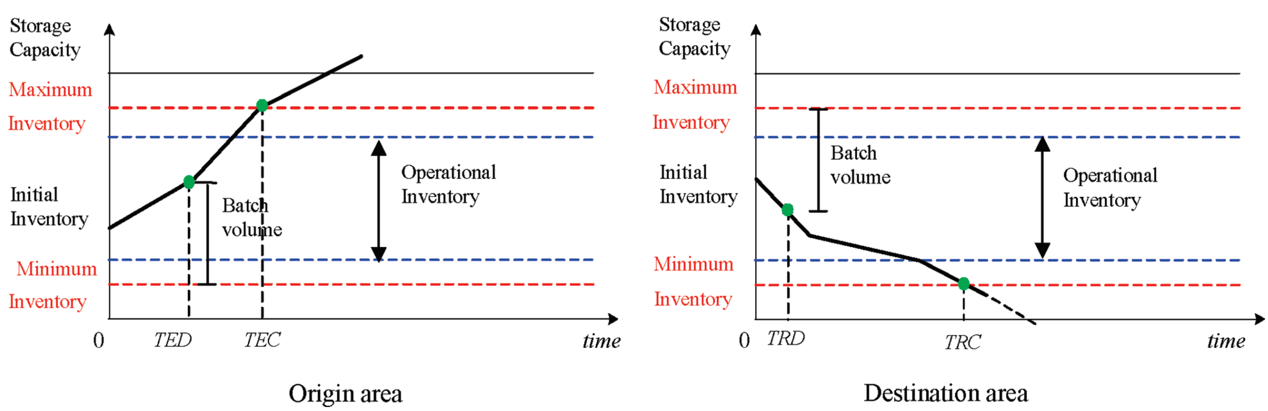


Figure 3. Time windows constraints.

Table 1. Parameters Determined by the Simulation Block

b	d	bo_b	p_b	p_{bo}	$part_{bo,p}$ (vu)	$flw_{bo,p}$ (vu/h)
14	18	40	1	1	1700	500
14	**18**	**40**	**2**	**2**	**300**	**600**
24	**5**	**40**	**1**	**2**	**300**	**600**
24	5	40	2	3	1700	500
24	5	40	3	4	350	550
24	5	41	4	1	3350	600

product, and the time windows for each batch is determined. Table 5 (in the Results section) illustrates the piece of information given by the assignment/sequencing blocks.

The **simulation block** evaluates the information derived from assignment/sequencing blocks and calculates volumetric and flow rate limits (bounds). These limits present a preliminary indication on the viability of the programming. The simulation evaluates, step by step, the batch that is being pumped and analyses the influence of this pumping operation on other batches into the pipelines. Moreover, the simulation allows for the MILP model identify conditions in which a batch can remain stopped into the pipeline and it provides indications of significant flow rate changes due to network constraints. As a result, the batch movement along the pipeline network is analyzed, and "volumetric parts" (portions of this batch) with similar pumping conditions are identified. For example, suppose a batch with 10000 volumetric units (vu) that can be "divided" by the simulation algorithm into 3 parts (sub-batches) with, for instance, 2000 vu, 5000 vu, and 3000 vu. Each part can have its specific pumping flow rate. Between sub-batches, pumping stoppages can (or cannot) occur. In this way, the simulation block provides valuable information, which is used as parameters by the MILP model. Table 1 illustrates parameters determined by the simulation block, as discussed in section 3.3.

Finally, all the parameters calculated by the previous blocks are used in a continuous time MILP model, the **timing block**. This block determines the operational short-term scheduling for the pipeline network using the parameters (e.g., time windows and sequences of batches) determined by previous blocks. This block is described in detail in the next section.

These models are integrated in a collaborative way and are sequentially executed. A fundamental characteristic of this approach is that each model can be considered as a "preprocessing routine" to the next model in a hierarchical precedence. Thus, the output of a block can be used as an input to the next block. Additionally, the computational tool incorporates functionalities that allow visualizing reports, inventory graphics (e.g., Figure 16), and Gantt charts (e.g., Figures 8−10) related to proposed solutions.

3. MILP Model—Timing Block

A continuous time mixed integer linear programming approach is used to create the optimization model. The general guidelines used to state this model are summarized as follows:

1. The overall planning of production/consumption during a month for each product in each node is established in advance by the company. However, during the scheduling horizon, the total volume for each product has to be split into smaller volumes (batches), in order to be pumped. This task is done by the assignment block.
2. Products can be pumped from different origin areas. For each area a set of pumping batches is computed, and a list of all batches must be ordered (batch sequence) for the entire network. The generated sequence of batches depends on the scheduling horizon. This task is done by the sequencing block.
3. The batch volume for each product should be an integer number of the tank capacity available in batch origin/destination area or equal to the volume of a specific pipeline within the batch route (see item 4). In case of reverse flow operation (see item 9), an additional batch with the same volume of the "pipeline segment" under reversion must be inserted by the assignment block.
4. Due to operational practices, there are favorite routes for each product for each pair origin/destination area. The routes used by each batch are selected by the assignment block.
5. The choice of the routes considers that incompatible products should be sent by different pipelines, so that undesirable contamination is avoided. In other situations, if incompatible products have to be sent by the same pipeline, *plugs* should be added between these batches. The assignment block assigns the *plug* in between the incompatible products.
6. For each batch there are specified its used route, the product, and the total pumping volume, and the calculated time windows are input data supplied by the assignment block.
7. For each product and/or route, there is a "typical flow rate value" (pumping rate) in each origin area. However, this pumping rate can be affected by the self-products to be pumped. The simulation block analyses this feature. It supplies a table with the connections between different batches, where the pumping of one batch influences the movement of another one. Also, the table provides each fraction of volume, here known as "part", with the volume, flow rate, and indices of each part. Table 1, and its explanation, further exploits the simulation functionalities.
8. A batch can be received at a specific flow rate in a tank and it can be simultaneously pumped from this tank to another pipeline at a different flow rate (surge tank operation). To consider the surge tank operation, the route of the batch is split in two routes. The first route considers the origin area until the intermediate area with the tanks used in the operation; the second one, from the area in surge tank operation until the final destination area.
9. A series of pipelines can have the flow direction reverted. Specific procedures are required to manage such operational conditions. In order to implement a "flow reversion" procedure, an entire set of batches is pumped in one direction. Afterward, it is necessary to insert an auxiliary batch with the same volume of the considered pipeline to push out (finalize pumping) of batches in this flow direction. The pumping direction can then be reverted, and a set of products can be pumped according to the new direction. The auxiliary batch that will be used to accomplish a flow reversion operation must be specified *a priori* in the sequence of batches. Auxiliary batches can be injected to push previous pumped batches out of the line. This procedure was also used by Cafaro and Cerdá.[15]
10. Each area (node) of the considered pipeline network has particular limitations that need to be addressed in order to obtain useful solutions from the model. These limitations are called "local constraints". They restrict the simultaneous pumping/receiving procedures. The limitations are different for each area. Because of this, in many situations, the areas must be considered in an independent form, and constraints should be created for each case.
11. The tanks are considered in an aggregated manner for each area and product. The upper and lower limits to the overall

product inventory in each node are managed in the MILP model through the time windows. It is considered that if the MILP model satisfies the time windows previously calculated, which define the release and the due dates for sending operations and for receiving operations, the inventory management requirements will be also satisfied.

12. Pipelines always operate in full, and some of them have a considerable volume. Thus, they can "store" different products during pumping procedures. These products can be pumped to tanks or routed to other pipelines.
13. All batches previously "stored" in the pipelines at the beginning of the scheduling horizon (*in transit* batches) are sequenced in the beginning of the list of batches created by the assignment block (start-up of network scheduling).
14. Although a new product being pumped has a particular destination, previously "stored" products should be pushed out to their original destination.
15. When a new product is pumped from an origin area, the products previously "stored" into the pipelines are pushed, in accordance with the "new flow rate".
16. The pumping should be periodically interrupted in some operational areas during the scheduling horizon due to electrical energy constraints.

The mixed integer linear programming model representation will use the notations for indices, sets, parameters, and variables as shown in Nomenclature.

As previously defined, the assignment block determines the routes that will be used by the products' transfer. However, the choice of routes is, in practice, limited to a small number of possibilities since, generally, there exist just one or two pipelines (or possible routes) connecting two areas that can transport a specific product. In addition, the sequencing block tries to manage interface losses and forbidden sequences of products are also avoided. It is important to emphasize that the MILP formulation assumes that both (*i*) a detailed sequences of batch injections at input stations and (*ii*) associated batch movements and batch deliveries to receiving terminals have been heuristically generated and are model data.

Furthermore, the optimization model must satisfy a series of constraints. Some of them describe inventory management issues (e.g., the time windows violation: nonzero values are attributed to variables: ao_{b,no_b}, do_{b,no_b}, ad_{b,nd_b}, and dd_{b,nd_b}, indicating that the suggested windows were violated). Timing constraints are also added, and they relate timing variables (*e.g.* $fb_{b,\bar{n},n,d,p}$ and $ib_{b,\bar{n},n,d,p}$), calculating the time at which each part of a batch will be pushed into the pipelines.

The constraints are constructed in the optimization model considering a one-to-one correspondence between the specific part of batch (bo_b) that is being pumped at its origin area with another part of batch (b) that is moved by this pumping procedure. This one-to-one correspondence is provided by the results obtained through the simulation block, which are exemplified on Table 1. Finally, other constraints are able to identify which batch will increase the total pumping time because of on-peak demand hours. Time windows can be violated, but a penalty is added in the objective function in order to minimize these violations.

3.1. Defining Index Sets for Variables. The indices of the variables are constructed as part of sets that contain only specific combinations of each tuple. For example, for variables *ib*, *fb*, *ir*, and *fr*, the tuple $\langle b, \bar{n}, n1, d1, p\rangle$ contains only the areas and pipelines that will be used by batch b. If a batch b is sent by route N4 → 1 → N8 → 2 → N11, only the following indices are generated: $\langle b,$ N4, N8, 1, $p\rangle$ $\forall p \in PB_{b,\mathrm{N4},\mathrm{N8},1}$, and $\langle b,$ N8, N11, 2, $p\rangle$ $\forall p \in PB_{b,\mathrm{N8},\mathrm{N11},2}$. The same idea is used to create the indices of all batches in the model.

3.2. Calculating the Pumping Time. The parameter $tb_{b,no_b,n,d,p}$, that represents the pumping time, is obtained by the division of volume and flow rate, both estimated by the simulation block. In order to determine the pumping duration ($tb_{b,no_b,n,d,p}$) of part p of the batch b, in an origin area no_b, eq 2 is considered.

$$tb_{b,no_b,n,d,p} = part_{b,p}/flw_{b,p} \quad \forall b \in B, \{no_b, n\} \in N, d \in D, p \in PB_{b,no_b,n,d} \tag{2}$$

In this equation, $flw_{b,p}$ is the constant flow rate, of a specific part p of a batch b, and $part_{b,p}$ is the volume of the part p of the batch b. The flow rate value was determined by the simulation, considering the minimum between the flow rate indicated by the assignment block and the one supported by the pipelines that will contain products moved by the pumping of the batch bo_b. The flow rate used for the receiving procedure of the batch b is the flow rate $flw_{b,p}$ of the batch bo_b, which is being pumped at the origin area.

3.3. Problem Constraints. Once the main indices, sets, parameters, and variables are defined, the MILP model is built considering the problem constraints and objective function. Different types of constraints are defined, describing the problem characteristics.

3.3.1. Sequencing Constraints. Equations 3 and 4 determine the temporal precedence between parts p of the same batch b. This temporal precedence is defined for the sending (eq 3) and the receiving (eq 4) variables.

$$ib_{b,\bar{n},n,d,p} \geq fb_{b,\bar{n},n,d,p-1} \quad \forall b \in B, \{\bar{n}, n\} \in N, d \in D, p \in PB_{b,\bar{n},n,d} | p > pb^{first}_{b,\bar{n},n,d} \tag{3}$$

$$ir_{b,\bar{n},n,d,p} \geq fr_{b,\bar{n},n,d,p-1} \quad \forall b \in B, \{\bar{n}, n\} \in N, d \in D, p \in PR_{b,\bar{n},n,d} | p > pr^{first}_{b,\bar{n},n,d} \tag{4}$$

Equation 5 identifies the instant when the pumping of all sequenced batches in their origin areas is finished, including origin areas that stop during the on-peak hours. High electrical costs are often adopted by electrical companies in on-peak demand periods of the day. Moreover, this constraint considers the time interval α that the batch remains stopped in its origin area. This stoppage was considered mandatory for this model. This assumption was made on the basis of the operational practice of specialists. The stoppage duration is added to the pumping time, into the temporal block ($fb - ib$).

$$fb_{b,no_b,n,d,p} \geq ib_{b,no_b,n,d,p} + tb_{b,no_b,n,d,p} + \alpha\cdot \sum_{h\in HP} z_{b,p,no_b,h} \quad \forall b \in B, \{no_b, n\} \in N, d \in D, p \in PB_{b,no_b,n,d} \tag{5}$$

For a better understanding of the link among the parameters used by the constraints of the model, an example is presented based on Table 1. Table 1 shows the parameters determined by the simulation block. Analyzing the data, it is possible to notice that the pumping procedure of batch 40 (bo_b) was broken into four parts ($p_{bo_b} = 1, 2, 3, 4$). Each part of batch 40 influences the receiving (or moving) of one or more batches into the pipeline network. For example, as highlighted in Table 1 and illustrated in Figure 4, the second part ($p_{bo_b} = 2$) pumped of the batch 40 with 300 vu ($part_{bo_b,p_{bo}}$) causes: (i) the receiving of the same volume of the part 2 (p_b) of the batch 14 (b) in the

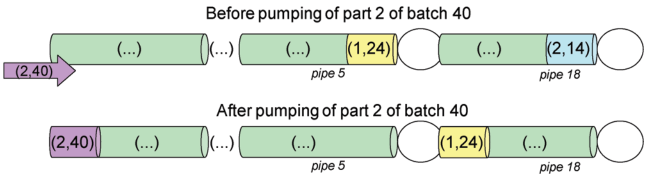


Figure 4. Example of movements by pumping of batch 40.

end of pipeline 18 (d), on its destination area; (ii) the moving of the part 1 (p_b) of the batch 24 (b) (with the same volume) from pipeline 5 (d) to another pipeline (in this case, pipe 18). Thus, when it is said that a pumping of a part influences the receiving of another part with the same volume, it is implicit that the pipeline volume balance is being respected. The parameter $tb_{bo,no,n,d,p}$ is calculated through expression 2 by the division of $part_{bo_b,p_{bo}}$ by $flw_{bo_b,p_{bo}}$.

