Computers and Chemical Engineering 174 (2023) 108242

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

A scalable optimization framework for refinery operation and management

Mayank Baranwal [a,*], Mayur Selukar [a], Rushi Lotti [a], Aditya A. Paranjape [a], Sushanta Majumder [b], Jerome Rocher [c]

[a] TCS Research, Tata Consultancy Services Ltd, India
[b] Engineering and Industrial Services, Tata Consultancy Services Ltd, India
[c] Total Energies, France

ARTICLE INFO

Keywords:
Process optimization
Refinery scheduling
Mixed-integer linear programs
Crude blending
Throughput maximization

ABSTRACT

End-to-end refinery management is a complex scheduling problem requiring simultaneous optimization of coupled subprocesses at several stages. In the specific context of this paper, a planner needs to ascertain (i) how best to store incoming crude at a port, (ii) schedule its transfer, after dewatering, to downstream refinery tanks, and (iii) schedule further processing in the crude distillation units (CDUs). The movement and storage of crude is subjected to various physico-chemical and operational constraints. The resulting optimization problem is combinatorial in nature and scales exponentially with the number of tanks, types of crude, and modes of operation. The problem becomes particularly challenging with stochasticity in crude receipt, requiring the planner to modify their decisions in real-time. In this paper, we develop a scalable, hierarchical framework to address the end-to-end refinery management for throughput maximization. The framework relies on an innovative approach to decoupling the decision-making at port and refinery, reducing significantly the complexity of the overall optimization problem. The proposed approach also results in a significant improvement over the schedules generated by an expert human planner for throughput maximization. It takes only a few minutes to execute the entire optimization routine, over a 30 day planning window, on a standard computer, making it possible to use implement our approach in a time-critical, real-time operational setting.

1. Introduction

In this paper, we describe an optimization framework for refinery operations, based on combining heuristics and mixed-integer linear programming (MILP). The refinery operations considered here essentially involve moving and blending crude subject to a set of constraints. The present problem can be viewed as a constrained flow optimization problem over networks.

The setting for our problem, introduced in a preliminary form in our prior work (Paranjape et al., 2022) and illustrated in Fig. 1, can be described as follows. Crude is received at a port and needs to be transferred to refinery tanks that feed the crude distillation units (CDUs). Crude may be blended at the port as well as the refinery tanks. The primary challenge stems from the fact that a single pipeline connects the port and the refinery. Its maximum transfer rate, after accounting for temporal operating constraints, matches the maximum throughput rate of the CDUs. Moreover, the transfer of crude through the pipeline needs to preserve grade continuity to the extent possible, i.e., there should be as few switches between crude grades as possible during an operational window. The setting of the present paper builds upon Paranjape et al. (2022) by adding the complexity of crude blending as well as realistic holding constraints (Hou et al., 2015; Yang et al., 2017). While MILP can solve the entire problem in principle, it is computationally burdensome with 60k integer decision variables. Therefore, following the framework in our prior paper, we adopt a hierarchical approach which combines heuristics and MILP.

1.1. Overview of the literature

Refinery scheduling problems are primarily based on linear mass balance equations, with nonlinearities arising on account of chemical mixing, flow constraints, and the dynamics of chemical processing units. The decision variables to be solved for could lie in a continuous field (setting flow rates as functions of time) or in a discrete integral field (turning flow valves on or off). Typically, decision problems involve variables of both types. The order of the decision problem (i.e., the number of decision variables) can be as high as 10^4, depending on the temporal discretization applied to the problem.

Refinery scheduling problems have been addressed in the literature primarily using linear or mixed-integer programming (LP or

* Corresponding author.
E-mail address: baranwal.mayank@tcs.com (M. Baranwal).

https://doi.org/10.1016/j.compchemeng.2023.108242
Received 17 November 2022; Received in revised form 17 March 2023; Accepted 23 March 2023
Available online 31 March 2023
0098-1354/

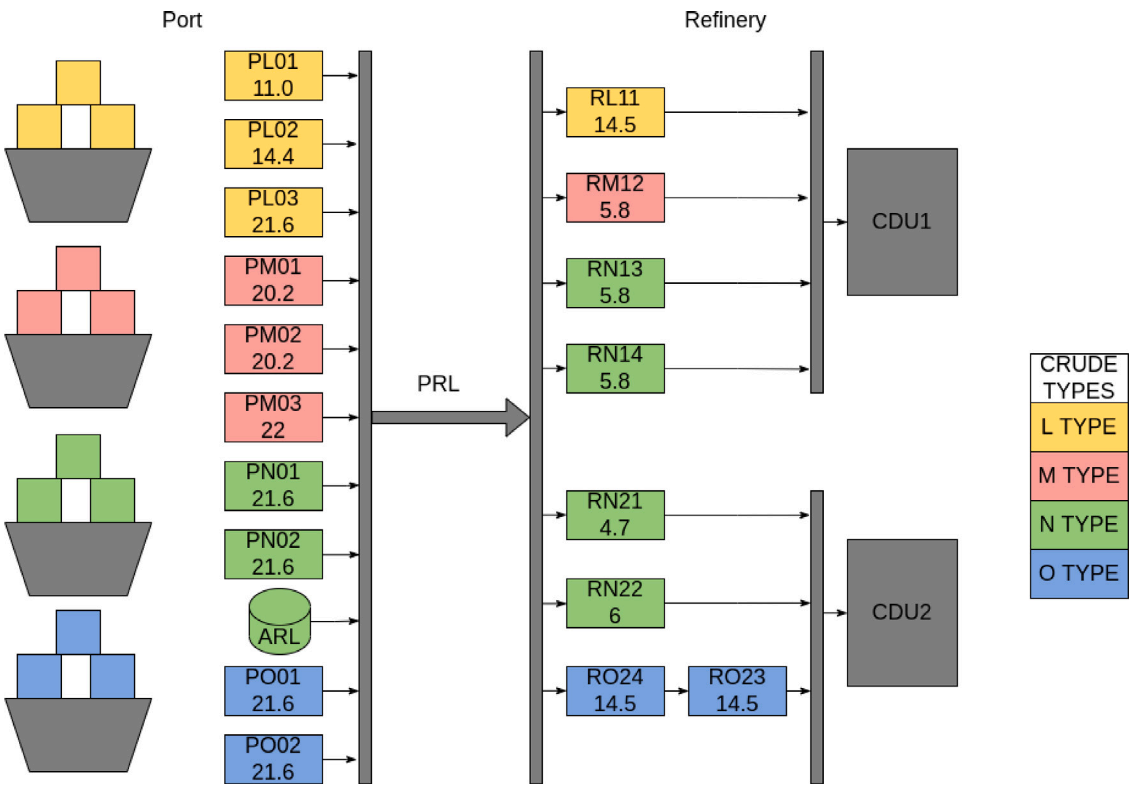

Fig. 1. Overview of the problem.

MILP) (Koenig, 1963; Baker and Lasdon, 1985; Lee et al., 1996; Kallrath, 2005). It is generally accepted that, despite the linear mass balance equations, nonlinear terms are quite fundamental in a refinery model. Bilinear terms arise frequently to account for blending (Andrade et al., 2016; Uribe-Rodriguez et al., 2020). Nonlinearities that arise as part of the processing dynamics can be difficult to model and there have been attempts to capture these using data-driven approaches (Li et al., 2016; Boukouvala et al., 2016; Demirhan et al., 2020). Since nonlinearities frequently introduce non-convexity in the problems, convex relaxation techniques have been concurrently introduced to deal with them in a computationally efficient manner. These include the use of McCormick envelopes (Andrade et al., 2016; Uribe-Rodriguez et al., 2020) for dealing with bilinear terms. Integrated refinery planning under uncertainties arising from stochasticity in production processes and demand forecasts, supply chain disruptions and transportation costs has been discussed extensively (Kallrath, 2002; Shah, 2005; Tong et al., 2012; Lima et al., 2018).

Nonlinear optimization problems are generally difficult to solve computationally, and a combination of linear and nonlinear programming has been used traditionally to simplify the computational complexity (Biegler, 2010). MILP using branch and bound and MINLP using reduced gradient was employed in Pinto et al. (2000) and Neiro and Pinto (2004). Sequential MILP approximation was employed in Mendez et al. (2006) to solve the planning problem in the presence of blending-induced nonlinearities. Heuristic initial guesses, found by solving relaxed problems, have been demonstrated in Andrade et al. (2016) for reducing the complexity of the parent MINLP. In Kolodziej et al. (2013), a radix-based discretization method was employed to discretize one variable in the bilinear terms, leading to the creation of efficient MILP relaxations for solving the pooling problem. The model incorporated inventory, flow, and quality constraints, and despite the increase in the number of binary variables, it was able to produce solutions in a shorter amount of time. Recently, a technique which combines multiparametric disaggregation and optimality-based bound tightening was demonstrated in Castro (2016) and Zhang et al. (2021) for solving MINLP problems. MINLP problems have also been solved "directly" using commercial solvers such as BARON (Siamizade, 2019) and ANTIGONE (Li et al., 2016; Boukouvala et al., 2016).

A further fundamental feature of refinery planning is the presence of multiple time scales (Castro et al., 2018) and spatiotemporal clustering (Uribe-Rodriguez et al., 2020), both of which have been used fruitfully to simplify the MINLP. For instance, in Uribe-Rodriguez et al. (2020), the refinery was broken into functional clusters. A MILP relaxation technique, followed by optimality-based bound tightening, was used to solve the optimization problem in each cluster. An associated challenge is that of uncertainties – these can manifest in terms of price variations, variations in demand and crude delivery, or processing unit down times – over short as well as long-time scales. Receding horizon techniques have been used in Wagle and Paranjape (2020) and Paranjape et al. (2022) to deal with uncertainties essentially by periodic replanning using a combination of the most recent information and updated estimates. Robust optimization approaches have been employed in Zhang et al. (2021) for dealing with price variations by including these explicitly in the cost function.

