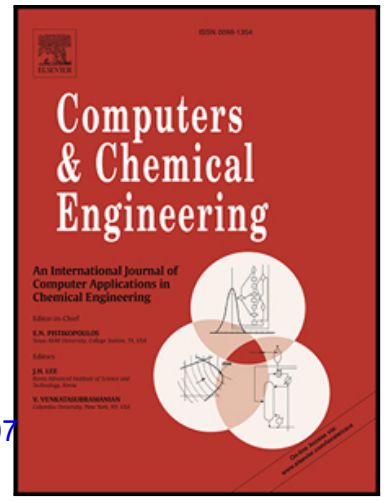

An Integrated Data-Driven Modeling & Global Optimization Approach for Multi-Period Nonlinear Production Planning Problems

C. Doga Demirhan, Fani Boukouvala, Kyungwon Kim, Hyeju Song, William W. Tso, Christodoulos A. Floudas, Efstratios N. Pistikopoulos

PII: S0098-1354(20)30300-8
DOI: https://doi.org/10.1016/j.compchemeng.2020.107007
Reference: CACE 107007

To appear in: *Computers and Chemical Engineering*

Received date: 18 March 2020
Revised date: 12 June 2020
Accepted date: 5 July 2020

Please cite this article as: C. Doga Demirhan, Fani Boukouvala, Kyungwon Kim, Hyeju Song, William W. Tso, Christodoulos A. Floudas, Efstratios N. Pistikopoulos, An Integrated Data-Driven Modeling & Global Optimization Approach for Multi-Period Nonlinear Production Planning Problems, *Computers and Chemical Engineering* (2020), doi: https://doi.org/10.1016/j.compchemeng.2020.107007

An Integrated Data-Driven Modeling & Global Optimization Approach for Multi-Period Nonlinear Production Planning Problems

C. Doga Demirhan[a,b], Fani Boukouvala[c], Kyungwon Kim[b,d], Hyeju Song[b], William W. Tso[a,b], Christodoulos A. Floudas[a,b], Efstratios N. Pistikopoulos[a,b,*]

[a] *Artie McFerrin Department of Chemical Engineering, Texas A&M University, College Station, TX, USA*
[b] *Texas A&M Energy Institute, Texas A&M University, College Station, TX, USA*
[c] *School of Chemical & Biomolecular Engineering, Georgia Institute of Technology, Atlanta, GA, USA*
[d] *Hyundai Oilbank Company Ltd., Seoul, Korea*

Abstract

In this work, we present an integrated data-driven modeling and global optimization-based multi-period nonlinear production planning framework that is applied to a real-life refinery complex. The proposed multi-period framework significantly extends and improves previous works based on single-period planning formulation by better managing and optimizing inventories. The framework features (i) automatic generation of nonlinear and sparse data-driven process models where yields and properties of the process models are based on input properties and compositions, (ii) estimation of model parameters using two years of real-life plant data from the Daesan Refinery in South Korea, and (iii) global optimization of the large-scale nonlinear and multi-period production planning model using commercial global solvers. Computational results for multiple case studies show that the optimal multi-period plans outperform the actual plan by 57-94% in each period.

Keywords: Data-driven modeling; multi-period nonlinear planning; global optimization; real plant data

1. Introduction

Production planning operations in the refining industry focus on both the long-term strategic decisions like purchasing the best crude oil mixture for future plans that cover a few months to a year and the short-term scheduling decisions like allocating streams or deciding on distillation cutpoints that cover the span of several days [1, 2]. Production planning problems are often large-scale optimization problems and the modeling efforts have traditionally relied on linear programming (LP) or mixed-integer linear programming (MILP) principles

*Corresponding author
Email address: stratos@tamu.edu (Efstratios N. Pistikopoulos)

by using fixed-yield planning models and swing-cut models due to tractability concerns [3, 4]. Many refinery operations such as crude distillation, hydrocracking, or hydrotreating, however, are complex processes that display strong nonlinear relationships between process inputs and output. Linear models often fail to capture the inherent nonlinearities in the refinery operations [5].

As a result, there has been significant efforts in the industry and academia to develop nonlinear models for refinery processes since the 1980s [6]. For this purpose, Mobil has developed a proprietary lumping technique called structure-oriented lumping (SOL) in early 1990s to predict physical properties using group contribution methods [7]. SOL models have been used to model the reaction networks in vacuum residua conversion [8] and heavy oil hydroprocessing [9]. Commonly investigated topics by the academia include blending [10, 11, 12], crude distillation, and fluid catalytic cracking processes[13, 14, 15]. Nonlinear models can be first principle-based or empirical [15]. First principle models consist of mass and energy balance equations as well as phase equilibrium conditions along an entire column. Such rigorous models also include flow rates and compositions of all internal and external streams as well as operating conditions such as tray temperatures and pressures. Optimization formulations of such systems present inherently nonlinear and nonconvex problems. Although the computational power has increased tremendously in the last decade and powerful commercial global optimization solvers such as BARON [16] and ANTIGONE [17] are now available, the use of such high-fidelity models in planning optimization is restricted due to the arising computational complexity [5].

Commercially available planning software like Aspen PIMS (AspenTech) and GRTMPS (Haverly Systems) rely mostly on linear models, even though they can handle nonlinear equations and large-scale problems [6]. These tools use sequential linear programming (SLP) techniques to solve nonlinear programming (NLP) models, where NLP models are linearly approximated around a reference operating point. While SLP is a well-established method in the industry to solve large-scale nonlinear problems, it suffers from yielding locally optimum solutions when handling nonconvex NLP models [3]. As a result, the use of nonlinear terms in the such software is somewhat restricted. Moreover, surveys on the use of planning and scheduling software indicate that such tools are inherently challenging to master for the engineers and the economists without proper optimization background [18]. For this reason, stream allocation in chemical industries is, more often than not, made based on company experts' experience and manual calculations [19]. While operator expertise can ensure feasible operation for individual units, without an optimization strategy, it is highly likely that the overall refinery operation can be at best suboptimal or nonprofitable if not infeasible. Therefore, academic literature conducts studies to efficiently incorporate accurate nonlinear models in optimization of the refinery process operations [20, 21]. Studies focusing on nonlinear and mixed integer nonlinear programming (MINLP) problems often also examine necessary global optimization strategies to solve the resulting problems [22, 12, 23, 24].

A promising way to model nonlinearity in complex processes with low computational cost is to use data-driven models [25]. Such models are often referred to as *surrogate*, *meta*, or *input-output* models [26, 27]. Data-driven models can be *grey-box* models with users having some information on the underlying input-output relationships or *black-box*

where the analytical relationships are unknown [28]. These models are used in a plethora of disciplines such as chemical engineering, financial management, mechanical engineering, geosciences, etc. [29, 30, 31, 28]. The analytical form of data-driven models are known and they are especially useful if the rigorous model of a process is computationally expensive or the finding an analytical expression for the required input-output relationship is nontrivial [32]. Data-driven models can be trained with simulation or operational data [33]. Some of the earlier applications of data-driven modeling in refinery processes include use of the non-linear fractionation index (FI) Alattas et al. [34], Yang and Barton [23], which approximates crude distillation as a series of flash separation units or swing-cut models [15]. Data are also used to train empirical nonlinear models for various processes including hydrocracking, hydrodesulfurization, delayed coking, etc. [35]. There are also hybrid models, where mass and energy balance equations are solved simultaneously with product quality constraints [36, 37].