Through the example illustrated on Figure 4, we can build up eq 6 that indicates the sending time on intermediate areas. The sending end time (fb) of a batch b is calculated considering the sending start time (ib) in the same pipeline, added to the pumping time of another batch (bo_b) that has influenced the movement of the batch b (e.g., Table 1). Thus, we have to consider the time interval during which the part p of batch b is inserted into another pipe $d1$ (an intermediate pipe in the route assigned to batch b) and the pumping time of part p_{bo} of batch bo_b, causing the movement of part (p,b), injected at its origin area no_{bo} into pipe d. This pipe d is the first one in the route assigned to batch bo_b. On the other hand, $d1$ is an intermediate pipe in the route assigned to batch b connecting nodes $\bar{n}$ and $n1$. The batch bo_b is pumped in its origin area no_{bo_b} and can remain stopped because of on-peak demand hours.

$$fb_{b,\bar{n},n1,d1,p_b} \geq ib_{b,\bar{n},n1,d1,p_b} + tb_{bo_b,no_{bo_b},n,d,p_{bo_b}} + \alpha \cdot \sum_{h \in HP} z_{bo_b,p_{bo_b},no_{bo_b},h}$$
$$\forall \{b, bo_b\} \in B, \{no_{bo_b}, n, \bar{n}, n1\} \in N, \{d, d1\} \in D, p_b \in PB_{b,\bar{n},n1,d1}, p_{bo_b} \in PB_{bo_b,no_{bo_b},n,d} \quad (6)$$

The one-to-one correspondence obtained through Table 1 is also used to create constraints 7 and 8. The start of receiving time of part p_b of batch b in area $n1$ (eq 7) is obtained using the start of pumping time of another part p_{bo_b} of batch bo_b in its origin area no_{bo_b} (e.g., Table 1).

$$ir_{b,\bar{n},n1,d1,p_b} = ib_{bo_b,no_{bo_b},n,d,p_{bo_b}}$$
$$\forall \{b, bo_b\} \in B, \{no_{bo_b}, n, \bar{n}, n1\} \in N, \{d, d1\} \in D, p_b \in PR_{b,\bar{n},n1,d1}, p_{bo_b} \in PB_{bo_b,no_{bo_b},n,d} \quad (7)$$

Equation 8 indicates the receiving end time of a batch b on area $n1$. The end of receiving will occur after the receiving start, adding the pumping time of bo_b in its origin area no_{bo_b}. It is also considered that pumping stoppages of the batch bo_b, due to on-peak demand hours, influence the receiving of batch b at area $n1$.

Table 2. Possibilities of Movements in a Pipeline d between $\bar{n}$ and n

b	$\bar{n}$	n	d
1	N7	N8	3
2	N7	N7	3
3	N8	N7	3
4	N8	N8	3

$$fr_{b,\bar{n},n1,d1,p_b} \geq ir_{b,\bar{n},n1,d1,p_b} + tb_{bo_b,no_{bo_b},n,d,p_{bo_b}} + \alpha \cdot \sum_{h \in HP} z_{bo_b,p_{bo_b},no_{bo_b},h}$$
$$\forall \{b, bo_b\} \in B, \{no_{bo_b}, n, \bar{n}, n1\} \in N, \{d, d1\} \in D, p_b \in PR_{b,\bar{n},n1,d1}, p_{bo_b} \in PB_{bo_b,no_{bo_b},n,d} \quad (8)$$

The batch sequence in all areas is already established before the sending procedure and it has to be respected. That is, the pumping, sending and receiving, not only in origin areas, but also in intermediate areas, should occur in sequential order, considering batches that share the same pipeline. The batch sequence was previously determined by the assignment block, or heuristic block. Equation 9 enables this temporal precedence among different batches. In this case, a batch $\bar{b}$ just starts the sending procedure in a pipeline d after the batch b had finished its sending procedure in the same pipeline. The initial ($\bar{n}$,$\bar{n}1$) and final (n, $n1$) areas to each batch (b, $\bar{b}$) can be different or not. Both conditions should be considered ($\bar{n} \neq n$ or $\bar{n} = n$) because of the possibility of reverting the flow direction of pipeline d (e.g., Table 2). Figure 5 also illustrates a flow reversion procedure.

$$ib_{\bar{b},\bar{n}1,n1,d,p_{\bar{b}}} \geq fb_{b,\bar{n},n,d,p_b} \quad \forall \{b, \bar{b}\} \in B, | b < \bar{b}, \{n, \bar{n}, n1, \bar{n}1\} \in N, d \in D,$$
$$p_b \in PB_{b,\bar{n},n,d} | p_b = pb^{last}_{b,\bar{n},n,d} \wedge p_{\bar{b}} \in PB_{\bar{b},\bar{n}1,n1,d} | p_{\bar{b}} = pb^{first}_{\bar{b},\bar{n}1,n1,d} \quad (9)$$

The previously established sequence must be also respected in the batch receiving (eq 10). Thus, a batch starts its receiving in the end of a pipeline always after the previous batch being totally received in the destination area. The start and final area to each batch can be different or not. When the pipeline flow reverted direction is considered (e.g., Table 2) then $\bar{n} = n$ (b is the auxiliary batch of reversion) or $\bar{n}1 = \bar{n}$ ($\bar{b}$ is the auxiliary batch of reversion).

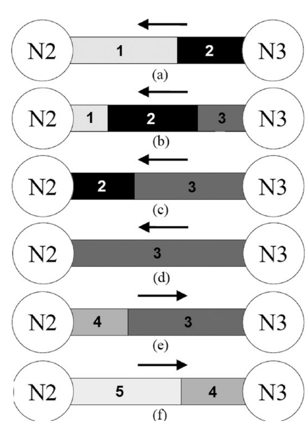


Figure 5. Reverted flow direction example.

$$ir_{\bar{b},\bar{n}1,n1,d,p_{\bar{b}}} \geq fr_{b,\bar{n},n,d,p_b} \quad \forall\{b,\bar{b}\} \in B, | b < \bar{b}, \{n,\bar{n},n1,\bar{n}1\} \in N, d \in D, p_b \in PR_{b,\bar{n},n,d} | p_b = pr^{last}_{b,\bar{n},n,d} \wedge p_{\bar{b}} \in PR_{\bar{b},\bar{n}1,n1,d} | p_{\bar{b}} = pr^{first}_{\bar{b},\bar{n}1,n1,d} \tag{10}$$

Consider three areas and two pipelines distributed in the route of batch b, organized as follows: $\bar{n} \to d \to n \to d1 \to n1$. The sending start ($ib$) of a batch b in the area n must be equal to the receiving start instant of the same batch in the area n. In this case, the tanks of area n are not used. Equation 11 specifies this condition.

$$ib_{b,n,n1,d1,p} = ir_{b,\bar{n},n,d,p} \quad \forall b \in B, \{\bar{n},n,n1\} \in N, | \bar{n} \neq n1, \{d,d1\} \in D, p \in PB_{b,n,n1,d1} \cap PR_{b,\bar{n},n,d} \tag{11}$$

Similarly to eq 11, constraint 12 identifies the final sending of a batch in an intermediate area n, which should occur after, or the same time as, the receiving end time in area n.

$$fb_{b,n,n1,d1,p} \geq fr_{b,\bar{n},n,d,p} \quad \forall b \in B, \{\bar{n},n,n1\} \in N, | \bar{n} \neq n1, \{d,d1\} \in D, p \in PB_{b,n,n1,d1} \cap PR_{b,\bar{n},n,d} \tag{12}$$

3.3.2. Reverse Flow Direction Constraints. Equations 9 and 10 already took into account the reversed flow possibility in a pipeline. However, an additional constraint is necessary to relate the receiving and the sending of an additional batch. This constraint is specifically created to contemplate reverted direction in pipelines. The additional batch has the origin area equal to the destination area ($\bar{n} = n$), and has the same volume of the reverted pipeline. Figure 5 shows a reverted flow example. Figure 5a presents batches 1 and 2 into the pipeline at the initial time. Parts b and c of Figure 5 show the pumping procedure of the additional batch 3 moving the batches 1 and 2 on direction to area N2. Figure 5d presents the occupation of the entire pipeline by batch 3, before the reverse flow starts. Finally, parts e and f of Figure 5 illustrate the inverse flow direction operation and the movement of batches 4 and 5 to destination area N3.

Equation 13 identifies the additional batch used for flow reversion purposes and guarantees that the receiving of a batch b in its destination, or origin ($\bar{n} = n$), is performed after the pumping end of the batch in the same area.

$$ir_{b,\bar{n},n,d,p} \geq fb_{b,\bar{n},n,d,p} \quad \forall b \in B, \{\bar{n},n\} \in N, | \bar{n} = n, d \in D, p \in PB_{b,\bar{n},n,d} \cap PR_{b,\bar{n},n,d} \tag{13}$$

3.3.3. Surge Tank Constraints. In some areas, a batch can be received at a specific flow rate in a tank and simultaneously pumped from this tank to another pipeline at a different flow rate. This procedure is characterized as a surge tank operation, a tank that simultaneously receives and sends product. To solve the surge tank operations, the route of a batch is split. Figure 6 shows an example of this operation. The "first batch" has a route including the origin area (A) until the area in which a set of tanks can be used (area B). The "second batch" has a route beginning in area (B), until the final destination (area C). In the case illustrated in Figure 6, the flow rate increases from B to C (from 200 vu/h to 400 vu/h). However, surge tank operations can also involve cases where the flow rate decreases.

The surge tank operations have implications within the mathematical formulation, since a route needs to be decomposed to contemplate this operation. Additional constraints had to be created in order to make a connection of these "partial-routes". Following this reasoning, the receiving time in an area with a surge tank operation and the pumping to the final destination must be made coherently. In the set of batches B, the two batches will remain together (in sequence). Thus, it is possible to make a connection between these batches ($\bar{b}$ and b) analyzing the sequence ($b = \bar{b} + 1$), the product ($prod_b = prod_{\bar{b}}$), and the area of the route ($no_b = nd_{\bar{b}}$), besides product and area that belong to the PM matrix.

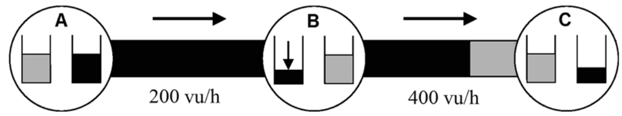


Figure 6. Surge tank operation.

Equation 14 establishes a relation between the receiving start of a batch with the pumping start in the area with surge tank operation. The pumping start of b cannot be made before inequality

$$\sum_{k=pb^{first}_{\bar{b},no_b,n,d}}^{p_{\bar{b}}} part_{\bar{b},k} \geq part_{p_b}$$

is verified. This fact occurs because the volume of each part is different and the pumping procedure can only be made in case of having received sufficient volume.

$$ib_{b,no_b,n,d,p_b} \geq ir_{\bar{b},\bar{n},nd_{\bar{b}},d1,p_{\bar{b}}} \quad \forall\{b,\bar{b}\} \in B, | b = \bar{b} + 1, \{no_b, n, \bar{n}, nd_{\bar{b}}\} \in N | no_b = nd_{\bar{b}} = PM_{i,1}, \{d, d1\} \in D, prod_b = prod_{\bar{b}} = PM_{i,2}, vol_b = vol_{\bar{b}}, \sum_{k=pb^{first}_{\bar{b},no_b,n,d}}^{p_{\bar{b}}} part_{\bar{b},k} \geq part_{p_b}, p_{\bar{b}} \in PR_{\bar{b},\bar{n},nd_{\bar{b}},d1}, p_b \in PB_{b,no_b,n,d}, i \in \{1, ..., npm\} \tag{14}$$

The end of receiving and pumping in the area with surge tank operation is established by inequality 15.

$$fb_{b,no_b,n,d,p_b} \geq fr_{\bar{b},\bar{n},nd_{\bar{b}},d1,p_{\bar{b}}} \quad \forall\{b,\bar{b}\} \in B, | b = \bar{b} + 1, \{no_b, n, \bar{n}, nd_{\bar{b}}\} \in N | no_b = nd_{\bar{b}} = PM_{i,1}, \{d, d1\} \in D, prod_b = prod_{\bar{b}} = PM_{i,2}, vol_b = vol_{\bar{b}}, \sum_{k=pb^{first}_{\bar{b},no_b,n,d}}^{p_{\bar{b}}} part_{\bar{b},k} \geq part_{p_b}, p_{\bar{b}} \in PR_{\bar{b},\bar{n},nd_{\bar{b}},d1}, p_b \in PB_{b,no_b,n,d}, i \in \{1, ..., npm\} \tag{15}$$

3.3.4. Seasonal Cost Constraints. The application of seasonal costs is analyzed within the areas where the product is pumped, that is, in the origin areas. The pumping in on-peak demand hours is not desirable because of high electrical costs during on-peak demand periods of the day. Thus, such pumping is likely to be interrupted in these moments. This stoppage was considered mandatory for this model based on the operational practice of specialists. As a consequence, other batches already inside of the pipelines can suffer quality degradation during pumping stoppages, due to interface propagation.