More recently, reinforcement learning (RL) techniques have been explored to deal with uncertainties inherent in the planning problem (Hubbs et al., 2020; Paranjape et al., 2022). Policies obtained using RL are computationally light as far as real-time implementation is concerned, generalizable, and capable of dealing with hidden models for external variables such as delays in crude delivery or time-dependent variation for product demand Wagle and Paranjape (2020). However, RL requires ample time-consuming training and, moreover, adequate exposure to a vast range of operating scenarios. Although training is a one-off exercise in principle, it may need to be carried out periodically in practice to ensure that the RL algorithm has been exposed to changes in the dynamics of the plant or the uncertainty models. In contrast to RL, solutions based on integer programming can be calculated relatively quickly using any of several well-recognized solvers such as BARON or ANTIGONE, and these techniques enjoy wider acceptance in the industry (on par with what might be termed as a fully "explainable solution") because of the confidence built over several decades of deployment.

1.2. Contribution

- Our approach is hierarchical: Decoupling port and refinery sides by first identifying optimal schedule through PRL (Port-to-Refinery Pipeline).
- Once decoupled, we handle the port and refinery side separately, building dewatering constraint, optimal crude blending, tank switching logic at the refinery side.
- Use of heuristics to handle crude storage at the port side.
- Combination of the simplest of heuristics for crude storage, along with MILPs for hierarchically handling various hydraulic, physical, and chemical constraints make this overall approach extremely scalable (takes less than a minute for total execution) allowing for real-time manipulation in events of uncertainties in crude-delivery and/or refinery operations

2. Problem description

Fig. 1 represents a scaled version of a real industrial refinery system.[1] The refinery system comprises of three key components — (a) Port, (b) Port-to-Refinery pipeline (PRL), and (c) Refinery. The PRL is a shared, single pipeline that transfers crude from the port to the refinery which is equipped with storage tanks and two crude distillation units (CDUs).

We consider four basic types of crude oil: (a) L-type — low sulphur and low metal content, (b) M-type — moderate sulphur and low metal content, (c) N-type — large sulphur and moderate metal content, (d) O-type — large sulphur and large metal content. In addition, the Arab Light crude, also commonly known as the ARL, is an N-type crude. Each storage tank is equipped with holding only one type of crude. The crude arrives at the port in ships, the arrival schedules for which are known a priori.

In addition to an exclusivity constraint that allows each tank to store only one type of crude, the storage tanks at the port-side are subjected to three additional constraints:

- **Capacity constraints** that limit the amount of crude each tank can hold.
- **Inflow/Outflow constraints** that prohibit simultaneous inflow and outflow to and from the tanks, respectively.
- **Dewatering constraints** to allow the excess water mixed in the crude to get separated naturally while the crude is stored in the tanks. The PRL is allowed to draw crude from a specific tank only after 24 h have elapsed since the last time the tank was supplied with crude.
- **Exclusion constraints** A ship's contents can be transferred to multiple port tanks, but not at the same time. The tanker can only connect to the inlet of one port tank at a time. If one port tank becomes full while the tanker still has remaining contents, the excess can be transferred to another available port tank. If no port tanks are available to receive the ship's contents, the tanker must wait, causing demurrage costs for the operator.

Once the crude is sufficiently dewatered, it is ready to be transferred to the refinery through the PRL. The PRL is a shared pipeline and is accessible only during a single block of 19.8 h during a 38 h window. The restriction of only being able to use the pipeline for 19.8 h out of every 38 h window is because the same pipeline is shared by two separate organizations, one using it for 19.8 h and the other for 18.2 h. While the PRL can draw dewatered crude from multiple tanks simultaneously, it cannot draw crudes of two different types at the same instant. The maximum flow rate through the PRL is limited to 1 k m^3/h.

Like the storage tanks at the port, the storage tanks at the refinery site are also subjected to exclusivity, capacity, and simultaneous inflow/outflow constraints. However, unlike the tanks located near the port, the refinery-side tanks are not subjected to dewatering constraints; however, dewatering is recommended for high-quality crude, such as the L and M-type. Subject to the physical constraints, these tanks supply crude to the CDUs for further processing. The maximum inflow rates at CDUs are limited to 0.22 k m^3/h and 0.32 k m^3/h, respectively. Depending on the availability of various crude types through the course of a month, the planner comes up with a daily mode-wise schedule for processing crudes in the CDUs. CDU1 can process crudes of types L, M and N, while CDU2 is dedicated for processing crudes of types N and O. It is also required to process N-type crude on all days. Thus, the production plan for CDUs is complementary to each other. For instance, if CDU1 is operating in L-mode (i.e., processing L-type crude) on a given day, then CDU2 must operate in N-mode. Similarly, if CDU1 operates in N-mode, then CDU2 must operate in O-mode.

Depending on the mode of operation, each CDU is subjected to further set of hydraulic constraints, as well as constraints on maximum sulphur and metal content. The constraints are broadly classified into two categories — (a) the maximum atmospheric residue yield (AtRes), and (b) the maximum overhead (OVH). Crudes with different set of properties but of the same type are often blended to meet the hydraulic and metal content constraints. In an event where it is impossible to create a blend that meets the hydraulic constraints, the inflow rate to CDUs must be reduced to meet these constraints. The reduction in throughput leads to shortfall in targeted crude production. The primary objective is to minimize the total shortfall in targeted production over the period of 30 days. CDU1 is subjected to an additional operational constraint, which is related to processing low-sulphur crude soon after processing the high-sulphur crude. Crude with low-sulphur content is of the finest quality and should not be diluted by letting it to mix with large sulphur content crude. Consequently, it is not desirable for CDU1 to be operated in L-mode soon after finishing up processing N-type crude.

Decision variables: Our objective is to minimize the total shortfall in production at the CDUs for a period of 30 days subjected to various physical, chemical, and operational constraints. In terms of the known quantities, the refinery operator has access to the initial states (crude levels) of all the storage tanks, as well as arrival schedule for different crude types with heterogeneous set of properties. The decision-making is done primarily at three stages:

- **Allocating crude receipt to storage tanks near the port**: The incoming crude on each day needs to be stored in the corresponding tanks near the port. Depending on the state of each tank, the operator must choose the right tank to be filled keeping in mind the aforementioned 24 h dewatering window. The operator also needs to limit the demurrage cost that is incurred due to the unavailability of sufficient storage in the tanks located near the port.
- **Scheduling crude transfer through PRL**: Depending on the availability of the crude at port-side storage tanks, active status of the PRL, states of storage tanks near the refinery, and the crude demands from the CDUs, the operator needs to schedule transfer of flow through the PRL. It is also desirable that these transfer schedules are contiguous to avoid frequent switching across tanks.
- **Transferring crude from refinery tanks to CDUs**: Depending on the mode of operation on a given day, the CDUs are supplied with crudes from the respective tanks. The operator schedules these transfers subject to crude availability through PRL, mode of operation at the CDUs, and other hydraulic and metal content constraints. While blending *across* crude types is prohibited, it is possible to downgrade N-type crude by blending it with some amount of O-type crude in extreme circumstances although it incurs significant cost to the operator.

[1] Name withheld due to NDA.

- **Optimal production plan**: The final set of decision variables are related to finalizing the production plan (or the mode of operation of CDUs). Production plan must account for meeting the targeted demand for each crude-type, as well as operational constraints and availability of sufficient crude in refinery storage tanks at the beginning of each day.

3. Baseline approach based on business rules

As described earlier, our ultimate objective is to optimize the throughput of CDUs. In this section, we assume that a production plan is available and prescribes the crude being processed on any given day and the ideal PRL schedule which, when accompanied by optimal blending, allows 100 percent of the desired throughput at the CDU. The business rules presented in this section deal with this decoupled problem. The business rules are found by breaking the problem in to 4 separate parts in the following order: draining one or more port tanks; filling one or more port tanks; filling one or more refinery tanks, and draining one or more refinery tanks.

3.1. Business rules for the port

We recall the operating schedule for the PRL is available along with that of crude receipt and the initial state of the tanks. We use this information to prescribe the ideal drain and fill amount from each tank and generate the actual delivery schedule.

1. **Draining the port tanks:** For every crude type, we are given the volume of crude to be drained to PRL along with the target CDU. We start by finding the subsets of tanks available to drain (T_{drain}) and use an ILP to formulate a blend amongst T_{drain}; this blend is then drained as per PRL availability.
2. **Filling the port tanks:** Once the draining operation for all crude types is completed, the tank availability is updated and based on incoming crude, we find tanks available to fill (T_{fill}). The tank with the most empty capacity is picked and filled until either the tank is full or the incoming crude is consumed. This process is repeated until the incoming crude has been fully consumed.

The algorithm for the port side drain and fill is described in the flow chart of Fig. 2.

Exceptions may occur due to the constraints on the available tanks and crude. These may result in an inability to either form an optimal blend or send the desired amount of a particular blend through the PRL. In either case the CDUs throughput is reduced, and this in turn results in lower consumption this deficit is then deducted from future PRL demand to not have an overflow situation at the CDU/Refinery side. These reductions may also result in cases where the refinery is unable to consume the incoming fuel, based on economic factors and the future drain schedule. In such cases, the remaining excess fuel might be downgraded to a lower crude grade or some demurrage costs may be incurred. These exceptions are flagged for manual handling on a case-by-case basis.

3.2. Business rules for the refinery

The business rules for the refinery are formulated assuming that the incoming PRL crude schedule, the initial tank levels and the CDU demand schedule are known. The rules are as follows:

1. Fill the incoming crude in the tanks and if tanks are available to drain, drain them based on the CDU demand.
2. If more than 2 tanks are available to fill, fill the tank with the maximum fill volume and move on to other tanks if needed.
3. If more than 2 tanks are available to drain, drain the tank with the maximum drain volume.
4. If there is no incoming crude, transfer crude between RO24 and RO23 if RO24 has enough volume to drain and RO23 has enough space to fill.

The logic for Refinery side is represented in the flowchart of Fig. 3.

