Short-Term Scheduling of a Large-Scale Oil-Refinery Operations: Incorporating Logistics Details

Nikisha K. Shah and Marianthi G. Ierapetritou
Dept. of Chemical and Biochemical Engineering, Rutgers University, Piscataway, NJ 08854

DOI 10.1002/aic.12359
Published online August 31, 2010 in Wiley Online Library (wileyonlinelibrary.com)

Refineries are increasingly concerned with improving the scheduling of their operations to achieve better economic performances by minimizing quality, quantity, and logistics give away. In this article, we present a comprehensive integrated optimization model based on continuous-time formulation for the scheduling problem of production units and end-product blending problem. The model incorporates quantity, quality, and logistics decisions related to real-life refinery operations. These involve minimum run-length requirements, fill-draw-delay, one-flow out of blender, sequence-dependent switchovers, maximum heel quantity, and downgrading of better quality product to lower quality. The logistics giveaways in our work are associated with obtaining a feasible solution while minimizing violations of sequence-dependent switchovers and maximum heel quantity restrictions. A set of valid inequalities are proposed that improves the computational performance of the model significantly. The formulation is used to address realistic case studies where feasible solutions are obtained in reasonable computational time. © 2010 American Institute of Chemical Engineers *AIChE J,* 57: 1570–1584, 2011
Keywords: short-term scheduling, refinery, large scale, logistics

Introduction

The refinery operations scheduling problem involves decisions that are related to quantity, quality, and logistics. Quantity decisions include lot sizes for raw material, intermediate and product tanks inventories, amount of material moving between production units and storage tanks, etc., whereas quality decisions deal with obtaining finished products that meet specific quality requirements. Logistics constraints include policies and procedures for production operations that deal with allocating resources to operations, sequencing or ordering of different modes of operations, and determining the durations of operations.

Correspondence concerning this article should be addressed to M. G. Ierapetritou at marianth@soemail.rutgers.edu.

Refinery scheduling has received a lot of attention in literature. Jia and Ierapetritou[1] proposed a solution strategy that decomposes refinery operations into three suboperations: (1) the crude-oil unloading and blending, (2) the production unit operations, and (3) the product blending and lifting. Problem 1 involves the crude-oil unloading, blending, and inventory control; Problem 2 includes production unit scheduling; and Problem 3 consists of finished product blending and lifting. The crude-oil unloading problem has been extensively studied.[2–6] The complex crude-oil blend scheduling optimization problem is decomposed into logistics and quality subproblems by Kelly and Mann.[7,8] They used successive linear programing to solve the quality subproblem. Models based on continuous-time representation are proposed for refinery production unit scheduling problem.[9–11] The purpose of blend scheduling optimization problem is to find the best way of mixing different semifinished products that have been rectified during various refinery processes with some additives so as to produce final products that meet quality specifications and demand while minimizing cost. Glismann and Gruhn[12] proposed a decomposition technique based on first solving the nonlinear (NLP) quality optimization model and then solving a mixed-integer linear programing (MILP) model to optimize temporal and resource decisions. A MILP optimization model based on continuous-time representation, using unit-specific event points, and fixed blend recipe was developed by Jia and Ierapetritou.[1] They modeled multipurpose product tanks, but they do not include certain features such as sequence-dependent switchovers constraints, fill-draw-delay at product tank, and one-flow out of blender. An iterative procedure is proposed by Mendez et al.[13] to deal with variable recipe and nonlinear properties for different grades of products by replacing MINLP with sequential MILP formulations. They enforced prepared blend recipe whenever it is possible. In their model formulation, they did not consider multipurpose product tanks, fill-draw-delay, and minimum run-length requirement. There is extensive literature on the refinery blending problem using nonlinear optimization tools.[14–16]

Kelly[17] emphasized the importance of logistics details in refinery blending and delivery problem and proposed a decomposition of the blend scheduling problem into two subproblems, logistics and quality. The logistics subproblem considers only the quantity- and logistics-related variables and the problem constraints, whereas the quality subproblem considers product specifications, quantity constraints, and bounds. Their work is based on discrete-time representation, and their formulation includes many logistics details such as minimum run length, sequence-dependent changeovers, and fill-draw-delay. They observed that incorporating logistics details into scheduling problem can yield substantial improvements in efficiency and productivity.

The work referenced in the previous paragraphs has been limited to the off-line blending problem. However, because of economic pressure for reducing on-site storage inventory and safety concerns for reducing inventory of volatile materials, today, most products are made by blending components together in an in-line blender where the process consists of simultaneously adding several streams of intermediate components into a common header without using component storage tanks. With more product grades, stricter specifications, new government regulations, and fewer feasible blends, refineries face increasing challenges to maintain, let alone increase, profitability. What is needed is a comprehensive model for the entire refinery scheduling problem that addresses quality, quantity, and logistics issues in a unified, yet flexible approach. The comprehensive model is complex, hard to build and solve, and there is sparse work in literature in this area. Moro et al.[18] proposed a planning model for refinery diesel production where the emphasis is on blending relations. Pinto et al.[19] proposed a planning and scheduling model for refinery production and distribution operations. Their formulation is based on discretization of time, and the model includes features such as sequence-dependent transition cost of products within oil pipeline. Luo and Rong[20,21] developed a two-tiered decision-making hierarchical scheduling model for overall refinery. The upper level optimization model is based on discrete-time formulation, and it is used to determine sequencing and timing of operations modes and to decide the quantities of materials produced/consumed at each operations mode. The upper decisions level uses aggregated tanks storage capacity, whereas the lower level uses heuristics to obtain a detailed schedule. They consider multipurpose product tanks via an iterative procedure that allows to readjust aggregated tank capacity at the lower level by changing multipurpose tank service mode and then to recalculate corresponding optimal solution at the upper level. The logistics details are ensured through heuristics at the lower level. There are also several commercial tools available for refinery scheduling such as Aspen Petroleum Scheduler, Aspen Refinery Multi-Blend Optimizer, Honeywell's Production Scheduler, and Honeywell's Blend. Honeywell's Production Scheduler has a logistics solver to optimize logistics and quantity problems and a quality solver to solve quantity and quality problems. Most of the models used in these commercial tools are based on discrete-time formulation.

In this work, we present a comprehensive integrated optimization model for the production units and end-product blend scheduling problem that incorporates quantity, quality, and logistics decisions related to real-life refinery operations. The model is driven by the shipment plan and accounts for the tradeoffs between costs of keeping inventory and changing run modes. The goals of refinery operations scheduling are to maximize the profit and performance while minimizing the penalties subject to quantity, quality, and logistics giveaways and nonattainment.[22] The outline of this article is as follows. Section ''problem definition'' presents the problem definition, Section ''proposed model'' presents the mathematical formulation that is applied to a realistic case study example to illustrate the applicability of proposed model to large-scale model in Section ''results on the case studies,'' and the article concludes with Section ''summary.''

Problem Definition

The production scheduling determines the detailed schedule of each production unit and each demand order unloading for a short time period (typically 10 days–1 month) by taking into account the operational constraints of the plant. The schedule defines which products should be produced and which materials should be consumed in each time interval over a given small time horizon; hence, it defines which run mode to use and when to perform changeovers to meet the market needs satisfying the demand and product specifications. Large-scale scheduling problems arise frequently in oil refineries where the main objective is to assign sequence of tasks to processing units within certain time frame such that the demand of each product is satisfied before its due date while minimizing the cost or maximizing the total profit.

The refinery production system considered here is composed of raw material storage tanks, production units, blending units, intermediate tanks, and final product tanks. Each production unit is defined as a continuous processing element that transforms the input streams into several products according to the variable production recipe. For reasons of operating flexibility and cost effectiveness, refinery unit operations can generate a range of intermediate streams that are blended into finished products. For simplicity, we separate the final products into two groups: (a) products that are stored in tanks and (b) products that are not stored in tanks but are supplied to the market directly from production units. In this work, we limit the use of term "demand order" only for the first group of products and each demand order corresponds to only one kind of product because each multiproduct order can be decomposed into several single product orders. The characteristics of the problem considered in this article are given in detail in the next subsection.

Problem characteristics

The key information available for the refinery includes the following:

1. Maximum and minimum proportion of material produced or consumed at each production unit
2. Maximum and minimum production flow rates for each production unit
3. Maximum and minimum inventory capacities for each storage tank, identity of material type that each tank can service, and initial holdup in each tanks
4. Upper limit on the flows out of finished product tanks
5. Demand orders for Group A products and their delivery time windows
6. Total demand for Group B products
7. Maximum allowable heel quantity for multipurpose product tanks
8. Minimum run length for units and maintenance time for multipurpose tanks between run modes
9. Quality specifications limit on blend product properties
10. Available scheduling time horizon of 10 days

The goal of optimization is to determine:

1. The sequencing of tasks for production units
2. Each product pool that satisfies demand orders
3. Durations of tasks at production units and duration of unloading tasks from product pools
4. The inventory levels in component and product pools
5. Production rates for units and unloading rate for product pools
6. Composition of material produced and consumed

The problem is also restricted by a series of logistics details as follows:

1. At any given time, only one task can take place at production unit
2. Minimum run-length constraint for production units
3. Noncontiguous product order fulfillment for product in Group A
4. Blend unit can send product to multiple pool sequentially, not simultaneously
5. Product tanks cannot distribute and receive material at the same time
6. Fill-draw-delay restriction for product pools enforces certain amount of downtime on tanks after product loading event has taken place
7. Multipurpose tanks can store different types of materials over time, but only one type of material at any given time
8. Sequence-dependent switchover for multipurpose tanks where higher quality product is stored before lower grade products
9. Maximum heel requirement restriction does not allow product heel to exceed specified maximum heel quantity when the multipurpose tank is switched to a different mode
10. Downgrading of the higher grade product to lower grade product if necessary

Although a realistic case study is modeled, the following assumptions had to be made:

1. Unlimited supply of raw materials
2. Fixed recipe for crude distillation units (CDUs)
3. Constant blend components properties
4. Perfect mixing in the blender
5. Product tank cannot satisfy multiple demand orders simultaneously
6. Each demand order involves only one product
7. For production units, the amount of time required for run-modes change is neglected
8. Changeover times in multipurpose tanks from higher to lower grade product are negligible

Before presenting the mathematical model, a case study with realistic data provided by Honeywell Process Solutions is presented in the next section. The refinery produces diesel fuels, jet fuel, and components for gasoline production. This case study will be used to illustrate the applicability of the proposed model in Section "proposed model."