Despite all the interest in improving planning operations in both academia and the petrochemical industries, collaborations are quite rarely published [38]. This limited research is often due to the confidentiality restrictions [39, 40]. In their work, Li et al. [5] use a data-driven approach to optimize the production plan of an existing petrochemical complex in China owned by PetroChina Company Ltd. They use the operational plant data provided by the company to train nonlinear process models. Their work highlights that if companies are willing to share their data with academia, state-of-the-art modeling and optimization techniques can help the industry to improve their operations.

With this study, we propose an integrated data-driven and global optimization approach for nonlinear multi-period production planning that significantly extends the previously proposed framework by Li et al. [5], featuring: (i) automatic generation of nonlinear and sparse data-driven process models where yields and properties of the process models are based on input properties and compositions, (ii) estimation of model parameters using real-plant data, and (iii) global optimization solution strategy of the large-scale nonlinear and multi-period production planning model using ANTIGONE. The automated model selection aims to achieve a certain degree of sparsity in the models. Sufficiently sparse models can improve the computational efficiency in multi-period planning formulations and allow for the use of commercial global optimization software. This integrated modeling and planning approach is first described and later applied to optimize the production planning problem of Hyundai Oilbank Company Ltd.'s Daesan Refinery in South Korea, where the processing units are modeled using the historical operational data provided by the company. Various data processing, model training, and single- and multi-period formulations are analyzed and the optimal production plans for selected days of operation are compared with the actual plan to show the effectiveness of the current approach.

2. Problem Formulation

2.1. Integrated Data-Driven Modeling and Production Planning

The framework is best described in four major steps as illustrated in Figure 1. **Step 1** comprises collection, grouping, and processing of the raw data provided by the industrial

partner. In **Step 2**, data-driven yield and property prediction models for all the processing units are developed. Lasso and elastic net regularization methods are compared to obtain sparse prediction models. In **Step 3**, a process superstructure is generated with all possible connections in the refinery. With the addition of mass balance, capacity, inventory, and demand constraints, lower and upper bound values on all decision variables, and the objective function to the processing models, a discrete-time multi-period planning problem is obtained. Finally, **Step 4** is when the resulting optimization problem, that is a large-scale nonconvex, constrained NLP, is solved to ε-global optimality using commercial global optimization solvers. The details of each step are elaborated in the following sections of this chapter.

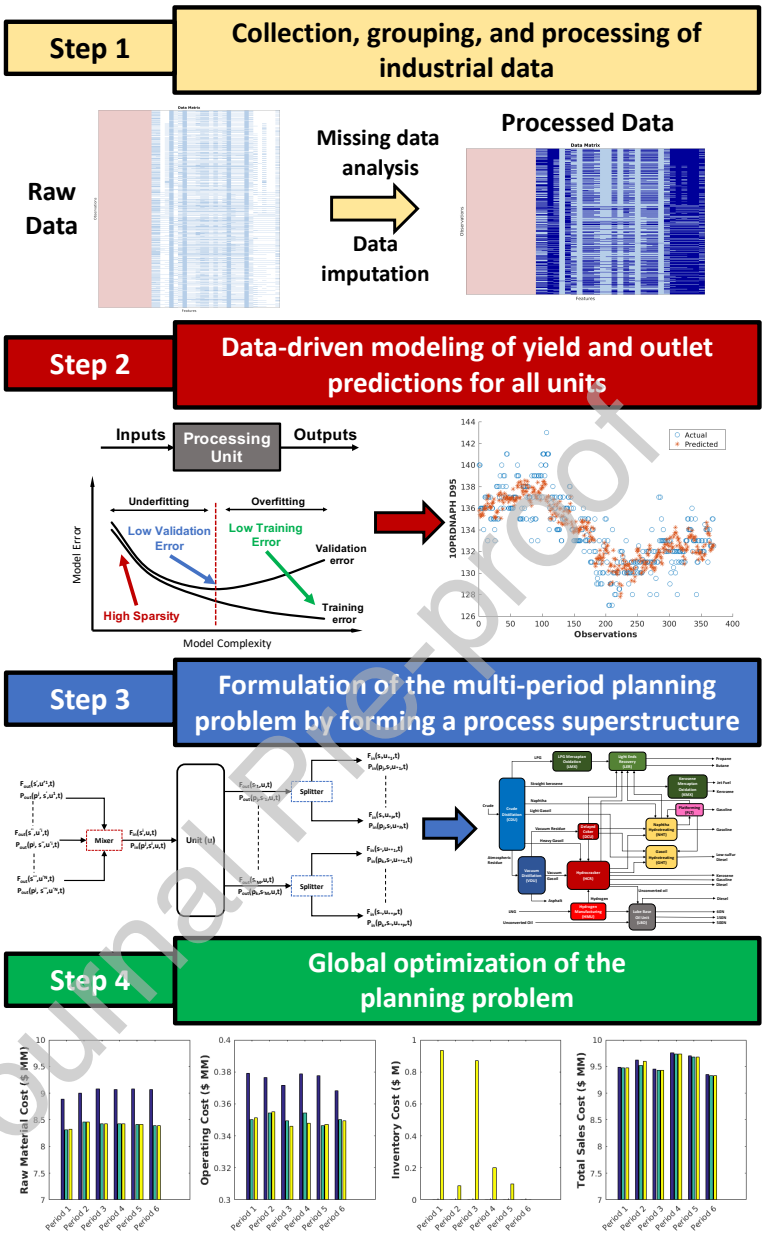

Figure 1: Integrated data-driven modeling and global optimization approach for production planning.

2.2. Data Processing

Data collection, grouping, and processing is the most important step in this framework. Since modeling is purely data-driven, the raw data taken from a plant often need to be checked for consistency and reliability prior to any modeling efforts. The data sets for each processing unit are formed as data matrices where each column consists of either flow rates

or properties (features) and each row consists of the corresponding measurements made at each operating day (observations). Each processing unit has its own data matrix.

The original raw data set consists of various property measurements for every stream in the plant. While using all the stream flow rates is essential to model all the possible connections, reducing the number of property measurements down to the smallest essential amount of features is important in reducing the size of the optimization problem. The initial list of predicted properties for each processing unit are decided by the company experts. Additional properties that do not belong to the initial list are also included to the properties to be predicted, since they need to be later used as input variables to the connected processing units. Once formed, the mass balances and the measurements in the data set are checked for consistency by the company experts. However, this alone is not sufficient to declare the data set as reliable.