4. Hierarchical MILP-based framework

We now propose and formulate the hierarchical framework for addressing several aspects of refinery management and scheduling. The approach aims to first formalize the crude transfer plan through the PRL at a day-level resolution, i.e., in the first pass, the hierarchical framework concerns with coarser level decision making disregarding the details on the exact schedule for crude transfer within each day and the properties of the crude blend generated inside the PRL. This is done to decouple the port and refinery optimization problems, and significantly reduces the complexity of the overall refinery management. Post decoupling, the optimization problems at the level of port and refinery are nearly identical. The objective in either scenario is to allocate incoming crude to one of the storage tanks, and depending on the demand at the outlet, engage tanks for crude withdrawal. This work describes scalable MILP formulation for each of these sub-problems, involving crude, tank, and operational-level constraints at required stages. Below we discuss these formulations in detail.

4.1. MILP for optimal production plan generation

As stated earlier, we first aim to decouple the port and refinery optimization problems. This is achieved by identifying an optimal crude transfer plan from port-side tanks to the PRL and from the refinery-side tanks to the CDUs at a day-level resolution. Consequently, the primary decision variables include daily crude transfer plan through the PRL and daily mode of operation at the CDUs. i.e., the type of crude to be processed by CDUs on each day.

To leverage a hierarchical approach for generating optimal transfer and production plan, we disregard finer level constraints at this stage. For instance, all tanks tied to storing a specific type of crude are combined to form a single *zone* with a storage capacity of total of all the tanks in that zone. Thus, at this stage, the optimization routine at this stage only considers allocating incoming crude to a specific zone, which reduces the computational complexity of the overall optimization problem. The exact tying of crude to specific tanks is handled later in the next step of our hierarchical framework (see Fig. 4 for detailed schematic). We specify four unique zones corresponding to storing four different types of crude, in addition to the ARL zone at the port side. The maximum and minimum holding capacities of the ith-zone are depicted by $S_{\max}$ and $S_{\min}$, respectively. The zones $\{1,2,3,4\}$ correspond to crude-types L, M, N and O, respectively. The initial state of these zones is depicted by S_0, while the daily incoming crude arrival plan is depicted by S_{in}. Not all N-type crude is ARL, and thus the binary input is_{ARL} is used to depict if the incoming N-type crude is indeed of class ARL. Recall that ARL is available only in packets of $4.44\,\text{k}-\text{m}^3$. We use integer variables $g \in \{0,1,\dots,4\}$ to represent the number of packets of ARL being transferred through the PRL.

We have a similar nomenclature for zones located at the refinery. Zones $\{1,2,3\}$ at the refinery correspond to crude-types L, M and N, respectively, while zones $\{4,5\}$ correspond to crude-types N and O, respectively. Note that we make a distinction between N-type crude being fed to CDU1 and CDU2. The maximum and minimum holding capacities of the ith-zone at the refinery are depicted by $Q_{\max}$ and $Q_{\min}$, respectively. The initial state of these zones is depicted by Q_0. The maximum rates at which crude can be fed to the CDU1 and CDU2 are represented by C_1 and C_2, respectively.

The PRL is a shared single pipeline which remains active only for a duration of 19.8 h over each period of 38 h. This implies that while the PRL is active for the first 19.8 h on day 1 of the planning, it is

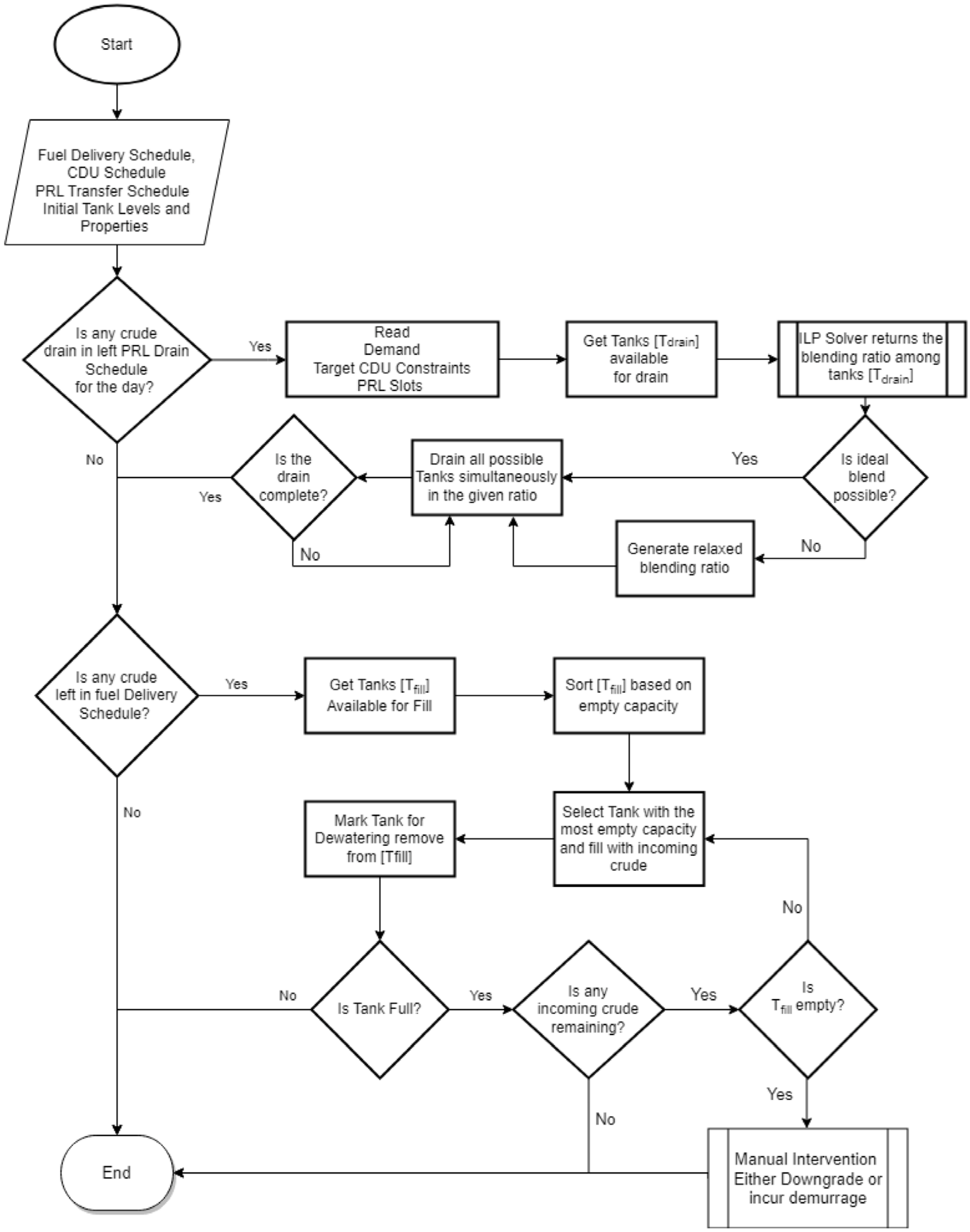

Fig. 2. Logic for port side.

only going to be active for the last 10 h on the following day, leading to non-uniform maximum crude transport capacity denoted by $P_{\max}$. While intermixing crude of different types is largely prohibited, it is still possible (though not preferred) to blend crude of types N and O to avoid shortfall in throughput to the CDUs. We use the decision variables O2O and O2N to denote how much of crude of type O is used daily to feed the refinery zones of types O and N, respectively. Likewise, decision variables N2O and N2O depict the amount of crude of type N that is fed into O-type zone at the refinery. The amount of crude transferred through the PRL is represented by the decision variable u, where indices $\{1,2,3,4,5\}$ correspond to crude types L, M, N (for CDU1), N (for CDU2) and O, respectively. It is also desirable to ensure continuity of operations, i.e., every time the planner decides to push a specific crude type through the PRL, it is desirable to transfer at least 2.22 k m^3 of crude before the planner decides to transfer crude of a different type. We use the binary variable a to capture such contingencies.

The primary objective of our planner is to ascertain the daily mode of operation of CDUs, i.e., the planner decides on the type of crude that must be processed on a given day inside CDU1 and CDU2. The decision variable corresponding to mode of operation of CDU1 is denoted by w, where modes $\{1,2,3\}$ represent processing of crude types L, M