Case study description

The production process at Honeywell refinery consists of two blender units, 13 other processing units, and two nonidentical parallel CDUs that process two different type of crude oils. The schematic of production system is shown in Figure 1. There are two charging tanks for one CDU and a charging pipeline for another CDU. The CDUs concurrently transform crude oil into several distillation cuts. These distillation cuts from the CDUs are then sent to other production units for fractionation and reaction to produce blend components for finished products. The oil refinery under case study has the following production units: vacuum tower, coker conversion unit, continuous catalytic reforming process unit, isomax, fluid catalytic cracking (FCC) unit, penex, De-C5, and alkylation process unit. Current regulatory requirements to produce ultra-low-sulfur fuels require the use of hydrotreating technology. Thus, the refinery also includes three hydrodesulfurization (HDS) units: Naphtha HDS, Diesel HDS, and FCC HDS.

Honeywell refinery uses Jet blender and Diesel blender units to blend components produced by other production units to produce final products. Jet blender unit blends straight run jet, coker jet, and isomax jet streams to produce jet fuel, which can be stored in two product tanks. Diesel blender unit produces three different grades of fuel: CARB diesel, EPA diesel, and red-dye diesel. Light cycle oil, HDS diesel, and hydrocracked diesel are blended to produce these three grades of diesel products using three different run modes. There are two dedicated tanks for each grade of diesel products and one multipurpose tank that can service CARB and EPA diesel.

There is 6 h of cleaning or maintenance downtime when the multipurpose tank service switches from lower grade of diesel product to higher grade of product. This cleaning downtime is essential to remove any sulfur contamination

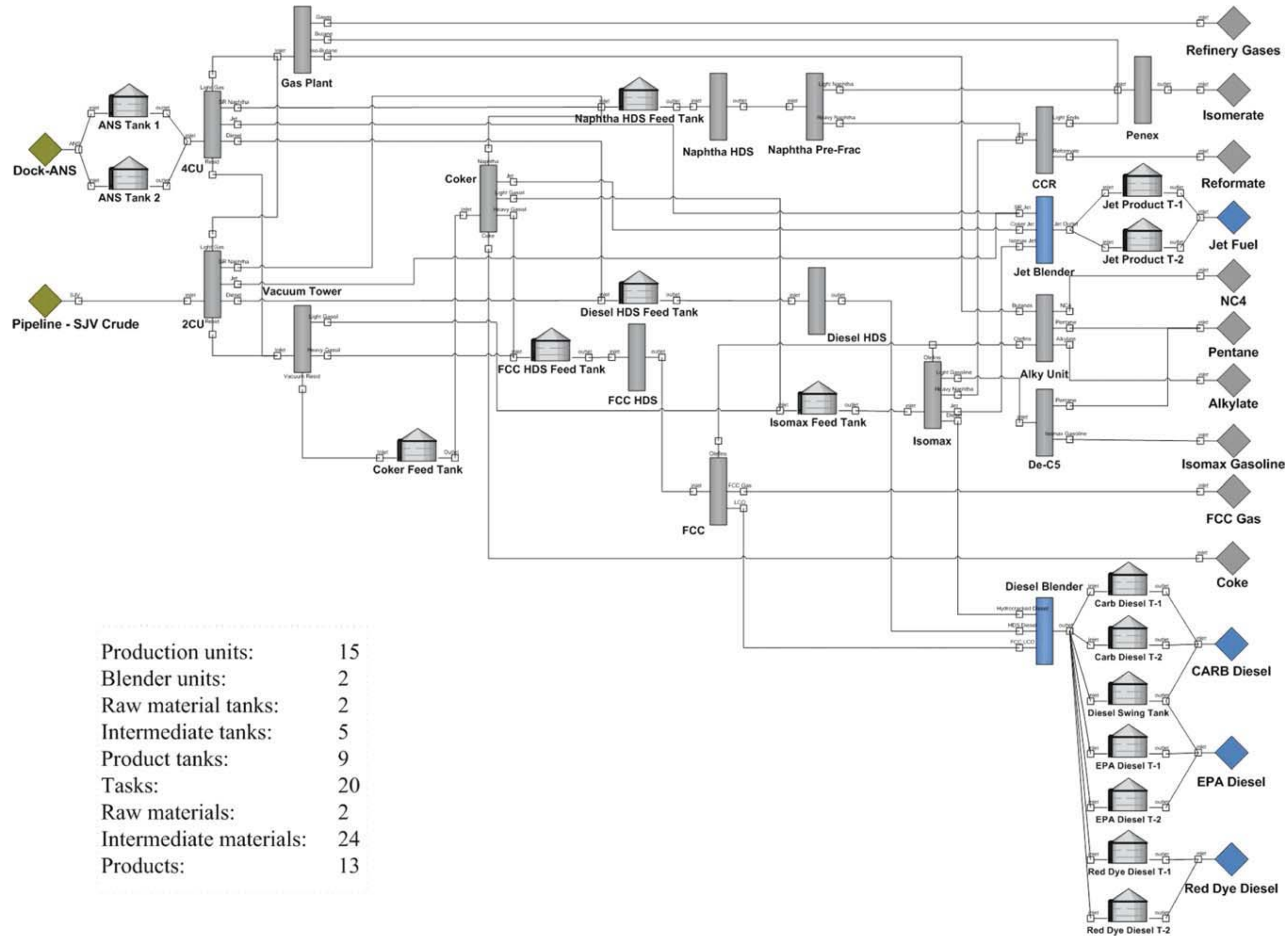

Figure 1. Schematic of the refinery production plant for reference example.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

present in the tank before low sulfur product is sent for storage. The product tank has 4 h of downtime called fill-draw-delay for certificate of analysis preparation and to let the product settle down and mix before is shipped to the market.

Proposed Model

In this section, we present the mathematical formulation for the refinery production scheduling problem based on continuous-time representation and the idea of unit-specific event points.[23,24] A state-task network representation introduced by Kondili et al.[25] is used to describe the refinery operations. The model involves material balance constraints, capacity constraints, demand constraints, quality constraints, logistics constraints, and setup constraints. Material balance constraints connect the amount of material at one event point to next event point; storage and production capacity limit is enforced by capacity constraints; demand constraints ensure that all the products demand is satisfied; and quality constraints ensure product quality specifications. Logistics constraints include all the logistics details presented in the previous section. If a feasible solution that satisfies all the quantity, quality, and logistics constraints cannot be obtained, then it is essential to produce a schedule that can still be implemented in real-life refinery sacrificing model feasibility. In this case, we introduced artificial variables to treat any infeasibility present, and these variables are subsequently penalized in the objective function to obtain an optimal solution that satisfies as many as possible from the quantity, logistics, and quality constraints by minimizing giveaways. A detailed description of the each variable and parameter used in the model can be found in the Notation section.

Variable recipe constraints

Upper and lower bounds are forced on the individual components volumetric flow rates processed at each production units. Here, $B_{i,j,n}$ is a total amount of material processed at unit j performing task i at event point n. Constraint 1a enforces the bound for the amount produced, whereas constraint 1b ensures that the amount consumed is restricted by the imposed recipe.

$$\rho_{s,i}^{p,\min} B_{i,j,n} \le bp_{s,i,j,n} \le \rho_{s,i}^{p,\max} B_{i,j,n}, \quad \forall s \in S, i \in I_s^P, j \in J_i, n \in N \tag{1a}$$

$$\rho_{s,i}^{c,\min} B_{i,j,n} \le bc_{s,i,j,n} \le \rho_{s,i}^{c,\max} B_{i,j,n}, \quad \forall s \in S, i \in I_s^C, j \in J_i, n \in N. \tag{1b}$$

Furthermore, the amount of material processed is equal to the total amount of material consumed or produced. Therefore, constraints 1a and 1b can be replaced by 2a–2c, and we can eliminate variable $B_{i,j,n}$. Constraint 2c satisfies material balance at each production unit j, which states that the total amount of material consumed is equal to the total amount of material produced.

$$\rho_{s,i}^{p,\min}\sum_{s'\in S_i^p} bp_{s',i,j,n} \le bp_{s,i,j,n} \le \rho_{s,i}^{p,\max}\sum_{s'\in S_i^p} bp_{s',i,j,n},\quad \forall s\in S, i\in I_s^P, j\in J_i, n\in N \tag{2a}$$

$$\rho_{s,i}^{c,\min}\sum_{s'\in S_i^c} bc_{s',i,j,n} \le bc_{s,i,j,n} \le \rho_{s,i}^{c,\max}\sum_{s'\in S_i^c} bc_{s',i,j,n},\quad \forall s\in S, i\in I_s^C, j\in J_i, n\in N \tag{2b}$$

$$\sum_{s\in S_i^p} bp_{s,i,j,n} = \sum_{s\in S_i^c} bc_{s,i,j,n},\quad \forall j\in J, i\in I_j, n\in N. \tag{2c}$$

In our work, we assume that the CDUs have the same lower and upper bounds which means that the distillation cuts are assumed to be known.

