Reprinted from:
HYDROCARBON PROCESSING®
June 2003, pgs 47–53 and July 2003, pgs 72–79
Used with permission.
www.HydrocarbonProcessing.com

PROCESS/PLANT OPTIMIZATION

SPECIALREPORT

Crude oil blend scheduling optimization: an application with multimillion dollar benefits

The ability to schedule the crude oil blendshop more effectively provides substantial downstream benefits

J. D. KELLY* and J. L. MANN, Honeywell Industry Solutions, Toronto, Ontario, Canada

Economic and operability benefits associated with better crude oil blend scheduling are numerous and significant. The various crude oils that arrive at a refinery to be processed into the different refined products must be carefully handled and mixed before they are charged to the atmospheric and vacuum distillation units or pipestill. Many details are involved in optimizing scheduling of a refinery's crude oil feedstocks from receipt to charging of the pipestills.

Every refinery that charges a mix of crude oils to a pipestill has a crude oil blend scheduling optimization opportunity. Producing and updating a schedule of when, where, what and how much crude oil to blend can be difficult. Although crude oils are often planned, purchased, procured and have a delivery schedule set long before they arrive at the refinery, details of scheduling crude oil off-loading, storing, blending and charging to meet pipestill feed quantity and quality specifications must always be prepared based on current information and very short-term anticipated requirements. However, the rewards for performing better crude oil blend scheduling optimization can be substantial depending on complexity and uncertainty of the particular crude oil blendshop** operation.

Crude oil blendshop scheduling has historically been carried out using fit-for-purpose spreadsheets and simulators with most decisions made manually. The user attempts to create a feasible schedule that meets flowrate and inventory bounds, operating practices and quality targets over a near-term horizon, typically 1 to 10 days (up to 30). Actual schedule length depends on the reliability or certainty of the crude oil delivery quantities and timing, and the pipestill production mode runs. Farther into the future, the input data become more uncertain and the scheduling work is typically cut off.

The goal of production scheduling optimization is to automate many of these manual decisions by taking advantage of recent advances in computer power and mathematical programming codes and solving techniques. The main advantages of this approach are that many thousands of scheduling scenarios can be evaluated as part of the optimization in comparison to perhaps only one schedule found by a user, a substantial reduction in time required to generate better schedules and the ability to incrementally rerun the optimization when different what-if scenarios are required (i.e., evaluating distressed crude oil cargos).

The focus of this article is four-fold. First, we delineate the business problem of crude oil blend scheduling using a simple but revealing motivating example. Second, we highlight the hard and soft benefit areas of improved blend scheduling to "whet the appetite" for the impending details. Third, we provide a description of the new scheduling approach for crude oil blending including the theory, explicit problem formulation and related aspects such as how to segregate crude oils when there are not enough tanks for dedicated storage. And fourth, we discuss key elements of the scheduling solution to enlighten the reader on the nuances and the challenges of solving large combinatorial problems.

Before further discussion, it is important to highlight the difference between production planning and scheduling and to discuss the underlying need for continuous improvement of the scheduling function. There will always be a planning activity and a scheduling activity. Together, they form a hierarchical decision-making framework that is very much a part of the organizational structure of every corporation.

Planning is *forecast driven* and typically *aggregates* resources such as equipment, materials and time to model and solve the breadth of the problem. Planning generates simplified activities or tasks consistent with these aggregations. Scheduling is *order driven* and uses the *decomposed* equipment, materials and time to model and solve the depth of the problem. It should be appreciated that a great number of planning decisions are made long before any scheduling decisions are generated. This implies that good scheduling can only result from good planning.

The subtlety between forecasts and orders must also be appreciated. At the planning stage firm and reliable customer orders are rarely available over any significant planning time horizon except

* Author to whom all correspondence should be addressed.

** The term blendshop is used to describe the network of equipment and piping found in a refinery to support handling and blending of crude oils or refined products such as gasolines and distillates.

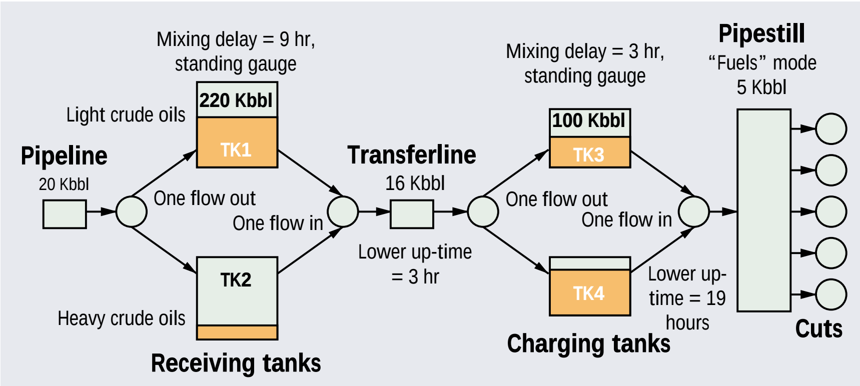

FIG. 1. This blendshop problem involves one pipeline, two receiving tanks, two charging tanks and one pipestill.

Blendshop example. *Marine-access* blendshops are usually characterized by having a set of storage or receiving tanks and a set of feed or charging tanks with either a continuous- or batch-type blend header in the middle. *Pipeline-access* blendshops often only have receiving tanks because settling of the unloaded crude oil for free water removal after a marine-vessel has unloaded is not required.

Fig. 1 shows a small blendshop problem with one pipeline, two receiving tanks, one transferline (or batch-type blend header), two charging tanks and one pipestill. Tank inventory capacities and names are shown inside the tank objects and flowrate capacities of the semicontinuous equipment are shown above the pipeline, transferline and pipeline objects. The bold arrows in the figure indicate the flow or movement variables for the blendshop scheduling problem. Connections between equipment are typically material based in the sense that only certain crude oils or mixtures are allowed to flow between a source and destination. These material based connections permit use of crude oil segregations or pooling by directing certain crude oils to be stored into specific tanks. Segregations are useful when controllable equipment or movements can be used to prepare a blend recipe or formulation to be specified for charging the pipestill.[2] Segregations are useful to reduce problem complexity by reducing the number of decisions to be made in terms of where crude oils should be stored. In our example we do not impose any batch recipe such as a 50:50 blend for flow out of each segregation. That is, we do not impose a 50% volume fraction from the light crude oil pool (TK1) and a 50% fraction from the heavy crude oil pool (TK2). We have omitted this detail so as not to detract from the main focus of the article.

Table 1 provides the information on crude oil receipts for four different types of crude oils that are segregated into light and heavy crude oil pools (i.e., crude oils #3 and #4 are light and #1 and #2 are heavy). The start and end times are in hours from the start-of-schedule which is set at the zero hour; the end-of-schedule is on the 240th hr (10 days). Crude oil mixture liftings from the charging tanks to the pipestill are continuously set at 5 Kbbl/hr and the pipestill's *fuels* production mode is unchanged over the scheduling horizon. This defines the crude oil mixture demand schedule.

Fig. 1 also describes various operating rules that must be respected for the blendshop for it to operate as a crude oil blendshop (more details on these and others are given in Part 2). For this blendshop we only allow one flow out of or in to the pipeline and transferline at a time. Also, the tanks must be in standing-gage operation where there can be only flow in or out at a time but not both. A generalization of standing gage is the "mixing-delay" restriction. It imposes a time delay after the last flow into a tank has finished before a movement out is allowed. The receiving tanks have a 9-hr delay and the charging tanks a 3-hr delay.

The last logic constraint for this example is that all flows are semi-continuous or disjunctive. This means that a flowrate must either be zero or lie between lower and upper bounds.

Table 2 displays a subset of the assay information for the four crude oils being stored and delivery into the blendshop. Only three cuts are included: whole crude oil, kerosene and heavy gas oil. Blending of the cuts and assigned properties are based on volume or weight depending on the property. Blending numbers or indices would be

TABLE 1. **Crude oil receipt or supply orders over scheduling horizon (cycle data).**

Crude Oil #	Start time (hr)	End time (hr)	Flowrate	Duration	Flow	Valid destination
1	82	91	20 Kbbl/hr	10 hr	200 bbl	TK2
1	163	172	20	10	200	TK2
2	7	16	20	10	200	TK2
3	43	52	20	10	200	TK1
3	221	230	20	10	200	TK1
4	132	141	20	10	200	TK1

for wholesale agreements or contracts. Forecast or best-guess aggregate demands and capabilities are used to optimize projections of plant operations. Thus, the plans generated are used to set directions and not production orders. Scheduling on the other hand is primarily based on orders. Orders are much more concrete both in quantity and time and have the highest reliability for the immediate future. Scheduling generates detailed tasks and activities to meet the immediate orders and scheduling is typically updated whenever significant changes to the order or plant capabilities occur.