In order to address this operational condition, binary variables identify when on-peak demand hours occur in any area that stops pumping at on-peak demand hours ($N_{hor} \subset N$). Then, it is possible to identify the batches that will suffer application of on-peak costs by means of the relation between a batch b with the batch that is being pumped in the origin area bo_b. Equations

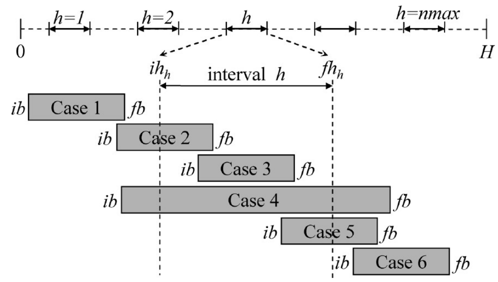


Figure 7. Cases to be analyzed for on-peak demand periods.

5, 6, and 8 increase the pumping time of the batches that are moved into these on-peak demand hour instants.

Through constraints 16 and 17, the binary variable $x_{b,p,no_b,h}$ is set to one if the pumping start of a batch b occurs before the start of an on-peak demand hour.

$$ih_h - ib_{b,no_b,n,d,p_b} \geq -M\cdot(1 - x_{b,p_b,no_b,h}) \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{16}$$

$$ih_h - ib_{b,no_b,n,d,p_b} \leq ((M + eps)\cdot x_{b,p_b,no_b,h}) - eps \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{17}$$

Constraints 18 and 19 indicate that $y_{b,p,no_b,h}$ is set to one if the pumping end of a batch b occurs after the start of an on-peak demand hour.

$$fb_{b,no_b,n,d,p_b} - ih_h \geq -M\cdot(1 - y_{b,p_b,no_b,h}) \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{18}$$

$$fb_{b,no_b,n,d,p_b} - ih_h \leq ((M + eps)\cdot y_{b,p_b,no_b,h}) - eps \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{19}$$

Constraints 20 and 21 determine if $w_{b,p,no_b,h}$ is set to one whenever the pumping start of a batch b occurs before the end of an on-peak demand hour.

$$fh_h - ib_{b,no_b,n,d,p_b} \geq -M\cdot(1 - w_{b,p_b,no_b,h}) \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{20}$$

$$fh_h - ib_{b,no_b,n,d,p_b} \leq ((M + eps)\cdot w_{b,p_b,no_b,h}) - eps \quad \forall b \in B, n \in N, no_b \in N_{hor}, d \in D, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{21}$$

In order to determine whether a pumping procedure should be stopped or not (variable $z_{b,p,no_b,h}$ is used to indicate this condition), six cases of possible positions of a temporal block ($fb - ib$) related to an interval of high electrical cost (h) must be analyzed, according to Figure 7 and Table 3.

Table 3 shows all cases of on-peak demand hours. The value of the binary variable $z_{b,p,no_b,h}$ is decided according to the value of other three variables: $x_{b,p,no_b,h}$, $y_{b,p,no_b,h}$, and $w_{b,p,no_b,h}$. If $x_{b,p,no_b,h} = 1$, the pumping start occurs before the start of an on-peak demand hour; $y_{b,p,no_b,h} = 1$ indicates that the pumping end occurs after the start of an on-peak demand hour; and, $w_{b,p,no_b,h} = 1$ if the pumping start occurs after the end of an on-peak demand hour. Thus, $z_{b,p,no_b,h}$ will indicate if the temporal block ($fb - ib$) of the batch will increase α hours. In particular, for the considered application problem, the condition $x_{b,p,no_b,h} \neq w_{b,p,no_b,h}$ is not accepted (please refer to constraint 25 herein defined). In this case, the pumping procedure is forced to be made after the end of the on-peak demand hour, as in case 6.

Table 3. Binary Variables for On-Peak Demand Periods

	$x_{b,p,no_b,h}$	$y_{b,p,no_b,h}$	$w_{b,p,no_b,h}$	$z_{b,p,no_b,h}$
Case 1	1	0	1	0
Case 2	1	1	1	1
Case 3	0	1	1	([a])
Case 4	1	1	1	1
Case 5	0	1	1	([a])
Case 6	0	1	0	0

[a] Cases not accepted.

Expressions 22−25 establish the relation of the binary variables $x_{b,p,no_b,h}$, $y_{b,p,no_b,h}$, and $w_{b,p,no_b,h}$, determining the value for variable $z_{b,p,no_b,h}$ for each case.

$$z_{b,p_b,no_b,h} \leq x_{b,p_b,no_b,h} \quad \forall b \in B, no_b \in N_{hor}, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{22}$$

$$z_{b,p_b,no_b,h} \leq y_{b,p_b,no_b,h} \quad \forall b \in B, no_b \in N_{hor}, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{23}$$

$$z_{b,p_b,no_b,h} \geq x_{b,p_b,no_b,h} + y_{b,p_b,no_b,h} - 1 \quad \forall b \in B, no_b \in N_{hor}, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{24}$$

$$x_{b,p_b,no_b,h} = w_{b,p_b,no_b,h} \quad \forall b \in B, no_b \in N_{hor}, h \in HP, p_b \in PB_{b,no_b,n,d} \tag{25}$$

3.3.5. Constraints on Time Windows. The lower and upper time bounds to send (TED_b and TEC_b, respectively) or receive (TRD_b and TRC_b, respectively) a batch b indicate inventory bounds to be observed. These limits are previously estimated by the assignment block.[14] Within the MILP model, violations of these values are allowed and will be minimized through the objective function. These violations can be used as feedback to analyze the proposed scheduling, thus providing improvements.

Considering a production area (e.g., nodes N4, N6), the lower (TED_b) and upper (TEC_b) time bounds to send a batch b should be respected. However, as above referred, such restrictions can be violated, and a penalization is included in the objective function. The variables ao_{b,no_b} and do_{b,no_b} will provide the difference between the original planned time windows and the possible violation on sending operations (constraints 26 and 27). Basically, ao_{b,no_b} indicates earliness, and do_{b,no_b}, the delays.

$$ib_{b,no_b,n,d,p} \geq TED_b - ao_{b,no_b} \quad \forall b \in B, \{no_b, n\} \in N, d \in D, p \in PB_{b,no_b,n,d} | p = pb^{first}_{b,no_b,n,d} \tag{26}$$

$$ib_{b,no_b,n,d,p} \leq TEC_b + do_{b,no_b} \quad \forall b \in B, \{no_b, n\} \in N, d \in D, p \in PB_{b,no_b,n,d} | p = pb^{first}_{b,no_b,n,d} \tag{27}$$

A similar condition exists in a destination area. The lower (TRD_b) and upper (TRC_b) time bounds to receive a batch b should be respected, but again they can also be violated. The variables ad_{b,nd_b} and dd_{b,nd_b} give the difference between the original planned destination time windows and the possible violation on receiving (constraints 28 and 29).

$$ir_{b,\bar{n},nd_b,d,p} \geq TRD_b - ad_{b,nd_b} \quad \forall b \in B, \{\bar{n}, nd_b\} \in N, d \in D, p \in PR_{b,\bar{n},nd_b,d} | p = pr^{first}_{b,\bar{n},nd_b,d} \tag{28}$$

$$ir_{b,\bar{n},nd_b,d,p} \leq TRC_b + dd_{b,nd_b} \forall b \in B, \{\bar{n}, nd_b\} \in N, d \in D, p \in PR_{b,\bar{n},nd_b,d} | p = pr^{first}_{b,\bar{n},nd_b,d} \quad (29)$$

3.4. The Objective Function. The scheduling horizon to complete sending and receiving operations has not been previously defined. In this way, any of the constraints imposes a maximum time for all the pumping operations. However, the objective function (expression 30) tries to minimize the scheduling horizon. In addition, it also minimizes violations on time windows.

$$\begin{aligned}&\text{minimize}\\ &\sum_{b\in B}\sum_{\bar{n}\in N}\sum_{n\in N}\sum_{d\in D}\sum_{p\in PB_{b,\bar{n},n,d}}\left(ib_{b,\bar{n},n,d,p} + fb_{b,\bar{n},n,d,p}\right) + \\ &\quad\sum_{b\in B}\sum_{\bar{n}\in N}\sum_{n\in N}\sum_{d\in D}\sum_{p\in PR_{b,\bar{n},n,d}}\left(ir_{b,\bar{n},n,d,p} + fr_{b,\bar{n},n,d,p}\right) + \\ &\sum_{b\in B}\sum_{no_b\in N}\left(ao_{b,no_b} + do_{b,no_b}\right)\cdot k + \\ &\quad\sum_{b\in B}\sum_{nd_b\in N}\left(ad_{b,nd_b} + dd_{b,nd_b}\right)\cdot k\end{aligned} \quad (30)$$

Constraints 3−29 optimized under the objective function 30 will return the multiproduct scheduling of batches within a general pipeline network. Other local constraints can be included in the model so as to deal with specific operational practices in each area and/or pipes that may characterize different networks. In section 4, this model is tested in real-world case studies involving, for instance, 323 batches, which are expected to be moved within a scheduling horizon of one month.

4. Case Studies

The network studied is shown in Figure 1. As described in section 2 the considered pipeline network involves 14 areas (nodes) and comprises 4 refineries (nodes N3, N4, N5, and N6), 2 harbors (N7 and N10), 2 final clients (N2 and N14), and 6 depots (N1, N8, N9, N11, N12, and N13), which receive or send products through 30 bidirectional pipelines. For more details please see section 2.

Due to the characteristics of the case studies, additional constraints had to be created in the generic model presented in section 3 in order to account for local restrictions of some areas. The next subsection presents these local constraints. The model is then solved for the case studies presented, and in subsection 4.2 the obtained results are shown. Also, a comparison between a real case study and the solution of the same case through a simplified MILP model presented by Neves-Jr et al.[11] is performed.

4.1. Local Constraints. Each area (node) of the considered pipeline network has particular limitations that need to be addressed in order to obtain functional solutions from the model to a real-world case study. These limitations, here identified as local constraints, restrict the simultaneous pumping/receiving procedures. The limitations are different for each area. Because of this, in many situations the areas must be considered as independent forms, and a new constraint should be created for each case.

These constraints can be used to particularize the network and, in our case, are used together with other constraints previously suggested in section 3 for a general formulation for pipelines networks. Thus, a unique model including general and local constraints is generated and run.

Constraints 31 and 32 do not allow simultaneous pumping to adjacent pipelines of area N8, except pipeline 26. In this case, if another product is passing through N8, it is possible to simultaneously start the pumping in N8. To consider this case, the constraints compare all parts of two batches b and $\bar{b}$ ($\bar{b} > b$). The model decides the better case to activate one or the other constraint through the binary variable $lp_{b,\bar{b},pb,p_{\bar{b}}}$. If $lp_{b,\bar{b},pb,p_{\bar{b}}} = 1$, part $p_{\bar{b}}$ of batch $\bar{b}$ starts to be pumped (ib) in the area no_b after the pumping end (fb) of part p_b of batch b (eq 30); otherwise, if $lp_{b,\bar{b},pb,p_{\bar{b}}} = 0$, part p_b of b starts to be pumped (ib) in the area no_b after the pumping end (fb) of the part $p_{\bar{b}}$ of batch $\bar{b}$ (constraint 31).

$$\begin{aligned}&fb_{b,no_b,n,d,p_b} \leq ib_{\bar{b},no_b,n1,d1,p_{\bar{b}}} + M\cdot\left(1 - lp_{b,\bar{b},p_b,p_{\bar{b}}}\right)\\ &\forall\{b,\bar{b}\} \in B | b < \bar{b}, \{no_b, n, n1\} \in N | no_b = 8, \{d, d1\} \in D | d \neq 26 \wedge d1 \neq 26,\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (31)$$

$$\begin{aligned}&fb_{\bar{b},no_b,n1,d1,p_{\bar{b}}} \leq ib_{b,no_b,n,d,p_b} + M\cdot lp_{b,\bar{b},p_b,p_{\bar{b}}}\\ &\forall\{b,\bar{b}\} \in B | b < \bar{b}, \{no_b, n, n1\} \in N | no_b = 8, \{d, d1\} \in D | d \neq 26 \wedge d1 \neq 26,\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (32)$$

Constraints 33 and 34 limit the pumping procedures in areas N11, N12, and N13. No simultaneous pumping is permitted to adjacent pipelines in any one of these areas. In this way, the start of pumping of a part $p_{\bar{b}}$ of batch $\bar{b}$ should wait for the pumping end of part p_b of batch b, in the same area.

$$\begin{aligned}&fb_{b,no_b,n,d,p_b} \leq ib_{\bar{b},no_b,n1,d1,p_{\bar{b}}} + M\cdot\left(1 - lp_{b,\bar{b},p_b,p_{\bar{b}}}\right)\\ &\forall\{b,\bar{b}\} \in B, | b < \bar{b}, \{no_b, n, n1\} \in N | no_b \in \{11, 12, 13\}, \{d, d1\} \in D,\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (33)$$

$$\begin{aligned}&fb_{\bar{b},no_b,n1,d1,p_{\bar{b}}} \leq ib_{b,no_b,n,d,p_b} + M\cdot lp_{b,\bar{b},p_b,p_{\bar{b}}}\\ &\forall\{b,\bar{b}\} \in B, | b < \bar{b}, \{no_b, n, n1\} \in N | no_b \in \{11, 12, 13\}, \{d, d1\} \in D,\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (34)$$

Inequalities 35 and 36 prohibit the simultaneous pumping procedure of products 5 and 8 from area N3. Then, if one of these products is being pumped, the other one should respect the pumping end of the first to initialize its pumping from N3.