Material balance constraints for production units

Constraints 3a connect the material produced at production units to subsequent storage tanks, production units, and end-product delivery to market. Constraints 3b represent that the consumption at a production unit is equal to the amount of material coming from preceding storage tanks, previous units, and raw material supply.

$$\sum_{i\in I_j} \mathrm{bp}_{s,i,j,n} = \sum_{k\in K_j^{pk}\cap K_s} \mathrm{Kif}_{s,j,k,n} + \sum_{j'\in J_j^{seq}\cap J_s^c} \mathrm{JJf}_{s,j,j',n} + \mathrm{Uof}_{s,j,n},\quad \forall s\in S, j\in J_s^p, n\in N \tag{3a}$$

$$\sum_{i\in I_j} \mathrm{bc}_{s,i,j,n} = \sum_{k\in K_j^{kp}\cap K_s} \mathrm{Kof}_{s,k,j,n} + \sum_{j'\in J_j^{seq}\cap J_s^p} \mathrm{JJf}_{s,j',j,n} + \mathrm{Uif}_{s,j,n},\quad \forall s\in S, j\in J_s^c, n\in N. \tag{3b}$$

Material balance constraints for storage tanks

The material balance constraints for storage tanks are given by Eqs. 4a and 4b. The equations state that the inventory of a tank at one event point is equal to that of previous event point adjusted by the input and output streams amount and by taking into account the downgraded products amount.

$$\mathrm{st}_{s,k,n} = \mathrm{sto}_{s,k} + \sum_{j\in J_k^{pk}} \mathrm{Kif}_{s,j,k,n} + \mathrm{Rif}_{s,k,n} - \sum_{j\in J_k^{kp}} \mathrm{Kof}_{s,j,k,n}, - \sum_{o\in O_s} \mathrm{Lf}_{k,o,n} + \sum_{s'\in S_k} \mathrm{std}_{s',s,k,n} - \sum_{s'\in S_k} \mathrm{std}_{s,s',k,n},\quad \forall s\in S, k\in K_s, n=1 \tag{4a}$$

$$\mathrm{st}_{s,k,n} = \mathrm{st}_{s,k,n-1} + \sum_{j\in J_k^{pk}} \mathrm{Kif}_{s,j,k,n} + \mathrm{Rif}_{s,k,n} - \sum_{j\in J_k^{kp}} \mathrm{Kof}_{s,j,k,n}, - \sum_{o\in O_s} \mathrm{Lf}_{k,o,n} + \sum_{s'\in S_k} \mathrm{std}_{s',s,k,n} - \sum_{s'\in S_k} \mathrm{std}_{s,s',k,n},\quad \forall s\in S, k\in K_s, 1<n\le N. \tag{4b}$$

The variable $\mathrm{std}_{s',s,k,n}$ is defined as the tank heel of material s' present at the end of event point $n-1$ that is downgraded to material s during event point n.

$$\mathrm{std}_{s',s,k,n} = \begin{cases} \ge 0 & \text{if product } s' \text{ is downgraded to } s \text{ in tank } k \text{ at event point } n \\ 0 & \text{otherwise.}\end{cases}$$

When there is a changeover from higher grade product to lower grade, the tank heel present in the tank would be transformed into lower grade product without violating any product property specifications of lower grade product.

Capacity constraints for production units

Constraint 5 enforces that the material processed by unit j performing task i is bounded by the maximum and minimum rate of production. Constraint 6 gives an upper bound on total amount of material processed at unit j over the entire time horizon.

$$R_{i,j}^{\min}\left(\mathrm{Tf}_{i,j,n} - \mathrm{Ts}_{i,j,n}\right) \le \sum_{s'\in S_i^p} \mathrm{bp}_{s',i,j,n} \le R_{i,j}^{\max}\left(\mathrm{Tf}_{i,j,n} - \mathrm{Ts}_{i,j,n}\right),\ i\in I, j\in J_i, n\in N \tag{5}$$

$$\sum_{s'\in S_i^p} \mathrm{bp}_{s',i,j,n} \le \mathrm{UH}\times R_{i,j}^{\max}\times wv_{i,j,n},\quad \forall i\in I, j\in J_i, n\in N. \tag{6}$$

Capacity constraints for storage tanks

Constraints 7a–7c are capacity constraints for storage tanks, and they define the binary variables associated with flow in and out of the tanks.

$$\mathrm{Kif}_{s,j,k,n} \le V_k^{\max}\times \mathrm{in}_{s,j,k,n},\quad \forall j\in J, k\in K_j^{pk}, n\in N \tag{7a}$$

$$\mathrm{Kof}_{s,j,k,n} \le V_k^{\max}\times \mathrm{out}_{s,j,k,n},\quad \forall j\in J, k\in K_j^{kp}, n\in N \tag{7b}$$

$$\mathrm{Lf}_{s,k,o,n} \le V_k^{\max}\times l_{k,o,n},\quad \forall s\in S, k\in K_s, o\in O_s, n\in N. \tag{7c}$$

Constraints 8a–8c represent that the material present in the tank should not exceed maximum storage capacity. The multipurpose tanks can store different grades of products, and if the higher grade product is present in the tank when it is being serviced for lower grade product, then the high-quality product will be downgraded into lower quality. The downgraded products are taken into consideration for storage capacity limit in constraints 8a and 8b.

$$\mathrm{sto}_{s,k} + \sum_{j \in J_k^{pk}} \mathrm{Kif}_{s,j,k,n} + \mathrm{Rif}_{s,k,n} + \sum_{s' \in S_k} \mathrm{std}_{s',s,k,n} - \sum_{s' \in S_k} \mathrm{std}_{s,s',k,n} \le V_k^{\max} y_{s,k,n}, \quad \forall s \in S, k \in K_s, n = 1 \tag{8a}$$

$$\mathrm{st}_{s,k,n-1} + \sum_{j \in J_k^{pk}} \mathrm{Kif}_{s,j,k,n} + \mathrm{Rif}_{s,k,n} + \sum_{s' \in S_k} \mathrm{std}_{s',s,k,n} - \sum_{s' \in S_k} \mathrm{std}_{s,s',k,n} \le V_k^{\max} y_{s,k,n}, \quad \forall s \in S, k \in K_s, 1<n \le N \tag{8b}$$

$$\sum_{j \in J_k^{kp}} \mathrm{Kof}_{s,k,j,n} + \sum_{o \in O_s} \mathrm{Lf}_{k,o,n} \le V_k^{\max} y_{s,k,n}, \quad \forall s \in S, k \in K_s, n \in N. \tag{8c}$$

The maximum and minimum unloading (lift) rate for product storage tanks must be bounded as specified by constraint 9.

$$\mathrm{RU}_k^{\min}\left(\mathrm{Tof}_{k,o,n} - \mathrm{Tos}_{k,o,n}\right) \le \mathrm{Lf}_{k,o,n} \le \mathrm{RU}_k^{\max}\left(\mathrm{Tof}_{k,o,n} - \mathrm{Tos}_{k,o,n}\right), \quad \forall k \in K_p, o \in O, n \in N \tag{9}$$

Quality constraints

The final products produced by the blenders should satisfy the quality specifications. These product qualities are assumed to be computed by volumetric average to maintain model linearity. As the blend components properties are assumed to be constant, linearity of the model is preserved. Constraint 10 guarantees that final product leaving the outlet port of the blender satisfies set product quality range. Here, $P_{s,p}^{\min}$ and $P_{s,p}^{\max}$ are the upper and lower limit of property p for final blend product s.

$$P_{s,p}^{\min} \sum_{i \in I_s^p} \mathrm{bp}_{s,i,j,n} - \mathrm{pg}_{s,p,n}^{l} \le \sum_{i \in I_s^p, s' \in S_i^c} P_{s',p} \mathrm{bc}_{s',i,j,n} \le P_{s,p}^{\max} \sum_{i \in I_s^p} \mathrm{bp}_{s,i,j,n} + \mathrm{pg}_{s,p,n}^{u}, \quad \forall s \in S_b, j \in J_s^p, n \in N \tag{10}$$

When the final products produced by the blenders cannot meet the quality specifications at event point n, we introduced positive slack variables $\mathrm{pg}_{s,p,n}^{l}$ and $\mathrm{pg}_{s,p,n}^{u}$, which are penalized in objective function to minimize giveaways.

Demand constraints

Demand of each finished product must be satisfied during the entire scheduling horizon. Constraint 11 guarantees that sufficient amount of product will be available to meet the demand.

$$D_{o,s}^{-} + r_s - \mathrm{dg}_o^{l} - \mathrm{rg}_s + \le \sum_{k,n} \mathrm{Lf}_{k,o,n} + \sum_{j,n} \mathrm{Uof}_{s,j,n} \le D_{o,s}^{+} + \mathrm{dg}_o^{u}, \quad \forall s \in S_b, o \in O_s \cup \forall s \in S_f. \tag{11}$$

Because of production capacity limitation, sometimes the demand order of finished product cannot be satisfied during the entire scheduling horizon. A feasible solution can be obtained by introducing the positive artificial variables dg_0^{l}, dg_0^{u}, and rg_s in the demand constraint, which are penalized in the objective function to minimize quantity giveaway.

Logistics constraints

Allocation Constraints. Constraint 12 expresses that if a task i starts at event point n, then it must be performed in one of the suitable units j. It also satisfies the operating detail that a unit can physically perform only one task at any given time.

$$\sum_{i \in I_j} wv_{i,j,n} \le 1, \quad \forall j \in J, n \in N. \tag{12}$$

The requirement that multipurpose storage tanks can store only one type of products at any time is enforced by Eq.13.

$$\sum_{s \in K_s} y_{s,k,n} \le 1, \quad \forall k \in K^m, n \in N. \tag{13}$$

Minimum Run Lengths. The minimum run length for each task is enforced by constraint 14. Here, the minimum run length (RL_i) is 15 h.

$$\mathrm{Tf}_{i,j,n} - \mathrm{Ts}_{i,j,n} \ge \mathrm{RL}_i wv_{i,j,n} - \mathrm{UH}\left(1 - wv_{i,j,n}\right), \quad \forall i \in I, j \in I_j, n \in N. \tag{14}$$

Similar to minimum run-length constraint, maximum run-length restriction can be imposed if necessary.

Loading and Unloading Constraints. Product tanks cannot load and unload material at the same time, and this restriction is enforced by Eq. 15. Furthermore, we restrict that product tanks can only satisfy one demand order at any given time.

$$\sum_{s \in S_k} \mathrm{in}_{s,j,k,n} + \sum_{o \in O} l_{k,o,n} \le 1, \quad \forall k \in K, j \in J_k^{pk}, n \in N. \tag{15}$$

A One-Flow-Out Restriction for Blender Unit. A one-flow-out restriction given by Eq. 16 is required for all operation tasks on a blender to ensure that the product output from the blend unit can only go to one product tank. This restriction is imposed because the refinery has a blend property online controller that is set up to fill a specific product tank by taking into account the tank heel properties. If a blend unit does not comply with this restriction and send the output from the blender to multiple tanks simultaneously and not sequentially, then this miss-operation can result in a significant off-specification of product stocks.

$$\sum_{k \in K_s \cap K_j^{pk}} \mathrm{in}_{s,j,k,n} \le 1, \quad \forall s \in S_b, j \in J_s^p, n \in N. \tag{16}$$

Setup Constraints. To model tank setup, we use the binary variables $\beta_{k,n}$, which are defined by constraints 17a and 17b.

$$\beta_{k,n} = \begin{cases} 1 & \text{if storage tank } k \text{ becomes active at event point } n \text{ for first time} \\ 0 & \text{otherwise} \end{cases}$$

$$\beta_{k,n} \le \sum_{s\in S_k} y_{s,k,n} + \sum_{s\in S_k, n'<n} y_{s,k,n'}, \quad \forall k \in K^h, n \in N, \sum_s yo_{s,k} = 0 \tag{17a}$$

$$\beta_{k,n} \ge \sum_{s\in S_k} y_{s,k,n} - \sum_{s'\in S_k, n'<n} y_{s',k,n'}, \ \forall k \in K^h, n \in N, \sum_s yo_{s,k} = 0. \tag{17b}$$

Similarly for production unit, the setup variables are $\alpha_{j,n}$, and constraints 18a and 18b represent the utilization of unit at event n for the very first time during the production time horizon.

$$\alpha_{j,n} = \begin{cases} 1 & \text{if unit } j \text{ becomes active at event point } n \text{ for first time} \\ 0 & \text{otherwise} \end{cases}$$

$$\alpha_{j,n} \le \sum_{i\in J_i} wv_{i,j,n} + \sum_{i\in J_i, n'<n} wv_{i,j,n'}, \quad \forall j \in J^h, n \in N \tag{18a}$$

$$\alpha_{j,n} \ge \sum_{i\in J_i} wv_{i,j,n} - \sum_{i'\in J_{i'}, n'<n} wv_{i',j,n'}, \quad \forall j \in J^h, n \in N. \tag{18b}$$

As the refinery operates in a continuous mode, we only define the setup variables for identical parallel production units and tanks. Setup variables are penalized in the objective function to minimize the total number of units and storage tanks that are used during refinery operation.