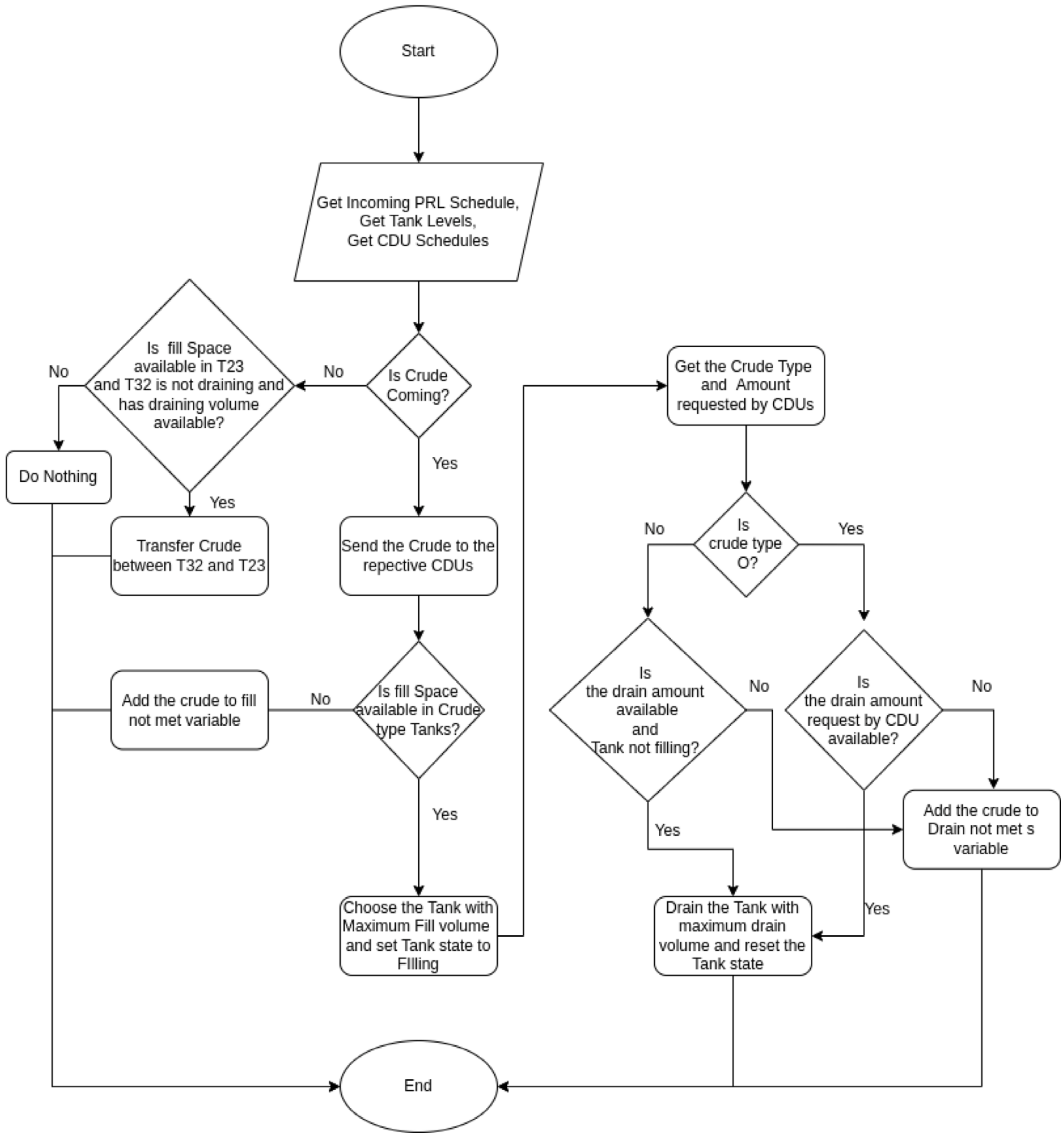

Fig. 3. Logic for refinery side.

and N, respectively, inside CDU1. The mode of operation of CDU2 is complementary to that of CDU1: it processes O whenever CDU1 processes N, and it processes N otherwise.

When processing crude of different types inside the CDU1, adequate consideration must be considered to avoid degradation of processed high-quality crude. For instance, the crude type L is a low-sulphur crude, and it is required to avoid processing high-sulphur crude of type N on the preceding day, since the leftover residue inside CDU1 degrades the overall quality of the incoming L-type crude. The daily change in mode of operation of CDUs is represented by the binary decision variable s, and it is desirable to minimize cumulative mode-switches subject to operational constraints. Despite the relaxation of tank and crude blend specific constraints, it may still be not possible to run the CDUs at maximum throughput for each day. This may also have to do with the fact that there is insufficient crude available over a period of D days. Any such event of shortfall is represented by the variable s_f. Table 1 summarizes all the above input and decision variables.

The MILP for generating the optimal production and transfer plan with zone-based mass-balance and contiguity constraints is formulated in (OptPlan-MILP). The objective is to minimize the combination of total number of mode switches, amount of N and O-type crude being blended with O and N-type crude, respectively, and the total shortfall in throughput at the CDUs.

The first set of constraints require that on each day d the total amount of crude transferred through the PRL must not exceed the maximum allowable limit $P_{\max}[d]$. The second constraint requires that on each day, CDU1 operates in one of the three modes — L, M or N. The third set of constraints simply states that the N-type crude can be sourced from the O-type zone, N-type zone, and ARL, respectively. Similarly, the O-type crude can be either be sourced from the N-type zone and O-type zone, respectively.

The fourth and fifth sets of constraints enforce the mass-balance requirements on each day for the port and refinery zones, respectively. Note that the mass-balance constraints for the zones located at the refinery are modelled through a fictitious shortfall variable s_f. If the total shortfall is zero, then the constraints translate to simple mass-balance constraints, however, introduction of fictitious shortfall variable s_f allows us to describe mass-balance constraints at maximum throughput rates C_1 and C_2. In an event, where it is impossible to run the CDUs

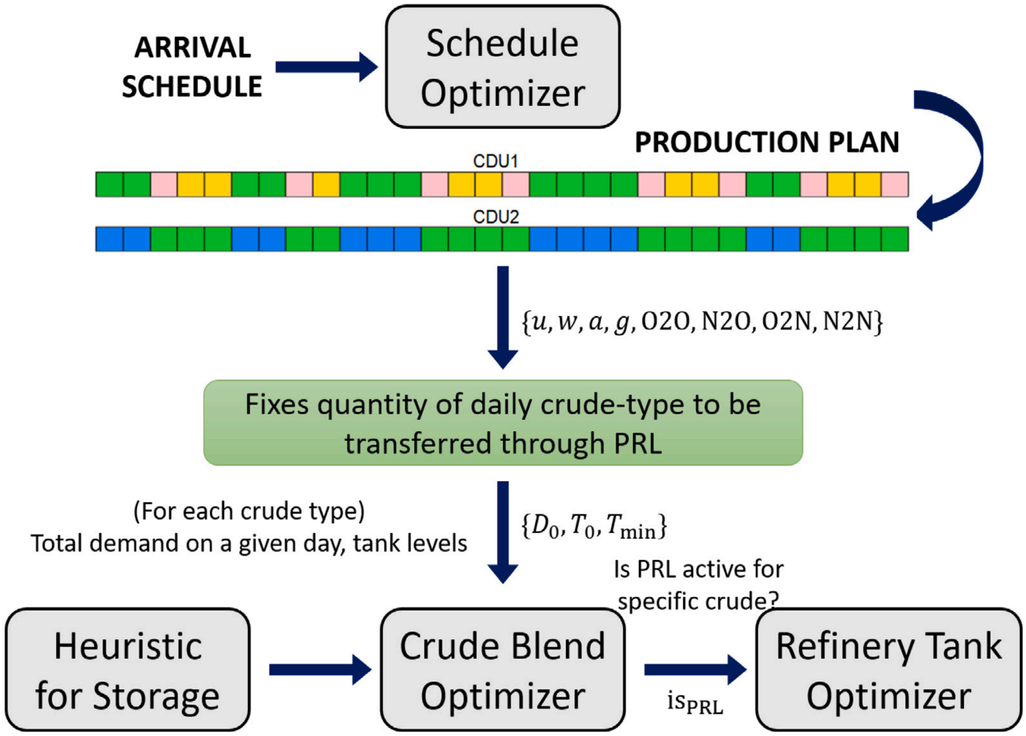

Fig. 4. Schematic of the proposed hierarchical framework. Schedule optimizer is used to generate feasible schedules for mode of operation, as well as flow through the PRL. This decouples the process optimization for port and refinery tanks. In addition, the crude blend optimizer aims to create a blend at each instant that minimizes the total shortfall in production. The variables located at the input to various sub-blocks indicate outputs of the previous sub-blocks, and are used to solve the next optimization problem in hierarchy.

Table 1
Description of symbols for OptPlan-MILP.

Symbol	Description	Type	Sub-type	Size
D	# of days	Input	Real	Scalar
$P_{\max}$	Daily max flow through PRL	Input	Real	D
S_0	Initial qty. of crude at port	Input	Real	4
$S_{\max}$	Max capacity of zone at port	Input	Real	4
S_{in}	Qty. of crude arriving at port	Input	Real	$4 \times D$
Q_0	Initial qty. of crude at refinery	Input	Real	5
$Q_{\max}$	Max capacity of zone at refinery	Input	Real	5
$Q_{\min}$	Min capacity of zone at refinery	Input	Real	5
C_1	Rate of processing at CDU1	Input	Real	Scalar
C_2	Rate of processing at CDU2	Input	Real	Scalar
is_{ARL}	Indicates arrival of ARL at port	Input	Binary	D
w	Mode of operation of CDU	Variable	Binary	$3 \times D$
s	Mode-switch at CDU	Variable	Binary	D
u	Transfer quantity through PRL	Variable	Real	$5 \times D$
a	Min. PRL transfer indicator	Variable	Binary	$5 \times D$
g	Packets of ARL	Variable	Integer $\in \{0, 1, \dots, 4\}$	D
O2O	Qty. of O-type crude being used as O-type	Variable	Real	D
N2O	Qty. of N-type crude being used as O-type	Variable	Real	D
O2N	Qty. of O-type crude being used as N-type	Variable	Real	D
N2N	Qty. of N-type crude being used as N-type	Variable	Real	D
s_f	Daily shortfall in production	Variable	Real	$5 \times D$

at maximum throughput, s_f assumes non-zero value and can be used directly to estimate total shortfall in production.

The next set of constraints requires that CDU1 cannot operate in the mode L immediately after operating in the mode N. The next of constraints enforce contiguity of mode of operations, i.e., if the CDUs operate in each mode, they continue to do so for a minimum of MUT days. The next set of constraints requires to transfer at least 2.22 k m^3 crude of a specific type continuously through the PRL. As shown in Fig. 1, the refinery side is equipped with only one tank each for crude types L and M. It is desirable that if a crude block is transferred to one of these tanks, it must be rested at least for a day (dewatering constraint) before it can be consumed by the CDU1. This is ensured through the final set of constraint equations in (OptPlan-MILP). Since, $u[i,d] \geq 0$ and $w[i,d] \in \{0,1\}$ for $i \in \{1,2\}$, the top-2 constraints among the final set of constraints ensure that whenever $w[i,d]$ is 1, i.e., the CDU1 is being operated in the ith-mode, there is no flow into the corresponding refinery tank on the same day. Similarly, the next set of constraints ensure that whenever $w[i,d]$ is 1, there is no flow into the corresponding refinery tank on the previous day, too, thereby allowing at least a day for the crude to rest in the refinery tanks (see Box I).