The raw data consist of noise and occasional missing points. There are potential reasons listed in the literature for sources of noise in actual data such as variability in operating personnel, data acquisition devices of the processing units, and random error [5]. Missing data are commonly caused by sensor breakdown, data acquisition system malfunction, data recording errors [41], and often by decisions and/or mistakes by the operation personnel [42]. While the stream flow rates and yields are measured every day, some property measurements are only taken on a weekly basis, often after crude oil change. While removing a mostly empty row (or a column) is one way to get rid of the missing data problem [43], more often than not, the missing data points are isolated cells in a big data matrix. Removing an entire row can mean losing precious information on other useful features.

Missing data imputation is a way to regain some of the information missing in the original data set [44]. Previous work by Li et al. [5] uses k-nearest neighbor (k-NN) algorithm to replace the missing data point with the corresponding value from the nearest-column, where the nearest column is the closest column in Euclidean distance. For this purpose, they use MATLAB's *knnimpute* function. In this study, in addition to the k-NN algorithm, available imputation techniques available in MATLAB are tested to find the best performing technique for this data set. The alternative methods are the probabilistic principal component analysis (PPCA) using MATLAB's *ppca* function, the iterative algorithm (IA), the nonlinear iterative partial least squares regression algorithm (NIPALS), the known data regression (KDR), and the trimmed scores regression (TSR) methods. The latter four methods are available in MATLAB through the Missing Data Imputation (MDI) Toolbox that is described in the works of Folch-Fortuny et al. [42, 45]. After the missing data are imputed, the data set is normalized and then ready for the data-driven modeling.

2.3. Data-Driven Modeling and Feature Selection

2.3.1. Model Training

The aim in data-driven modeling is to establish relationships between several explanatory input variables and the response variables that are to be predicted using the available data [46]. All refinery units have significant variety in the outputs with respect to changing input conditions, often showing nonlinear relationships. To capture the relationships we allowed the prediction models to include linear, quadratic, and bilinear interaction terms. Such

models are referred to as quadratic models or polynomial response surface models (RSM). RSMs with higher order polynomial, exponential, or logarithmic terms are also tested in preliminary studies, however, their contributions were found to be insignificant to the model performance. On the contrary, higher order terms often can lead to overfitting considering the existing noise in the data set. Linear, quadratic, and bilinear terms are handled well with commercial global optimization solvers [47, 17] while capturing the underlying trends in the presence of noise and uncertainty [5]. In this study, individual parameter estimation problems are solved for each yield and property prediction model. The inputs are tested for correlation by checking Pearson correlation coefficients. A cutoff value of 0.7 is used for elimination inputs and no significant correlations in the data set are observed.

There are important challenges related to using data-driven models, which need to be addressed in any study. Two important decisions in data-driven modeling are the input-output relationships and the adequate model complexity. There are trade-offs between finding (i) a model that best fits the training data set, (ii) a model that makes the best predictions when tested with data outside the training set, and (iii) a sparse model that ensures computational efficiency when used in an optimization application. In order to use nonlinear models in multi-period optimization problems, it is beneficial to have a model that simultaneously satisfies all the three points to a degree. While objectives (i) and (ii) can often be achieved by k-fold cross-validation (CV) approach, addressing objective (iii) requires a somewhat different approach. Sparsity in models is often obtained by selecting a subset of inputs and omitting a set of predictors in the model. Since the size of multi-period optimization problems scales linearly with the number of periods, sparsity in models becomes crucial to reduce the size of a problem.

There are various variable selection methods in the literature such as Akaike Information Criteria (AIC), Bayesian Information Criteria (BIC), lasso, ridge, or elastic net types of regularization [32], or nonlinear methods such as support vector regression (SVM) [48, 49]. Regularization methods use a nonnegative hyperparameter, the regularization coefficient λ, to penalize the coefficients of the predictors. Lasso regularization penalizes the L^1-norm of the coefficients, whereas ridge regularization penalizes the L^2-norm. Lasso method pushes the coefficients to be zero, effectively eliminating predictors, if they are not relevant. While ridge method penalizes large values of coefficients, it does not necessarily push them to be zero. Elastic net is a mixture of both methods, having both L^1- and L^2-norm in its objective function to be minimized [50]. Regularization is a fast method and easy to implement. The objective of the regularization problem is given in Equation 1:

$$\min_{\beta,\beta_0} \quad \left(\frac{1}{2N} \sum_{k=1}^{N} (y_k - \beta_0 - x_k^T \beta)^2 + \lambda \left(\frac{(1-\alpha)}{2} \|\beta\|^2 + \alpha \|\beta\| \right) \right) \tag{1}$$

Equation 1 becomes the lasso regularization problem for $\alpha = 1$ and ridge for $\alpha = 0$. In elastic net regularization, α can take any value between 0 and 1. In any case, a larger λ penalizes more terms, resulting in a sparser model.

In this study, two regularization methods are used to obtain sparse models: (1) lasso-regularization and (2) elastic net-regularization. For elastic net $\alpha = 0.5$ is used. For both

methods, MATLAB's *lasso* function is used with 5-fold cross-validation. The function solves a parameter estimation problem with the objective of minimizing the cross-validated mean square error (MSE), that is shown in Equation 2, where y_k is the observed value and $\hat{y}_k$ is the predicted value.

$$MSE = \frac{\sum_{k=1}^{N}(\hat{y}_k - y_k)^2}{N} \tag{2}$$

The *lasso* function performs regularization using a geometric sequence of λ's, resulting in 100 discrete values of λ and 100 parameter estimation problems solved. The function reduces the number of non-zero regression coefficients gradually by using larger values of λ in each step. This results in 100 sets of parameters for each prediction model. Among those 100 sets, two are highlighted by MATLAB: the one with the minimum cross-validation MSE (minMSE) and the one with minimum MSE plus one standard error (minMSE+1SE), a sparser model due to larger regularization coefficient in expense of a larger cross-validation MSE.