Changeovers Constraints. Different mode of production and storage tasks are specified by different types of product being produced or stored. Changeovers between modes of operations cause disturbances and additional costs. Thus, few changeovers (long sequences of the same mode of operations) are desired. Continuous variables $\chi_{i',i,j,n}$ and $\eta_{s',s,k,n}$ denote changeover of task at production unit j and changeover of service mode at product pool k, respectively.

$$x_{i',i,j,n} = \begin{cases} 1 & \text{if mode changes from } i' \text{ at event point } n \text{ to } i \text{ at later event point} \\ 0 & \text{otherwise} \end{cases}$$

$$\eta_{s',s,k,n} = \begin{cases} 1 & \text{if mode changes from } s' \text{ at event point } n \text{ to } s \text{ at later event point} \\ 0 & \text{otherwise} \end{cases}$$

Changeover constraints proposed by Shaik et al.[26] are used in this work. Constraints 19–20c are thus used to force the changeover variables to 1 if there is a change in operations mode from event point n to any later event point.

$$\chi_{i',i,j,n} \le wv_{i',j,n}, \quad \forall j \in J^m, i \in I_j, i' \in I_j, i' \ne i, n < N \tag{19a}$$

$$\chi_{i',i,j,n} \le 1 + wv_{i,j,n'} - \sum_{i''\in I_j} wv_{i'',j,n'} + \sum_{i''\in I_j, n''\in n<n''<n'} wv_{i'',j,n''}, \quad \forall j \in J^m, i \in I_j, i' \in I_j, i' \ne i, n < N, n < n' \le N \tag{19b}$$

$$\chi_{i',i,j,n} \ge wv_{i',j,n} + wv_{i,j,n'} - 1 - \sum_{i''\in I_j, n<n''<n'} wv_{i'',j,n''}, \quad \forall j \in J^m, i \in I_j, i' \in I_j, i' \ne i, n < N, n < n' \le N \tag{19c}$$

$$\eta_{s',s,k,n} \le y_{s',k,n}, \quad \forall k \in K^m, s \in S_k, s' \in S_k, s' \ne s, n < N \tag{20a}$$

$$\eta_{s',s,k,n} \le 1 + y_{s,k,n} - \sum_{s\in S_k} y_{s',k,n'} + \sum_{s''\in S_k, n''\in n<n''<n'} y_{s'',k,n''}, \quad \forall k \in K^m, s \in S_k, s' \in S_k, s' \ne s, n < N, n < n' \le N \tag{20b}$$

$$\eta_{s',s,k,n} \ge y_{s',k,n} + y_{s,k,n'} - 1 - \sum_{s''\in S_k, n<n''<n'} y_{s'',k,n''}, \quad \forall k \in K^m, s \in S_k, s' \in S_k, s' \ne s, n < N, n < n' \le N. \tag{20c}$$

The changeover variable $\eta o_{s',s,k}$ in Eq. 20d is active if there is material s' present in the tank at the beginning of the time horizon and then service is changed over to new material s at event point $n = 1$.

$$yo_{s,k} + y_{s',k,n} \le \eta o_{s,s',k} + 1, \quad \forall k \in K^m, s \in S_k, s' \in S_k, s' \ne s, yo_{s,k} = 1. \tag{20d}$$

Changeovers variables $\chi_{i',i,j,n}$,$\eta o_{s',s,k}$, and $\eta_{s',s,k,n}$ are penalized in the objective function to minimize the changeovers.

Heel Requirement. When changeover occurs from higher to lower quality product, the holdup in the tank must be less than equal to the maximum heel quantity specified. The maximum heel requirement can be enforced on $\mathrm{std}_{s,s',k,n}$ as soft constraint using Eqs. 21a and 21b and positive slack variable $\mathrm{mh}_{s,k,n}$. The artificial variable $\mathrm{mh}_{s,k,n}$ is penalized in the objective function to minimize the heel.

$$\mathrm{std}_{s,s',k,n} - \mathrm{mh}_{s,k,n} \le V_k^{\mathrm{heel}} + V_k^{\max}\left(1 - \eta o_{s,s',k}\right), \quad \forall k \in K, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n = 1 \tag{21a}$$

$$\mathrm{std}_{s,s',k,n+1} - \mathrm{mh}_{s,k,n+1} \le V_k^{\mathrm{heel}} + V_k^{\max}\left(1 - \eta_{s,s',k,n}\right), \quad \forall k \in K, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n < N. \tag{21b}$$

Product Downgrading. The downgrading of product happens, (1) there is a changeover of service at used multipurpose tanks or (2) it is required to meet lower quality product lifting demand order due to production capacity limitation. The downgrading of product is captured by Eqs. 22a–22e. The variable associated with downgrading $\mathrm{std}_{s,s',k,n}$ is zero when the switchover occurs from lower to higher quality product.

$$\mathrm{std}_{s,s',k,n} \leq 0, \quad \forall k \in K^m, s \in K_s, s' \in K_s, \varphi_{s'} > \varphi_s, n \in N \tag{22a}$$

$$\mathrm{std}_{s,s',k,n} \leq V_k^{\max} \eta o_{s,s',k}, \quad \forall k \in K^m, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n = 1 \tag{22b}$$

$$\mathrm{std}_{s,s',k,n+1} \leq V_k^{\max} \eta_{s,s',k,n}, \forall k \in K^m, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n < N \tag{22c}$$

$$\mathrm{std}_{s,s',k,n} - V_k^{\max}\left(1 - \eta o_{s,s',k}\right) \leq \mathrm{sto}_{s,k} \leq \mathrm{std}_{s,s',k,n} + V_k^{\max}\left(1 - \eta o_{s,s',k}\right), \forall k \in K^m, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n = 1 \tag{22d}$$

$$\mathrm{std}_{s,s',k,n+1} - V_k^{\max}\left(1 - \eta_{s,s',k,n}\right) \leq \mathrm{st}_{s,k,n} \leq \mathrm{std}_{s,s',k,n+1} + V_k^{\max}\left(1 - \eta_{s,s',k,n}\right), \ \forall k \in K^m, s \in K_s, s' \in K_s, \varphi_{s'} < \varphi_s, n < N. \tag{22e}$$

Timing constraints

Sequence Constraints for Production Units. Finishing time of any task must be greater than the starting time of that task, as represented by constraint 23a. Constraint 23b expresses that if task i starts at event point $n + 1$, then it must start after the end of the same task happening at event point n, while Eq. 23c enforces the time sequence constraint for different tasks happening in the same unit.

$$\mathrm{Tf}_{i,j,n} \geq \mathrm{Ts}_{i,j,n}, \quad \forall i \in I, j \in J_i, n \in N \tag{23a}$$

$$\mathrm{Tf}_{i,j,n+1} \geq \mathrm{Tf}_{i,j,n}, \quad \forall i \in I, j \in J_i, n \in N, n < N \tag{23b}$$

$$\mathrm{Ts}_{i,j,n+1} \geq \mathrm{Tf}_{i',j,n} - UH\left(1 - wv_{i',j,n}\right), \quad \forall j \in J, i \in I_j, i' \in I_j, i' \neq i, n \in N, n < N \tag{23c}$$

Constraints 24a–24d represent that two consecutive productions with no storage in between, happen at the same time because production units operate as continuous processes. Here, unit j consumes the material produced by unit j'.

$$\mathrm{Ts}_{i,j,n} \leq \mathrm{Ts}_{i',j',n} + \mathrm{UH}\left(1 - wv_{i,j,n} - wv_{i',j',n}\right), \quad \forall j' \in J, j \in J_{j'}^{\mathrm{seq}}, i \in I_j, i' \in I_{j'}, n \in N \tag{24a}$$

$$\mathrm{Ts}_{i,j,n} \geq \mathrm{Ts}_{i',j',n} - \mathrm{UH}\left(1 - wv_{i,j,n} - wv_{i',j',n}\right), \quad \forall j' \in J, j \in J_{j'}^{\mathrm{seq}}, i \in I_j, i' \in I_{j'}, n \in N \tag{24b}$$

$$\mathrm{Tf}_{i,j,n} \leq \mathrm{Tf}_{i',j',n} + \mathrm{UH}\left(1 - wv_{i,j,n} - wv_{i',j',n}\right), \quad \forall j' \in J, j \in J_{j'}^{\mathrm{seq}}, i \in I_j, i' \in I_{j'}, n \in N \tag{24c}$$

$$\mathrm{Tf}_{i,j,n} \geq \mathrm{Tf}_{i',j',n} - \mathrm{UH}\left(1 - wv_{i,j,n} - wv_{i',j',n}\right), \quad \forall j' \in J, j \in J_{j'}^{\mathrm{seq}}, i \in I_j, i' \in I_{j'}, n \in N. \tag{24d}$$

Sequence Constraints for Storage Tanks. Finishing time of the inlet, outlet transfer service has to be greater than or equal to the start time of that service. Constraints 25–27 enforce the sequence time requirement for movement transfer task from one event point to next event point for same unit-tank connection.