There is always a requirement to continuously improve scheduling utility. The underlying driving force for this is related to the notion of innovation in industry. Three known innovations are outlined by Norman *et al*.[1] The first is the manufacture of *replaceable or interchangeable parts* that comprise the bill-of-materials of any given product. The second is to produce many products within a single facility and is sometimes referred to as *product diversification.* And the third innovation, the one being implemented today in industry, is *production for final demand* or *demand-driven production* (DDP). This is the prevailing concept that we should only produce product that satisfies actual product demand (demand orders). Speculative or provisional production must be inventoried and, hence, can be considered to be inefficient and potentially risky because a real customer's purchase order is not secured. Many popular production paradigms such as just-in-time (JIT), Kanban, tach time, single minute change of a die (SMED), theory-of-constraints (TOC), etc., are all examples of striving toward the goal of DDP. The scheduling optimization solution presented in this article is a critical step in the on-going struggle to improve efficiency and profitability of an oil refinery or petrochemical plant with respect to the DDP innovation.

TABLE 2. **Crude oil assay information (model data).**

Cut/property	Crude oil #1	Crude oil #2	Crude oil #3	Crude oil #4
Whole crude oil/specific gravity	0.870	0.872	0.856	0.851
Kerosene/yield, vol%	9.26	8.71	10.44	9.60
Kerosene/pour point	–27.0	–42.0	–31.0	–27.0
Heavy gas oil/yield, vol%	9.16	9.75	10.17	9.51
Heavy gas oil/specific gravity	0.864	0.878	0.859	0.849
Heavy gas oil/sulfur, wt%	1.83	0.77	1.67	1.28

TABLE 3. **"Fuels" production mode cut/property specifications (model data).**

Cut/property	Minimum	Target	Maximum
Whole crude oil/specific gravity	–	0.863	–
Kerosene/yield volume	–	–	–
Kerosene/pour point	–	–30.833	–
Heavy gas oil/yield volume	–	–	–
Heavy gas oil/specific gravity	–	–	–
Heavy gas oil/sulfur	–	1.508	–

TABLE 4. **Crude oil opening inventories and compositions (cycle data).**

Tank	Inventory	Crude oil #1	Crude oil #2	Crude oil #3	Crude oil #4
TK1	100 Kbbl	0%	0	100	0
TK2	3	100%	0	0	0
TK3	50	100%	0	0	0
TK4	100	50%	0	50	0

used for those properties that blend nonlinearly such as Reid vapor pressure (Rvp) and viscosity. Synergistic and antagonistic nonlinear blending such as evidenced for octane are not considered further for the crude oil blendshop problem.

Table 3 displays the minimum, target and maximum quality specifications for the mix of crude oils required by the fuel's operating mode. For this example we do not concern ourselves with bounds on the quality variables though the scheduling optimizer can be configured to respect these bounds over the scheduling horizon. The target values are those typically found in the planning optimizer. Table 4 completes the required information for the blendshop by providing the opening tank inventories and crude oil compositions in each tank.

The information provided for the example characterizes a simple yet typical crude oil blendshop scheduling optimization problem. It has been deliberately organized into two main data themes: *model* and *cycle*. Model data are generally static and do not change within the scheduling time horizon. They define the material/flow/capacity network, desired operating logic and crude oil assay information and mixture quality specifications. Cycle data define the dynamic data that can change every time the next schedule is made such as tank opening inventories and compositions, supply, demand and maintenance orders, and any actual or logged movements. The problem's model and cycle data can be further segmented into what we call the *quantity, logic* and *quality* aspects of the problem and will be discussed in Part 2.

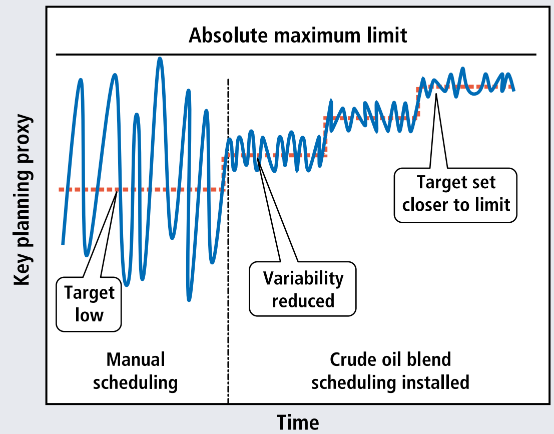

FIG. 2. **Quality variability was reduced with automated crude oil blend scheduling.**

Use of the word cycle is taken from the well-known hierarchical planning and scheduling philosophy of Bitran and Hax[3] who advocate a rolling-horizon framework or scheduling cycle to mitigate uncertainty due to such effects as order reliability, measurement inaccuracies and execution errors. For more recent details see S. C. Graves in the *Handbook of Applied Optimization,* 2002.[4]

Potential benefits. Of course it makes sense to pursue only those aspects of a solution to a problem that have value. Before we describe how we formulate and solve the crude oil blend scheduling optimization problem it seems prudent to analyze why we would want to solve it. This leads into the discussion of expected benefits.

Three major types of disturbances arguably affect a refinery at any time during its production or operation: *crude oil mixture quality variability, ambient temperature changes* and *unreliable* or *faulty equipment.* Processing equipment malfunctions can cause serious production outages and safety concerns, and are normally mitigated by sound maintenance practices. Seasonal and diurnal ambient temperature swings also disturb stability of operations and are mitigated by providing increased cooling or heating capability and improving controls.

The molecules from each crude oil receipt are eventually processed at every unit within a refinery. Variation in crude oil mixture quality charged to the pipestill is perhaps the single most influential disturbance to a refinery. It is the foremost reason why reducing variability around the many quality targets by blending crude oils is of tremendous importance.

Crude oil blendshop scheduling optimization is a relatively inexpensive and timely way to seriously improve performance of almost any refinery. Five benefit areas are all aided by applying better blend scheduling:

Reducing quantity and quality target variability—As mentioned, reducing quality target variance should be at the top of the list for refinery improvements. Deviations from quality targets should be minimized to charge the pipestill with a steady mixture of crude oil. Steadier quality crude oil mixtures charging the pipestills will also translate into steadier operations for downstream production units.

It also makes good sense to run pipestills with a constant flowrate for as long as possible. An example of an improved quality target or key planning proxy is shown in Fig. 2.

Improving the ability to generate more than just feasible schedules—For those blendshops that are tightly resource constrained due to previous cost-cutting initiatives, it may be arduous to generate a feasible schedule for the immediate future. For these blendshops it is valuable to have an automated scheduling application generate in seconds what would take a human scheduler hours to construct. Multiple "better-than-feasible" or what we call optimized schedules may be presented that meet the production goals for selection by the scheduler. The effect of not being able to generate feasible schedules means that either the supply scenario must be changed by distressing crude oil deliveries or demand must be altered by decreasing or increasing the flowrate of crude oil mixtures charging the pipestill. Unfortunately, both of these alternatives are undesirable for various reasons.

Rapid acceptance and inserting spot supply and demand opportunities—Typically, a refinery will be some mix of contract (wholesale, nondiscretionary, strategic or base) versus spot (retail, discretionary, tactical or incremental) crude oil purchases. Faster and better ability for a refinery to assess whether a particular spot crude oil purchase will result in a feasible operation the better the refinery can capitalize on short-term market opportunities.

Consistency of schedules—A common problem in production scheduling is that usually only one skilled scheduler can schedule a refinery's crude oil blendshop effectively. When the main scheduling individual is sick or on vacation it is very difficult to backfill with another appropriately trained scheduler. Hence, if this occurs, schedules generated by the two individuals can be widely different to the point where unfortunately schedules generated by the relief may actually be infeasible. Using an automated scheduling tool alleviates some of these issues since schedules are made to satisfy the same business logic and reflecting the same constraints and limitations.

Production schedule visibility throughout the refinery—Finally, given the use of Internet technologies, it should be standard now and in the future to disseminate the official schedules online so that managers, operators and engineers can all view the same production program for the next several days or weeks. Although this can be easily accomplished with spreadsheets and simulators, it is not always possible with these solutions to look out into the future many days or weeks and show the longer-term schedules to those who can take advantage of greater look-ahead.