$$\begin{aligned}&fb_{b,no_b,n,d,p_b} \leq ib_{\bar{b},no_b,n1,d1,p_{\bar{b}}} + M\cdot\left(1 - lp_{b,\bar{b},p_b,p_{\bar{b}}}\right)\\ &\forall\{b,\bar{b}\} \in B, | b < \bar{b}, \{no_b, n, n1\} \in N | no_b = 3,\\ &\{d, d1\} \in D, prod_b \in \{5, 8\}, prod_{\bar{b}} \in \{5, 8\}\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (35)$$

$$\begin{aligned}&fb_{\bar{b},no_b,n1,d1,p_{\bar{b}}} \leq ib_{b,no_b,n,d,p_b} + M\cdot lp_{b,\bar{b},p_b,p_{\bar{b}}}\\ &\forall\{b,\bar{b}\} \in B, | b < \bar{b}, \{no_b, n, n1\} \in N | no_b = 3,\\ &\{d, d1\} \in D, prod_b \in \{5, 8\}, prod_{\bar{b}} \in \{5, 8\}\\ &p_{\bar{b}} \in PB_{\bar{b},no_b,n1,d1}, p_b \in PB_{b,no_b,n,d}\end{aligned} \quad (36)$$

A similar consideration can be made for receiving purposes. Two simultaneous receiving procedures can occur in area N11, if the products are different (constraints 37 and 38). For equal products, the receiving cannot occur in simultaneous form. In this case, the product of batch b is compared with the product of batch $\bar{b}$. If $prod_b = prod_{\bar{b}}$, only one receiving procedure is allowed.

$$fr_{b,n,nd_b,d,p_b} \le ir_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} + M\cdot(1 - lr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 11, \{d, d1\} \in D, prod_b = prod_{\bar{b}}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (37)$$

$$fr_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} \le ir_{b,n,nd_b,d,p_b} + M\cdot lr_{b,\bar{b},p_b,p_{\bar{b}}}$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 11, \{d, d1\} \in D, prod_b = prod_{\bar{b}}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (38)$$

In node N12 there is one restriction for receiving/passing procedures. This restriction occurs when pipelines 9 and 17 are simultaneously used for receiving (or passing through) products different from product 4. This operation is prohibited and the inequalities 39 and 40 restrict this case.

$$fr_{b,n,n1,d,p_b} \le ir_{\bar{b},\bar{n},n1,d1,p_{\bar{b}}} + M\cdot(1 - lr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{n, \bar{n}, n1\} \in N|n1 = 12, \{d, d1\} \in D \cap \{9, 17\}, prod_b \ne 4, prod_{\bar{b}} \ne 4, p_{\bar{b}} \in PR_{\bar{b},\bar{n},n1,d1}, p_b \in PR_{b,n,n1,d} \quad (39)$$

$$fr_{\bar{b},\bar{n},n1,d1,p_{\bar{b}}} \le ir_{b,n,n1,d,p_b} + M\cdot lr_{b,\bar{b},p_b,p_{\bar{b}}}$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{n, \bar{n}, n1\} \in N|n1 = 12, \{d, d1\} \in D \cap \{9, 17\}, prod_b \ne 4, prod_{\bar{b}} \ne 4, p_{\bar{b}} \in PR_{\bar{b},\bar{n},n1,d1}, p_b \in PR_{b,n,n1,d} \quad (40)$$

In N8, N11, and N12 areas, only 2-simultaneous-receivings are permitted. Constraints 41−48 contemplate the 2-simultaneous-receiving-cases to three areas. In particular, the condition of three general batches that will be received in each area had to be analyzed ($b < \bar{b} < b2$). For instance, the first batch can be simultaneously received with the second batch. But the next batch has to wait until the end of receiving of one of the two previous batches in order to start its receiving in N8, N11, or N12. In this case, the model decides the better condition to activate: either one or none of the two constraints to the same area, through the binary variables $lr_{b,\bar{b},p_b,p_{\bar{b}}}$ and $lrr_{b,\bar{b},p_b,p_{\bar{b}}}$. If $lr_{b,\bar{b},p_b,p_{\bar{b}}} = 1$, part $p_{\bar{b}}$ of $\bar{b}$ starts to be received (ir) in area nd_b after the end of the receiving (fr) of part p_b of batch b (constraint 41, 43, or 45). If $lrr_{b,\bar{b},p_b,p_{\bar{b}}} = 1$, part p_b of b starts to be received (ir) in the area nd_b after the end of the receiving (fr) of the part $p_{\bar{b}}$ of batch $\bar{b}$ (constraint 42, 44 or 46).

Equations 41 and 42 contemplate the 2-simultaneous-receiving-cases to N8.

$$fr_{b,n,nd_b,d,p_b} \le ir_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} + M\cdot(1 - lr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 8, \{d, d1\} \in D, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (41)$$

$$fr_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} \le ir_{b,n,nd_b,d,p_b} + M\cdot(1 - lrr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 8, \{d, d1\} \in D, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (42)$$

The receiving of products 13 and 14 is limited in N12. Constraints 43 and 44 contemplate this situation and allow, simultaneously, only two receiving procedures.

$$fr_{b,n,nd_b,d,p_b} \le ir_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} + M\cdot(1 - lr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 12, \{d, d1\} \in D, prod_b \in \{13, 14\} = prod_{\bar{b}} \in \{13, 14\}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (43)$$

$$fr_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} \le ir_{b,n,nd_b,d,p_b} + M\cdot(1 - lrr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 12, \{d, d1\} \in D, prod_b \in \{13, 14\} = prod_{\bar{b}} \in \{13, 14\}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (44)$$

Constraints 45 and 46 contemplate the 2-simultaneous-receiving-case at N11. The constraints are made in a similar form of constraints 41and 42. However, N11 can only simultaneously receive two products, if they are different ($prod_b \ne prod_{\bar{b}}$).

$$fr_{b,n,nd_b,d,p_b} \le ir_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} + M\cdot(1 - lr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 11, \{d, d1\} \in D, prod_b \ne prod_{\bar{b}}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (45)$$

$$fr_{\bar{b},n1,nd_b,d1,p_{\bar{b}}} \le ir_{b,n,nd_b,d,p_b} + M\cdot(1 - lrr_{b,\bar{b},p_b,p_{\bar{b}}})$$
$$\forall\{b,\bar{b}\} \in B, |b < \bar{b}, \{nd_b, n, n1\} \in N|nd_b = 11, \{d, d1\} \in D, prod_b \ne prod_{\bar{b}}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (46)$$

Constraint 47 translates the relationship between binary variables $lr_{b,\bar{b},p_b,p_{\bar{b}}}$ and $lrr_{b,\bar{b},p_b,p_{\bar{b}}}$. If the group of constraints 41, 43, and 45 is activated by $lr_{b,\bar{b},p_b,p_{\bar{b}}} = 1$, the group of constraints 42, 44, and 46 cannot be activated ($lrr_{b,\bar{b},p_b,p_{\bar{b}}} = 0$). Otherwise it is also true. It is possible that both constraints are not activated ($lr_{b,\bar{b},p_b,p_{\bar{b}}} = lrr_{b,\bar{b},p_b,p_{\bar{b}}} = 0$). In this case b is received simultaneously with $\bar{b}$.

$$lr_{b,\bar{b},p_b,p_{\bar{b}}} + lrr_{b,\bar{b},p_b,p_{\bar{b}}} \le 1 \quad \forall\{b,\bar{b}\} \in B, |b < \bar{b}, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d} \quad (47)$$

When analyzing three batches b, $\bar{b}$, and $b2$, at least one situation should be activated; that is, it is not possible to receive three batches simultaneously. Consequently, one of the binary variables in the constraint 48 should be equal to 1.

$$lr_{b,\bar{b},p_b,p_{\bar{b}}} + lr_{b,b2,p_b,p_{b2}} + lr_{\bar{b},b2,p_{\bar{b}},p_{b2}} + lrr_{b,b2,p_b,p_{b2}} + lrr_{b,\bar{b},p_b,p_{\bar{b}}} + lrr_{\bar{b},b2,p_{\bar{b}},p_{b2}} \ge 1$$
$$\forall\{b,\bar{b},b2\} \in B, |b < \bar{b} < b2, p_{\bar{b}} \in PR_{\bar{b},n1,nd_b,d1}, p_b \in PR_{b,n,nd_b,d}, p_{b2} \in PR_{b2,n2,nd_b,d2} \quad (48)$$

As it can be observed through constraints 34−48, to deal with local constraints is indeed a complicating feature that is addressed by the MILP proposed model.

4.2. Computational Results. The MILP model was implemented and solved using ILOG OPL Studio 5.5.1.[16] The solver CPLEX 11 was used. The computational hardware was an Intel Core 2/2.13 GHz processor with 2GB of RAM memory. Using the data presented in Tables 4, 5, and 6 the model described was run. Table 4 presents the values of demands, production, initial storage, and minimum/maximum storage capacity of each area and product.

Table 5 presents part of the data obtained by the allocation block. For this scenario we have a total of 323 batches and

Table 4. Data (10^3 volumetric units) at Each Area for a Month of Planning

area	N2	N3									N4									
product	4	2	3	4	5	6	7	8	10	14	1	3	4	5	8	9	11	12	13	14
demand	109	–	135	–	–	–	–	12	2	43	–	271	117	0	285	35	–	21	979	60
production	–	35	135	26	–	4	92	85	136	308	44	272	112	106	326	19	30	–	818	97
initial storage	7	5	3	1	8	7	–	42	40	46	15	9	13	37	68	10	14	1	144	12
minimum capacity	1	2	–	–	–	–	–	9	12	13	–	–	–	–	13	3	–	–	100	15
maximum capacity	12	6	10	4	9	9	18	33	43	41	30	37	27	72	80	20	58	11	160	44

area	N5				N6										N7				N8				
product	2	4	8	13	3	4	5	7	8	9	11	12	13	14	1	5	9	13	1	8	11	13	14
demand	44	53	–	–	207	66	–	–	29	–	14	–	37	44	–	–	–	150	–	–	–	–	–
production	10	26	93	118	206	65	131	173	39	172	–	–	257	44	–	4	13	–	–	–	–	–	–
initial storage	5	4	16	27	28	12	20	3	49	42	2	8	75	41	–	1	7	48	51	7	16	50	–
minimum capacity	–	1	–	–	–	4	–	–	11	7	–	–	–	10	–	–	–	–	–	–	–	–	9
maximum capacity	8	5	25	62	87	19	61	76	98	52	4	34	150	60	50	25	36	78	63	63	64	128	63

area	N9			N10							N11				N12			N13						N14
product	5	8	14	3	4	7	8	10	11	14	8	9	11	14	8	13	14	1	6	8	11	13	14	5
demand	–	–	–	68	17	–	89	146	0	116	47	172	–	43	67	–	90	36	1	58	15	–	78	228
production	–	–	–	–	137	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
initial storage	–	–	19	1	13	–	9	17	5	4	23	9	3	26	30	4	47	13	2	13	9	15	14	86
minimum capacity	–	–	–	–	9	–	–	2	–	–	8	6	–	8	13	–	15	–	–	4	–	–	5	55
maximum capacity	5	5	29	17	66	17	17	36	17	7	26	26	10	35	34	6	48	23	5	20	10	16	21	75

Table 5. Example of Data Supplied by the Allocation Block

b	*prod*$_b$	*no*$_b$	*nd*$_b$	*vol*$_b$(vu)	*TED*	*TEC*	*TRD*	*TRC*	routes								
...																	
61	P13	12	13	5178	0	10.56	0	21.11	N12	19	N13						
62	P2	3	5	2700	0	34.36	8.03	42.40	N3	10	N9	8	N5				
63	P14	3	10	6000	0	21.86	24.46	30.05	N3	12	N9	15	N10				
64	P4	10	2	5400	0	33.89	36.72	46.17	N10	14	N9	8	N5	7	N12	29	N2
65	P13	13	4	13695	0	26.86	0	53.71	N13	22	N4						
66	P11	8	6	12000	0	40.82	11.50	52.32	N8	26	N6						
67	P10	3	10	18000	0	49.38	1.67	51.05	N3	12	N9	15	N10				
68	P9	6	11	12300	0	47.23	16.76	64.00	N6	24	N8	28	N10	4	N11		
69	P2	3	5	2700	16.46	91.76	40.79	99.79	N3	10	N9	8	N5				
70	P14	3	9	11000	0	37.61	0	60.12	N3	12	N9						
248	P14	9	12	11000	0	37.61	0	60.12	N9	17	N12						
252	P13	12	13	9000	0	137.91	0	237.57	N12	19	N13						
249	P14	12	13	11000	0	37.61	59.71	60.12	N12	19	N13						
71	P8	5	12	8000	0	53.84	0	68.08	N5	6	N12						
250	P8	12	13	8000	0	53.84	20.09	68.08	N12	19	N13						
72	P4	10	2	5400	0	67.90	70.73	80.18	N10	14	N9	8	N5	7	N12	29	N2
73	P9	6	11	12300	0	47.97	52.33	64.73	N6	24	N8	28	N10	4	N11		
74	P4	3	5	2700	45.02	69.65	54.21	78.83	N3	10	N9	8	N5				
75	P7	6	7	8000	19.77	49.57	49.57	109.17	N6	25	N8	3	N7				
76	P14	3	10	6000	0	65.69	61.67	67.26	N3	12	N9	15	N10				
77	P4	10	5	2700	35.13	142.81	44.34	152.02	N10	14	N9	8	N5				
78	P8	3	10	9000	0	81.12	35.77	82.97	N3	11	N9	30	N10				
79	P4	10	2	5400	19.91	101.91	104.74	114.19	N10	14	N9	8	N5	7	N12	29	N2
80	P7	6	7	8000	52.84	82.64	82.64	142.24	N6	25	N8	3	N7				
81	P9	6	11	12300	0	96.87	101.23	113.63	N6	24	N8	28	N10	4	N11		
82	P14	3	10	6000	0	102.90	98.88	104.47	N3	12	N9	15	N10				
83	P2	3	5	2700	70.77	148.13	98.19	156.16	N3	10	N9	8	N5				
84	P5	4	14	18000	0	107.11	93.15	155.54	N4	1	N8	2	N11	5	N12	18	N14
85	P4	3	2	2700	116.78	136.35	128.63	148.20	N3	10	N9	8	N5	7	N12	29	N2
86	P7	6	7	8000	85.91	115.71	115.71	175.31	N6	25	N8	3	N7				
87	P10	3	10	18000	0	148.39	79.33	150.05	N3	12	N9	15	N10				
88	P8	5	12	10500	0	145.83	76.65	160.93	N5	6	N12						
89	P4	10	2	5400	65.57	152.92	155.76	165.20	N10	14	N9	8	N5	7	N12	29	N2
90	P14	3	10	5000	0	129.83	129.89	141.67	N3	12	N9	15	N10				
91	P13	5	12	9000	0	137.91	0	237.57	N5	20	N12						
315	P13	5	5	1300	0	10000	0	10000	N5	20	N5						
251	P13	13	4	9000	0	137.91	38.63	237.57	N13	22	N4						
317	P13	13	13	9400	0	10000	0	10000	N13	22	N13						
92	P8	3	10	9000	0	150.07	104.71	151.92	N3	11	N9	30	N10				
93	P13	6	7	21500	0	144.98	93.78	207.25	N6	25	N8	3	N7				
94	P8	5	12	8000	76.46	153.07	76.46	167.30	N5	6	N12						
253	P8	12	13	8000	76.46	153.07	119.32	167.30	N12	19	N13						
95	P14	3	9	11000	65.01	138.23	65.01	160.64	N3	12	N9						
254	P14	9	12	11000	65.01	138.23	65.01	160.64	N9	17	N12						
255	P14	12	13	11000	65.01	138.23	160.22	160.64	N12	19	N13						
96	P9	6	11	12300	65.95	145.77	150.14	162.54	N6	24	N8	28	N10	4	N11		
97	P1	4	13	18000	48.46	155.90	142.32	179.40	N4	22	N13						
318	P13	4	4	9400	0	10000	0	10000	N4	22	N4						
98	P2	3	5	2700	125.23	203.94	154.60	211.98	N3	10	N9	8	N5				
99	P14	3	9	12000	0	82.23	0	375.04	N3	12	N9						
.. .																	