Depending on the scale of the variables y_k the range of MSE can be very different. While this does not affect the optimal solution of the parameter estimation problem, normalizing the MSE for all the models can be useful is comparing the relative accuracy of different models (e.g. yield vs. property predictions). Taking the square root of the MSE and then dividing it by the range of the measured data is commonly used to normalize MSE as shown in Equation 3 and the obtained quantity is called normalized root mean squared error (NRMSE).

$$NRMSE = \frac{\sqrt{MSE}}{y^{max} - y^{min}} \tag{3}$$

2.3.2. Yield and Property Prediction Models

In this section, the generic expressions for the prediction models are presented. The full list of all the variables and parameters used in the notation is given in the Nomenclature chapter. The outlet flow rate of a stream from a unit, $F_{out}(s,u,t)$, is calculated from its yield and the total input flow rate to that unit using Equation 4.

$$\begin{aligned} F_{out}(s,u,t) = \frac{Yield(s,u,t)}{100}\sum_{s'} F_{in}(s',u,t) \\ \forall u \in U^{pro}, s \in S_u^{out}, s' \in S_u^{in} \end{aligned} \tag{4}$$

The yields are predicted from inlet flow rates and inlet properties. Equations 5 and 6 show the mapping between the inputs variables and the output yields, and the general form of the yield prediction equations, respectively.

$$\begin{aligned} Yield(s,u,t) = f[F_{in}(s',u,t), P_{in}(p',s',u,t)] \\ \forall u \in U^{pro}, s \in S_u^{out}, s' \in S_u^{in}, p' \in P_{s',u}^{in}, t \end{aligned} \tag{5}$$

$$Yield(s,u,t) = C_{yield}(s,u)\{\beta_{yield,0}(s,u) + \sum_{i=1}^{I_{yield,u}} \beta_{yield,i}(s,u)x_{yield,i}(u,t) + \sum_{i=1}^{I_{yield,u}} \sum_{j=1,j\geq i}^{I_{yield,u}} \beta_{yield,i,j}(s,u)x_{yield,i}(u,t)x_{yield,j}(u,t)\} \quad \forall u \in U^{pro}, s \in S_u^{out}, s' \in S_u^{in}, p' \in P_{s,u}^{in}, t \tag{6}$$

where $x_{yield,i}(u,t)$ and $x_{yield,j}(u,t)$ can either be $F_{in}(s',u,t)$ or $P_{in}(p',s',u,t)$ and $C_{yield}(s,u)$ is the factor used to scale the value of $Yield(s,u,t)$. $I_{yield,u}$ is the set of inputs for yield predictions for unit u, respectively that are associated with the unit u.

Outlet properties are predicted from inlet flow rates, inlet properties, and outlet flow rates. Equations 7 and 8 describe the mappings between the input variables and the output properties or outlet yields for each unit.

$$P_{out}(p,s,u,t) = f[F_{in}(s',u,t), F_{out}(s,u,t), P_{in}(p',s',u,t)] \quad \forall u \in U^{pro}, s \in S_u^{out}, s' \in S_u^{in}, p \in P_{s,u}^{out}, p' \in P_{s',u}^{in}, t \tag{7}$$

Equation 8 shows the form of property prediction equations:

$$P_{out}(p,s,u,t) = C_{Pout}(p,s,u)\{\beta_{prop,0}(p,s,u) + \sum_{i=1}^{I_{prop,u}} \beta_{prop,i}(p,s,u)x_{prop,i}(u,t) + \sum_{i=1}^{I_{prop,u}} \sum_{j=1,j\geq i}^{I_{prop,u}} \beta_{prop,i,j}(p,s,u)x_{prop,i}(u,t)x_{prop,j}(u,t)\} \quad \forall u \in U^{pro}, s \in S_u^{out}, s' \in S_u^{in}, p \in P_{s,u}^{out}, p' \in P_{s',u}^{in}, t \tag{8}$$

where $x_{prop,i}(u,t)$ and $x_{prop,j}(u,t)$ can either be $F_{in}(s',u,t)$, $F_{out}(s,u,t)$, or $P_{in}(p',s',u,t)$ and $C_{Pout}(p,s,u)$ is the factor used to scale the value of $P_{out}(p,s,u,t)$. $b_{yield,0}(s,u)$, $b_{yield,i}(s,u)$, and $b_{yield,i,j}(s,u)$ are the parameters of the yield prediction equation for stream s leaving unit u. $I_{prop,u}$ is the set of inputs for yield predictions for unit u, respectively that are associated with the unit u.

2.3.3. Feature Selection

A feature selection approach to reduce the number of terms in the regression models, in other words increasing the model sparsity, serves a multitude of goals [49]. Sparsity in this work is defined in Equation 9:

$$Sparsity = \frac{Number\ of\ nonzero\ parameters}{Number\ of\ all\ possible\ parameters} \tag{9}$$

MATLAB's *lasso* function is called for four different type of models: (a) linear terms only (LM), (b) linear plus interaction terms (LIM), (c) linear plus quadratic terms (LQM), (d) linear plus interaction and quadratic terms (LIQM). For each type of model, we get

two versions: the minMSE and minMSE+1SE, with different sparsity and CV errors. The procedure for feature selection to generate sparse data-driven models is presented below:

Procedure Feature selection to obtain sparse models	
Step 1	Initialize the model sparsity cutoff criteria (CC) for yields and properties as 85% and 90%, respectively
Step 2	Train regularized LM using 5-fold CV
Step 3	Pick the best regularized model
Step 4	Use the subset of variables selected by LM and train regulzarized LIM, LQM, and LIQM using 5-fold CV
Step 5	Eliminate any model where: # possible terms ≥ 0.1(# observations) to prevent overfitting
Step 6	Pick the model with the minimum CV-MSE that obeys: Model sparsity $\geq$ CC
Step 7	If no model is selected, then relax the CC by 5% and go to Step 6; else continue
Step 8	If the linear terms of all the features do not appear in the model, retrain the model including the linear terms; else continue
Step 9	Print the model and continue to Step 1 for the next prediction model

This procedure aims to create a subset of relevant features first by comparing the cross validation MSE (CV-MSE) of the regularized LM models (Steps 1 and 2). Later on, interaction and quadratic terms of the selected features are also included in the model to see if performance of the model is improved (Steps 3 and 4). Overfitting is a significant concern that is possible to run into if the number of observations are not large enough. A rule of thumb is to use at least an order of magnitude more observations than the number of terms. Models that do not obey this conditions are eliminated (Step 5). Then, the model with the minimum CV-MSE that obeys the sparsity criterion is selected (Step 6). If no model obeys the initial sparsity cutoff criterion, the criterion is relaxed by 5% (Step 7). The models including the nonlinear terms are also regularized to eliminate insignificant nonlinear terms. This can potentially give models that include nonlinear combinations of a feature while its linear terms are eliminated (Step 4). As per convention for the surface response model training [46], the linear term of any feature that appears in any nonlinear form is forced to be included in the final (Step 8).

2.4. Multi-Period Planning Problem Formulation

Multi-period formulation allows the production facility to produce and store products over a planning horizon and inventory constraints tie the production plans of adjacent periods to each other. The additional degrees of freedom coming from inventory management provides room for improvement in the objective function value. Since the planning problem is solved for all periods simultaneously, the problem grows large in size compared to singe-period formulation where each period is solved individually. While single-period models are easier to solve due to smaller problem size, they provide less flexibility to the schedulers by ignoring the inventory levels, resulting in possible infeasibility when considering in scheduling operations [10, 20]. Multi-period formulation is a way to overcome this problem and to have more flexible production plans. The optimal solution of a multi-period problem is at

least as good as the single-period problem. However, since the number of decision variables and constraints are scaled linearly with respect to the number of periods, finding a global solution to a nonlinear and nonconvex multi-period problem can be difficult. For this reason, multi-period planning models are mostly reported to use linear equations for the processing units [51].