$$\mathrm{Tsf}_{j,k,n} \geq \mathrm{Tss}_{j,k,n}, \quad \forall k \in K, j \in J_k^{pk}, n \in N \tag{25a}$$

$$\mathrm{Tss}_{j,k,n+1} \geq \mathrm{Tsf}_{j,k,n}, \quad \forall k \in K, j \in J_k^{pk}, n \in N, n < N \tag{25b}$$

$$\mathrm{Tsf}_{k,j,n} \geq \mathrm{Tss}_{k,j,n}, \quad \forall k \in K, j \in J_k^{kp}, n \in N \tag{26a}$$

$$\mathrm{Tss}_{k,j,n+1} \geq \mathrm{Tsf}_{k,j,n}, \quad \forall k \in K, j \in J_k^{kp}, n \in N, n < N \tag{26b}$$

$$\mathrm{Tof}_{k,o,n} \geq \mathrm{Tos}_{k,o,n}, \quad \forall k \in K_p, o \in O, n \in N \tag{27a}$$

$$\mathrm{Tos}_{k,o,n+1} \geq \mathrm{Tof}_{k,o,n}, \quad \forall k \in K_p, o \in O, n \in N, n < N. \tag{27b}$$

Start time sequence constraints for tanks receiving/sending material from/to multiple production units are given by constraints 28a and 28b, whereas constraints enforcing time sequence requirement for different type of demand orders satisfied by the tank are given by Eq. 28c.

$$\mathrm{Tss}_{j',k,n+1} \geq \mathrm{Tsf}_{j,k,n} - \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{pk}, j' \neq j, n \in N, n < N \tag{28a}$$

$$\mathrm{Tss}_{k,j',n+1} \geq \mathrm{Tsf}_{k,j,n} - \mathrm{UH}\left(1 - \mathrm{out}_{k,j,n}\right), \quad \forall k \in K, j \in J_k^{kp}, j' \in J_k^{kp}, j' \neq j, n \in N, n < N \tag{28b}$$

$$\mathrm{Tos}_{k,o',n+1} \geq \mathrm{Tof}_{k,o,n} - \mathrm{UH}\left(1 - l_{k,o,n}\right), \quad \forall k \in K_p, o \in O, o' \in O, o' \neq O, n \in N, n < N. \tag{28c}$$

Sequence constraint for material transfer in and out of intermediate tanks happening at the same event point is enforced by Eqs. 29a–29d.

$$\mathrm{Tss}_{j,k,n} + \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right) \geq \mathrm{Tss}_{k,j',n} - \mathrm{UH}\left(1 - \mathrm{out}_{s,k,j',n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N \tag{29a}$$

$$\mathrm{Tss}_{j,k,n} - \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right) \leq \mathrm{Tss}_{k,j',n} + \mathrm{UH}\left(1 - \mathrm{out}_{s,k,j',n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N \tag{29b}$$

$$\mathrm{Tsf}_{j,k,n} + \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right) \geq \mathrm{Tsf}_{k,j',n} - \mathrm{UH}\left(1 - \mathrm{out}_{s,k,j',n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N \tag{29c}$$

$$\mathrm{Tsf}_{j,k,n} - \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right) \leq \mathrm{Tsf}_{k,j',n} + \mathrm{UH}\left(1 - \mathrm{out}_{s,k,j',n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N. \tag{29d}$$

Constraints 30a and 30b connect material transfer from one event point to next event point. Product tanks cannot simultaneously load and unload material, and this restriction is enforced by Eqs. 31a and 31b. Variable ta_k is a fill-draw-delay parameter for tank k. The tank unloading happens anytime after the material flow into the tank is over and fill-draw-delay downtime has elapsed. Product flow into the tank starts after the end of the product unloading.

$$\mathrm{Tss}_{j,k,n+1} \geq \mathrm{Tss}_{k,j',n} - \mathrm{UH}\left(1 - \mathrm{out}_{s,k,j',n}\right), \quad \forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N, n < N \tag{30a}$$

$$\mathrm{Tss}_{k,j',n+1} \geq \mathrm{Tss}_{j,k,n} + \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right),$$
$$\forall k \in K, j \in J_k^{pk}, j' \in J_k^{kp}, n \in N \quad (30b)$$

$$\mathrm{Tos}_{k,o,n+1} \geq \mathrm{Tsf}_{j,k,n} - \mathrm{UH}\left(1 - \mathrm{in}_{s,j,k,n}\right) + \mathrm{ta}_k \mathrm{in}_{s,j,k,n},$$
$$\forall s \in S, k \in K_s, j \in J_k^{pk}, o \in O_s, n < N \quad (31a)$$

$$\mathrm{Tss}_{j,k,n+1} \geq \mathrm{Tof}_{k,o,n} - \mathrm{UH}\left(1 - l_{k,o,n}\right),$$
$$\forall k \in K, j \in J_k^{pk}, o \in O, n < N. \quad (31b)$$

The tank needs to go through cleaning maintenance to store higher grade product after servicing lower grade product. This maintenance downtime requirement is captured by constraints 31c. Constraints are relaxed if there is no switchover in the service from lower grade to higher grade product type.

$$\mathrm{Tss}_{j,k,n+1} \geq \ \mathrm{Tof}_{k,o,n} - \mathrm{tclean}_k\left(2\eta_{s',s,k,n} - 1\right),$$
$$\forall k \in K, s \in S_k, s' \in S_s, \varphi_{s'} < \varphi_s, j \in J_k^{pk} o \in O_{s'}, n < N. \quad (31c)$$

Sequence Constraints for Production Units and Storage Tanks. Upstream production and material flow into storage tank happen at the same time, which is imposed by constraints 32a–32d.

$$\mathrm{Ts}_{i,j,n} \leq \mathrm{Tss}_{j,k,n} + \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{in}_{s,j,k,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{pk}, i \in I_j, n \in N \quad (32a)$$

$$\mathrm{Ts}_{i,j,n} \geq \mathrm{Tss}_{j,k,n} - \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{in}_{s,j,k,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{pk}, i \in I_j, n \in N \quad (32b)$$

$$\mathrm{Tf}_{i,j,n} \leq \mathrm{Tsf}_{j,k,n} + \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{in}_{s,j,k,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{pk}, i \in I_j, n \in N \quad (32c)$$

$$\mathrm{Tf}_{i,j,n} \geq \mathrm{Tsf}_{j,k,n} - \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{in}_{s,j,k,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{pk}, i \in I_j, n \in N. \quad (32d)$$

For intermediate storage tanks, downstream production and material flow out of feed tank occur simultaneously. This constraint is imposed by Eqs. 33a–33d.

$$\mathrm{Ts}_{i,j,n} \leq \mathrm{Tss}_{k,j,n} + \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{out}_{s,k,j,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{kp}, i \in I_j, n \in N \quad (33a)$$

$$\mathrm{Ts}_{i,j,n} \geq \mathrm{Tss}_{k,j,n} - \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{out}_{s,k,j,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{kp}, i \in I_j, n \in N \quad (33b)$$

$$\mathrm{Tf}_{i,j,n} \leq \mathrm{Tsf}_{k,j,n} + \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{out}_{s,k,j,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{kp}, i \in I_j, n \in N \quad (33c)$$

$$\mathrm{Tf}_{i,j,n} \geq \mathrm{Tsf}_{k,j,n} - \mathrm{UH}\left(1 - wv_{i,j,n} - \mathrm{out}_{s,k,j,n}\right),$$
$$\forall s \in S, k \in K_s, j \in J_k^{kp}, i \in I_j, n \in N. \quad (33d)$$

All tasks should start and finish before the end of the scheduling time horizon as stated in 34a–34d.

$$\mathrm{Ts}_{i,j,n} \leq H, \quad \mathrm{Tf}_{i,j,n} \leq H, \quad \forall i \in I, j \in J_i, n \in N \quad (34a)$$

$$\mathrm{Tss}_{j,k,n} \leq H, \quad \mathrm{Tsf}_{j,k,n} \leq H, \quad \forall i \in J, k \in K_j^{pk}, n \in N \quad (34b)$$

$$\mathrm{Tss}_{k,j,n} \leq H, \quad \mathrm{Tsf}_{k,j,n} \leq H, \quad \forall i \in J, k \in K_j^{kp}, n \in N \quad (34c)$$

$$\mathrm{Tos}_{k,o,n} \leq H, \quad \mathrm{Tof}_{k,o,n} \leq H, \quad \forall o \in O, k \in K_p, n \in N. \quad (34d)$$

Intermediate Due Dates. Intermediate due dates for Group A products, which are stored in product pools, are given by constraints 35a and 35b. Orders can start unloading anytime after the vessel arrival time and finish unloading anytime before vessel departure time. The due date requirements are enforced as inequality constraints to consider demurrage by using slack variables Tearly_o and Tlate_o, which are penalized in the objective function.

$$\mathrm{Tos}_{k,o,n} + \mathrm{UH}\left(1 - l_{k,o,n}\right) \geq \mathrm{times}_o - \mathrm{Tearly}_o,$$
$$\forall o \in O, k \in K_p, n \in N \quad (35a)$$

$$\mathrm{Tof}_{k,o,n} + \mathrm{UH}\left(1 - l_{k,o,n}\right) \leq \mathrm{timef}_o + \mathrm{Tlate}_o,$$
$$\forall o \in O, k \in K_p, n \in N. \quad (35b)$$

Valid inequalities

Valid inequalities are added in the proposed model to improve the computational efficiency of the proposed model. Constraints 36a and 36b enforce that if the material s is flowing into the tank at event point n, then the binary variable $y_{s,k,n}$ is 1, and similarly, if the material is flowing out of the tank, then the binary variable is also 1.

$$\sum_{j \in J_k^{pk}} \mathrm{in}_{s,j,k,n} \leq \sum_{j \in J_k^{pk}} y_{s,k,n}, \quad \forall k \in K, s \in S_k, n \in N \quad (36a)$$

$$\sum_{j \in J_k^{kp}} \mathrm{out}_{s,j,k,n} \leq \sum_{j \in J_k^{kp}} y_{s,k,n}, \quad \forall k \in K, s \in S_k, n \in N. \quad (36b)$$

If the tank is sending or receiving the material from a production unit, then that unit is active and these requirements are represented by Eqs. 37a and 37b.

$$\sum_{k \in K_j^{pk}} \mathrm{in}_{s,j,k,n} \leq \sum_{k \in K_j^{pk}} \sum_{i \in I_j \cap I_s^p} wv_{i,j,n}, \quad \forall j \in J, s \in S_j^p, n \in N \quad (37a)$$

$$\sum_{k \in K_j^{kp}} \mathrm{out}_{s,k,j,n} \leq \sum_{k \in K_j^{kp}} \sum_{i \in I_j \cap I_s^c} wv_{i,j,n}, \quad \forall j \in J, s \in S_j^c, n \in N. \quad (37b)$$