4.2. MILP for optimal crude blending

The (OptPlan-MILP) is concerned with only zone-level decision making given the mode of operation at the CDUs. These zones are obtained by merging tanks tied to storing crude of the same type.

$$
\begin{aligned}
&\text{minimize} \quad \sum_{d=1}^{D} s[d] + \sum_{d=1}^{D} \mathrm{N2O}[d] + \sum_{d=1}^{D} \mathrm{O2N}[d] + 20 \sum_{d=1}^{D}\sum_{j=1}^{5} s_f[j,d],\\
&\text{s.t.} \quad \boxed{\sum_{j=1}^{5} u[j,d] \le P_{\max}[d]}\\
&\boxed{\sum_{j=1}^{3} w[j,d] = 1}\\
&\boxed{\begin{array}{l} \mathrm{O2N}[d] + \mathrm{N2N}[d] + 4.44g[d] = u[3,d] + u[4,d] \\ \mathrm{N2O}[d] + \mathrm{O2O}[d] = u[5,d] \end{array}}\\
&\boxed{\begin{array}{ccccc}
S_{\min}[i] & \le & S_0[i] + \sum_{k=1}^{d}\left\{S_{\text{in}}[i,k] - u[i,k]\right\} & \le & S_{\max}[i], \quad i \in \{1,2\} \\
S_{\min}[3] & \le & S_0[3] + \sum_{k=1}^{d}\left\{(1-\mathrm{is}_{\mathrm{ARL}})\cdot S_{\text{in}}[3,k] - \mathrm{N2O}[k] - \mathrm{O2O}[k]\right\} & \le & S_{\max}[3] \\
0 & \le & \sum_{k=1}^{d}\left\{\mathrm{is}_{\mathrm{ARL}}\cdot S_{\text{in}}[3,k] - 4.44g[k]\right\} & & \\
S_{\min}[4] & \le & S_0[4] + \sum_{k=1}^{d}\left\{S_{\text{in}}[4,k] - u[5,k]\right\} & \le & S_{\max}[4]
\end{array}}\\
&\boxed{\begin{array}{ccccc}
Q_{\min}[i] & \le & Q_0[i] + \sum_{k=1}^{d}\left\{u[i,k] + s_f[i,k] - C_1 w[i,k]\right\} & \le & Q_{\max}[i], i \in \{1,2,3\} \\
Q_{\min}[4] & \le & Q_0[4] + \sum_{k=1}^{d}\left\{u[4,k] + s_f[4,k] - C_2(1-w[3,k])\right\} & \le & Q_{\max}[4] \\
Q_{\min}[5] & \le & Q_0[5] + \sum_{k=1}^{d}\left\{u[5,k] + s_f[5,k] - C_2 w[3,k]\right\} & \le & Q_{\max}[5]
\end{array}}\\
&\boxed{\begin{array}{ccll}
s[d] & \ge & w[1,d] + w[3,d+1] - 1, & d \in \{1,2,\ldots,D-1\} \\
1 & \ge & w[1,d+1] + w[3,d] - 1, & d \in \{1,2,\ldots,D-1\}
\end{array}} \qquad \text{(OptPlan-MILP)}\\
&\boxed{\begin{array}{ccll}
\sum_{k=1}^{\mathrm{MUT}} w[3,k] & \ge & \mathrm{MUT}\cdot w[3,1] & \\
\sum_{k=d}^{d+\mathrm{MUT}-1} w[3,k] & \ge & \mathrm{MUT}\cdot (w[3,d] - w[3,d-1]), & d \in \{2,\ldots,D-\mathrm{MUT}+1\} \\
\sum_{k=1}^{\mathrm{MDT}} (1-w[3,k]) & \ge & \mathrm{MDT}\cdot (1-w[3,1]) & \\
\sum_{k=d}^{d+\mathrm{MDT}-1} (1-w[3,k]) & \ge & \mathrm{MDT}\cdot (w[3,d-1] - w[3,d]), & d \in \{2,\ldots,D-\mathrm{MDT}+1\}
\end{array}}\\
&\boxed{\begin{array}{rcl}
u[i,d] - M\cdot a[i,d] & \le & 0 \\
u[i,d] & \ge & 2.22\cdot a[i,d]
\end{array}}\\
&\boxed{\begin{array}{rcl}
u[1,d] - M(1-w[1,d]) & \le & 0 \\
u[2,d] - M(1-w[2,d]) & \le & 0 \\
u[1,d-1] - M(1-w[1,d]) & \le & 0 \\
u[2,d-1] - M(1-w[2,d]) & \le & 0
\end{array}}
\end{aligned}
$$

Box I.

However, each crude type is further categorized into various sub-types with slightly different properties. Consequently, the properties of the crude of the same type stored across different tanks also differ. These properties include the sulphur and metal contents, the overhead (OVH) content and the atmospheric residue yield (AtRes). The properties of the crude play significant role in determining the throughput at the CDUs. If the crude blend being processed inside the CDUs does not meet the hydraulic constraints, $\mathrm{OVH}_{\max}$ and $\mathrm{AtRes}_{\max}$, the throughput must be reduced resulting in shortfall in production. Additionally, it is required that the crude blend meets the mode-level operational constraints specified in terms of sulphur content ($\mathrm{S}_{\max}$) and metal contents ($\mathrm{Ni}_{\max}$, $\mathrm{V}_{\max}$, $\mathrm{Fe}_{\max}$). Given the daily demand of crude of a specific type to be transferred through the PRL, the task of the planner is to create a crude blend by drawing crude from multiple tanks of the corresponding zone, such that the reduction in throughput is minimized.

To this end, we assume that there are N_t tanks available for the crude to be drawn from. Their initial level is denoted by T_0, while $T_{\min}$ represents their minimum holding capacities. The properties of crude in

Table 2
Description of symbols for CrudeBlend-MILP.

Symbol	Description	Type	Sub-type	Size
$\mathrm{Ni}_{\mathrm{max}}$	Max Nickel content	Input	Real	Scalar
$\mathrm{V}_{\mathrm{max}}$	Max Vanadium content	Input	Real	Scalar
$\mathrm{Fe}_{\mathrm{max}}$	Max Iron content	Input	Real	Scalar
$\mathrm{S}_{\mathrm{max}}$	Max Sulphur content	Input	Real	Scalar
$\mathrm{OVH}_{\mathrm{max}}$	Max overhead content	Input	Real	Scalar
$\mathrm{AtRes}_{\mathrm{max}}$	Max Atm. residue yield	Input	Real	Scalar
N_t	# of tanks	Input	Integer	Scalar
T_{min}	Min tank levels	Input	Real	N_t
T_0	Initial tank levels	Input	Real	N_t
T_{OVH}	Overhead content of crude in each tank	Input	Real	N_t
T_{AtRes}	Atm. residue yield of crude in each tank	Input	Real	N_t
T_{Ni}	Nickel content of crude in each tank	Input	Real	N_t
T_{V}	Vanadium content of crude in each tank	Input	Real	N_t
T_{Fe}	Iron content of crude in each tank	Input	Real	N_t
T_{S}	Sulphur content of crude in each tank	Input	Real	N_t
D_0	Total crude demand	Input	Real	Scalar
q_{OVH}	Overhead content of the blend	Variable	Real	Scalar
q_{AtRes}	Atm. residue yield of the blend	Variable	Real	Scalar
f	Fraction of demand from each tank	Variable	Real	N_t
δ	Does blend property exceed max OVH/AtRes?	Variable	Binary	2
θ	Cost of creating blend	Variable	Real	2

each tank is depicted by the tuple $(\mathrm{T}_{\mathrm{AtRes}}, \mathrm{T}_{\mathrm{OVH}}, \mathrm{T}_{\mathrm{Ni}}, \mathrm{T}_{\mathrm{V}}, \mathrm{T}_{\mathrm{Fe}}, \mathrm{T}_{\mathrm{S}})$. We use D_0 to depict the total crude demand to be transferred through the PRL obtained by solving the (OptPlan-MILP). The planner must decide on the fraction f of the total demand that is supplied by each of the tanks. The reduction in throughput can occur due to crude blend exceeding the CDU (a) OVH or (b) AtRes. We use binary variable δ to capture these events, where the first index corresponds to the OVH event, and the second index corresponds to the AtRes event. The objective is to minimize the total cost θ of creating the crude blend. The cost is zero if the blend meets the hydraulic limits. In case it is impossible to meet the hydraulic limits without reducing the throughput, minimizing the cost is synonymous to penalizing the reduction in throughput. Table 2 summarizes all the above input and decision variables.

As stated earlier, the objective of (CrudeBlend-MILP) is to minimize the total cost of creating blend subject to hydraulic and operational constraints. The first constraint requires that the sum of all fractions is unity. The second set of constraints ensure that the tanks cannot be emptied beyond the minimum holding limits and that the total crude amount withdrawn from a tank cannot exceed the initial tank level. The next set of constraints require that the sulphur and metal content of the crude blend do not exceed the maximum allowable limit. The next set of equations update the OVH and AtRes contents of the crude blend. The next set of constraints require that if the blend OVH content does not exceed $\mathrm{OVH}_{\mathrm{max}}$, the cost variable $\theta[1]$ is set to zero. In an event of blend OVH exceeding the $\mathrm{OVH}_{\mathrm{max}}$, the cost variable $\theta[1]$ is set to a positive value $\left(\frac{q_{\mathrm{OVH}}-\mathrm{OVH}_{\mathrm{max}}}{\mathrm{OVH}_{\mathrm{max}}}\right)$, which needs to be minimized. Note that the higher the ratio is above the maximum content, the higher is the cost ($\theta[1]$), and thus, even if there are violations to keep the blend overhead content within the maximum allowable limits imposed by hydraulic constraints, the optimization will try to keep it as close to the maximum value as possible. A similar set of constraints are formulated for the blend AtRes content. It must be noted that the cost variables (θ) are nonnegative.

$$\text{minimize} \quad \theta[1] + \theta[2],$$

$$\text{s.t.} \quad \boxed{\sum_{n=1}^{N_t} f[n] = 1}$$

$$\boxed{T_{\mathrm{min}}[n] \leq T_0[n] - D_0 \cdot f[n] \leq 0}$$

$$\boxed{\begin{array}{lllll} 0 & \leq & \sum_{n=1}^{N_t} f[n] \cdot T_{\mathrm{Ni}}[n] & \leq & \mathrm{Ni}_{\mathrm{max}} \\ 0 & \leq & \sum_{n=1}^{N_t} f[n] \cdot T_{\mathrm{V}}[n] & \leq & \mathrm{V}_{\mathrm{max}} \\ 0 & \leq & \sum_{n=1}^{N_t} f[n] \cdot T_{\mathrm{Fe}}[n] & \leq & \mathrm{Fe}_{\mathrm{max}} \end{array}}$$

$$\boxed{\begin{array}{lll} q_{\mathrm{OVH}} & = & \sum_{n=1}^{N_t} f[n] \cdot T_{\mathrm{OVH}}[n] \\ q_{\mathrm{AtRes}} & = & \sum_{n=1}^{N_t} f[n] \cdot T_{\mathrm{AtRes}}[n] \end{array}}$$