If we take for the purpose of discussion a medium-sized 100,000 bpd refinery, or equivalently 35,000,000 bpy with a 350-day yr production schedule, it is possible to make a list of some of the expected benefits and their value. Here we only detail the tangible benefits. However, a range of intangible benefits can translate into significant value. Each benefit cited is incremental over what would be achievable using spreadsheets or simulators.

Quality target variability improvement such as whole crude oil sulfur: *$2,000,000/yr.* This number was estimated from the benefits identified when a similar application for crude oil blend scheduling was developed and applied by the first author to a sweet crude oil processing 100,000 bpd refinery. The benefits were captured at the planning feedstock selection activity level because the proxy constraint on bulk crude oil sulfur was raised from 0.55 to 0.85 %wt sulfur over a three-month period (Fig. 2). This resulted in a cheaper slate of crude oils being purchased while still being able to meet the quality specifications for all of the finished products refined and blended.

Reduced chemical injection: *$100,000/yr.* Further and unexpected savings on corrosion control chemicals were also observed for the refinery over a one-year period due to the fact that less inhibitor needed to be added given the improved regulation of crude oil bulk sulfur concentration.

Distressed sale of crude oils from the refinery: *5-incidents/yr × $1.30/bbl × 50,000 bbl = $325,000/yr.* Here we assume that five times in one year the refinery needs to distress a 50,000 bbl batch of crude oil at a loss, in terms of lost netback production (opportunity) and loss in selling price of $1.30 per bbl. This means that the refinery lost the opportunity to process the crude oil and make netback $1.00/bbl and sold the crude oil batch at a loss of $0.30/bbl.

Pipeline penalty charge for changes in sequence or timing: *3 incidences/yr × $25,000/incidence = $75,000/yr.* A penalty of $25,000 per incidence is realized for altering the start or end time of a batch of crude oil before it can be received at the refinery.

Spot opportunity for crude oil trades: *5 opportunities/year × 50,000 bbl × $1.00 net margin/bbl = $250,000/yr.* There are five extra opportunities per year to run batches of 50,000 bbl at a netback or net margin of $1.00/bbl due to better crude oil blendshop scheduling.

Reduced working capital (decommissioning of a crude oil tank): *50,000 bbl cycle stock × $20/bbl × 10% cost of capital/yr = $100,000/yr.* This is a result of sustaining a long-term reduction of safety or cycle stocks of reserve crude oil in the crude oil blendshop.

Total savings: *$2,000,000 + $100,000 + $325,000 + $75,000 + $250,000 + $100,000 = $2,850,000/yr!* These hypothetical and somewhat anecdotal benefit calculations translate into over two-and-a-half million dollars worth of the potential savings to the refinery profit and loss that would not have been achieved with manual scheduling alone. It must be emphasized that this number only provides a benchmark or yardstick to "Pareto," at least qualitatively, the priorities in terms of choosing between other possible and competing capital investment projects at the refinery. From the perspective of overall crude oil costs of the refinery over one year, this $2.85 million in savings is less than 0.41% of the total feedstock cost (i.e., $2,850,000 / (35,000,000 bbl) / ($20/bbl) × 100 = 0.407%).

Formulating problem logistics and quality details.

Modeling the crude oil blendshop is the cornerstone of being able to capture the potential benefits outlined. Although the modeling must ultimately reside as a collection of complex mathematical expressions relating variables and constraints to offer some level of optimization, we supply a qualitative description of the model only. We do this so as not to detract from the general understanding of the overall problem and reasons for solving it. Some of the more specific details around the mathematical modeling of the crude oil blendshop scheduling optimization problem can be found in Lee *et al.*,[5] Shah[6] and Jia *et al.*[7]

At the core of our formulation is the hierarchical† decomposition of the problem into *logistics* and *quality* subproblems. The logistics subproblem is very similar to the supply chain logistics problem except that our logistics problem has less spatial scope. It considers only the crude oil blendshop (inside the production chain) and not the entire supply chain but has more of an in-depth operational view of the crude oil handling and blending. The logistics subproblem only

† Use of the word hierarchical means we decompose the problem not along the spatial or temporal dimensions but along the decision-making dimension such as planning then scheduling then execution.

considers the *quantity* and *logic* variables and constraints of the problem and ignores the other *quality* variables and constraints. The quality subproblem is solved after the logistics subproblem whereby the logic variables are fixed from the logistics solution and the quantity and quality variables are adjusted to respect both the quantity and quality bounds and constraints. The quality optimizer is very similar to commercially available oil refinery and petrochemical planning software formulations which are used to select the crude oils, for example, that will be processed at the refinery.

The reason for the logistics and quality subproblem decomposition is three-fold. First, commercially available optimization software, optimization theory and computer horsepower have not progressed to the point where we could solve a full-blown simultaneous quantity, logic and quality crude oil blendshop scheduling problem in reasonable time. Second, theory tells us that if you cannot find a feasible solution to the logistics subproblem then you will not be able to find a feasible solution to the quality subproblem. Hence, the decomposition provides a very useful problem-solving aid because if the logistics subproblem is infeasible for whatever reason (i.e., bad input data, overly aggressive production plans, etc.) then there is no use spending time solving the quality subproblem until something has changed in terms of the quantity and logic aspects of the problem. However, even if the logistics subproblem is feasible there is no guarantee that the quality subproblem is feasible. Third, theory also tells us that if the logistics subproblem is globally optimal (i.e., the very best solution has been found) and the quality subproblem is feasible then we have found the global optimum for the overall problem. Hence, our decomposition provides a very powerful structure given that it is easier to check for quality feasibility than quality global optimality after the best logistics solution has been found. Unfortunately if the quality subproblem is infeasible then there must be a mechanism to send back special constraints to the logistics subproblem to force it away from those regions of the search space that are known to cause quality infeasibilities.

We find that this break down of the overall problem into two subproblems is in fact very intuitive for the scheduling users who are using spreadsheets, especially the aspects of the quantity and quality. The logic details are known by the users but rarely included in formulating their spreadsheets due to the discrete nature (i.e., requires some search mechanism and/or trial-and-error). Logic aspects are usually resolved ad hoc after a quantity-quality solution has been circumscribed, but only for the immediate near term of the schedule.

There could also be further decomposition within each of the logistics or quality subproblems. For example, in the logistics subproblem it is possible to decompose the scheduling into assignment and sequencing stages. The assignment stage can be solved relatively easily to optimality by assigning orders or jobs to equipment and ignoring the equipment job sequencing. The assignment decisions are then fixed and the sequencing stage is solved. If a feasible sequencing solution can be found then the overall logistics subproblem is feasible. If the lower-level sequencing is infeasible then extra constraints are added to the assignment stage problem to guide the higher-level solution away from those assignments that are known to cause problems for the sequencing.[8]

A strong parallel to the logistics and quality decomposition is found in discrete parts manufacturing refined by the Japanese. That is the decomposition of JIT, Kanban, SMED, TOC, etc., with the statistical quality control philosophies of W. E. Deming and S. Taguchi. A clear separation between the two is brought together in the end to guide the manufacturing machine to produce quality products efficiently, effectively and punctually.

Finally, formulating crude oil blend scheduling optimization is in the class of production scheduling known as a *closedshop*.[‡] Definition of a closedshop, and its counter-part an *openshop*, can be found in the review paper by Graves.[9] In an openshop all production orders are by customer request and no inventory is necessarily stocked. In a closedshop all customer requests are serviced from inventory and a production activity is generally a result of inventory replenishment decisions. These definitions really state that closedshops involve quantity variables and inventory balances whereas openshops typically don't. Closedshops are generally associated with *lot-sizing* problems (requiring a flow path or network to be defined) and almost always are formulated using some form of scheduling horizon segmentation into time periods. Even when continuous-time closedshop formulations are used (see Jia *et al.*)[7] the number of time-event points is required as an input. Segmenting the time horizon is necessary to perform inventory or material balances. Solving the industrial-scale closedshop problem has been attempted by first determining lot, batch or blend sizes and then making the assignment, sequencing and timing decisions (or logic decisions) using an openshop framework. However, this decomposition has been met with limited success. A unique formulation of the logistics subproblem is to model and solve the blendshop as a closedshop explicitly by including both quantity and logic decisions simultaneously in one optimization.

To facilitate more specific information on the formulation we must first talk about the problem variables. These can be classed into continuous and combinatorial variables. Continuous variables are the quantity and quality variables, and the combinatorial variables are the logic or discrete variables. There are also auxiliary or intermediate variables such as startup (and shutdown or switchover) and flow times yield variables that are used to support solving both the logistics and quality subproblems. Bounds and constraints associated with these variables follow.