As loading and unloading cannot happen at the same event point, if the material is unloaded at event point $n + 1$, then tank is not empty at previous event point n. This feature of model is captured by Eq. 38.

$$\sum_{o} l_{k,o,n+1} \leq \sum_{s \in S_k} y_{s,k,n}, \quad \forall k \in K, n \in N, n < N. \quad (38)$$

If two units are consecutive without any storage tank between them, then constraint 39 imposes the simultaneous

Table 1. Price of Raw Materials and Final Products

Material	Price	Material	Price	Material	Price
ANS crude oil	25	Jet fuel	70	Pentane	40
SJV crude oil	20	Refinery gases	50	NC4	40
Carb diesel	80	Isomerate	60	Alkylate	60
EPA diesel	60	Reformate	60	FCC gas	45
Red-dye diesel	50	Isomax gasoline	65	Coke	30

operation of these units due to the continuous operation mode. However, this constraint is not imposed on parallel production units that can produce the same type of products.

$$\sum_{j' \in J_j^{\text{seq}}/J^h, i' \in I_{j'}} wv_{i',j',n} \leq \sum_{j' \in J_j^{\text{seq}}/J^h, i \in I_j} wv_{i,j,n}, \quad \forall j \in J, j \notin J^h, n \in N. \tag{39}$$

Constraint 40 enforces the material balance constraint in addition to the constraint presented in Eq. 2c.

$$\sum_{k \in K_j^{kp} \cap K_s} \mathrm{Kof}_{s,k,j,n} + \sum_{j' \in J_j^{\text{seq}} \cap J_s^p} \mathrm{JJf}_{s,j',j,n} + \mathrm{Uif}_{s,j,n} \leq \sum_{k \in K_j^{pk} \cap K_s} \mathrm{Kif}_{s,j,k,n} + \sum_{j' \in J_j^{\text{seq}} \cap J_s^c} \mathrm{JJf}_{s,j,j',n} + \mathrm{Uof}_{s,j,n}, \quad \forall j \in J, n \in N. \tag{40}$$

Objective function

The objective function (41) is used to maximize the performance and profit of total production. The refinery performance is represented by the minimization of utilization of units and tanks, all the connection between production units and tanks, start up setups, changeovers, and production downgrading. The blending task is significantly improved by reducing the quantities of downgraded products. Logistics and quality giveaways, under- and overproduction, and demurrage are penalized. Profit term includes the costs of feeds and revenue of products. The penalty weights are assigned arbitrary to each term depending on its importance in schedule. The deviation from intermediate due dates is heavily penalized, quality giveaways are penalized the second most, while connection between unit and tank is least heavily penalized. It is favorable that the multipurpose product tanks store different grade products only in a certain order that is allowed by the sequence-dependent switchovers constraint. The favorable switchovers are from higher grade of product to lower grade, and unfavorable switchovers are from lower grade to higher grade product. Because of contamination issues, unfavorable switchovers are more heavily penalized than favorable. Similar to multipurpose tanks, there is a sequence-dependent switchover restriction for multipurpose blend units. The switchover from the run mode that produces better quality product to lower quality product is least penalized than vice versa.

$$\begin{aligned} z = & \sum_{i,j \in J_i, n} c_{i,j}^1 wv_{i,j,n} + \sum_{s,k \in K_s, n} c_k^2 y_{s,k,n} + \sum_{k \in K_p, o, n} c_k^3 l_{k,o,n} \\ & + \sum_{j, s \in S_j^p, k \in K_s, n} c_{j,k}^4 \mathrm{in}_{s,j,k,n} + \sum_{j, s \in S_j^c, k \in K_s, n} c_{k,j}^4 \mathrm{out}_{s,k,j,n} \\ & + \sum_{j \in J^h, n} c_j^5 \alpha_{j,n} + \sum_{k \in K^h, n} c_k^6 \beta_{k,n} + \sum_{j \in J^m, i \in I_j, i' \in I_j, i' \neq i, n} c_{i,i'}^7 \chi_{i,i',j,n} \\ & + \sum_{k \in K^m, s \in S_k, s' \in S_k, s \neq s'} c_{s,s'}^8 \eta o_{s,s',k} + \sum_{k \in K^m, s \in S_k, s' \in S_k, s \neq s', n} c_{s,s'}^8 \eta_{s,s',k,n} \\ & + \sum_{s,k \in K_s, n} c_k^9 st_{s,k,n} + \sum_{k \in K^m, s \in S_k, s' \in S_k, s \neq s', n} c_{k \in K^m}^{10} \mathrm{std}_{s,s',k,n} \\ & + \sum_{k \in K^m, s \in S_k, n} c_{s,k}^{11} \mathrm{mh}_{s,k,n} + \sum_{s \in S_b, p, n} c_{s,p}^{12} \mathrm{pg}_{s,p,n}^l + \sum_{s \in S_b, p, n} c_{s,p}^{13} \mathrm{pg}_{s,p,n}^u \\ & + \sum_{o \in O} c_o^{14} \mathrm{dg}_o^l + \sum_{o \in O} c_o^{15} \mathrm{dg}_o^u + \sum_{s \in S_f} c_s^{16} \mathrm{rg}_s + \sum_{o \in O} c_o^{17} \mathrm{Tearly}_o \\ & + \sum_{o \in O} c_o^{18} \mathrm{Tlate}_o + \sum_{k, s \in S_{k,n}} c_s^{19} \mathrm{Rif}_{s,k,n} + \sum_{s, j_s^c, n} c_s^{19} \mathrm{Uif}_{s,j,n} \\ & - \sum_{s \in S_f, j_s^p, n} c_s^{20} \mathrm{Uof}_{s,j,n} - \sum_{s \in S_b, k \in K_s, o \in O_s, n} c_s^{21} Lf_{s,k,o,n}. \end{aligned} \tag{41}$$

Note that the different penalty parameters have significant effect on the computational time required to obtain an optimal solution.

Results on the Case Studies

In this section, two problems based on the realistic case study presented in Section ''problem definition'' are solved to optimality and analyzed to show the effectiveness of the proposed model. Each case study includes different set of examples that differ in either demands, intermediate due dates, or initial holdup in the tank. All the problems are solved on a Dell Precision (Intel^{R} Xeon^{TM} with CPU 3.20 GHz and 2 GB memory) running on Windows XP using CPLEX 12.1.0/GAMS 23.2. The penalties parameter used in the objective function are given in Table 1.

Case study 1

The first case study is obtained by deleting three diesel product tanks to reduce the model size and to obtain medium-scale case study. The medium-scale refinery has only one dedicated tank for each three diesel product and one multipurpose tank that can service CARB and EPA diesel. For this problem, there are no product tanks receiving material from more than one blender unit and no raw material tank supplying material to multiple CDUs. Thus, we exclude constraints 28a and 28b from the model because the time

Table 2. Penalty Parameters in Objective Function

Penalty Parameter	Value	Penalty Parameter	Value
$C^1_{\text{carbnormal,dieselblender}}$	125	$C^{11}_{s,k}$	10
$C^1_{\text{EPAnormal,dieselblender}}$	105	$C^{12}_{s,p}$	1000
$C^1_{\text{Reddyenormal,dieselblender}}$	85	$C^{13}_{s,p}$	1300
C^2_k	1	C^{14}_o	1150
C^3_k	1	C^{15}_o	500
$C^4_{j,k}$	4 (Multipurpose tank: 6)	C^{16}_s	1100
C^5_j	1	C^{17}_o	1400
C^6_k	150	C^{18}_o	1250
$C^7_{i,i'}$	40 (Unfavorable: 50)	C^{19}_s	$\text{price}(s) \cdot (e+q)^{-1}$
$C^8_{s,s'}$	60 (Unfavorable: 70)	C^{20}_s	$\text{price}(s) \cdot e^{-1}$
C^9_k	$(V^{\max}_k)^{-1}$	C^{21}_s	$\text{price}(s) \cdot q^{-1}$
C^{10}_k	$(V^{\text{heel}}_k)^{-1}$		

where $e = 3\sum_s r_s$ and $q = 2\sum_{o,s} \text{order}_{o,s}$.

Table 3. Group A Products Demand Order Data for Case Study 1

Ex.	Orders [Product Type, Amount (kbbl), Delivery Window, Delivery Rate (kbbl/h)]						
	O1	O2	O3	O4	O5	O6	O7
1	P3	P2	P1	P4			
	[10,100]	[50,150]	[50,175]	[10,150]			
	[58,71]	[10,20]	[28.5,46.5]	[40,70]			
	3	10	10	3			
2	P3	P2	P1	P4	P4		
	[37,100]	[55,150]	[98,175]	[112,150]	[50,100]		
	[65,72]	[55,68]	[30,43]	[38,50]	[59.6,71.5]		
	10	10	10	10	10		
3	P3	P2	P1	P1	P4	P4	P2
	[5,15]	[5,21]	[15,50]	[10,75]	[50,150]	[25,75]	[10,38]
	[10,23]	[23,30]	[36,48]	[54,65]	[25,38]	[55,72]	[63,72]
	10	10	10	10	10	10	10
4	P3	P2	P1	P1	P4	P4	P2
	[5,15]	[5,21]	[15,50]	[10,75]	[50,150]	[25,75]	[10,38]
	[10,23]	[23,30]	[36,48]	[54,65]	[25,38]	[55,72]	[63,72]
	10	10	10	10	10	10	10

Table 4. Group B Products Demands Data for Case Study 1

		Group B Products Demand (kbbl)								
Ex.	Initial Holdup (Product, kbbl)	P5	P6	P7	P8	P9	P10	P11	P12	P13
1	P1–10	5	20	20	20	5	0	5	5	0
2	–	5	24	65	13	5	0	5	5	0
3	P1–7	5	20	15	8	5	3	5	4	9
4	P2–7	5	20	15	8	5	3	5	4	9

Table 5. Computational Performance of Case Study 1 (Without Valid Inequalities)

Ex.	n	0-1 Var.	Cont. Var.	Constraints	Nonzero Elements	Nodes	Iterations	CPU Time (s)	Obj. Value	% Gap
1	4	260	1683	4200	14,841	167,899	24,414,456	12504.70	1125.98	0.00
2	4	268	1719	4311	15,245	379,243	58,903,042	35273.88	1643.92	0.00
3	4	284	1793	4542	16,024	1,000,000	161,530,640	102634.73	1234.92	3.75*
4	4	284	1794	4541	16,021	1,000,000	204,485,505	121729.67	1193.73	5.01*
4	5	355	2242	5743	20,702	564,000	218,078,428	169094.88	1065.40	16.30†

*Nodes limit reached.
†Out of memory.