(CrudeBlend-MILP)

$$\boxed{\begin{array}{rcl} \theta[1] & \leq & M \cdot \delta[1] \\ \theta[1] + M(1-\delta[1]) & \geq & \left(\dfrac{q_{\mathrm{OVH}} - \mathrm{OVH}_{\mathrm{max}}}{\mathrm{OVH}_{\mathrm{max}}}\right) \\ \theta[1] - M(1-\delta[1]) & \leq & \left(\dfrac{q_{\mathrm{OVH}} - \mathrm{OVH}_{\mathrm{max}}}{\mathrm{OVH}_{\mathrm{max}}}\right) \end{array}}$$

$$\boxed{\begin{array}{rcl} \theta[2] & \leq & M \cdot \delta[2] \\ \theta[2] + M(1-\delta[2]) & \geq & \left(\dfrac{q_{\mathrm{AtRes}} - \mathrm{AtRes}_{\mathrm{max}}}{\mathrm{AtRes}_{\mathrm{max}}}\right) \\ \theta[2] - M(1-\delta[2]) & \leq & \left(\dfrac{q_{\mathrm{AtRes}} - \mathrm{AtRes}_{\mathrm{max}}}{\mathrm{AtRes}_{\mathrm{max}}}\right) \end{array}}$$

Note that with the proposed hierarchical framework, there is no need to estimate the blending properties of mixtures travelling through the system. The properties are always accounted as given parameters, and the constraints are concerned with keeping the Sulphur and metal contents below the maximum limits. At the same time that the resulting blend is forced to meet the hydraulic constraints as much as possible.

4.3. MILP for optimal tank receipt and withdrawal

It is required to process type-N crude daily. This requirement translates to maintaining adequate supply of type-N crude in the refinery tanks — RN13 and RN14 (for CDU1), and RN21 and RN22 (for CDU2). Since the tanks cannot be both filled and emptied at the same time, adequate supply of crude to the CDUs can only be managed by ensuring that at any moment, only one of the tanks is used for storage, while the other tank is used to supply crude to the CDU. When the tank supplying the crude gets emptied or the tank being used to store crude gets nearly full, the planner must optimally switch the roles of the two tanks. In our work, we automate the process of switching the roles of tanks using another MILP formulation (see (OptAlloc-MILP)).

Let binary variables δ_{in} and δ_{out} represent the events when the first tank is supplied with the crude type N and the crude is withdrawn from it, respectively. Consequently, $(1-\delta_{\mathrm{in}})$ and $(1-\delta_{\mathrm{out}})$ represent the corresponding decision variables for the second tank. Let $\{1, \dots, H\}$ represent the time-discretization of an entire day allocated towards processing type-N crude. In our implementation, we have worked with a time-step of 3 min, i.e., $H = 480$. We use r_1 and r_2 to represent states (tank levels) of the two tanks. The binary input $\mathrm{is}_{\mathrm{PRL}}$ indicates whether

$$
\begin{aligned}
&\text{minimize} \quad 0, \\
&\text{s.t.} \quad \boxed{\delta_{\text{in}}[h] + \delta_{\text{out}}[h] = 1, \quad h \in 1, 2, \ldots, H} \\
&\boxed{\begin{array}{lcl}
r_1[1] & = & R_{1,0} + f_{\text{in}} \cdot \delta_{\text{in}}[1] \cdot \text{is}_{\text{PRL}}[1] - f_{\text{out}} \cdot \delta_{\text{out}}[1] \\
r_1[h] & = & r_1[h-1] + f_{\text{in}} \cdot \delta_{\text{in}}[h] \cdot \text{is}_{\text{PRL}}[h] - f_{\text{out}} \cdot \delta_{\text{out}}[h], \quad h \in \{2, \ldots, H\} \\
r_2[1] & = & R_{2,0} + f_{\text{in}} \cdot (1 - \delta_{\text{in}}[1]) \cdot \text{is}_{\text{PRL}}[1] - f_{\text{out}} \cdot (1 - \delta_{\text{out}}[1]) \\
r_2[h] & = & r_2[h-1] + f_{\text{in}} \cdot (1 - \delta_{\text{in}}[h]) \cdot \text{is}_{\text{PRL}}[h] - f_{\text{out}} \cdot (1 - \delta_{\text{out}}[h]), \quad h \in \{2, \ldots, H\}
\end{array}} \\
&\boxed{\begin{array}{lcll}
R_{1,\min} & \leq & r_1[h] & \leq R_{1,\max}, \quad h \in \{1, \ldots, H\} \\
R_{2,\min} & \leq & r_2[h] & \leq R_{2,\max}, \quad h \in \{1, \ldots, H\}
\end{array}} \qquad \text{(OptAlloc-MILP)}
\end{aligned}
$$

Box II.

Table 3
Description of symbols for OptAlloc-MILP.

Symbol	Description	Type	Sub-type	Size
H	# of indices in a day	Input	Integer	Scalar
is_{PRL}	Indicates if PRL is active	Input	Binary	H
$R_{1,0}$	Tank-1 initial level	Input	Real	Scalar
$R_{2,0}$	Tank-2 initial level	Input	Real	Scalar
$R_{1,\min}$	Tank-1 minimum level	Input	Real	Scalar
$R_{2,\min}$	Tank-2 minimum level	Input	Real	Scalar
$R_{1,\max}$	Tank-1 maximum level	Input	Real	Scalar
$R_{2,\max}$	Tank-2 maximum level	Input	Real	Scalar
f_{in}	Flowrate into tanks	Input	Real	scalar
f_{out}	Flowrate out of tanks	Input	Real	scalar
r_1	Tank-1 level	Variable	Real	H
r_2	Tank-2 level	Variable	Real	H
δ_{in}	Indicates if tank-1 is fed	Variable	Binary	H
δ_{out}	Indicates if tank-1 is emptied	Variable	Binary	H

the PRL is active at a given time instant and supplies the N-type crude. The various input and decision variables are summarized in Table 3.

(OptAlloc-MILP) describes the MILP formulation for the optimal tank receipt and withdrawal for the N-type crude. Interestingly, optimizing tank allocation (for receipt and withdrawal) is largely a satisfiability problem, the objective function is simply set to 0. The first constraint ensures that a tank cannot be simultaneously filled and emptied. The next set of constraints impose mass-balance requirements on the tank levels, while the last set of constraints require that the tank levels are within the required minimum and maximum tank capacities (see Box II).

A note on feasibility: Before decomposing the original optimization problem into smaller sub-problems, we conduct a preliminary feasibility check. This includes:

(1) For a 30-day period, the sum of total supply for each crude type (specifically for types L and M) and the current tank holding must not exceed the total demand for corresponding crude types. This can be ensured in a constant time (non-negligible).

(2) There are additional instances of potential infeasibilities that arise when enforcing a maximum sulphur/metal content for a potential blend, especially when all incoming crude has a higher sulphur/metal content than the allowable threshold. In these cases, it is advised to increase the threshold beyond the normal value.

(3) Finally, there is a possibility of encountering infeasibility when either the incoming crude cannot be accommodated or the required demand cannot be met. In the former situation, this would lead to additional demurrage costs, while in the latter case, introducing a fictitious shortfall variable enables us to ensure the solution remains feasible.

It is important to emphasize that the problem of maximizing throughput is always feasible, provided that the constraints on tank holding capacity and the sulphur and metal content of crude blend are satisfied. The other requirements, such as hydraulic constraints, availability of shared pipeline, continuous supply of sufficient crude to meet refinery demand, and dewatering constraints, only impact the total throughput to the refineries. Hence, aside from the basic feasibility check, our approach always provides a solution that is feasible, and its quality is determined by the total shortfall in production.

5. Experiments

We now benchmark the proposed hierarchical framework on the refinery system shown in Fig. 1. As stated previously, the refinery system is a scaled version of a real industrial refinery, subjected to various physico-chemical, hydraulic and operational constraints (described in detail in Section 2). The section also lists the numerical values of different problem parameters, such as the maximum flow-rate through PRL, incoming flow-rate at the port, and maximum flow-rates at the CDUs (see Table 4). The capacities of individual tanks have been depicted in Fig. 1. The hierarchical framework is implemented in Python using Gurobi (Gurobi Optimization, LLC, 2022) on a 16 GB Core-i7 2.8 GHz CPU.

For evaluation, we consider a realistic scenario for a 30-day period, and compare the performance of the proposed hierarchical method against the business rules adopted by the expert human operator. Both the incoming crude schedule, as well as the initial tank levels and the properties of the starting crude blend in these tanks is directly adopted from the historic data. The arrival schedule and the corresponding crude properties are shown in Tables 5 and 6, respectively. Recall that the L and M-type crude are processed only inside CDU1, and hence the overhead (OVH) and the atmospheric residue yield (AtRes) values are depicted only for CDU1. Moreover, there are no constraints on the metal content for processing the L-type crude. Likewise, the M and N-type crude do not have any constraints on the sulphur content. O-type is the lowest grade crude and there are no constraints on the sulphur and metal content. However, in the absence of availability of sufficient N-type crude, it is possible to blend small quantities of O-type crude with N-type crude (see (OptPlan-MILP)) to an extent that it does not violate the metallicity constraints required for processing the N-type crude (see Table 7). .

Subject to various operational, hydraulic and physico-chemical constraints, and the incoming crude arrival schedule, we generate solutions using both the business rules (see Fig. 5a) and the proposed hierarchical framework (Figs. 5b and c). Recall that the primary objective is to

Table 4
Refinery management: Problem parameters.