Quantity details (hydraulic capacities). There are essentially three types of hydraulically related quantity bounds: flowrate, flow and inventory. Each of these has continuous variables associated with them in both the logistics and quality optimizer formulations. All inventory variables are related to the flows through the material balances on each piece of equipment.

Flowrate bounds are capacity bounds associated with a movement's process and transfer-type equipment such as pipestills, headers, line segments, pumps, valves, etc. They specify how much material can flow within a certain amount time through the piece of equipment and are defined by an upper and lower bounds.

Flow bounds specify a quantity of material that can be transferred from one piece of equipment to another. They extend the flowrate bound to fully describe a supply or demand order. Knowing the rate and the quantity determines the duration. Both flow and flowrate bounds are associated with a connection between a source and destination piece of equipment and ultimately relate to the underlying limiting or shared transfer-type piece of equipment that moves the material from the source to the destination.

Inventory bounds are capacity bounds for inventory-type equipment such as spheres, tanks or drums. They specify how much material can be stored in a piece of equipment and are defined by an upper and lower bound.

‡ A blendshop is a classed as closedshop.

Logic details (operating rules). Fourteen different kinds of logic constraints are typical of a crude oil blendshop operation. This list is not exhaustive but is a very reasonable starting point. As mentioned, to model these constraints we need to have logic variables or combinatorial variables. These are also referred to as 0-1 or binary variables and are associated specifically with a flow between source and destination equipment. Zero indicates the flow is inactive and one implies the flow is active and must be between its lower and upper flow bounds. We also have two other logic variables to indicate when a flow route has been started up (time it is made active) or has been shut down (time it is been made inactive); these variables are also used to model transition or switchovers.

Semicontinuous (SC) constraints represent a flow that can be zero or between a lower or upper bound. Without SC constraints the logistics problem would become a linear program and not a mixed-integer linear program.

Standing gage (SG) constraints enforce the practice for tanks where there can be flow in or out but not both at the same time (mutually exclusive). SG constraints are useful to decouple the production chain from the supply chain upon receipt of a crude oil delivery for example, and to enable tank level differences to be used as a cross check for custody transfer meters.

Mixing delay (MD) constraints restrict flow out of a tank until a certain amount of time has past after the last flow in. A tank must have SG constraints set for mixing delay to be used. The MD constraints are useful to allow separating ballast-free water after a marine vessel unload.

One flow in (OFI) constraints prevent more than one flow into a piece of equipment at a time. OFI constraints are useful to model cases where a pipestill can only be fed from one tank at a time for example.

One flow out (OFO) constraints prevent more than one flow out of a piece of equipment at a time. OFO constraints are useful to model cases where a pipeline can only discharge to one tank at a time.

Contiguous order fulfillment (COF) constraints define a flow to be fixed quantity and fixed rate over specified start and end times. They are typical of pipeline receipt and delivery orders. In these cases the flowrate is equal to the quantity divided by the difference between the end and start times. These order fulfillment types are such that there is a contiguous or consecutive flow between the order start and end time (i.e., an uninterrupted or non-preemptive flow).

Noncontiguous order fulfillment (NOF) constraints define order fulfillment to be the opposite of the COFs. Arrival and departure dates are specified for supply orders and release and due dates are specified for lifting orders. The NOFs are defined with a specified order quantity such that between the arrival and departure date or release and due date, the cumulative quantity of material that has flowed from the pipeline or to the pipestill equals that specified by the order. This implies that there can be noncontiguous or nonconsecutive flows from or to a piece of equipment (i.e., an interruptible or preemptive flow). Arrival and departure dates are useful for handling marine vessel unloading when arrival due to inclement weather conditions causes higher than normal uncertainty levels. Release and due dates are useful for specifying pipestill production mode orders because the planning solutions will typically say how much crude oil to process within a particular time horizon with the detailed flow scheduling to be determined by the scheduling optimization program.

Lower up-time (LUT) constraints are identical to minimum production run length-type constraints. They are used to specify a minimum time a particular movement needs to be up or active before shutting down or becoming inactive.

Upper up-time (UUT) constraints are used to specify a maximum contiguous time a movement can be up before it is required to be shut down.

Equal flow (EF) constraints force the same flow value for a collection of time periods where the movement is contiguously or consecutively active. Either a lower or upper up-time must be specified before the equal-flow constraints will be added for that particular source-destination pair.

Switch-over-when-empty (SWE) constraints indicate a movement cannot switch over or shut down until tank inventory is less than some specified threshold. This is useful when a charge or feed tank must be near empty before it can be shut down or before another tank can be used to charge the pipestill.

Switch-over-when-full (SWF) constraints are very similar to the SWE constraint where a movement cannot be shut down until the tank is full. This is useful for receiving or storage tanks when being fed from a pipeline because it tries to fill a tank before moving on to another one if the volume or quantity of the delivery order is greater than the available ullage.

Startup opening (SUO) bounds are applied to a particular startup variable for a movement and are used to restrict the time of day when that movement can be started up. For example, it may be useful to only have a switchover to a different tank of crude oils feeding a pipestill during the day shift (between 8:00 a.m. and 4:00 p.m.).

Shutdown opening (SDO) bounds are similar to the SUO except that they tell the logistics optimizer when a possible movement shutdown can occur.

Quality details (property specifications). The intensive properties of the crude oil mixtures charging the refinery must be carefully regulated for the pipestill to meet the downstream quality stipulations or specifications when operated in a particular production mode. These qualities are associated with the temperature cutpoints or cuts of the different hydrocarbon streams being separated by the pipestill and must be modeled as continuous variables in the quality optimizer formulation. Quality balances or equations must be associated with each quality throughout the entire blendshop where we model tanks as perfectly mixed vessels. The quality balances force the subproblem to be nonlinear due to the product of quantity (flow and inventory) times quality. Quality splitter equations model the situation of multiple simultaneous flows out of equipment to ensure that each outlet stream has the same quality as all of the other outlet streams. Following is a somewhat complete list of the many streams produced by the pipestill or atmospheric and vacuum distillation unit with typical properties that could be typically assigned or measured for the pipestill output streams.

Wet or saturated gas cut/properties include both the volume and weight yields of the pure components methane, ethane, propane, iso- and normal-butane, specific gravity, etc.

Light and heavy straight-run naphtha cut/properties include both the volume and weight yields, paraffins, olefins, naphthenes and aromatics (PONA), Rvp, octane, specific gravity, sulfur, etc.

Jet fuel and kerosene cut/properties include both the volume and weight yields, cloud point, freeze point, pour point, specific gravity, sulfur, etc.

Diesels and middle distillates cut/properties include both the volume and weight yields, cloud point, flash point, pour point, specific gravity, sulfur, viscosity, etc.

Heavy distillates cut/properties include both the volume and weight yields, basic nitrogen, metals (nickel, vanadium, iron), refractive index, specific gravity, sulfur (total and reactive), viscosity, etc.

Light and heavy vacuum gas oils cut/properties include both the volume and weight yields, base oils, basic nitrogen, metals (nickel, vanadium and iron), refractive index, specific gravity, sulfur (total and reactive), viscosity, etc.

Vacuum residue or pitch cut/properties include both the volume and weight yields, asphaltenes, base oils, carbon number, metals (nickel, vanadium and iron), penetration, specific gravity, sulfur (total and reactive), viscosity, etc.

Logistics and quality objective function details. Now that we have enumerated the variables and constraints of the two subproblems it is important to talk about the driving force for optimization. This underlying forcing function is the objective function that is continuously being maximized during the course of the logistics and quality searches over the entire scheduling horizon (start-to-end of schedule). Both the logistics and quality objective functions are separated into three terms.

The first term is *profit* defined as revenue of crude oil mixtures minus the feedstock costs of the delivered crude oils and any inventory holding or carrying costs for both types of tanks. The profit function is identical to both the logistics and quality subproblems although the quality profit term can be extended to include individual revenue generated from the cut yield flows. The second term is required to maximize *performance*. Performance for the logistics subproblem is defined so as to minimize the number of active movements and the number of movement startups and shutdowns (i.e., transitions or switchovers). Another term in the performance category is to minimize deviation of any tank inventory from a closing inventory target specified by the user. This is also used in the quality subproblem but is extended to include deviations from user-specified quality targets on the crude oil compositions and cut properties. Ad hoc performance weights are usually used for each performance type and can be tuned based on the priority level dictated by the scheduling user.