This work uses a discrete-time model with each time period represented uniformly by a single-day. This choice is made in accordance with the way our industrial partner makes their planning decisions, however, the duration of a period in the model can be nonuniform and weights of each period can be accounted for by modifying the inventory constraints as shown in literature [51]. The multi-period planning problem is formulated by including mass balance equations, capacity constraints, inventory constraints, operational restrictions, demand constraints, and the objective function along with the process models. These constraints provide the necessary physical restrictions to the process. The full list and definition of variables, sets, and parameters are presented in the Nomenclature section.

2.4.1. Connections and Unit Mass Balances

A typical refinery operation is summarized in Figure 2. The refinery process under investigation begins with the blending of various crude oil feeds and Daesan Refinery blends up to seven different crudes to obtain their crude oil mixture. The refinery changes the crude oil in every 4-5 days. Depending on the length of the planning horizon and the length of each period, multi-period plans can deal with multiple crude oil blends.

The blended crude oil is fed to the crude distillation unit (CDU) to be separated into fractions of different products. The rest of process can be summarized as further separation and upgrading of the crude oil distillates. In this study, crude oil blending is not included in the model, since a different branch of the refinery experts is working on finding the optimum blend. Hence the main input to the refinery is blended crude oil.

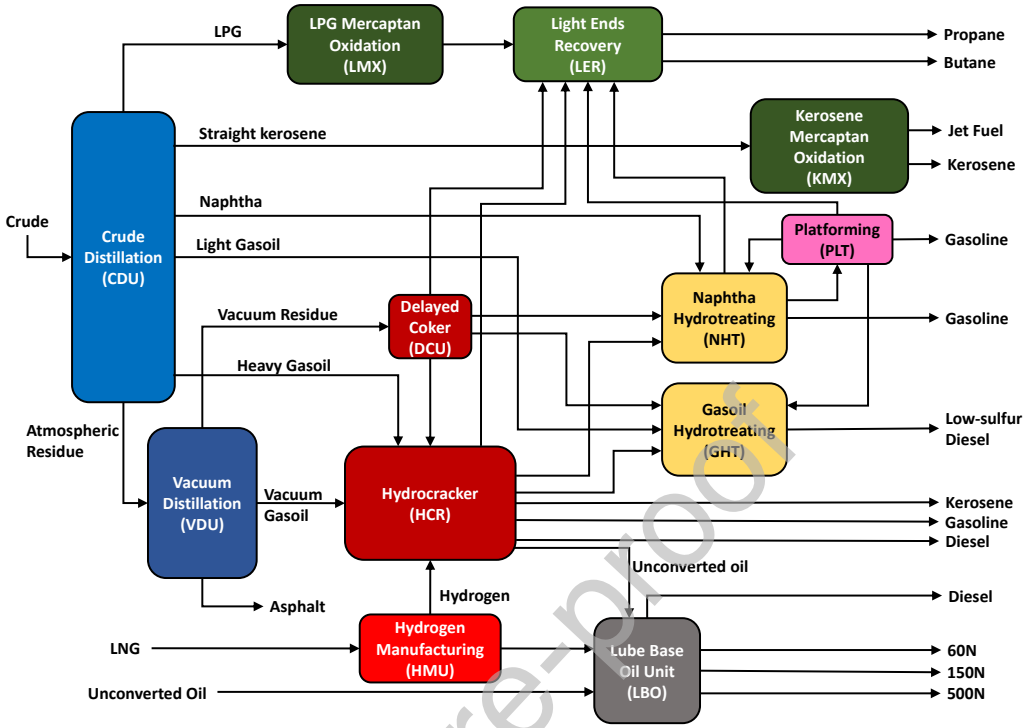

Figure 2: Simplified flowsheet of the refinery processes in Plant #1.

The plant topology with all possible stream connections is represented by a process superstructure. The stream connections are formulated as shown in Equations 10 and 11. Only allowable connections are made possible by constraining the equations using the stream connections subset UC. A pictorial representation of the inputs and outputs to unit u is shown in Figure 3.

$$\sum_{u^*}\sum_{s^*} F(s^*,u^*,s',u,t) \geq F_{in}(s',u,t)$$
$$\forall (u,u^*) \in U^{pro}, s^* \in S^{out}_{u^*}, s' \in S^{in}_{u}, (s^*,u^*,s,u) \in UC, t \tag{10}$$

$$F_{out}(s,u,t) \geq \sum_{(s',u')} F(s,u,s'',u'',t)$$
$$\forall (u,u'') \in U^{pro}, s \in S^{out}_{u}, s'' \in S^{in}_{u''}, (s,u,s'',u'') \in UC, t \tag{11}$$

where the $F_{out}(s,u,t)$ is calculated using Equation 4. Equations 10 and 11 are not strict equality constraints, hence the flow of streams entering the unit can be adjusted if needed. Note that, these inequality constraints allow for stream mass to be discarded or lost during transfer between two processing units, they do not allow for mass to be generated. Loss of stream mass in connections is common and acknowledged by the industrial partner.

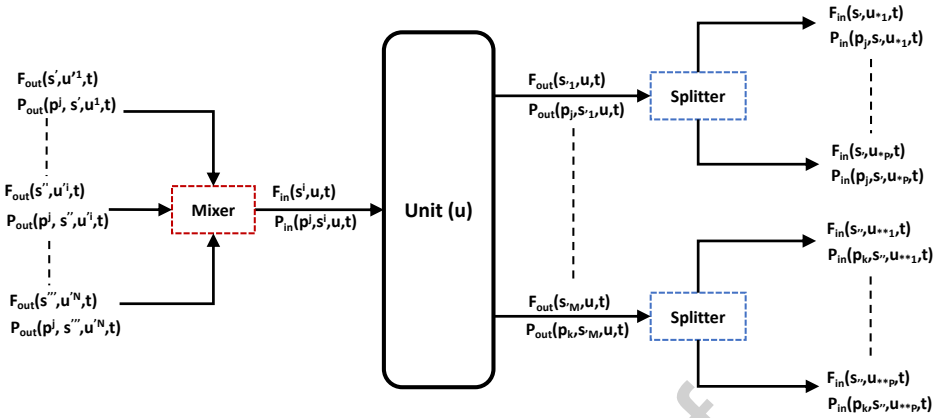

Figure 3: Schematic diagram showing how inputs and outputs are related from different units.