Table 6. Parameters Considered for the MILP Model

number of batches:	323
days in seasonal costs:	7
areas in seasonal costs:	N6, N7 and N8
areas in surge tank operation:	N8, N9, N12, and N13
products in surge tank operation:	P1, P3, P4, P5, P6, and P7
α:	3
eps:	0.001
k:	10000
M:	1500.1

information of some of the batches is shown. Great part of the batches in Table 5 can be visualized in the pumping Gantt chart of the results for 10-first days (Figure 10; the column "*b*" at Table 5 is the label on Gantt charts). The MILP model considers the sequence suggested on this table. The first batch showed is batch 61 of product P13 that has to be pumped from node 12 to node 13 (by pipeline 19). The pumping volume is equal to 5178 vu; the sending time window spans from hour 0 to hour 10.56, and the receiving time window spans from hour 0 to hour 21.11.

By analyzing the table, one can observe that the next batches pumped from node 12 to node 13 (by pipeline 19) are, respectively, batches 252 and 249. Thus, the precedence of batch 61 over batch 252 has to be respected, and in an analogous way, the precedence of batch 252 over batch 249 has also to be respected. Thus, numerical values of batches (the label "*b*") are not a valid indicator of precedence but rather the relative position of batches within Table 5. Another example can be given by observing batches 317 and 97. Batch 317 appears before batch 97 on the list, and both batches share pipeline 22. Then, batch 97 should be pumped after batch 317. In particular, batch 317 uses the route N13→22→N13 and batch 97 uses the route N4→22→N13. On the other hand, if a pair of batches does not share the same pipeline, the order is not necessarily respected. The routes used by batches 90 and 92, for instance, are, respectively, N3→12→N9→15→N10 and N3→11→N9→30→N10.

The sending time windows (from *TED* to *TEC*) and the receiving time windows (from *TRD* to *TRC*) are also specified in Table 5. In some cases (e.g., batches 315, 317, 318) the suggested time intervals can be larger than the typical scheduling horizon (720 h). Within these cases, the batches can be pumped (or received) in any time of the scheduling horizon. The MILP model is in charge of such temporal decisions, which typically involve surge tank operations.

Table 6 shows some parameters considered in the MILP Model. The scheduling time of the batches is measured in hours. Thus, in a 30-day planning horizon, the pumping time will be typically less than 720 h. Thus, we penalized violations on time windows by a factor greater than 10 times ($k = 10000$) of the scheduling horizon. The M value was chosen by considering more than two times of the planning horizon, and *eps* is a small tolerance value used only to avoid equality situations on constraints.

The results obtained were validated by expert operators (schedulers) that deal with the scheduling procedures within this network. Several insights can be derived from the obtained solutions.

Figures 8 and 9 present the Gantt charts that translate the pumping and receiving procedures of the case study with 323 batches (including additional reversion batches) for one month of planning. Figure 10 presents in a more detailed form the 10 first-days of the scheduling horizon.

The scheduling is obtained in a detailed form, indicating various parts of pumping for the same batch (timing blocks). The timing block of the same batch is divided when the flow rate is changed or even when a stoppage is necessary. The stoppages frequently occur since the pipelines are shared among various different routes and products. Different colors are given for timing blocks of products.

Figure 10 shows the results for pumping procedures during the 10 first-days of the scheduling horizon. This figure presents all the used pipelines, giving an overview on how the pipeline

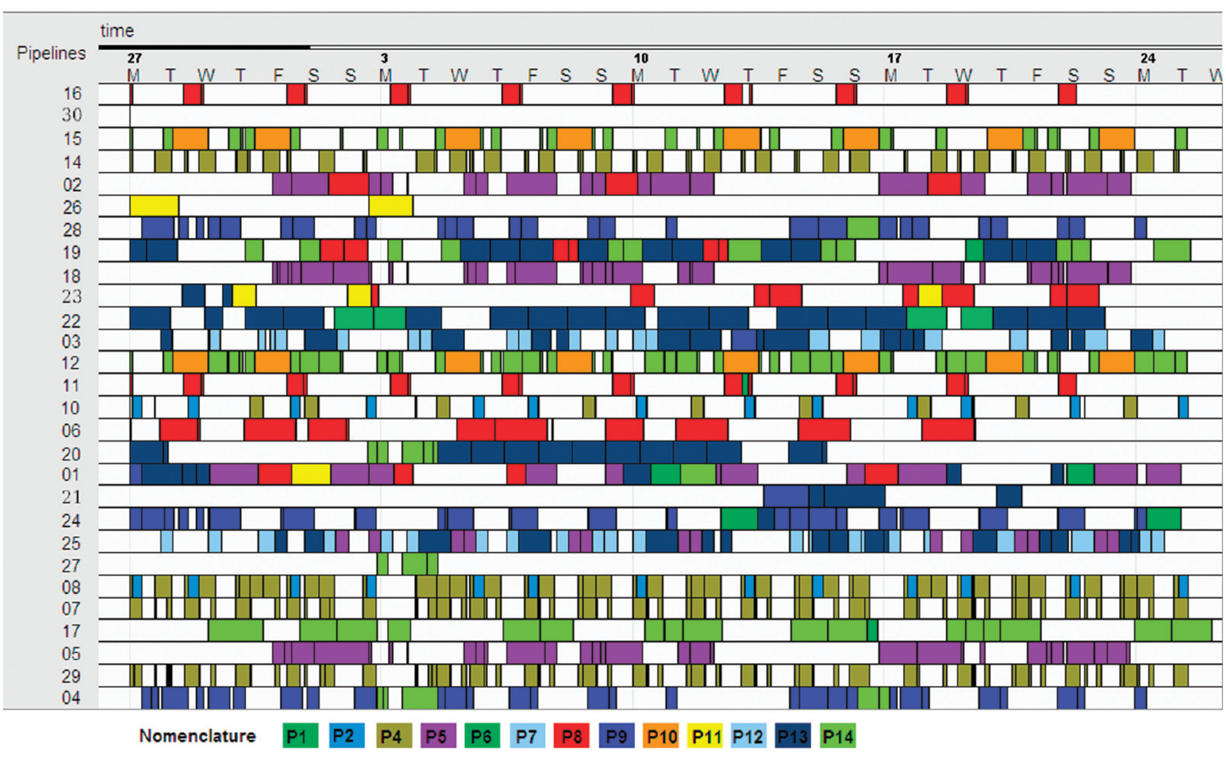


Figure 8. Sending Gantt chart.

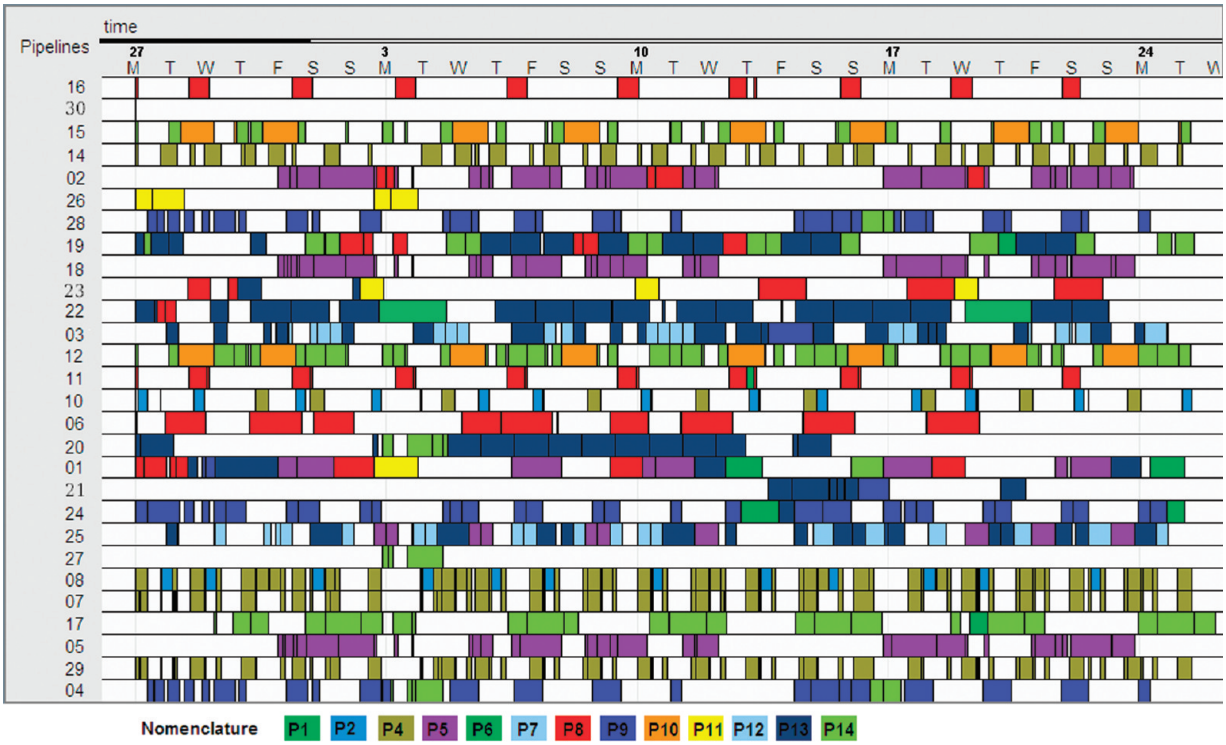


Figure 9. Receiving Gantt chart.

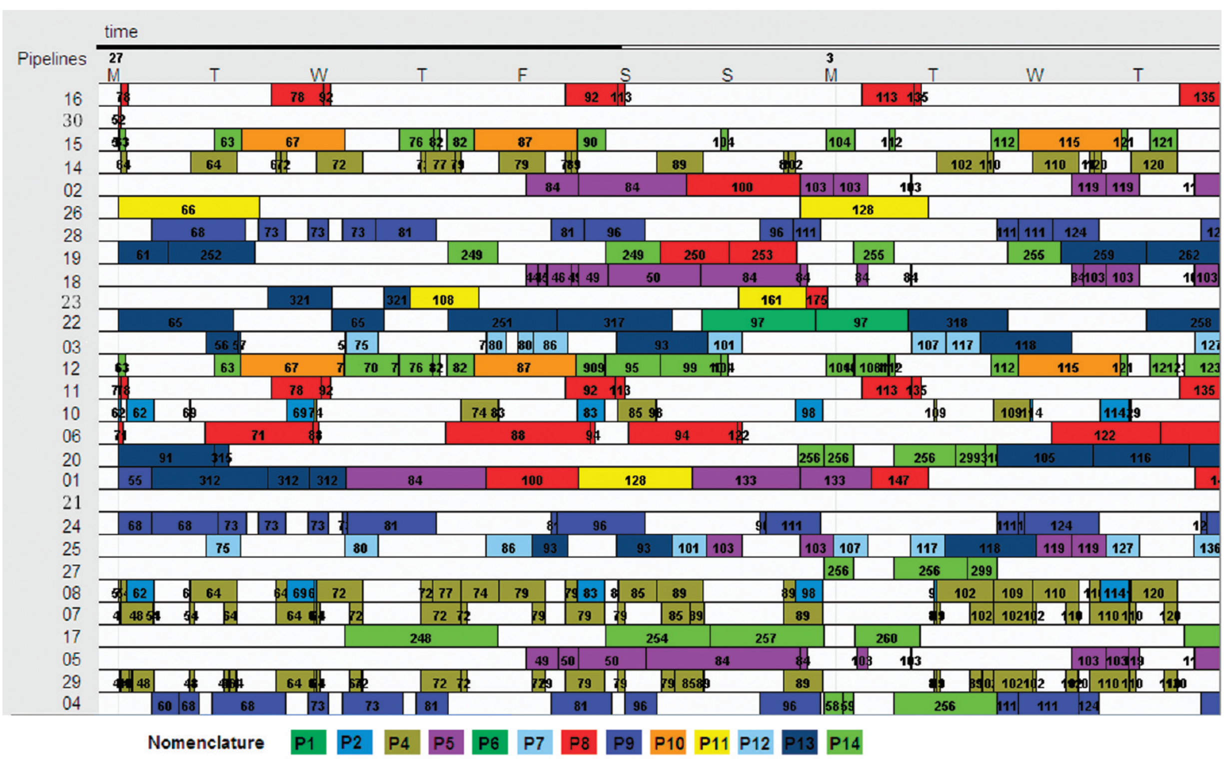


Figure 10. Ten first-days of the sending Gantt chart.

network management has to be conducted. In the Gantt chart, each batch has an identifying number, which remains the same as the batch passes through different pipelines (except for surge tank operations). For example, batch 84 passes through pipelines 01, 02, 05, and 18.