The third term is very important when solving real-world problems. Not all input data required to solve for optimized schedules is *good* or free of gross errors (see Kelly[10] for a list of possible sources of error). Therefore, we must always anticipate that some infeasibilities may occur before the data have been optimized and carefully cross-checked for validity. In light of this, all quantity, logic and quality constraints have artificial or *penalty* variables associated with them. Each penalty variable is weighted and minimized in the objective function so that the most important business practices at a site are respected when they can't all be met. If the problem data are free of gross errors or flaws (as some people refer to them) then the penalty variables will be driven to zero by the optimizer meaning all business requirements are satisfied. The penalties are also known in the planning domain as infeasibility breakers or safety valves.

Ultimately, the scheduling optimization objective function is used to balance the three costs of manufacturing: *cost of renewable and nonrenewable resources* (i.e., materials, equipment, labor, utilities, chemicals, etc.), *inventory* (i.e., it costs money to store materials and equipment) and *transitions* (i.e., startups, shutdowns, changeovers, switchovers, sequencing, etc.).

Typical planning optimization systems only include the resource and inventory costs and do not model transition costs. The major reason is due to the mathematical intractability of solving simultaneously for quantity, logic and quality given today's state of optimization technology. Consequently, transition costs are excluded from the planning models and only quantity and quality details are formulated, except for minor logic details concerning cargo or batch size increments for feedstock availability. Because transition costs are relegated to the scheduling layer, all planning solutions are overoptimized. This implies that all *plan versus schedule* or *plan versus actual* analysis will have inherent biases or offsets even if measurement, model, solution and execution errors are negligible[10] and strongly suggests that these biases be interpreted carefully.

Time modeling. Both planning and scheduling involve time considerations. There are principally two types of time modeling. The first and most used and studied is time discretization into predefined fixed duration time periods but not necessarily of equal duration over the scheduling horizon. All activities are defined to start and end at the time period boundaries and are piece-wise continuous over the time period duration.

The second time model is the most elegant and is that of continuous-time modeling whereby activity start and end times are included explicitly as optimization variables. An example of continuous-time formulation of the crude oil blend scheduling optimization can be found in Jia *et al.*[7] Continuous-time models also have the notion of time periods except that these have variable durations determined by the optimizer.

The recognized disadvantage of discrete-time formulations are that they require a large number of time periods to model the smallest duration activities, however, continuous-time modeling enables each piece of equipment to have its own timetable. This removes the need to artificially synchronize all equipment to be on the same timetable and thus reduces the number of logic or binary variables. There are nonetheless advantages of discrete time in that it scales well when long time horizons are required for what-if studies because larger time period durations can be used and it can handle easily time-varying quantity bounds and out-of-service orders. With continuous-time formulations, time-varying tank inventory bounds, for example, require extra binary variables to be generated for the optimizer to assign which time period the tank inventory capacity change is to take place even though we know explicitly the event time of the change. Therefore, for the immediate future, discrete-time formulations seem to have value over continuous-time formulations given the previous discussion, yet in the end both discrete- and continuous-time formulations should be available to the scheduling user.

One final note on time models, the popular distinction now between production planning models and production scheduling models, with underlying structures of the lot-sizing problem, is through the notion of *big buckets* and *small buckets* to discretize time. This can be found in Belvaux and Wolsey[11] who also have LOTSIZELIB, a library of diverse lot-sizing problems. The fundamental difference between big and small buckets, where big buckets are used typically to model planning problems, is that big buckets are those in which several materials can be produced on a *convergent-flow-path*[§] piece of equipment, such as a blend header, during a single time period. Small buckets are typically used to model scheduling problems where only one material can be produced on a single piece of equipment at a time (single-use or unary resource logic constraints). Small time buckets are used to model startups, switchovers and shutdowns as is the case in our formulation of the crude oil blend scheduling optimization problem.

Segregating crude oils into tanks. A salient aspect of crude oil handling and blending is that of segregating crude oils into specific tanks. Segregation is used is to separate disparate crude oil types into different tanks to maintain the flexibility or controllability to blend to specific cut property values (i.e., specification blending as opposed to recipe blending). The first requirement of crude oil segregation is to understand the key cut property constraints.

From a degrees-of-freedom analysis, the number of key constraints must be less than or equal to the number of tanks used to blend the crude oil mixtures (i.e., typically the number of receiving tanks). For example, in the example where there are two receiving tanks, at most two cut properties can be controlled at any given time. Since there is a supply order of 5 Kbbl/hr this reduces the number of degrees-of-freedom by one and hence, only one cut property at any time can be controlled. Once the key cut properties have been identified then the crude oils should be separated according to the level of each in the crude oil. All in all, effective segregation can be difficult to figure out but can be automated following the control and optimization techniques found in Kelly.[2] Usually very simple isolation rules are applied based on crude oil bulk sulfur or density levels. Another relevant reason segregation is used is to reduce complexity of the logistics subproblem. When we preassign specific crude oils into tanks the number of choices where an individual crude oil receipt can be stored is circumscribed by the segregation. In our scheduling formulation we handle segregations by pruning the available connections between pipelines and receiving tanks. For instance, in the example with two receiving tanks and two segregations, light and heavy crude oils, only light crude oils 3 and 4 are allowed in TK1 and only heavy crude oils 1 and 2 are allowed in TK2. Hence, of the possible eight crude oil-based connections (two tanks times four crude oils) only four are allowed.

Continuous and batch blending. When most people think of blending in the process industries they envision simultaneous mixing of the blend constituents or components in some mixer or blend header. This is known as *continuous* blending. When we are solving the logistics subproblem, a fixed recipe or bill-of-materials is required that relates the blend volume size to the fractions of each component material feeding the blend header. This is known as *recipe* blending. In the quality subproblem, *specification* blending is performed whereby the recipe is determined based on the property specifications of the blended product. Continuous blending is relatively straightforward to model because at every time period we impose either the recipe constraints in the logistics optimizer or the quality constraints in the quality optimizer. However, in the quality optimizer, specification blending makes the problem nonlinear.

Batch blending can be considered as the opposite to continuous blending similar to batch distillation or separation. Batch blending mixes the required components sequentially in a destination tank with the components typically being fed one after the other. Both recipe and specification blending are achievable using batch blending similar to the continuous blending. Yet instead of the blending constraints being set-up for each time period, batch blending requires the constraints to be specified over a time window made up of two or more time periods so that the component additions are the equations cumulatively. In our example we employ batch blending at the transferline with the restriction that components can flow into the transferline one at a time. The time window we use for our example is arbitrarily chosen at 20 hr.

It is also important to mention that components included in the blending equations are not the individual crude oils[§§] but the crude oil segregations or mixtures. For instance in our example, the two blending components are light and heavy crude oils.

Last of all, if we could solve the overall problem simultaneously for quantity, logic and quality then we would not have to concern ourselves with the side issues of segregating crude oils into the receiving tanks and specifying a nominal recipe for the blend headers. These aspects would be dealt with effectively by the single optimizer and it would determine where to put the crude oils upon delivery and how much of each crude oil mixture from each receiving tank should be set to through the blend header. The only other effect that would preclude us from achieving almost perfect crude oil blend scheduling optimization would be the type, sequence and amount of each crude oil supply order and potentially the production run schedule on the pipestills. Unfortunately simultaneous quantity, logic and quality solutions are not attainable given the present state of optimization technology and, hence, puts the onus on the scheduling user to properly configure the system to help overcome the solver limitations and to go on to generate better-than-spreadsheet or simulator-type schedules.

Solving the problem for logistics and quality. Since both the logistics and the quality subproblems have been carefully formulated as mathematical programs, solving them using commercially available optimization codes is our next step to achieve better crude oil blend scheduling optimization. From the perspective of finding optimized solutions, we can class all solutions coming out of both the logistics and the quality optimizers as *infeasible, feasible, approximate* (locally optimal) and *globally optimal,* given that both subproblems are known to be nonconvex.[§§§] Infeasible solutions do not satisfy all of the problem constraints, feasible solutions do satisfy all of the constraints, approximate solution are feasible and are deemed to be of reasonable worth (best within some neighborhood) and globally optimal solutions are the best overall. For our purposes we concern ourselves with approximate solutions given that running the optimization searches to find the global optimum may take a very long time (i.e., more time than we are willing to wait for an answer).

Before we begin our discussion on solvers, an essential procedure known as *roll-forward* is required to determine tank opening inventories and compositions at start-of-schedule (SoS).