The property information is also transferred using the connections subsets $P^{in}_{s,u}$, $P^{out}_{s,u}$, and $UC(s,u,s',u')$ as shown in Equation 12:

$$\begin{aligned} P_{in}(p,s',u',t) = P_{out}(p,s,u,t) \\ \forall (u,u') \in U^{pro}, s \in S^{out}_u, s' \in S^{in}_u, \\ p \in (P^{in}_{s,u} \cup P^{out}_{s,u}), (s,u,s',u') \in UC, t \end{aligned} \tag{12}$$

The yield and property prediction Equations (5 and 7) are modified with slack variables and they take the form of Equations 13 and 14.

$$\begin{aligned} Yield(s,u,t) = f[F_{in}(s',u,t), P_{in}(p',s',u,t)] \pm YieldSlacks(s,u,t) \\ \forall u \in U^{pro}, s \in S^{out}_u, s' \in S^{in}_u, p' \in P^{in}_{s,u}, t \end{aligned} \tag{13}$$

$$\begin{aligned} P_{out}(p,s,u,t) = f[F_{in}(s',u,t), F_{out}(s,u,t), P_{in}(p',s',u,t)] \pm PropSlacks(p,s,u,t) \\ \forall u \in U^{pro}, s \in S^{out}_u, s' \in S^{in}_u, p \in P^{out}_{s,u}, p' \in P^{in}_{s,u}, t \end{aligned} \tag{14}$$

By the addition of slack variables, we are allowing slight violations of the upper or lower bounds of the predicted quantities. Since yield and property prediction constraints are in forms of equalities, adding slack variables to these equations relaxes the problem. Note that the slack variables are constrained to be no more than 5% of the upper bound of the predicted variable. To give an example, if the upper bound of a yield prediction is 40% and the model predicts the yield as 40.8%, then the negative slack will have the value 0.8% and the level value of the prediction will be 40%. Since nonzero slacks are essentially not desired, the sum of all the slacks is later added to the objective function to be minimized.

The throughput of the unit is the total flow rate leaving and it must not exceed the unit capacity. Equation 15 ensures that condition.

$$\begin{aligned} CAPmin(u) \leq \sum_s F_{out}(s,u,t) \leq CAPmax(u) \\ \forall u \in U^{pro}, s \in S^{out}_u, t \end{aligned} \tag{15}$$

2.4.2. Variable Bounds

For each variable there are upper and lower bounds coming from two years of operational data. Equations 16 to 20 show the upper and lower bounds on the decision variables.

$$F_{in}^{low}(s,u) \leq F_{in}(s,u,t) \leq F_{in}^{up}(s,u) \quad \forall u \in (U^{pro} \cup U^{hyp}), s \in S_u^{in}, t \tag{16}$$

$$F_{out}^{low}(s,u) \leq F_{out}(s,u,t) \leq F_{out}^{up}(s,u) \quad \forall u \in (U^{pro} \cup U^{hyp}), s \in S_u^{out}, t \tag{17}$$

$$Yield^{low}(s,u) \leq Yield(s,u,t) \leq Yield^{up}(s,u) \quad \forall u \in U^{pro}, s \in S_u^{out}, t \tag{18}$$

$$P_{in}^{low}(p,s,u) \leq P_{in}(p,s,u,t) \leq P_{in}^{up}(p,s,u) \quad \forall u \in (U^{pro} \cup U^{hyp}), s \in S_u^{in}, p \in P_{s,u}^{in}, t \tag{19}$$

$$P_{out}^{low}(p,s,u) \leq P_{out}(p,s,u,t) \leq P_{out}^{up}(p,s,u) \quad \forall u \in (U^{pro} \cup U^{hyp}), s \in S_u^{in}, p \in P_{s,u}^{out}, t \tag{20}$$

2.4.3. Demand Constraints

For each final product there is a different demand during the planning horizon. Upper and lower bounds on the demand for a product is addressed in Equation 21. All products are sent to the hypothetical $SALES$ unit.

$$MinDemand(s,t) \leq F_{in}(s,SALES,t) \leq MaxDemand(s,t) \quad \forall s \in S_{SALES}^{in}, t \tag{21}$$

2.4.4. Inventory Balance Constraints

The inventory variables $Inv(s,t)$ and $Inv^0(s,t)$ are used to connect the production variables to the sales variables. Equations 22 and 23 allow some of the refinery products to be stored in inventory.

$$Inv(s,t) = Inv^0(s,t-1) + \sum_u \sum_{s'} F(s',u,s,SALES,t) \quad \forall u \in U^{pro}, s \in S_{SALES}^{in}, s' \in S_u^{out}, (s',u,s,SALES) \in UC, t \tag{22}$$

$$Inv^0(s,t) = Inv(s,t) - F_{in}(s,SALES,t) - W(s,t) \quad \forall s \in S_{SALES}^{in}, t \tag{23}$$

$Inv(s,t)$ and $Inv^0(s,t)$ show the inventory level at the beginning of a period and at the end after the demands are satisfied, respectively. The inventory constraints and increased time horizon of the model allow products to be stored when either the demand or the crude prices are low. The excess production can later be used to meet the product demands when they are higher than refinery capacity. Alternatively, some excess products can be discarded if $W(s,t)$ variable is nonzero.

2.4.5. Objective Function

The objective of this project is to maximize the gross profit of the refinery, that is presented in Equation 24.

$$\begin{aligned} Profit = \sum_{t} \Big\{ & \sum_{s} Price(s, SALES, t) F_{in}(s, SALES, t) \\ & - \sum_{s'} Cost(s', PURC, t) F_{out}(s', PURC, t) \\ & - \sum_{u} OperatingCost(u) \sum_{s''} F_{out}(s'', u, t) \\ & - \sum_{s} InvCost(s) Inv^{0}(s, t) \Big\} \\ & \forall u \in U^{pro}, s \in S^{in}_{SALES}, s' \in S^{out}_{PURC}, s'' \in S^{out}_{u}, t \end{aligned} \tag{24}$$

where the cost of all raw materials that are purchased at the hypothetical $PURC$ unit and the operating costs of all processing units are subtracted from the total gross sales, which is the total revenue gained by sold products that are sent to the hypothetical $SALES$ unit. Total profit is calculated in $\$MM/day$ (millions of \$/day) basis.

The objective function for the main optimization problem is the sum of negative profit and slack variables. The addition of slack variables and minimizing the sum ensures that while maximizing the profit, we are also forcing the slack variables (thus the small violations of hard bounds) to be as small as possible. The objective function is shown in Equation 25.

$$\begin{aligned} Objective\ Function = & -Profit \\ & + \gamma \sum_{t} \Big\{ \sum_{u} \sum_{s} \sum_{p} PropSlacks(p, s, u, t) \\ & + \sum_{u} \sum_{s} YieldSlacks(s, u, t) \Big\} \\ & \forall u \in U^{pro}, s \in S^{out}_{u}, p \in P^{out}_{s,u}, t \end{aligned} \tag{25}$$

The coefficient γ is found by a process of trial. Different values for γ are tested, and the largest value that is ensuring that while the sums of slacks have a weight in the objective function they are not dominating the profit maximization objective. The optimization problem is presented in Equation 26:

$$\begin{aligned} \min \quad & Objective\ Function \\ \text{s.t.} \quad & Equations \quad 6, 8 \\ & Equations \quad 10 - 23 \end{aligned} \tag{26}$$

2.5. Global Optimization

Here, the global optimization software and algorithms are briefly explained. For more information, the readers are encouraged to read the work of Misener and Floudas [17] which

describes the novel components of the commercial solver ANTIGONE. The discrete-time, multi-period, planning model is a nonconvex, constrained NLP model where the nonlinearity comes from the quadratic and bilinear terms and the nonconvexity is caused by the bilinear interaction terms coming from property and yield prediction models.