As referred above, the MILP model approach was developed respecting the batch sequence determined by the assignment/sequencing block. Then, it is important to notice that the sequencing block includes batches to assist the reversion flow process into the pipelines. These batches are generated to guarantee the delivery of the batches that are already inside the pipeline, when the pipeline will suffer a reversion in flow direction. This may also be present in the results obtained through the MILP model. Figure 11 shows one of these cases, where it is depicted a pipeline where the flow reversion procedure occurs (pipeline 22 in Figure 1). The reversion

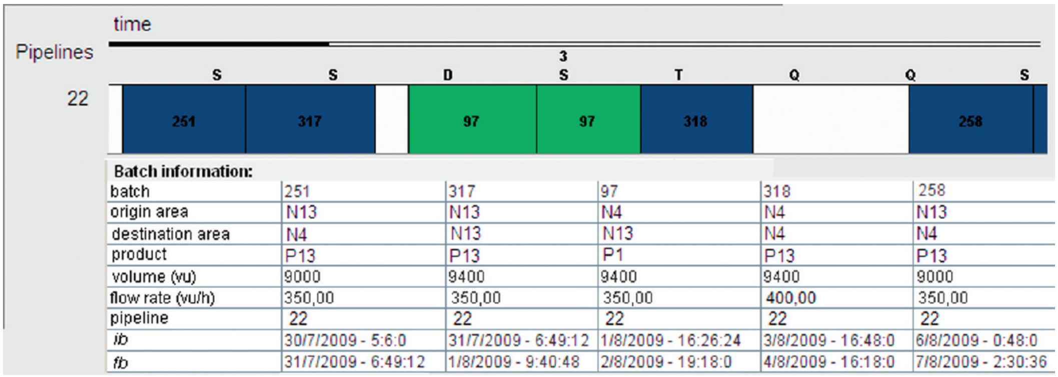


Figure 11. Example of flow reversion in pipeline 22.

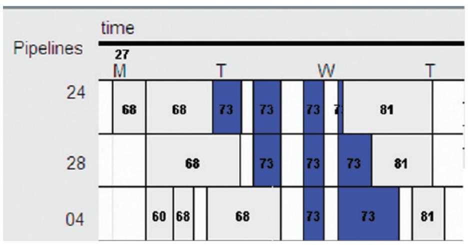


Figure 12. Gantt chart: batch 73.

involves the batches 317 and 318 (additional batches). As it can be visualized batch 251 and 258 are sent through pipeline 22 *from* N13, while batch 97 is sent *to* N13. In a first moment, batch 317 is pumped from area N13, fulfilling the pipeline and pushing batch 251 to its destination area (node N4). Batch 251 is totally received in its destination area due to batch 317 that pushes batch 251 to N4. At this moment, batch 317 is totally inside of pipeline 22. Then, batch 97 is sent from N4 through pipeline 22 to its destination area (node N13). At this procedure, pipeline 22 is operated in reverse flow. It is interesting to notice that batch 317 was sent back to node N13. So, batch 317 was sent from N13 and is, in a following step, pushed to N13. Batch 317 is an auxiliary batch used in the reversion procedure. Afterward, in a following moment, batch 318 totally pushes batch 97 to its destination area (node N13). Then, batch 258 pushes back batch 318 to node N4.

Figure 12 presents a three-pipeline example. Batch 73 is highlighted. The route of batch 73 contemplates the following areas and pipelines: N6 → 24 → N8 → 28 → N1 → 4 → N11 (see Figure 1). Thus, in a normal situation, a batch is pumped from its origin area, and it is received in its final destination, using all the pipelines of its route. Thus, batch 73 is pumped from N6 to N11, passing in sequence through pipelines 24, 28, and 4. It is possible to observe that there exist several timing blocks (parts) with the same batch index in Figure 12. For instance, it is important to notice that the number of parts that are pumped in an origin area can differ from the number of parts that are received in subsequent areas: there are 4 parts of batch 73 in pipeline 24, 3 parts of batch 73 in pipeline 28 and, in pipeline 4, the batch 73 is divided into 2 parts. The number of parts is different, in accordance with the batch and the pipeline. The simulation block divides a batch in various parts and calculates the number of parts that will be pumped (or received), supplying the MILP model through parameters. Some of these parameters are presented on Table 7. The difference in the size of the batches among pipelines is a consequence of the movement of the batches by another batch that is pumped in its origin area, since the flow rate and the dislocated volume vary in each pipeline.

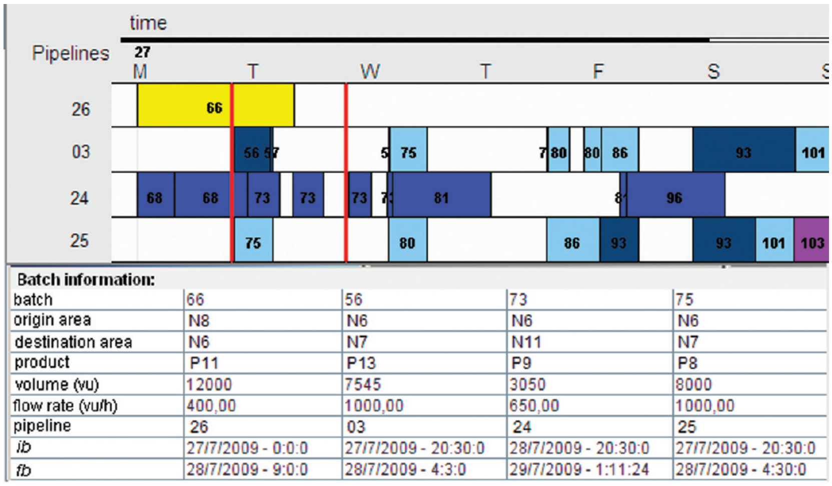
Table 7. Some Parameters Determined by the Simulation for the Example Highlighted in Figure 12

b	d	bo_b	p_b	p_{bo}
68	28	68	1	3
68	28	68	2	4
68	28	73	3	1
73	28	73	1	2
73	28	73	2	3
73	28	73	3	4
73	28	81	4	1
68	04	68	1	4
68	04	73	2	1
68	04	73	3	2
73	04	73	1	3
73	04	73	2	4
73	04	81	3	1
73	04	81	4	2

There are cases where the pumping procedure is interrupted and the batch is pumped later on. These stops can be visualized in Figure 12 through the gaps between parts of the same batch in a specific pipeline. This fact happens, for instance, with the batch 73 (highlighted). The reader can also identify the batches that influence the movement of parts of the batch 73. In pipeline 24, the second part of batch 73 starts to be pumped, pushing the first part of batch 73 to pipeline 28, influencing its movement. Batch 81 starts to be pumped to pipeline 24, pushing the last part of the batch 73 for the pipeline 28. Thus, it is possible to identify a connection between a batch b with another batch that is being pumped in its origin (bo_b) area.

Figure 12 and Table 7 indicate that the third part of batch 73 in pipe 28 is indeed composed by two smaller parts to get a one-to-one correspondence between the parts sending batches 73 and 81 in pipeline 24. A similar consideration can be made for the second part of batch 73 in pipeline 04. In addition, this figure shows two cases where the transfer of a part of batch 68 or 81 from pipeline 28 to pipe 04 is temporarily interrupted. On these situations, the stoppages at on-peak demand hours are represented in different forms: (*i*) on pipe 04 a clear interruption between batches can be identified; (*ii*) on pipes 24 and 28 the stoppage is incorporated inside the temporal blocks; that is, batches 68 and 81 stop in pipes 24 and 28 during on-peak demand hours, and the temporal blocks on these pipelines are increased and contain these stoppages. This fact can be understood by inspecting, for instance, inequalities 5 and 6. Thus, within the exemplified cases, the stoppages for on-peak demand hours are explicit on pipe 04 and implicit on pipes 24 and 28 (see explanation of Figure 13).

Figure 13 illustrates the occurrence of an on-peak demand period and how it influences the pumping of batches. Four batches are highlighted in this figure: batches 66, 56, 73, and 75. The same on-peak hour influences batches 66, 56, 68, and



batch	66	56	73	75
origin area	N8	N6	N6	N6
destination area	N6	N7	N11	N7
product	P11	P13	P9	P8
volume (vu)	12000	7545	3050	8000
flow rate (vu/h)	400,00	1000,00	650,00	1000,00
pipeline	26	03	24	25
ib	27/7/2009 - 0:0:0	27/7/2009 - 20:30:0	28/7/2009 - 20:30:0	27/7/2009 - 20:30:0
fb	28/7/2009 - 9:0:0	28/7/2009 - 4:3:0	29/7/2009 - 1:11:24	28/7/2009 - 4:30:0

Figure 13. Gantt chart: on-peak demand period.

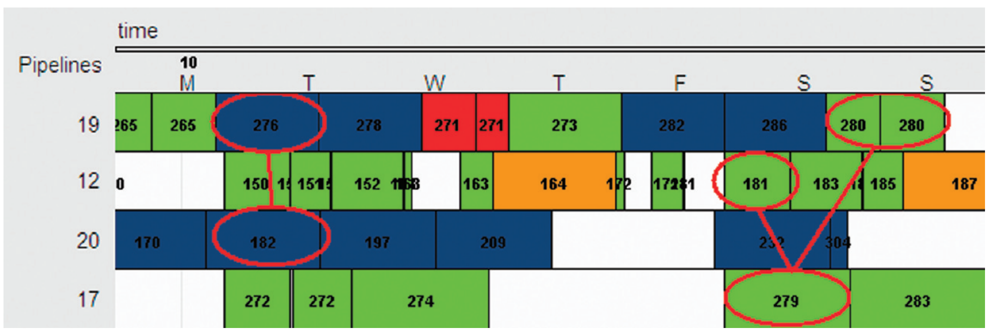


Figure 14. Gantt chart: surge tank operation.

75 on Monday (M) in different ways. Batch 66 has its temporal block increased. Its duration time is $tb_{b,no_b,n,d,p} = part_{b,p}/flw_{b,p} = 12000/400 = 30$ h plus 3 h of the on-peak demand. Since the pumping start (*ib*) is at Monday, 00:00, the end will occur 33 h later, that is, on Tuesday at 09:00. This means that during the on-peak period the pumping of batch 66 was stopped. In different operational conditions, the start of pumping of batch 75, which occurred at 20:30, was delayed until the end of the on-peak period. It is important to notice that, within the considered problem, during Saturdays and Sundays the on-peak demand period must not be considered.

Figure 14 presents two examples where surge tank operations happen. We first analyze the one that involves 3 pipelines (12, 17, and 19) and 2 areas. Within the route N3 → 12 → N9 → 17 → N12 → 19 → N13, the surge tank operation occurs at N9 and N12. As previously explained, in surge tank operations, the route of a batch is split into two routes. The first route considers the origin area until the intermediate area with the tanks used in the operation; the second one, from the area in surge tank operation until the next destination area. Thus, from N3 to N9 the batch labeled 181 is pumped. Then, from N9 to N12 the same batch now labeled 279 is pumped at a different flow rate. Intermediate storage was used in order to balance the difference between input and output flow rates. In similar reasoning, from N12 to N13 the same batch now labeled 280 is pumped. So, at this example, we have two consecutive surge operations. Since the flow rate of batch 279 is smaller than the flow rate of batch 181, the pumping of batch 279 starts when the batch 181 is received in N9 (constraint 14 is active). On the other hand, the flow rate of batch 280 is bigger than flow rate of batch 279. In this case, the pumping of batch 280 could finish when the batch 279 finishes its receiving time in N12. A similar case can be seen with batches 182 and 276. The flow rate of batch 276 is bigger than that of batch 182. The pumping of batch 276 is finished at the same time that batch 182 is entirely received (constraint 15 is active).

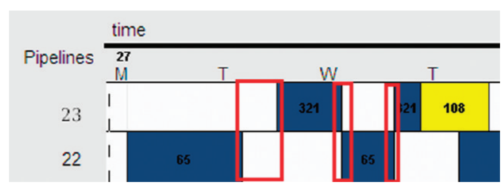


Figure 15. Gantt chart: local constraint.

One example of local constraint can be visualized in Figure 15. Pipelines 22 and 23 are adjacent to area N13. Batches 321 and 65 of product P13 cannot be simultaneously pumped from this area. The second part of batch 65 in pipeline 22 has to wait for the end pumping time of the first part of batch 321 in pipeline 23 to start its pumping procedure. Finally, the last part of batch 321 in pipeline 23 has to wait until the end pumping time of the second (in this case, the last) part of batch 65 in pipeline 22 to start its pumping procedure.