[§] Convergent-flow-path types of equipment can consume many inlet materials and produce only one outlet material. Conversely, divergent-flow-path units can consume many materials and produce two or more outlet materials.

[§§] Not unless a segregation contains only one crude oil.

[§§§] Nonconvex implies that the search space is dichotomous, discontinuous or disjunctive where a locally optimal solution in one subregion is not necessarily the global optimum over all of the other subregions. Convex problems imply that a local optimum is coincident with the global optimum because there is only one region over which to search.

Rolling the information forward to start-of-schedule using simulation. As mentioned previously, advanced planning and scheduling solutions are used within a rolling horizon construct to mitigate inherent effects of uncertainty in the exogenous information of the problem. This is a best-practice policy introduced by Bitran and Hax[3] as part of their hierarchical planning and scheduling approach (see also Clark and Clark[12] for a recent application to the lot-sizing problem). In the context of crude oil blend scheduling optimization the business problem of roll-forward is two fold and is typically carried out every business work day on a daily cycle except of course for weekends. The first operation of roll-forward, using prior opening information and actual movement data, is to predict current or baseline (e.g., 7:00 am) tank inventory data. This is checked to ensure measured data for tank inventories are consistent to what is simulated by the scheduling tool. If there is a discrepancy between the measurements and simulated values then it is up to the scheduling user to resolve the differences by cross-checking with other information.

The second aspect of roll-forward is to predict changes from baseline to SoS to arrive at the initial conditions for the next scheduling optimization. This is also a simulation-type function, which uses in-progress and any future movements that are or will be occurring between the baseline and SoS. Any movement activities that cross the baseline or SoS are truncated and the movement quantity is prorated so that only the amount within the time frame in question is used in the simulation.

Two types of simulation technology can be used to solve the roll-forward function: the sequential modular approach (SMA) or the simultaneous equation approach (SEA). The SMA is sometimes referred to as the *closed-form* approach found in process simulators. It requires external knowledge of the material flow path and simulates each piece of equipment individually, which can be somewhat complex when anywhere to anywhere-type of blendshop networks exist. The main disadvantage of SMA is that it does not handle reverse-, recycle- or recirculating-type flows well and requires an iteration loop to converge when they are present. It has the advantage of being able to handle discontinuous and complex nonlinear functions to model difficult reaction kinetics and fluid mechanics. The SEA is sometimes referred to as the *open form* approach found in process optimizers. It has the disadvantage that all nonlinear equations must be continuous and once differentiable but has the advantage of being able to handle the reversal-type flows easily. The SEA requires the topology to be an implicit part of the model to allow for easy handling of anywhere to anywhere-type of blendshop networks. For our scheduling application we use the SEA. Specifically, the SEA is well suited to crude oil blendshop simulations because we blend or mix linearly by either volume or weight.

Logistics solving methods. Although there is a paucity of literature documenting the quantity and logic formulation of continuous/semicontinuous (CSC)-type processes, there is, however, a remarkable amount of literature on the techniques being used to formulate batch/semi-batch (BSB) type processes both in the operations research (OR) literature and in the chemical engineering journals on process synthesis engineering. That said, the underlying mathematical programming theory used to aid in formulating the crude oil blendshop problem was mostly found in the OR literature[13, 14, 15] and relates to the classic problem formulations of the *fixed-charge network flow, lot scheduling* and *facility location* problems.

At the core of the logistics optimization is use of the branch-and-bound (B&B) search heuristic using linear programming (LP) as the underlying sub-optimization method; this is also commonly referred to as mixed-integer linear programming (MILP). It is well known and can be found in many textbooks. B&B is an exact search method in that if given enough time it would arrive at the global optimum. The B&B begins by solving an LP with all of the binary variables relaxed to lie between zero and one. Then the search begins to successively fix binary variables to either zero or one based on elaborate variable selection criteria and solving an LP for each newly bounded binary variable. After each LP, which are called the B&B nodes, another selection criterion is required to chose which node will be branched on next. The B&B will terminate, kill or fathom a branch of the search tree for two reasons. The first happens when a node along the branch is recognized to be infeasible. The second is called *value dominance* and happens when the node's objective function value is less than the value of the incumbent integer-feasible solution for maximization problems. The incumbent integer-feasible solution is the last solution found that has all binary variables at the extremes of either zero or one. Consequently there is no sense continuing a search on a branch that is infeasible and it does not seem beneficial to follow a branch that is not as good as the current integer-feasible solution found so far. This technique can have other flavors to the search such as breadth first and depth first with backtracking and more details can be found in standard textbooks on integer programming. Moreover, other enhancements to the B&B search include cutting planes, special ordered sets and variable prioritization which in general can speed the search to find good integer-feasible solutions.

Unfortunately even with the most efficient formulation, cleverest B&B search and fastest LP code, finding good integer-feasible or approximate solutions can be very time consuming. Hence, we must be somewhat more pragmatic from the perspective of the quality of the logistics solutions that can be found in reasonable time. To help speed the search, a myriad of heuristics have been the focus of much research in both OR and artificial intelligence (AI). These are referred to as primal and meta-heuristics. Primal heuristics use results of the LP solutions and successively round and fix binary variables to either zero or one. Examples are the pivot-and-complement,[13] relax-and-fix,[15] dive-and-fix,[15] smooth-and-dive[16] and chronological decomposition.[17] Meta-heuristics use a metaphor usually found in nature to devise a search strategy that exploits a particular nuance of the natural mechanism. Examples include the genetic algorithm, tabu search, scatter search, simulated annealing, ant colony optimization and squeaky wheel optimization.

Many other heuristics or approximation algorithms can be found in the OR and AI literature and are basically separated into two categories: *greedy search* and *local search*. Greedy searches are typically used to find quickly integer-feasible solutions in some greedy fashion with a myopic view of the search space. Greedy searches tend to exploit some detail of the problem to enable some fixing of the binary variables. Local searches are basically a refinement on top of greedy searches to try and find better solutions, essentially using a trial-and-error approach, in the neighborhood of the greedy search solutions. An interesting example of local search applied to the lot-sequencing can be found in Clark.[18] All in all, most relatively successful heuristics for practical size problems require some form of a B&B search with backtracking and will typically embed a commercial B&B code in the algorithm.

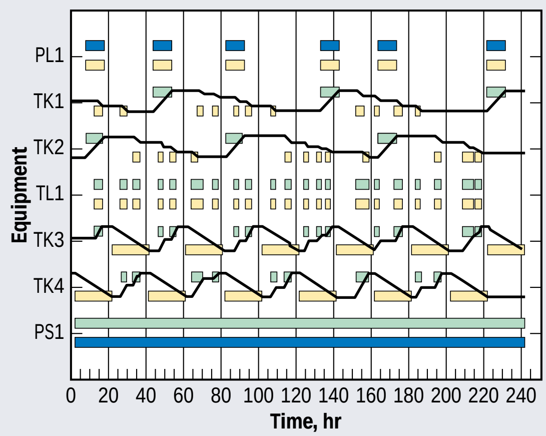

FIG. 3. This Gantt chart shows one penalty-free logistics solution with a 10-day time horizon.

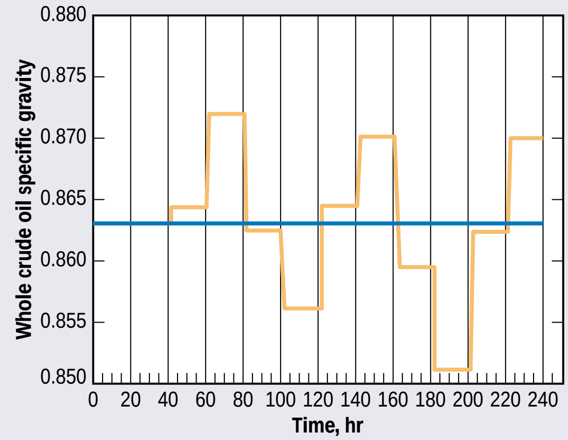

FIG. 4. Trend of whole crude oil specific gravity cut property.

Quality solving method. Solving for the quality variables of the problem is carried out using well-established successive linear programming (SLP). SLP technology is the cornerstone of all solving methods found in oil refinery and petrochemical large-scale planning systems. An example SLP algorithm can be found in Palacios-Gomez *et al.*[19] which in spirit is used by many of the SLP solvers today. Success of SLP as the method of choice for solving industrial-size planning and scheduling arises from it use of the LP. As LP technology improves SLP technology improves because the major iterations of the SLP are simply the LP solutions. Although SLPs are well documented to be more suitable for mildly nonlinear problems with either none or only a few degrees-of-freedom at the optimum (i.e., otherwise known as *superbasic* variables), the maturity of LP technology plays a major role in the SLP success over other nonlinear solvers such as successive quadratic programming or conjugate-gradient methods for example.