Table 6. Computational Performance of Case Study 1 (With Valid Inequalities)

Ex.	n	0-1 Var.	Cont. Var.	Constraint	Nonzero Elements	Nodes	Iterations	CPU Time (s)	Obj. Value	% Gap
1	4	260	1683	4474	15,675	10,552	1,487,406	713.28	1125.98	0.00
2	4	268	1719	4585	16,085	20,085	2,684,973	1309.78	1643.92	0.00
3	4	284	1793	4816	16,876	56,424	7,003,479	3473.47	1234.24	0.00
4	4	284	1794	4815	16,873	14,922	1,810,327	983.47	1193.54	0.00
4	5	355	2242	6087	21,772	78,318	17,482,935	11616.69	1055.45	0.00

Table 7. **Material Flow In and Out of Multipurpose Tank for Case Study 1, Example 1**

Initial Holdup (Product/Amount, kbbl)	Flow Direction	Event Points	
		$n1$	$n2$
P1/10	Loading	P2 (40 kbbl)	
	Unloading		P2 (50 kbbl)

sequence requirements for intermediate tanks are satisfied by other sequence constraints present in the formulation. Data for different set of examples are given in Tables 3 and 4, and results are shown in Tables 5 and 6. The initial holdup in Table 2 is stated for multipurpose tanks. Demand of Group B products has to be met before the end of the scheduling time horizon. In this case study, the scheduling horizon is 72 h (3 days). The products P1, P2, P3, and P4 correspond to red-dye diesel, EPA diesel, CARB diesel, and jet fuel, respectively. In this case study, we have included four product qualities requirements for blend products. The rule of thumb for the smallest number of event points needed to obtain an optimal solution is ($d + 1$), where d represents the total number of diesel products in the demand orders. For example, if only CARB and EPA diesel products are required, then three event points should be considered first.

For the examples addressed in the case study 1, the aforementioned valid inequalities allowed us to compute medium-scale scheduling problems with significantly less computational effort. The first integer solution is obtained within 15 s for all the examples studied. The computational performance with and without valid inequalities is presented in Tables 5 and 6, respectively. When the valid inequalities are included in the model, the number of variables remains the same, but the number of constraints and nonzero elements increase. Valid inequalities have no effect on quality of the optimal solution; rather their effect is concentrated in significantly reducing the computational effort needed to find the optimal solution. The CPU time required to reach optimal solution is reduced by 90% when the valid inequalities are present vs. when they are not included in the model. When example 3 is solved without valid inequalities, the optimal solution is not obtained even after 28 h, whereas when it is solved with valid inequalities, the optimal solution is obtained within 1 h. The size of the model increases as the event point increases, and time to obtain optimal solution also increases as observed for examples 4. In example 1, to satisfy the demand of order 2 (O2) within its delivery window, higher quality product P1 is downgraded to lower quality product P2. Multipurpose tank inventory data for example 1 are shown in Table 7. Product degradation is observed in optimal solution of example 3 to satisfy the product demand of order 2.

Case study 2

In this section, the proposed model is applied to Honeywell Hi-Spec refinery problem presented in section "problem definition." We exclude constraints 28a and 28b because of the reasons mentioned in the previous case study. The time horizon considered in this case study is 10 days (240 h), and there are four different quality restrictions placed on the

Table 8. **Group A Products Demand Order Data for Case Study 2**

Orders	Product Type, Amount (kbbl), Delivery Window, Delivery Rate (10 kbbl/h)					
	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5	Ex. 6
O1	P3 [50,100] [28.5,46.5]	P3 [75,100] [110,140]	P3 [15,50] [36,54]	P3 [15,50] [36,54]	P1 [15,50] [36,54]	P1 [15,50] [24,50]
O2	P2 [50,150] [65,89]	P2 [50,100] [63,89]	P2 [32,50] [63,92]	P2 [32,50] [63,92]	P2 [32,50] [63,92]	P1 [10,50] [63.73]
O3	P1 [50,200] [105,130]	P1 [75,125] [26.5,60]	P1 [45,70] [103,119]	P1 [45,70] [103,119]	P3 [45,70] [103,119]	P2 [5,70] [85,109]
O4	P1 [50,175] [216.5,235.5]	P1 [100,175] [215,235.5]	P1 [63,98] [135,148.5]	P1 [63,98] [135,148.5]	P1 [63,98] [135,148.5]	P3 [13,98] [125,138.5]
O5	P4 [75,200] [90,120]	P4 [100,200] [90,120]	P4 [50,150] [100,135]	P4 [50,150] [100,135]	P4 [50,150] [100,135]	P4 [20,150] [100,125]
O6	P4 [50,250] [174,200]	P4 [75,250] [204,230]	P4 [15,130] [200,236.5]	P4 [15,130] [200,236.5]	P4 [15,130] [200,236.5]	P4 [25,130] [200,236.5]
O7	P2 [50,100] [166,211.5]	P2 [50,100] [166,191.5]	P2 [10,48] [166,199.5]	P2 [10,48] [166,199.5]	P2 [10,48] [166,205]	P1 [10,48] [145,175]
O8			P3 [10,35] [226,240]	P3 [10,35] [226,240]	P1 [10,35] [226,240]	P2 [10,35] [195,220]
O9						P3 [10,35] [230,240]
O10						P4 [25,150] [225,240]

Table 9. Group B Products Demands Data for Case Study 2

Ex.	Initial Holdup (Product, kbbl)	Group B Products Demand (kbbl)								
		P5	P6	P7	P8	P9	P10	P11	P12	P13
1	–	10	50	50	50	10	10	15	15	0
2	–	5	50	200	50	50	10	40	10	50
3	P2–8	7	43	172	53	40	11	44	15	30
4	–	7	43	172	53	40	11	44	15	30
5	–	7	43	172	53	40	11	44	15	30
6	P1–10	2	13	72	30	10	5	22	5	0

blend products. The data for different demand orders are shown in Tables 8 and 9, and results for these data are shown in Table 10. All the examples reach the first integer solution within 20 s. In many instances, the optimal solution is reached fast and the rest of the time is spent proving the global optimality, which is a typical behavior of mixed-integer programing models. In example 1, the optimal solution is obtained with product P4 sulfur limit violations and due date violation (demurrage) of demand order 1. When only four event points are used for example 3, the optimal solution of 3866.22 is obtained that does not fully satisfy demand order 7 (O7). However, when five event points are used, the optimal solution of 1562.13 is obtained that satisfies all the demand orders within their due dates. In the case of example 5 with five event points, the first integer solution of 286630.74 and 99.78% gap is reached within 10 s, and the first integer solution that does not violate any demand requirements and due date restrictions is obtained within 1600 s with an objective value of 1741.81 and gap of 58.89%. Example 6 obtains the first integer solution with objective value of 287161.44 and 99.91% gap in 15 s. The solution without any quantity, quality, and demurrage violations is reached within 450 s with objective value of 1728.68 and 82.08% gap. A solution without product downgrading is obtained for example 6 when six event points are used. As the demand order increases, the event points needed to reach best solution also increase, thus size of the model increases too.

Summary

In this article, a short-term scheduling model is developed based on continuous-time representation for large-scale refineries. The model features logistics decisions such as start-up, minimum run length, fill-draw-delay, one-flow out of blender, sequence-dependent changeovers, maximum heel quantity, and downgrading of product. A set of valid inequalities are proposed that reduces the CPU resolution

Table 10. Computational Results for Case Study 2 (With Valid Inequalities)

Ex.	Event Points	Var. Int./Cont. (Constraints) Nonzero Elem.	Last Integer Solution				Optimal Solution		
			Nodes/Iterations	Obj. Value	CPU Time (s)	Gap (%)	Nodes/Iterations	Obj. Value	CPU Time (s)
1	4	328/1956 (5333) 18,614	103,384/9,312,323	34638.64	5000	0.45	249,153/14,282,981	34638.64	7910.20
2	4	328/1957 (5333) 18,622	20,760/3,626,885	1671.86	1800	6.57	35,489/4,741,152	1671.86	2449.83
3	4	336/1994 (5431) 18,947	79,969/15,494,045	3866.22	8300	3.64	207,687/23,440,793	3866.22	12762.77
3	5	420/2490 (6872) 24,466	419,747/119,257,871	1562.13	100,700	1.96	872,647/133,443,528	1562.13	112879.86
4	4	336/1993 (5438) 18,992	43,412/8,367,727	1743.87	4650	6.70	59,005/9,238,856	1743.87	5150.23
5	4	340/2009 (5518) 19,244	22,787/4,473,417	1649.75	2650	3.16	55,538/6,353,433	1649.75	3787.66
5	5	425/2510 (6976) 24,793	97,379/30,910,982	1511.44	21,500	44.30	509,585/178,283,024	1511.44	136811.42*
6	5	445/2597 (7246) 25,694	428,485/160,549,580	1137.57	132,500	38.10	916,986/281,680,876	1137.57	234163.58†
6	6	534/3114 (8778) 31,779	120,720/52,056,975	997.25	54,800	40.71	375,832/150,550,479	997.25	161182.4‡

*Out of memory. Optimal solution obtains with 33.13% gap.
†Out of memory. Optimal solution obtains with 12.62% gap.
‡Out of memory. Optimal solution obtains with 30.59% gap.

time by a significant factor for large-scale refinery problems. The model with valid inequalities is applied to different examples and was observed that valid inequalities result in up to 90% reduction in CPU performance time compared to model without inequalities. The model is applied to two case studies to illustrate the applicability of the proposed formulations to large-scale refinery operations. However, even with valid inequalities present in the model, the computational time required to reach optimal solution is still high. Thus, the focus for the future work is to use different decomposition approaches such as Lagrange decomposition, Benders decomposition, heuristics, or combination of both heuristics and mathematical decomposition to enable the solution of large-scale problems.

Acknowledgments

The authors thank Jeff Kelly from Honeywell for providing the case study presented in this article. The authors gratefully acknowledge financial support from National Science Foundation under Grants CBET 0625515, 0966861.