The initial storage level of each tank, refinery production campaigns, and local market consumption are known *a priori*. Thus, based on the determined pumping and receiving times, it is possible to obtain the storage profile in diverse areas for different products. Figure 16 shows the typical inventory profiles for all products in different areas. These inventories present the aggregated stock levels (1000 vu) for a 30-day horizon.

Through the analysis of Figure 16, increasing and decreasing "inventory curves" are observed. In the case of a specific area without local production then an increasing curve indicates, in this case, the receiving of a batch. For example, area N14 does not send batches of P5 (red line) to other areas. In fact, N14 is a consumption area of P5. So, the decreasing

Figure 16. Continued.

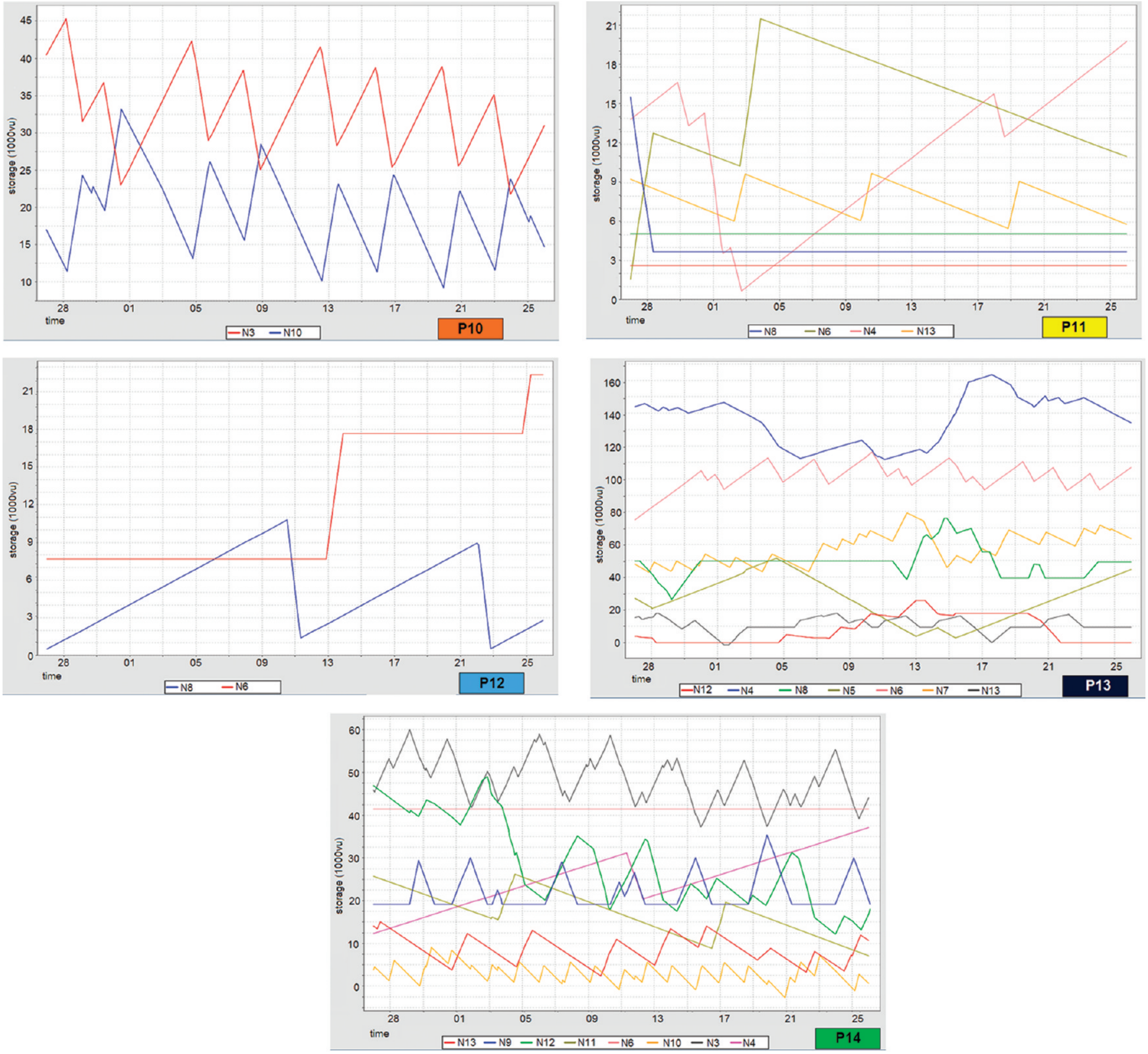


Figure 16. Inventory per product at different areas.

curves show the local consumption of P5 in this area. The different increasing inclinations indicate receiving of (parts of) batches at different flow rates.

The storage profile of product P9 indicates that the movements are frequent in areas N6 (blue line) and N11 (red line). Area N6 is a refinery that supplies P9 to depot N11, supplying its demand during the entire considered scheduling interval. The profiles of these two areas are almost symmetrical; at the same time that N6 is sending P9, area N11 is receiving, at an initial time, the product that was available inside the pipeline and, afterward, some parts of the pumped batch. This fact occurs because, in this case, the route is almost dedicated to this product and the pipelines are fulfilled with P9 or small volumes of another product.

Some violations on time windows are observed at the MILP model results. These violations are shown in Table 8 and can be illustrated by observing the graphics of P7, area N7, in Figure 16. For this product, at this scenario, the initial storage level was typically too low at the consumer and the producer. The smallest admissible batch from the producer (5000 vu) could only be sent in day 28. A given period was necessary to produce such a batch. In addition, if we consider the transit time from producer to consumer, (roughly 3 days), we can observe that there was no practical condition of the consumer to be supplied. Within the Figure 16, theoretical negative values are plotted. In practical terms, a product shortage would be observed from day 29 until the first sent batch arrives at the consumer (day 31). In Table 8, each value indicates the deviation of the time of pumping/receiving of a batch in relation to the previously estimated time windows.

According to Table 8, the majority of the violations are observed in receiving time in advance (*ad*) of the batches. Significant violations, which are defined as values greater than 24 h, are presented in 10 batches that represent about 3% of all the batches. The violations on both origin and destination areas can suggest changes on production rate, forecasting the lack of tanks to store the products, while they cannot be pumped or consumed. However, these changes are undesirable, and as

Table 8. Violation Values

origin area violations (pumping)				destination area violations (receiving)			
batch/node	advance (h)	batch/node	delay (h)	batch/node	advance (h)	batch/node	delay (h)
195/N6	9,04	76/N12	3,82	66/N10	2,14	225/N8	33,92
196/N6	36,8	113/N3	8,88	73/N9	6,82	226/N7	39,77
209/N6	53,68	114/N9	20,58	89/N10	22,05	245/N7	3,26
261/N6	15,92	115/N12	13,35	97/N10	21,06	261/N7	40,21
302/N6	5,54	150/N12	13,35	106/N9	46,17	281/N10	16,28
		153/N9	11,42	107/N12	33,28	284/N7	12,21
		225/N6	29,73	122/N10	1,38	302/N7	6,58
		238/N9	11,33	147/N9	12,88	316/N10	6,59
				170/N7	2,95		
				182/N9	20,38		
				183/N12	7,06		
				228/N9	25,02		
				229/N12	12,3		
				271/9	23,6		
				300/N10	9,66		

Table 9. Model Performance for Different Scenarios

number of batches	323	367	376
model status	optimal	optimal	optimal
time (s)	69.5	80.5	206.5
objective function	8.3375×10^6	2.1031×10^7	9.1617×10^6
iterations	4648	4148	28389
variables	17403	21891	24071
integer variables	7796	11653	13013
constraints	41065	48336	52548

future work, these violations can be used to aid the resource allocation/sequencing blocks to determine better pumping volumes/sequences. However, according to the opening condition of storage levels, it becomes quite hard to attend to production/consumer needs, as previously explained for P7 in area N7. Violations on time windows are, in some cases, unavoidable. In fact, if hard constraints on storage levels were implemented instead of allowing violations due to scenario "inconsistencies", this would have led to infeasible models.

4.2.1. Results for Other Scenarios. Table 9 compares the model performance for three different scenarios. The first one was detailed in previous sections. The parameters shown in Table 6 were used for the three studied cases, and a 30-day planning horizon is considered. The number of pumped batches differs from case to case, according to results of the allocation block. All scenarios were run until the status of the model was optimal.[16] No relative gap was considered.

Table 9 indicates that optimal solutions in all scenarios can be found in few minutes of running. In the first two cases in less than 2 min and in about 3 min in the last case. The growth of the number of pumped batches increases the number of variables and constraints, influencing the computational burden.

4.2.2. Comparison of Results. In order to evaluate the characteristics of the developed formulation, results are presented and compared to those of a former developed model.[11] This previous model is a simplified version of the current model and was only applied to a small part of the considered pipeline network. In order to be able to perform an adequate comparison, the simplified model was adapted to consider the network here presented (Figure 1). It is worthwhile to note that the batch sequence, time windows, and other parameters provided by the allocation and sequencing blocks are the same for both models.

The Gantt chart presented in Figure 17a shows the main results of the simplified model,[11] while the Gantt chart presented in Figure 17b, shows the scheduling for the temporization model considered in this work. Both figures indicate the pumping of batches within a month.

The main difference between a and b of Figure 17 is the number of timing blocks. Figure 17b presents more timing blocks than Figure 17a, although the number of batches sent for the two cases is equal. The proposed model (with results presented in Figure 17b), is able to identify potential pumping stops, splitting a batch in several parts, while the model proposed by Neves-Jr et al.[11] (results in Figure 17a), is not able to identify such operational conditions. Thus, the presented model gives a more detailed scheduling to the system operator. As can be seen in pipeline 16, for instance, the pumping of the batches for the present model was made in a more detailed manner, while in the former developed model,[11] the pumping of the products is made through an aggregated batch. The detailed pumping is better for the company, since the storage limits are more easily controlled. Also, the sizes of timing blocks for the proposed model are smaller than those of the simplified model.[11] This occurs because the new model shows only the real pumping time, where stoppages among the start and end pumping of a batch are considered.

An example of the difference between the formulations of the two models can be seen through expressions 49 and 50. As the simulation is not run for the model proposed by Neves-Jr et al.,[11] constraint 49 determines the pumping of variables *ib* and *fb* in intermediate areas. This constraint is used when the origin area of the batch $b1$ is $n3$, determining the sending time window of the bath b in the intermediate area $n3$. In this model, a constraint should be made for each case, for example, when a batch's volume ($volb_b$) is greater than a pipeline's volume (vol_d), a pipeline's volume is greater than a batch's volume, or when the sum of the route pipelines' volume is greater (or less) than the batch's volume. Despite that, the previous model[11] does not contemplate all the cases, resulting in wrong timing values for the respective variables.

$$\begin{aligned} & fb_{b,n3,n4,d3} - ib_{b,n3,n4,d3} \geq (volb_b - vol_d - vol_{d1})/flow_b \\ & +vol_{d3}/flow_{b1} + \alpha\cdot \sum_{h\in HP} z_{b,n,h} \\ & \forall\{b,b1\} \in B, |b < b1, \{n,n3,n4\} \in \\ & \quad N|n3 \neq n4 \wedge n3 = no_{b1}, n = no_b \\ & \{d,d1,d3\} \in D|d \neq d1 \neq d3, \\ & \text{for } volb_b \geq (vol_d + vol_{d1}), volb_{b1} \geq vol_{d3} \end{aligned} \tag{49}$$

These "comparisons" in the proposed model are made into the simulation block (Table 1), removing part of the computational burden of the MILP model. Moreover, all the cases of volumes can be determined through a unique constraint 50 for the new model (constraint 6 in the proposed MILP model).

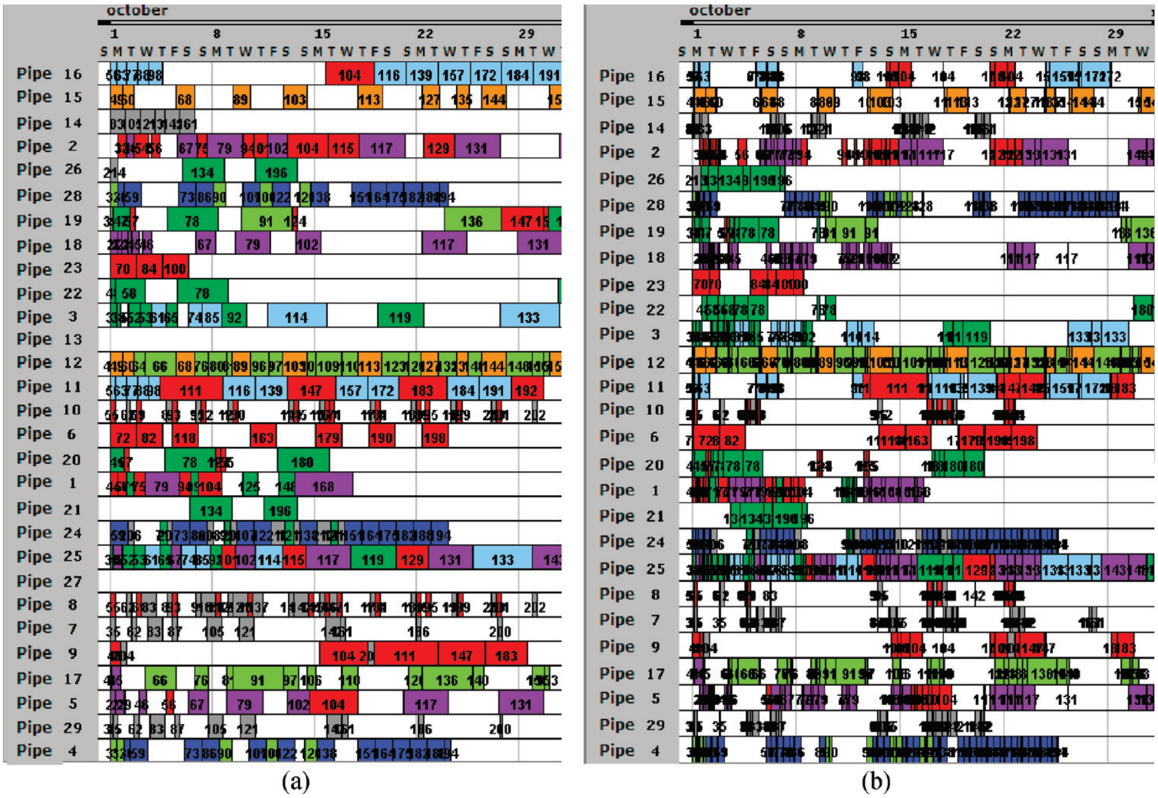


Figure 17. Comparison of models: Gantt charts.

$$fb_{b,\bar{n},n1,d1,p_b} \geq ib_{b,\bar{n},n1,d1,p_b} + tb_{bo_b,no_{bo_b},n,d,p_{bo_b}} + \alpha \cdot \sum_{h \in HP} z_{bo_b,p_{bo_b},no_{bo_b},h}$$

$$\forall \{b, bo_b\} \in B, \{no_{bo_b}, n, \bar{n}, n1\} \in N, \{d, d1\} \in D, p_b \in PB_{b,\bar{n},n1,d1}, p_{bo_b} \in PB_{bo_b,no_{bo_b},n,d} \quad (50)$$

Table 10 shows computational results for the considered case study, where the first column presents results for the model developed by Neves-Jr et al.,[11] named here as Simplified Model (SM), and the second column presents results for the proposed MILP model (PM).