One of the biggest advantages is the use of *presolve.*[14] Presolve is applied before any LP is solved and can dramatically reduce LP matrix size (i.e., fewer rows and columns) through clever tightening, consistency and probing techniques, and can remove easily vacuous and redundant constraints and variables; presolve is also used in the MILP solutions. While the other nonlinear solvers could also take advantage of presolve, these nonlinear solvers often do not employ third-party commercial LP codes that have many man-years of development implementing incredibly efficient and fast presolving techniques. A second advantage is the use of interior-point and simplex (both dual and primal) LP solving methods where needed in the SLP algorithm. Since commercial LP codes offer both interior-point and simplex methods, the SLP program can be tailored to use the appropriate LP method at each step. Nonlinear solving codes usually use only one solving technique. For example, for large problems it is appropriate to solve the initial LP using interior-point then any subsequent LP resolves use the dual-simplex; this is also true for MILP problems.

Need for the SLP formulation is of course borne out by the product of quantity times quality or a flow times a cut/yield for instance. When blending is performed linearly either by volume or weight, in the absence of any antagonistic or synergistic effects requiring nonlinear blend laws, this makes the problem both bilinear, trilinear and quadlinear. It's trilinear because of the flow times cut/yield times cut/property and quadlinear because of the density property required when performing the weight balances. Unfortunately this makes the problem nonconvex as mentioned, and to solve it to global optimality necessitates use of global optimization techniques found in Adya *et al.*[20] To solve to global optimality requires a *spatial* B&B search similar to the MILP B&B search except that the branching variables are continuous and not binary. In our case they would be the flow and quality variables. Due to the fact that global optimization is very slow and no commercial software is available, we claim only to search for locally optimal or approximate quality subproblem solutions.

A side benefit to solving for the logistics subproblem first in series, then solving for the quality subproblem, is the actuality that the SLP solves faster than if we were to solve for the qualities first (as in the planning systems that solve for quantity and quality). The reason is that the logistics solution provides us with an excellent starting position or local neighborhood for the flows and inventories. This aids the SLP where it is well known that all nonlinear programs do better when better initial guesses are provided.

Example results. Fig. 3 shows one penalty-free logistics solution with a 10-day time horizon. The blue horizontal bars are the supply and demand orders. The yellow bars are flow out of the equipment and the green bars represent flow into the equipment. The trend lines superimposed on the tank equipment show the inventory profiles that are within the limits of their respective upper and lower bounds. The major ticks on the x-axis are strategically spaced at a distance of 20 hr and the minor ticks are positioned at every 5 hr. The y-axis shows the renewable equipment resources starting from the pipeline at the top down to the pipestill displayed at the bottom. If either quantity or logic penalties were encountered they would be shown as red bars between the flow to and from bars for each equipment. It is clear that because there are no penalties there were no logic constraints (standing gage, mixing delay, minimum up-time, etc.) violated and all inventory and flow bounds were simultaneously respected without incident, i.e., this schedule satisfies 100% of the business practices and needs over the entire horizon.

As can be seen in the figure we have satisfied all of the six supply orders for the pipeline (PL1) and segregations are properly maintained in that only those crude oils that belong to a segregation can

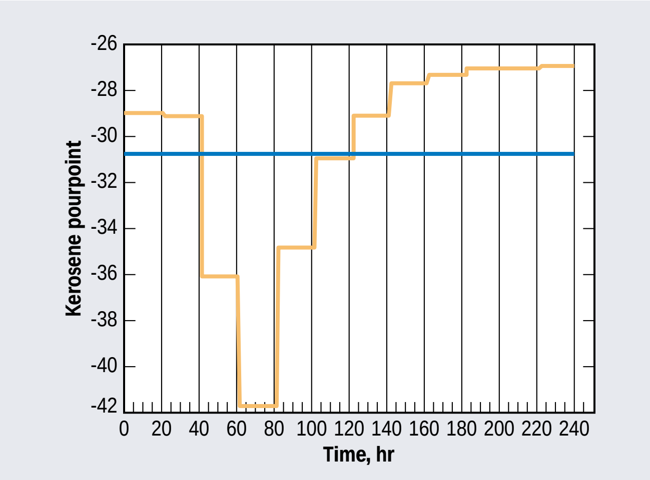

FIG. 5. Trend of crude oil kerosene pour point specific gravity.

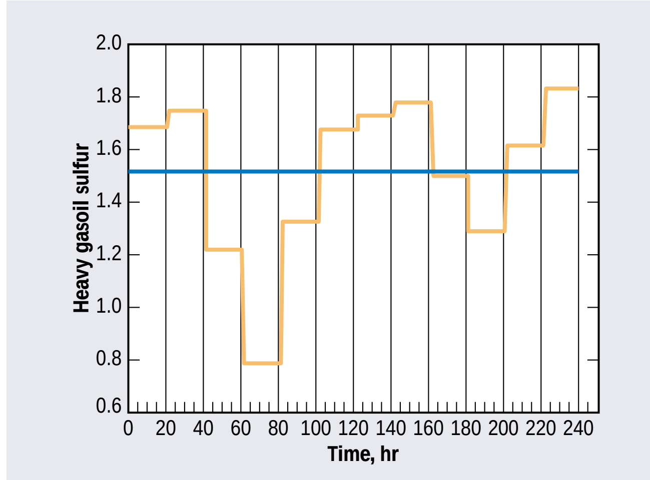

FIG. 6. Trend of heavy gas oil sulfur cut property.

fill a storage tank (TK1 and TK2). Flows from TK1 and TK2 to the transferline (TL1) comply with the 3-hr minimum run length as well as the 9-hr mixing delay specification. To observe mixing delay on tanks count the number of hours from the end of a green in-flow bar to the start of the first out-flow yellow bar. The long run lengths for flows from each of the two feed tanks (TK3 and TK4) charging the pipestill (PS1) also comply with the 19-hr up-time minimum constraint and 3-hr mixing delay. All standing gage restrictions were also obeyed since no green and yellow bars overlap for TK1, TK2, TK3 and TK4. The demand order of continuously charging 5 Kbbl/hr or 120 Kbpd to PS1 was additionally met.

This logistics solution took approximately 60 seconds to generate on a 1-gigaHz PC which involved solving an MILP. No special heuristics except for the default settings in the B&B search were used. Table 5 illustrates the power of presolve. The number of inequalities or rows is reduced by 58% and the number of continuous variables or columns is even more dramatically reduced by 77%. The number of nonzeros in the constraint matrix is correspondingly reduced by 61%. Thus, matrix density has increased or conversely, sparsity has in fact decreased—it has become less sparse after presolve.

The logistics solution was then used as input to the quality optimizer where the same 1-hr time period and 240-hr time horizon were used to generate the quality time profiles. The quality solver took approximately two seconds to solve. In this case study, only an LP is required to solve the quality optimization given that no flows were adjustable and hence, no nonlinearities present; the lower and upper flowrate bounds are equal. Figs. 4 to 6 trend the profiles of whole crude oil/specific gravity, kerosene/pour point and heavy gas oil/sulfur cut/properties respectively for only those flows leaving the charging tanks and entering the pipestill; we do not show any internal flow or tank qualities. The black line is the actual trace of the cut/property and the blue line is the planning proxy target found in Table 3.

For the whole crude oil/specific gravity we observe a step-type function that is due to the business practice of preparing mixes of crude oils in the feed tanks and then charging that mix from one tank at a time, emptying each tank before swinging to the other. The result is an approximate 19-hr run length given the feed tank capacity and pipestill charge rate. The approximately 0.011 maximum excursion from the proxy would be improved by using a 50:50 recipe

TABLE 5. **Logistics optimization problem statistics.**

# Rows	# Columns	# Non zeros	# 0–1 Variables	# SOS1
16391	33025	79611	1500	9540
6814	7651	30745	1500	9540

on the transferline. The 50:50 recipe is driven simply by the fact that the light and heavy crude oil segregations for whole crude oil specific gravities would mix to 0.862 if there were also 50:50 mixes in the receiving tanks of each appropriate crude oil. Unfortunately, this recipe would cause undue variation in the other two qualities. Because there are only two receiving tanks and three qualities, and total flow to the pipestill to potentially respect, at most we could only reasonably control two of the variables. Since throughput is rarely sacrificed for overall refinery stability and profitability, only one quality could potentially be controlled. The other two qualities would display an offset from target (i.e., only one quality can possess reset or integral action in the context of control theory).