Parameter	Description	Value
$OVH1_{max}$	Maximum Overhead (CDU1)	250
$OVH2_{max}$	Maximum Overhead (CDU2)	279.17
$AtRes1_{max}$	Maximum Atmospheric Residue Yield (CDU1)	420.83
$AtRes2_{max}$	Maximum Atmospheric Residue Yield (CDU2)	650
S_{max}-(L)	Maximum Sulphur content (L-type)	1.05
$(Ni_{max}, Va_{max}, Fe_{max})$-(M)	Maximum metal content (M-type)	(10, 14, 8)
$(Ni_{max}, Va_{max}, Fe_{max})$-(N)	Maximum metal content (N-type)	(20, 60, 10)
C_1	Maximum flow-rate into CDU1	0.22 km^3/h
C_2	Maximum flow-rate into CDU2	0.32 km^3/h
C_{PRL}	Maximum flow-rate through PRL	1 km^3/h
C_{in}	Maximum flow-rate into port tanks	1.33 km^3/h

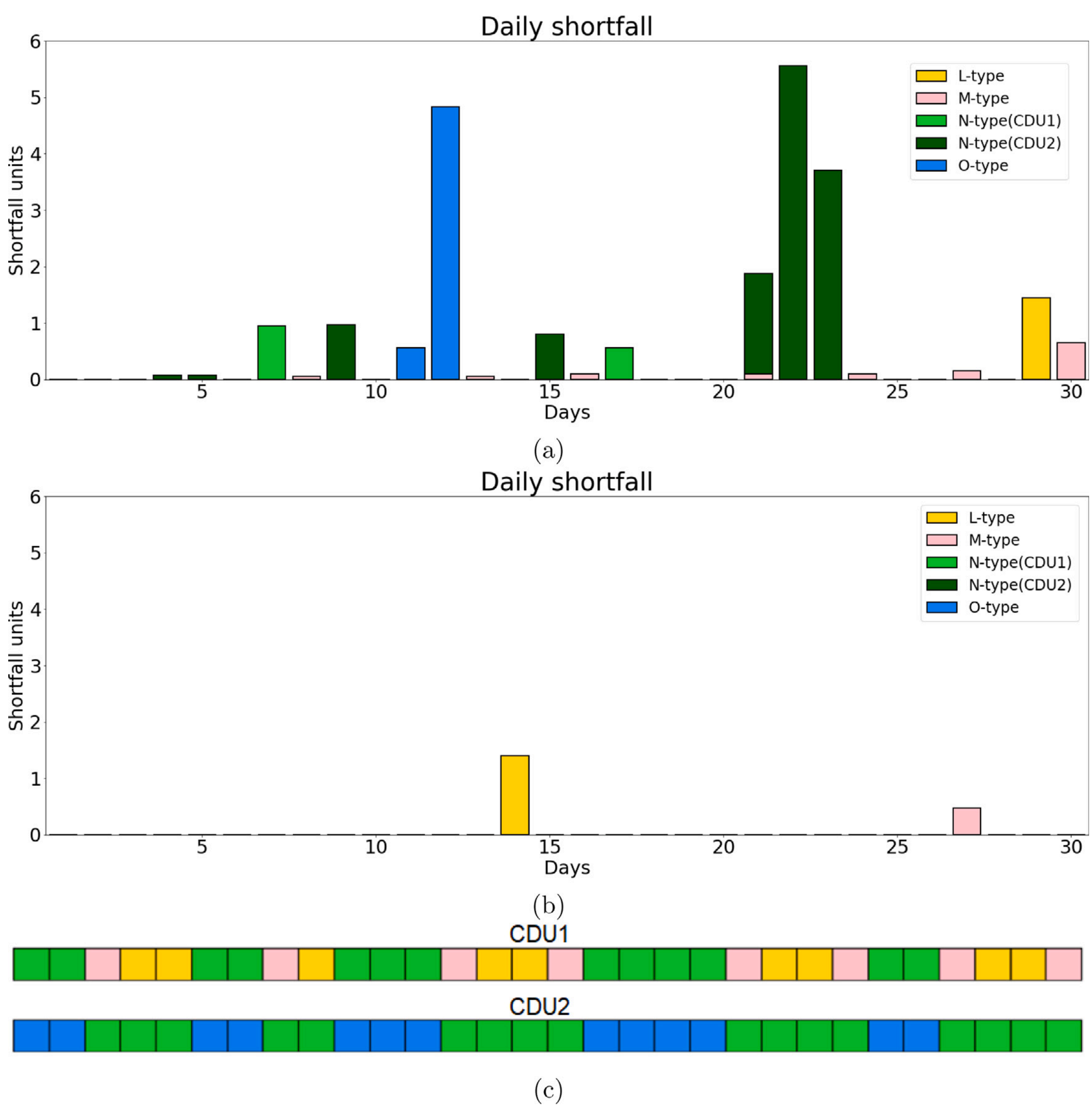

Fig. 5. Comparison of (a) business rules, and (b) the proposed hierarchical method for minimizing the total shortfall in production. (c) The daily production plan generated by the hierarchical framework satisfies all the mode-specific operational constraints. The proposed framework results in an improvement of $\sim 91.83\%$ in throughput maximization over the business rules.

Table 5
Daily arrival schedule.

Day	Crude	Quantity (km^3)	Day	Crude	Quantity (km^3)
1	–	–	16	L-2	16.7
2	L-1	22.2	17	ARL	17.8
3	M-1	11.1	18	–	–
4	ARL	17.8	19	M-1	11.1
5	O-1	11.1	20	O-1	24.4
6	–	–	21	N-2	24.4
7	N-1	22.2	22	–	–
8	–	–	23	ARL	17.8
9	ARL	17.8	24	–	–
10	O-1	24.4	25	ARL	17.8
11	M-2	11.1	26	O-1	24.4
12	–	–	27	M-2	22.2
13	ARL	17.8	28	ARL	17.8
14	L-2	16.7	29	–	–
15	O-1	11.1	30	–	–

Table 6
Properties of incoming crude.

Crude	T_{OVH1}	T_{AtRes1}	T_{OVH2}	T_{AtRes2}	(S, Ni, Va, Fe)
L-1	167	423	–	–	(1.0, –, –, –)
L-2	139	142	–	–	(0.64, –, –, –)
L-3	258	277	–	–	(0.57, –, –, –)
M-1	292	222	–	–	(–, 2.6, 4.8, 6.9)
M-2	187	399	–	–	(–, 10.1, 17.5, 2)
N-1	26	485	37.7	703.25	(–, 7.3, 22.3, 12.1)
N-2	131	494	190	716.3	(–, 9.6, 41.9, 2)
ARL	142	450	205.9	652.5	(–, 11.6, 33.9, 2)
O-1	133	499	192.8	723.6	(–, 29.8, 107.2, 8.1)
O-2	262	332	379.9	481.4	(–, 8.8, 26.1, 3.1)

minimize the total shortfall in crude production. The business rules (also used by the expert human operator) result in a total shortfall of 22.52 km^3 over a period of 30 days. On the other hand, the proposed hierarchical framework is capable of reducing the shortfall to just 1.84 km^3 over the same period, resulting in an improvement of nearly 91.83% over the baseline method for throughput maximization. In addition, the total runtime for the proposed hierarchical MILP framework is only 57 s, making it extremely scalable and amenable to real-time scheduling. The planning resolution is set at three minutes. This means that each day is divided into 480 intervals of three minutes each. Decisions such as which crude oil should be stored in which tank, which crude oil should be transferred through the PRL to which refinery tank, and what type of blend should be made at a given time, are updated at a resolution of three minutes.

Besides the constraints on processing of crude in CDUs and storing them in tanks, not every schedule is operationally feasible. For instance, an L-type mode of operation should not be preceded by N-type mode of operation. Fig. 5c shows the daily mode of operation for the two CDUs. It can be observed that the schedules are contiguous and all L-type mode of operations are preceded by M-type mode of operation. The schedules for CDU1 and CDU2 are complementary to each other, as mandated by the problem specifications. Fig. 6 depicts the daily flow (in km^3) of crude of different types through the pipeline PRL. As expected, the net flow amount varies daily due to the unequal daily maximum flow capacities through the PRL. This heterogeneity is a result of the PRL's availability for flow transfer being limited to only the first 19.8 h of every 38 h window.

We have also validated our approach on other realistic test scenarios, and obtained very similar analysis. While the detailed results have been excluded from the main text for brevity, we have included them in the accompanying supplementary information. The results across ten challenging test scenarios are summarized in Table 8. Our algorithm demonstrates a substantial reduction in shortfall, with scenarios A3 and A5 showing no shortfall at all. Additionally, the average shortfall reduction across all scenarios is remarkably high at 94.30%, considering the revenue gained by maximizing throughput. It can also be noted that, using our approach, the net shortfall in other test scenarios (with the exception of scenarios A4 and A10) is almost insignificant. If one takes a close look, the arrival schedule in scenario A4 (presented in the supplementary material) does not feature any crude arrival events during days 17–23, which may have resulted in a shortage of N-type crude over the 30-day period. Similarly, the arrival events of M-type crude are very limited in scenario A10, leading to non-negligible shortfall in production of M-type crude. Even with the difficulties posed by the demanding crude delivery schedule, our framework still significantly outperforms the schedule generated by the human expert in terms of throughput maximization.

6. Conclusion & future work

We propose a scalable framework for scheduling of complex operations in a refinery system subjected to various constraints. The entire problem of refinery management is decomposed into three smaller sub-problems, each of which are handled using efficient MILP-based framework. This hierarchical approach allows the problem to be solved very efficiently (in less than a minute) and produce results that significantly outperform the schedules generated by a human expert on the same realistic scenario. In future, we look to expand the scope of this work to address refinery management under stochasticity in crude arrival schedule using robust optimization framework (Bertsimas et al., 2011). In addition, a refinery may be under maintenance due to unforeseen circumstances leading to disruption in handling of crude as suggested otherwise by the MILP framework. Since the framework is amenable to real-time planning, we look to test the efficacy of our framework in dealing with such extreme situations. Finally, we would like to incorporate events of scheduled maintenance into our optimization framework itself.

CRediT authorship contribution statement

Mayank Baranwal: Methodology — hierarchical MILP framework, Evaluation, Writing and proof reading. **Mayur Selukar:** Methodology — business rules, Evaluation. **Rushi Lotti:** Methodology — business rules, Evaluation, Literature review. **Aditya A. Paranjape:** Literature review, Writing and proof reading, Conceptualization. **Sushanta Majumder:** Problem formulation and definition, Conceptualization. **Jerome Rocher:** Conceptualization, Proof reading.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data has been included in the main text itself.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compchemeng.2023.108242.