The planning problem is modeled in GAMS and solved with the solver ANTIGONE to ε-global optimality. ANTIGONE takes the user defined NLP, detects the special structures, and reformulates the problem. It uses term-based underestimators to create tight convex lower bound problems (underestimations) in the form of a mixed integer linear optimization (MILP) program. Then, the MILP is combined with the upper bound (original problem) NLP in a branch-and-cut algorithm to find the global optimum solution. The algorithm generates tight convex underestimators, dynamically generates separating hyperplanes, bounds the variables, branches on the search space, and finds feasible solutions. As time progresses the lower and upper bounds converge to global optimum solutions.

3. Results and Discussion

3.1. Data Processing Results

The data sets for each processing unit consists of daily stream flow rates for all unit inputs and outputs, connections between units, daily/weekly/biweekly property measurements, product demands, raw material costs, as well as unit operating costs. Since 18 processing units are modeled, 18 separate data sets are required. In addition to these, we are given unit capacities, allowable lower and upper limits on product qualities, and detailed descriptions of the refinery flow sheet. The Daesan Refinery consists of three subplants within their refinery complex. Among the three plants, Plant #1 is chosen for this study. After grouping the data for each unit and having it approved by the company experts, the imputation analysis is done.

The performance of the data imputation techniques can depend on the missingness mechanism and the ratio of observations to features. Severson et al. [44] lists some of the mechanism as (i) random missingness, (ii) missingness that is correlated in time which can often be due to sensor failure, (iii) missingness with a patter that can be caused by multi-rate data, and (iv) censorship. The data set we have consists of columns with various missingness mechanisms, mainly suffering from types (i) and (iii). In order to compare the six imputation techniques listed above, we create test sets with varying degrees of missing data by removing cells randomly to form matrices with missingness fraction ranging from 0.1 to 0.5. Afterwards, the squared error (SE) between imputed values and the actual values are compared.

Figure 4 shows the performance of these methods on data sets taken from six refinery units, i.e. CDU, DCU, GHT, HCR, LER, and NHT. Note that the results obtained from NIPALS algorithm are not included in the figures because for many cases NIPALS performed significantly worse than the other five. An analysis of the results showed that in 56.7% of the total cases of 30, k-NN algorithm is the best, followed by TSR, which performs best in 33.3% of the cases, and KDR, which is the best in only 10% of the cases. In our studies, PPCA and IA give much larger SE, especially with increasing fraction of missing data. Since

k-NN algorithm consistently performs well with high fractions of missing data (e.g. 0.4-0.5), it is chosen over the other methods.

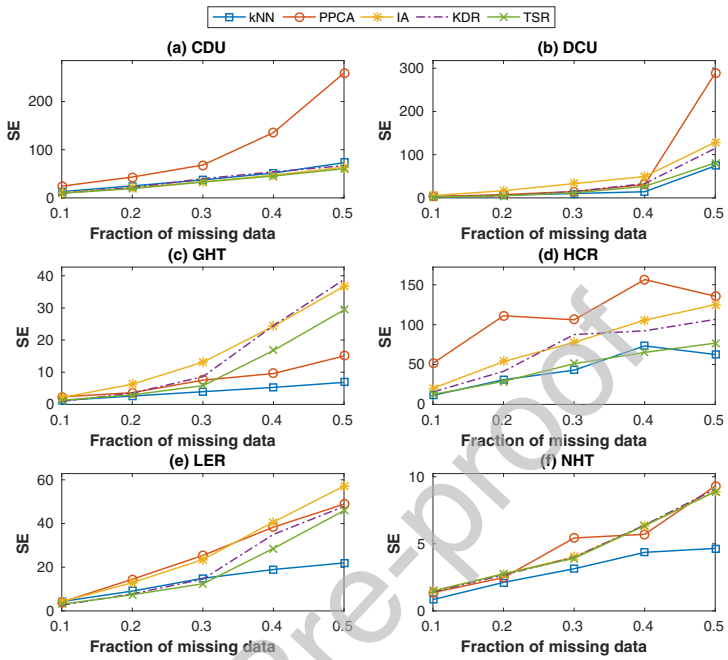

Figure 4: Imputation performance of k-NN, PPCA, IA, KDR, and TSR methods.

In order to restrict the degree of imputations, the amount of missing data on a row is limited to a cutoff value of 50% for yield inputs and 50% for property inputs. If a row (i.e. a day of observations) has more missing data that these cutoff values, it is removed completely from the data set. The overall missing data percentages before imputation ranged between 9 and 42% for all the units. The number of rows after removals ranged between 177 and 688. Additional reasons for row removal included larger than 10% mass balance error, a plant-wide shutdown, and maintenance shutdowns for individual units.

3.2. Parameter Estimation and Feature Selection Results

73 product yield and 181 outlet property prediction models are trained with the plant data. Regression analyses are done with lasso- and elastic net-regularization techniques using 5-fold cross validation. Sparse models are obtained with the feature selection procedure described earlier. 10% of the data is spared from the training data in order to be used for testing the models. The cross-validation NRMSE (CV-NRMSE) distributions of the lasso- and elastic net-regularized models are shown with box plots in Figure 5.