Although the computational time to obtain a solution increases in relation to the simplified model, this time still remains in the same order of magnitude. The number of continuous/binary variables and constraints increased significantly. This increase was a consequence of a more elaborated formulation that allows more detailed information about the operational scheduling. This new formulation provides a schedule that mimics real practices of network operation. Using the present model, the network scheduler is able to visualize usual programming details such as pumping stoppages not considered in the previous work.[11] A more realistic inventory profile was also obtained.

5. Final Considerations

This article deals with the development of a solution approach to schedule the operational activities in a pipeline network. Such approach took into account a decomposition strategy based on the integration of different heuristic blocks and a MILP model. The mixed integer linear programming model, which used a continuous time representation, was the core of a timing block and was described along this work. The tool developed was applied to a real case study that represents the operational conditions of a real multiproduct pipeline network that transports oil derivatives and organic products. The considered case study is particularly complex and involves many nodes and pipelines. The difficulty of obtaining a feasible scheduling of operational activities is a day-to-day problem faced by the company schedulers and such a model appears as a decision tool that can assist the decision makers.

Looking for an approach that treats a series of scheduling details (e.g., resources sharing, storage management, demand requirements, operational conditions, etc.) in a reasonable computational time, the use of a hierarchical strategy is required, which is an advantage when compared to approaches based on a single optimization technique. As a result, the scheduler can visualize in a reduced computational time scheduling details for generated solutions involving a horizon of approximately 1 month. The computational time to obtain a solution by all blocks of the hierarchical structure is about 3−5 min. On a first stage, a short-/middle-scale operational scheduling is analyzed in order to be implemented. Afterward, tendencies related to the lack of products in customers and supply excesses in the refineries can be analyzed through the time windows violations. If necessary, operational decisions are taken in order to avoid such tendencies to happen.

There is a series of functionalities and enhancements obtained with the developed approach. Some of them are herein highlighted:

Table 10. Comparison of Model Performances

	SM	PM
# batches	247	247
# variables	5692	9695
# binary variables	1680	4100
# constraints	16868	38672
# iterations	1624	1230
model status	optimal	optimal
CPU time (s)	22.3	60.0

- The proposed model was able to address a large-scale pipeline scheduling problem, obtaining a computational solution in nonelevated computational time.
- The approach has proved to be a functional tool to aid the decision-making process of schedulers.
- Through the development of the simulation block and using the parameters obtained by this block, the MILP formulation could address in a detailed way some complex operational conditions. For instance, the system can now indicate batches that are responsible for the movement of other batches within the pipeline network and, thus, provide a series of scheduling details.
- Details addressed within the optimization model allowed the identification of time instants for pumping and stops procedures that should be executed in the entire pipeline network. This approach indicates a set of operational details that were not addressed in previous published works.[11]
- The results allow the verification of pipelines usage throughout the scheduling time (approximately one month), identifying system bottlenecks, thus providing information to avoid such areas. In addition, the identification of inventory evolution along the scheduling is also possible (e.g., Figure 16).
- The time windows concept allowed a diagnosis that facilitates the negotiation between the production of refineries and the demand of customers with the pipeline scheduling. The generated schedule tries to minimize the number of system stoppages, through the minimization of the scheduling horizon, while respecting refineries/customers requirements.
- Finally, the proposed approach allows the evaluation of new operation modes for the pipeline system, thus suggesting large-scale modifications. Given the solution of one month, the results can provide evidence of new procedures for the pipeline network operation that can be adopted by the company schedulers. Such suggested operation can be compared with the real situation. In general, it has been observed that the suggested results tend to minimize the number of batches in relation to operational practice. This fact implies, for instance, reduction in generated interfaces and operational maneuvers.

Beyond the presented functionalities, the practical operation of the developed system, in phase of implementation by the oil industry, can provide benefits from a managerial point of view. The optimization of the transference procedures and the storage issues will certainly reduce operational losses, generating increases in the company profits.

In future developments, some further issues are intended to be addressed. The time windows limits should be previously estimated by the assignment block. Within the MILP model, violations of these values are allowed, but minimized. These violations can be used as a feedback to analyze the proposed scheduling, providing improvements. These analyses will be object of future studies. Also, constraints to deal with work shifts should be considered to prevent pumping starts in between the time periods where workers are changing their shifts. As the time to obtain a solution is short, if the input data set changes considerably during the scheduling horizon, a new scenario can be created from this date, and a new solution can be obtained. However, this issue can be better studied in future works, minimizing the change between the previous scheduling and the new one. Improvements in other blocks that compose the hierarchical approach (assignment/sequencing, simulation, heuristics) can decisively contribute to the overall approach performance.

Acknowledgment

We acknowledge financial support from ANP and FINEP (PRH-ANP/MCT PRH10 UTFPR), Brazilian Oil Company PETROBRAS (Grant 0050.0044324.08.4), CAPES/PDEE (Grant 3262/08-1) and CNPq (Grants 305188/2007-0 and 311877/2009-5). Special thanks to Petrobras Logistics Coordinator Paulo César Ribas.

Nomenclature

Indices/Sets

B = set of batches (determined by assignment block), where $b \in B$

D = set of pipelines, where $d \in D$

HP = set of temporal intervals where on-peak demand hours occur ($h \in HP$)

N = set of network nodes, where $n \in N$

$\bar{n}$ = index $\bar{n} \in N$ indicates the node immediately before a pipeline in the route of a batch

N_{hor} = set of areas (nodes) that stop the pumping at on-peak demand hour ($N_{hor} \subset N$)

$PB_{b,\bar{n},n,d}$ = set of sending parts of the batch b in the pipeline d between $\bar{n}$ and n. $PB_{b,\bar{n},n,d} = \{pb^{first}_{b,\bar{n},n,d}, ..., pb^{last}_{b,\bar{n},n,d}\}$ (determined by simulation block)

$PR_{b,\bar{n},n,d}$ = set of receiving parts of the batch b in the pipeline d between $\bar{n}$ and n. $PR_{b,\bar{n},n,d} = \{pr^{first}_{b,\bar{n},n,d}, ..., pr^{last}_{b,\bar{n},n,d}\}$ (determined by simulation block)

p = part of a batch, which is determined by the simulation block, where $p \in PB_{b,\bar{n},n,d}$ or $p \in PR_{b,\bar{n},n,d}$. To this specific part there exists a volume $part_{b,p}$ herein defined

General Parameters

α = number of hours during the on-peak demand period

eps = small value

fh_h = ending time of on-peak demand period (day h)

ih_h = starting time of on-peak demand period (day h)

k = constant used as a penalization weight within the objective function

M = large value

$PM_{i,j}$ = matrix of i lines and j columns ($j = 2$), where each position i shows a pair (area, product) and the flow rate can be changed by the "surge" operation

npm = maximum number of lines of the matrix $PM_{i,j}$ ($i = 1, ..., npm$)

$tb_{b,no_b,n,d,p}$ = pumping time of each part p of batch b in its origin area no_b (h)

Parameters (determined by assignment block)

no_b = origin area of batch b ($no_b \in N$)

nd_b = destination area of batch b ($nd_b \in N$)

$prod_b$ = product associated to batch b

TED_b = lower time bound to pump the batch b in the origin area (h)

TEC_b = upper time bound to pump the batch b in the origin area (h)

TRD_b = lower time bound to receive the batch b in the destination area (h)

TRC_b = upper time bound to receive the batch b in the destination area (h)

vol_b = batch volume (vu)

Parameters (determined by simulation block)

bo_b = batch in process of pumping that will influence the receiving of b

$pb^{first}_{b,\bar{n},n,d}$ = minimum index of $PB_{b,\bar{n},n,d}$
$pb^{last}_{b,\bar{n},n,d}$ = maximum index of $PB_{b,\bar{n},n,d}$
$pr^{first}_{b,\bar{n},n,d}$ = minimum index of $PR_{b,\bar{n},n,d}$
$pr^{last}_{b,\bar{n},n,d}$ = maximum index of $PR_{b,\bar{n},n,d}$
$part_{b,p}$ = pumping volume of each part p of batch b (vu)
$flw_{b,p}$ = pumping flow rate of each part p of batch b (vu/h)

Continuous Variables

ad_{b,nd_b} = advance in receiving time of the batch b in the destination area nd_b (h)
ao_{b,no_b} = advance in pumping time of the batch b in the origin area no_b (h).
do_{b,no_b} = delay in pumping time of the batch b in the origin area no_b (h).
dd_{b,nd_b} = delay in receiving time of the batch b in the destination area nd_b (h).
$fb_{b,\bar{n},n,d,p}$ = end time of sending the part p of batch b from node $\bar{n}$ to node n using pipeline d (h).
$fr_{b,\bar{n},n,d,p}$ = end time of receiving the part p of batch b from node $\bar{n}$ to node n using pipeline d (h).
$ib_{b,\bar{n},n,d,p}$ = start time of sending the part p of batch b from node $\bar{n}$ to node n using pipeline d (h).
$ir_{b,\bar{n},n,d,p}$ = start time of receiving the part p of batch b from node $\bar{n}$ to node n using pipeline d (h).

Binary Variables

$x_{b,p,no_b,h}$ = 1, if the pumping start of batch b occurs before the start of an on-peak hour
$y_{b,p,no_b,h}$ = 1, if the pumping ending of batch b occurs after the start of an on-peak hour
$w_{b,p,no_b,h}$ = 1, if the pumping start of batch b occurs before the ending of on-peak hour
$z_{b,p,no_b,h}$ = 1, if the pumping procedure of part p of batch b in the origin area no_b is stopped
$lp_{b,\bar{b},p_b,p_{\bar{b}}}$ = decides on the activation of constraints in the local pumping constraints
$lr_{b,\bar{b},p_b,p_{\bar{b}}}$ = decides on the activation of constraints in the local receiving constraints
$lrr_{b,\bar{b},p_b,p_{\bar{b}}}$ = decides on the activation of constraints in the local 2-simultaneously receiving constraints

Literature Cited

(1) Sasikumar, M.; Prakash, P. R.; Patil, S. M.; Ramani, S. Pipes: A Heuristic Search Model for Pipeline Schedule Generation. *Knowledge-Based Systems* **1997**, *10*, 169–175.

(2) 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.

(3) Moro, L. F. L.; Pinto, J. M. Mixed-Integer Programming Approach for Short-Term Crude Oil Scheduling. *Ind. Eng. Chem. Res.* **2004**, *43*, 85–94.

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

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

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

(7) Magatão, L.; Arruda, L. V R.; Neves-Jr, F. Using CLP and MILP for Scheduling Commodities in a Pipeline. In *Proceedings of the 15th Europe Symposium on Computer-Aided Process Engineering (ESCAPE 15), Barcelona, Spain*. May, 29−June, 1, 2005; *Comput.-Aided Chem. Eng.* Puigjaner, L., Espuña, A., Eds.; Elsevier: Amsterdam, NewYork, 2005; Vol. 20, pp 1027−1032.

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

(9) 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.

(10) 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* (17), 5659–5672.

(11) Neves-Jr, F.; Arruda, L. V. R.; 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. In *Proceedings of 17th European Symposium on Computer Aided Process Engineering (ESCAPE 17)*, Bucareste. 2007. *Comput.-Aided Chem. Eng.*; Vol 24; Elsevier: Amsterdam; pp 697−702.

(12) Reklaitis, G. V. Overview of Scheduling and Planning of Batch Process Operations. I. *Proceedings of the NATO Advanced Study Institute on Batch Processing Systems. Antalya, Turkey* **1992**, 660–705.

(13) Moura, A. V.; Souza, C. C.; Cire, A. A.; Lopes, T. M. T. Heuristics and Constraint Programming Hybridizations for a Real Pipeline Planning and Scheduling Problem. *11th IEEE International Conference on Computational Science and Engineering* **2008**, *44*, 5–462.

(14) Yamamoto, L. Metaheuristics Applied to Product Sequencing in Real-World Pipeline Networks. (English translation), Doctoral Thesis. Universidade Tecnológica Federal do Paraná, 2009.

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

(16) *ILOG OPL Development Studio 5.5: User's Manual*; ILOG S.A.: France, 2007.

Received for review April 29, 2009
Revised manuscript received April 12, 2010
Accepted April 21, 2010

IE900685V