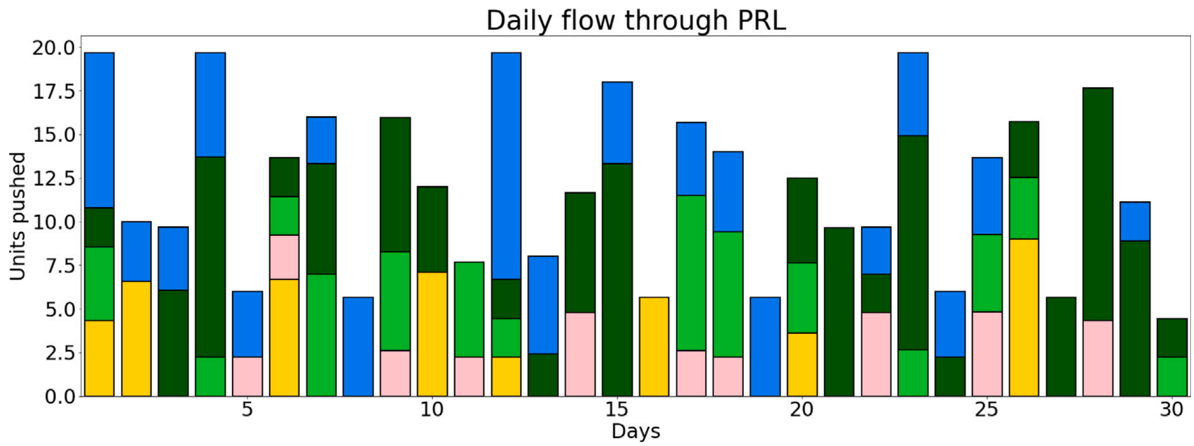

Fig. 6. Amount of crude in km^3 pushed through PRL for each crude-type. Colours yellow, pink, green, lime-green and blue depict crude of types L, M, N(CDU1), N(CDU2) and O, respectively.

Table 7
Initial tank level and blend properties.

Tank	Initial level (km^3)	T_{OVH1}	T_{AtRes1}	T_{OVH2}	T_{AtRes2}	S	(Ni, Va, Fe)
PL01	11.0	51.6	55.4	–	–	0.11	–
PL02	3.6	129	138.5	–	–	0.28	–
PL03	10.8	116.9	296.1	–	–	0.7	–
PM01	10.1	137.4	238	–	–	–	(1.5, 2.58, 0)
PM02	10.1	87.6	66.6	–	–	–	(.78, 1.44, 2.07)
PM03	8.9	20.5	383.2	29.8	555.6	–	(5.77, 17.62, 9.56)
PN01	21.6	12.7	237.6	18.5	344.6	–	(3.58, 10.93, 5.93)
PN02	16.2	87.6	66.6	–	–	–	(.78, 1.44, 2.07)
PO01	21.6	53.2	199.6	77.1	289.4	–	(11.92, 42.88, 3.24)
PO02	8.6	133	499	192.8	723.6	–	(29.8, 107.2, 8.1)
RL11	3.6	51.6	55.4	–	–	0.11	–
RM12	5.8	37.4	79.8	–	–	–	(2.02, 3.5, .4)
RN13	5.8	8.06	150.4	–	–	–	(2.26, 6.91, 3.75)
RN14	5.8	131	166	–	–	–	(4.4, 13.05, 1.55)
RN21	3.7	–	–	82.4	261	–	(4.64, 13.56, 0.8)
RN22	2.3	–	–	18.8	351.6	–	(3.65, 11.15, 6.05)
RO24	11.6	–	–	192.6	722.5	–	–
RO23	7.3	–	–	190.0	240.7	–	–

Table 8
Comparison between ours and the expert operator's schemes for reduction in shortfall.

Scenario	Shortfall (Ours) (km^3)	Shortfall (Expert) (km^3)	$\left(\frac{\text{Expert} - \text{Ours}}{\text{Expert}}\right) \times 100\%$
A1	1.84	22.52	91.83
A2	0.29	57.83	99.50
A3	0	42.85	100.0
A4	21.77	90.27	75.88
A5	0	74.03	100.0
A6	0.29	44.75	99.35
A7	0.29	33.63	99.14
A8	0.29	38.61	99.25
A9	0.49	38.81	98.74
A10	13.47	64.99	79.27

References

Andrade, T., Ribas, G., Oliveira, F., 2016. A strategy based on convex relaxation for solving the oil refinery operations planning problem. Ind. Eng. Chem. Res. 55 (1), 144–155.

Baker, T.E., Lasdon, L.S., 1985. Successive linear programming at exxon. Manage. Sci. 31 (3), 264–274.

Bertsimas, D., Brown, D.B., Caramanis, C., 2011. Theory and applications of robust optimization. SIAM Rev. 53 (3), 464–501.

Biegler, L.T., 2010. Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes. SIAM.

Boukouvala, F., Li, J., Xiao, X., Floudas, C.A., 2016. Data-driven modeling and global optimization of industrial-scale petrochemical planning operations. In: 2016 American Control Conference. ACC, IEEE, pp. 3340–3345.

Castro, P.M., 2016. Source-based discrete and continuous-time formulations for the crude oil pooling problem. Comput. Chem. Eng. 93, 382–401.

Castro, P.M., Grossmann, I.E., Zhang, Q., 2018. Expanding scope and computational challenges in process scheduling. Comput. Chem. Eng. 114, 14–42. http://dx.doi.org/10.1016/j.compchemeng.2018.01.020.

Demirhan, C.D., Boukouvala, F., Kim, K., Song, H., Tso, W.W., Floudas, C.A., Pistikopoulos, E.N., 2020. An integrated data-driven modeling & global optimization approach for multi-period nonlinear production planning problems. Comput. Chem. Eng. 141, 107007.

Gurobi Optimization, LLC, 2022. Gurobi Optimizer Reference Manual. URL https://www.gurobi.com.

Hou, J., Li, X., Sui, H., 2015. The optimization and prediction of properties for crude oil blending. Comput. Chem. Eng. 76, 21–26.

Hubbs, C.D., Li, C., Sahinidis, N.V., Grossmann, I.E., Wassick, J.M., 2020. A deep reinforcement learning approach for chemical production scheduling. Comput. Chem. Eng. 141, 106982.

Kallrath, J., 2002. Planning and scheduling in the process industry. OR Spectrum 24, 219–250.

Kallrath, J., 2005. Solving planning and design problems in the process industry using mixed integer and global optimization. Ann. Oper. Res. 140 (1), 339–373.

Koenig, W., 1963. The application of computers for refinery simulation (Refinery planning by linear programming). In: 6th World Petroleum Congress. OnePetro.

Kolodziej, S.P., Grossmann, I.E., Furman, K.C., Sawaya, N.W., 2013. A discretization-based approach for the optimization of the multiperiod blend scheduling problem. Comput. Chem. Eng. 53, 122–142.

Lee, H., Pinto, J.M., Grossmann, I.E., Park, S., 1996. Mixed-integer linear programming model for refinery short-term scheduling of crude oil unloading with inventory management. Ind. Amp Eng. Chem. Res. 35 (5), 1630–1641.

Li, J., Xiao, X., Boukouvala, F., Floudas, C.A., Zhao, B., Du, G., Su, X., Liu, H., 2016. Data-driven mathematical modeling and global optimization framework for entire petrochemical planning operations. AIChE J. 62 (9), 3020–3040.

Lima, C., Relvas, S., Barbosa-Póvoa, A., 2018. Stochastic programming approach for the optimal tactical planning of the downstream oil supply chain. Comput. Chem. Eng. 108, 314–336.

Mendez, C.A., Grossmann, I.E., Harjunkoski, I., Kaboré, P., 2006. A simultaneous optimization approach for off-line blending and scheduling of oil-refinery operations. Comput. Chem. Eng. 30 (4), 614–634.

Neiro, S.M., Pinto, J.M., 2004. A general modeling framework for the operational planning of petroleum supply chains. Comput. Chem. Eng. 28 (6–7), 871–896.

Paranjape, A.A., Baranwal, M., Wagle, S., Lotti, R., Majumder, S., Bullière, A.-L., 2022. Optimal schedule generation for single-channel crude transfer using a multi-model approach. Comput. Chem. Eng. 160, 107732.

Pinto, J.M., Joly, M., Moro, L.F.L., 2000. Planning and scheduling models for refinery operation. Comput. Chem. Eng. 24 (9–10), 2259–2276.

Shah, N., 2005. Process industry supply chains: Advances and challenges. Comput. Chem. Eng. 29 (6), 1225–1235.

Siamizade, M.R., 2019. Global optimization of refinery-wide production planning with highly nonlinear unit models. Ind. Eng. Chem. Res. 58 (24), 10437–10454.

Tong, K., Feng, Y., Rong, G., 2012. Planning under demand and yield uncertainties in an oil supply chain. Ind. Eng. Chem. Res. 51 (2), 814–834.

Uribe-Rodriguez, A., Castro, P.M., Gonzalo, G.-G., Chachuat, B., 2020. Global optimization of large-scale MIQCQPs via cluster decomposition: Application to short-term planning of an integrated refinery-petrochemical complex. Comput. Amp Chem. Eng. 140, 106883.

Wagle, S., Paranjape, A.A., 2020. Use of simulation-aided reinforcement learning for optimal scheduling of operations in industrial plants. In: 2020 Winter Simulation Conference. WSC, pp. 572–583. http://dx.doi.org/10.1109/WSC48552.2020.9383893.

Yang, Y., Vayanos, P., Barton, P.I., 2017. Chance-constrained optimization for refinery blend planning under uncertainty. Ind. Eng. Chem. Res. 56 (42), 12139–12150.

Zhang, L., Yuan, Z., Chen, B., 2021. Refinery-wide planning operations under uncertainty via robust optimization approach coupled with global optimization. Comput. Chem. Eng. 146, 107205.