The kerosene/pour point trend does not show any obvious cycle or periodicity as seen in Fig. 4 and there is a large excursion from target between hr 60 and 80 when crude oil #2 is delivered at hr 7, with a pour point of –42, starts to percolate through the blendshop to the pipestill. If it were the most important quality bottleneck, there would be four avenues to reduce this variability. The most powerful effect is to change the delivery schedule of crude oils to better manage the pour point quality. This is not always possible nor is it an option unless sufficient lead time is available to the crude oil traders and procurers. The second avenue would be to focus on a segregation recipe that would better control the pour point to the planning or operational target, although as mentioned this would be at the expense of the other two qualities. The third approach would be to alter the segregation policy. The current policy is based on the whole crude oil/specific gravity whereby crude oils #1 and #2 are deemed to be heavy and crude oils #3 and #4 are deemed to be light. If for example kerosene/pour point is the quality of most importance then it would seem prudent to segregate crude oils #2 and #3 together as a *low pour* segregation and crude oils #1 and #4 as a *high pour* segregation. This type of analysis can be found in more detail in Kelly and Forbes.[2]

The fourth avenue along the lines of changing the segregations is to add a third or even fourth receiving tank. This would be an expensive alternative but may provide a level of flexibility and controllability well worth the investment. For instance, if a third tank is added then instead of only being able to control theoretically without offset one quality, we would now be able to control two qualities. With four tanks we would be able to control without offset all three qualities. The somewhat intangible benefit of controlling more qualities implies that the downstream processes will have less disturbances to battle. A more tangible benefit quantifiable by the planning optimizer would be the ability to ride closer and closer to the real refinery constraints or quality bottlenecks as shown in Fig. 2.

Finally, we show the quality profile for the heavy gas oil/sulfur. An interesting consequence of this trend is the periodicity or cycle of the variation (i.e., up-down-up using the blue line as the datum). It appears to be in the range of 100 hr for this set of cycle data. This means that from a production standpoint, the heavy gas oil intermediate tankage must have sufficient capacity to store 100 hr worth of heavy gas oil production. The reasoning behind this is that given the relatively uncontrollable quality variation, due to the inherent delivery schedule disturbances and limitations in the blendshop, we need up and down or positive and negative variation around the planning target over some time frame to acquire reasonably on-specification quality in the intermediate tanks. The best alternative of course is to have constant or steady quality (i.e., the blue line) since shipment or blending of the intermediate can be performed at any time during production and there will be less likelihood of off- or over-specification product.

The next best thing is to have as short a perturbation cycle as possible. And for this example, it would seem that qualitatively we should have larger capacity intermediate tanks for kerosene/pour point than the heavy gas oil/sulfur due to the irregular nature of the pour point trend. This ultimately implies that the more a given cut varies in quality variation the more tankage is required to buffer it so that it is more consistent for blending or as charge to a downstream process unit. Minimal tankage will cause sharp swings in quality forcing controls to react to the upsets.

It should be emphasized that scheduling is an important decision-making tool not just to create pro-forma operational schedules but to also help answer tactical business questions such as "can we run feasibly by trading cargoes and delaying a crude oil delivery of Arabian light by two days" or "can we accept another crude oil delivery three days from now of Arabian heavy to fill out our fluidized catalytic cracking unit?" In the same way planning system users have more than just one planning model such as for facilities, budgetary, feedstock selection and operation, so can many different types of scheduling models be employed to answer these questions timely and accurately. **HP**

LITERATURE CITED

[1] Norman, A., K. Mahmood and M. Chowdhury, "The need for a paradigm for innovation," http://www.eco.utexas.edu/homepages/faculty/Norman/long/InnParadigm.html, Department of Economics, University of Texas at Austin, August, 1999.

[2] Kelly, J. D. and J. F. Forbes., "Structured approach to storage allocation for improved process controllability," *AIChE Journal,* 44, 8, 1998.

[3] Bitran, G. R. and A. C. Hax, "On the design of hierarchical production planning," *Decision Science,* 8, 28, 1977.

[4] Pardalos, P. M., and M. G. C.Resende, (editors), *Handbook of Applied Optimization,* Oxford University Press, London, U.K., 2002.

[5] Lee, H., J. M. Pinto, I. E. Grossmann and S. Park, "Mixed-integer linear programming model for refinery short-term scheduling of the crude oil unloading with inventory management," *Industrial Engineering Chemistry Research,* 35, 5, 1630–1641, 1996.

[6] Shah, N., "Mathematical programming techniques for crude oil scheduling," *Computers & Chemical Engineering,* 20, Suppl. B, S1227–S1232, 1996.

[7] Jia, Z., M. Ierapetritou and J. D. Kelly, "Refinery short-term scheduling using continuous-time formulation—crude oil operations," *Industrial Engineering Chemistry Research,* February 2002.

[8] Jain, V., and I. E. Grossmann, "Algorithms for hybrid MILP/CP models for a class of optimization problems," *INFORMS Journal of Computing,* 13, 258–276, 2001.

[9] Graves, S. C., "A review of production scheduling," *Operations Research,* 29, 4, 646–675, 1981.

[7] Jia, Z., M. Ierapetritou and J. D. Kelly, "Refinery short-term scheduling using continuous-time formulation—crude oil operations," *Industrial Engineering Chemistry Research,* February 2002.

[8] Jain, V., and I. E. Grossmann, "Algorithms for hybrid MILP/CP models for a class of optimization problems," *INFORMS Journal of Computing,* 13, 258–276, 2001.

[9] Graves, S. C., "A review of production scheduling," *Operations Research,* 29, 4, 646–675, 1981.

[10] Kelly, J. D., "The necessity of data reconciliation: some practical issues," NPRA Computer Conference, Chicago, Illinois, November 2000.

[11] Belvaux, G. and L. A. Wolsey, "Lot-sizing problems: modeling issues and a specialized branch-and-cut system bc-prod," *CORE Discussion Paper DP9848,* Universite Catholique de Louvain, February, 1998.

[12] Clark, A. R. and S. J. Clark, "Rolling-horizon lot-sizing when setup times are sequence-dependent," *International Journal of Production Research,* 38, 10, 2287–2308, 2000.

[13] Nemhauser, G. and L. A. Wolsey, *Integer and Combinatorial Optimization,* John Wiley, New York, 1988.

[14] Williams, H. P., *Model Building in Mathematical Programming,* 3rd Edition, John Wiley & Sons, 1993.

[15] Wolsey, L. A., *Integer Programming,* John Wiley & Sons, New York, 1998.

[16] Kelly, J. D., "Smooth-and-dive accelerator: a pre-milp primal heuristic applied to production scheduling problems," *Computers & Chemical Engineering,* 27, 827–832, 2003.

[17] Kelly, J.D., "Chronological decomposition heuristic for scheduling: a divide & conquer method," *AIChE Journal,* 48, 2995–2999, 2002.

[18] Clark, A. R., "A local search approach to lot sequencing and sizing," *IFIP WG5.7 Special Interest Group on Advanced Technologies in Production Planning and Control,* Florence, Italy, February 2000.

[19] Palacios-Gomez, F., L. Ladson and M. Enquist, "Nonlinear optimization by successive linear programming," *Management Science,* 28, 10, 1106, 1120, 1982.

[20] Adya, N., M. Tawarmalani and N V. Sahinidis, "A Lagrangian approach to the pooling problem," *Ind. Eng. Chem. Res.,* 38, 5, 1956–1972, 1999.

J. D. Kelly is a chemical engineer and has a master's degree in advanced process control from McMaster University. He has worked as an advanced control engineer at both Shell Canada and Imperial Oil including implementing real-time optimization programs and tactical planning and scheduling solutions in their refineries. Mr. Kelly has installed plant-wide data reconciliation packages in several oil refineries around the world and he has written many academic publications on the subject. He is now a solutions architect for advanced planning and scheduling at Honeywell Industry Solutions in Toronto, Canada.

J. L. Mann is a chemical engineer with a bachelor of applied science degree from the University of Toronto. He has worked as a design engineer and as a simulation engineer at Imperial Oil, including developing refinery-wide simulation tools to support planning and scheduling activities. Mr. Mann has worked on a number of plant information system projects with a focus on integrating plant data collection systems to plant-wide yield accounting systems. He now is a business architect for Honeywell Industry Solutions in Toronto, Canada.