Notation

Indices

i = tasks
j = production units
k = storage tanks
n = event points
o = product order
p = properties
s = states

Sets

I_j = tasks which can be performed in unit j
I_s^p = tasks which can produce material s
I_s^c = tasks which can consume material s
J = production units
J_s^c = units that consume material s
J_i = units which are suitable for performing task i
J^h = units that can produce all the same products as some other unit in the refinery
J^m = units which are suitable for performing multiple tasks
J_k^{kp} = units that consume material s stored in tank k
J_k^{pk} = units that produce material s stored in tank k
J_s^p = units that can produce material s
$J_{j'}^{seq}$ = units that follow unit j′ (no storage in between)
K = storage tanks
K^h = tanks that can store the same products as some other tank in the refinery
K_j^{kp} = tanks that store material consumed by unit j
K^m = multipurpose tanks that can store multiple materials
K_p = tanks that can store final products
K_j^{pk} = tanks that store material produced by unit j
K_s = tanks that can store material s
N = event point within the time horizon
O = orders for products that are stored in tanks
P = product properties
S = states
S_b = Group A final products, produced by blenders and stored in tanks
S_f = Group B final products, products that are not stored in tanks
S_k = materials that can be stored in tank k
S_i^c = materials that can be consumed by task i
S_j^c = materials that can be consumed by unit j
S_i^p = materials that can be produced by task i
S_j^p = materials that can be produced by unit j

Parameters

$D_{o,s}^+, D_{o,s}^-$ = demand limit requirement for order o and product s that is stored in tank
r_s = demand of the final product s at the end of the time horizon
$R_{i,j}^{\min}/R_{i,j}^{\max}$ = minimum/maximum rate of material be processed by task i in unit j
RL_i = minimum run length for task i
$\mathrm{RU}_k^{\min}/\mathrm{RU}_k^{\max}$ = minimum/maximum rate of product unloading at tank k
$\mathrm{sto}_{s,k}$ = amount of state s that is present at the beginning of the time horizon in k
ta_k = fill-draw-delay for product tank k
UH = available time horizon
$V_k^{\max}$ = maximum available storage capacity of storage tank k
V_k^{heel} = maximum heel available for storage tank k
$yo_{s,k}$ = 1 if the material s is present at the beginning of the time horizon in k
φ_s = Product grade index
$\rho_{s,i}^{\min}/\rho_{s,i}^{\max}$ = proportion of state s produced/consumed by task i

Variables

Binary Variables

$wv_{i,j,n}$ = assignment of task i in unit j at event point n
$\mathrm{in}_{s,j,k,n}$ = assigns the material flow of s into storage tank k from unit j at point n
$l_{k,o,n}$ = assigns the starting of product flow out of product tank k to satisfy order o at event point n
$\mathrm{out}_{s,k,j,n}$ = assigns the material flow of s out of storage tank k into unit j at point n
$y_{s,k,n}$ = denotes that material s is stored in tank k at event point n

Positive Variables

$\mathrm{bp}_{s,i,j,n}$ = amount of material s produced task i in unit j at event point n
$\mathrm{bc}_{s,i,j,n}$ = amount of material s undertaking task i in unit j at event point n
dg_o^l = minimum demand quantity giveaway term for order o
dg_o^u = maximum demand quantity giveaway term for order o
H = total time horizon used for production tasks
$\mathrm{JJf}_{s,j,j',n}$ = flow of state s from unit j to consecutive unit j' for consumption at point n
$\mathrm{Kif}_{s,j,k,n}$ = flow of material s from unit j to storage tank k event point n
$\mathrm{Kof}_{s,k,j,n}$ = flow of material s from storage tank k to unit j at point n
$\mathrm{Lf}_{o,k,n}$ = flow of final product for order o from storage tank k at event point n
$\mathrm{mh}_{s,k,n}$ = maximum heel giveaway term for product tanks
$\mathrm{pg}_{s,p,n}^l$ = lower limit giveaway of product quality p
$\mathrm{pg}_{s,p,n}^u$ = upper limit giveaway of quality p for product s
rg_s = minimum demand quantity giveaway term for Group B product s
$\mathrm{Rif}_{s,k,n}$ = flow of raw material to storage tank k event point n
$\mathrm{St}_{s,k,n}$ = amount of state s present in storage tank k at event point n
$\mathrm{std}_{s,s',k,n}$ = amount of state s that is downgraded to state s' in storage tank k at event point n
Tearly_o = early fulfillment of order o than required
$\mathrm{Tf}_{i,j,n}$ = time that task i finishes in unit j at event point n
Tlate_o = late fulfillment of order o than required
$\mathrm{Tos}_{k,o,n}$ = time that material starts to flow from tank k for order o at event point n
$\mathrm{Tof}_{k,o,n}$ = time that material finishes to flow from tank k for order o at event point n
$\mathrm{Ts}_{i,j,n}$ = time that task i starts in unit j at event point n
$\mathrm{Tsf}_{j,k,n}$ = time that material finishes to flow from unit j to tank k at event point n
$\mathrm{Tsf}_{k,j,n}$ = time that material finishes to flow from tank k to unit j at event point n
$\mathrm{Tss}_{j,k,n}$ = time that material starts to flow from unit j to storage tank k

$Tss_{k,j,n}$ = time that material starts to flow from tank k to unit j at event point n

$Uif_{s,j,n}$ = flow of raw material s to production unit j at point n

$Uof_{s,j,n}$ = flow of product material s from unit j at point n

$\alpha_{j,n}$ = for unit j, 1 if the unit becomes active for very first time at event point n

$\beta_{k,n}$ = for tank k, 1 if the tank becomes active for very first time at event point n

$\eta_{s,s',k,n}$ = continuous 0-1 variable, 1 if material in tank k switchover service from s at event point n to s' at later event point

$\eta o_{s,s',k}$ = continuous 0-1 variable, 1 if material in tank k switchover service from s to s'

$\chi_{i,i',j,n}$ = continuous 0-1 variable, 1 if task at unit j changes from i at event point n to i' at later event point.

Literature Cited

1. Jia Z, Ierapetritou M. Mixed-integer linear programming model for gasoline blending and distribution scheduling. *Ind Eng Chem Res*. 2003;42:825–835.
2. Lee H, Pinto JM, Grossmann IE, Park S. Mixed-integer programming model for refinery short term scheduling of crude oil unloading with inventory management. *Ind Eng Chem Res*. 1996;35:1630–1641.
3. Jia Z, Ierapetritou M, Kelly JD. Refinery short-term scheduling using continuous time formulation: crude-oil operations. *Ind Eng Chem Res*. 2003;42:3085–3097.
4. Reddy PCP, Karimi IA, Srinivasan R. A novel solution approach for optimizing crude oil operations. *AIChE J*. 2004;50:1177–1197.
5. Reddy PCP, Karimi IA, Srinivasan R. A new continuous-time formulation for scheduling crude oil operations. *Chem Eng Sci*. 2004;59:1325–1341.
6. Saharidis GKD, Minoux M, Dallery Y. Scheduling of loading and unloading of crude oil in a refinery using event-based discrete time formulation. *Comput Chem Eng*. 2009;33:1413–1426.
7. Kelly JD, Mann JL. Crude-oil blend scheduling optimization: an application with multi-million dollar benefits: Part I. *Hydrocarbon Process*. 2003;82:47–53.
8. Kelly JD, Mann JL. Crude-oil blend scheduling optimization: an application with multi-million dollar benefits: Part II. *Hydrocarbon Process*. 2003;82:72–79.
9. Joly M, Moro L, Pinto JM. Planning and scheduling for petroleum refineries using mathematical programming. *Braz J Chem Eng*. 2002;19:207–228.
10. Jia Z, Ierapetritou M. Efficient short-term scheduling of refinery operations based on a continuous time formulation. *Comput Chem Eng*. 2004;28:1001–1019.
11. Shah N, Saharidis GKD, Jia Z, Ierapetritou MG. Centralized-decentralized optimization for refinery scheduling. *Comput Chem Eng*. 2009;33:2091–2105.
12. Glismann K, Gruhn G. Short-term scheduling and recipe optimization of blending processes. *Comput Chem Eng*. 2001;25:627–634.
13. Mendez CA, Grossmann IE, Harjunkoski I, Kabore P. A simultaneous optimization approach for off-line blending and scheduling of oil-refinery operations. *Comput Chem Eng*. 2006;30:614–634.
14. Gounaris CE, Misener R, Floudas CA. Computational comparison of piecewise linear relaxations for pooling problems. *Ind Eng Chem Res*. 2009;48:5742–5766.
15. Audet C, Brimberg J, Hansen P, Digabel SL, Mladenovic N. Pooling problem: alternate formulations and solution methods. *Manage Sci*. 2004;50:761–776.
16. Adhya N, Tawarmalani M, Sahinidis NV. A Lagrangian approach to the pooling problem. *Ind Eng Chem Res*. 1999;38:1956–1972.
17. Kelly JD. Logistics: the missing link in blend scheduling optimization. *Hydrocarbon Process*. 2006;45–51.
18. Moro LFL, Zanin AC, Pinto JM. A planning model for refinery diesel production. *Comput Chem Eng*. 1998;22:S1039–S1042.
19. Pinto JM, Joly M, Moro LFL. Planning and scheduling models for refinery operations. *Comput Chem Eng*. 2000;24:2259–2276.
20. Luo C, Rong G. Hierarchical approach for short-term scheduling in refineries. *Ind Eng Chem Res*. 2007;46:3656–3668.
21. Luo C, Rong G. Hierarchical approach for short-term scheduling in refineries. *Ind Eng Chem Res*. 2010;49:4474–4474.
22. Kelly JD. Next-generation refinery scheduling technology. *Presented at the NPRA Plant Automation and Decision Support Conference*. San Antonio, TX, 2003.
23. Ierapetritou MG, Floudas CA. Effective continuous-time formulation for short-term scheduling. II. Continuous and semicontinuous processes. *Ind Eng Chem Res*. 1998;37:4360–4374.
24. Ierapetritou MG, Hene TS, Floudas CA. Effective continuous-time formulation for short-term scheduling. III. Multiple intermediate due dates. *Ind Eng Chem Res*. 1999;38:3446–3461.
25. Kondili E, Pantelides CC, Sargent RWH. A general algorithm for short-term scheduling of batch operations. I. MILP formulation. *Comput Chem Eng*. 1993;17:211–227.
26. Shaik MA, Floudas CA, Kallrath J, Pitz H-J. Production scheduling of a large-scale industrial continuous plant: short-term and medium-term scheduling. *Comput Chem Eng*. 2009;33:670–686.

Manuscript received Feb. 16, 2010, and revision received June 16, 2010.