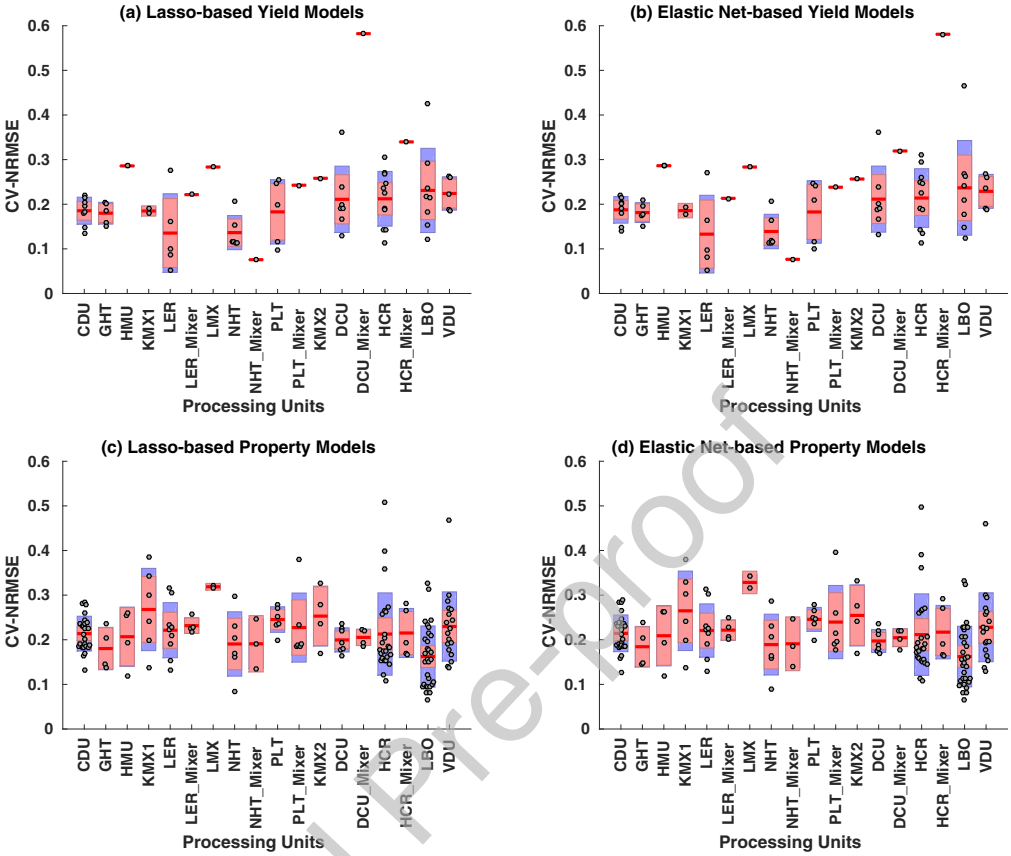

Figure 5: CV-NRMSE for property and yield prediction models (Red boxes show the 25^{th} and 75^{th} percentiles, blue whiskers show the 5^{th} and 95^{th} percentiles, grey dots show the CV-NRMSE of all individual prediction models associated with each unit).

The results show that there is no big difference between the CV-NRMSE of the lasso- and elastic net-regularization models. Figure 6 shows the histogram of % model sparsity in prediction models.

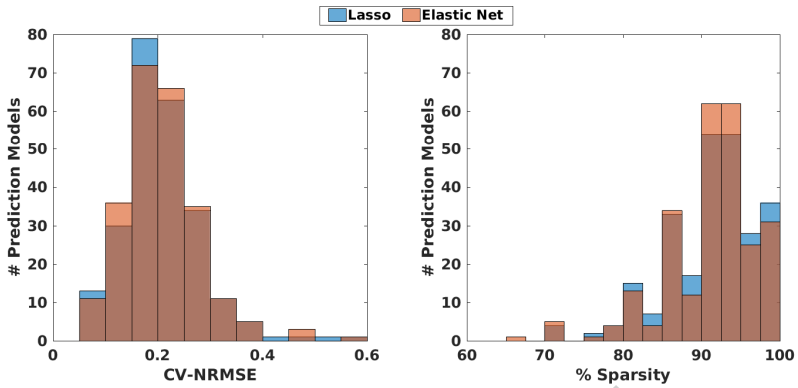

Figure 6: Histograms of CV-NRMSE and % sparsity of the data-driven prediction models.

Table 1: Single- and multi-period planning formulation statistics

Unit	# Yield Models	Selected Models	# Property Models	Selected Models
CDU	8	Elastic Net	25	Elastic Net
GHT	5	Lasso	4	Elastic Net
HMU	4	Elastic Net	4	Lasso
KMX1	2	Elastic Net	6	Elastic Net
LER	5	Lasso	9	Lasso
LER Mixer	1	Lasso	4	Lasso
LMX	1	Elastic Net	2	Elastic Net
NHT	6	Lasso	6	Elastic Net
NHT Mixer	1	Elastic Net	3	Elastic Net
PLT	5	Elastic Net	6	Lasso
PLT Mixer	1	Elastic Net	6	Lasso
KMX2	1	Elastic Net	4	Elastic Net
DCU	7	Lasso	7	Lasso
DCU Mixer	1	Lasso	4	Elastic Net
HCR	11	Lasso	27	Elastic Net
HCR Mixer	1	Elastic Net	6	Elastic Net
LBO	8	Elastic Net	32	Elastic Net
VDU	5	Lasso	26	Lasso

Results indicate that both regularization methods give the intended sparsity to the models. More than 87% of models have a sparsity greater than 85%. For deciding on which regularization model to use, 10% of the original data that is previously spared is used for testing. After comparing the testing NRMSE the list of selected models for each unit is presented in Table 1. Since the training data can contain noise and uncertainty, and the process units might go through certain changes over time, it is a good practice to retrain process models at regular intervals as new data become available. Since the selected models are surrogates, the models selected might be different depending on the data.

3.3. Production Planning Case Studies

3.3.1. Global Optimization

The planning problem is solved for selected days of October 2015. Both the single- and multi-period planning models are NLP. The problem is solved using ANTIGONE's advanced branch-and-bound algorithm, while CPLEX and CONOPT are selected as MILP and NLP solvers, respectively. All case studies are solved on a high-performance computing machine at Texas A&M High-Performance Research Computing (HPRC) facility using Ada IBM/Lenovo x86 HPC Cluster operated with Linux (CentOS 6) using 1 node (20 cores per node with 64 GB RAM). ANTIGONE 1.1 is used with GAMS 26.1.0 as the default solver. The solution time is limited to one hour and optimality criterion is set as 0.0001. Statistics of the single-period (SP) and multi-period (MP) optimization problems are given in Table 2. Different values (e.g. for the weights for the sum of slacks are tested to find a balance of profit and slack minimization. From a total of 4432 slack variables, 5.6% of them are found to be nonzero. Most of the nonzero slacks are observed in property predictions. These are the properties, where the data sets include higher rates of missing data. For problem size comparison, SP solution for period 1 is presented (SP-1) along with 2-, 4-, 6-, and 8-period MP solutions, that are MP-2, MP-4, MP-6, and MP-8, respectively.

Table 2: Single- and multi-period planning formulation statistics

Model Statistics		SP-1	MP-2	MP-4	MP-6	MP-8
Total continuous variables		1,349	2,718	5,434	8,150	10,866
Total equations		859	1,716	3,430	5,144	6,858
Total nonlinear terms		442	884	1,768	2,652	3,536
Solver Statistics		SP-1	MP-2	MP-4	MP-6	MP-8
ANTIGONE	Profit ($ MM)	0.356	0.742	1.677	2.152	2.334
	Solution time (s)	1,566	3,600	3,600	3,600	3,600
	Relative gap	0.00	0.52	N/A	N/A	0.98
BARON	Profit ($ MM)	0.356	0.742	1.677	2.154	2.330
	Solution time (s)	3,600	3,600	3,600	3,600	3,600
	Relative gap	0.59	0.66	N/A	N/A	N/A
IPOPT	Profit ($ MM)	0.356	0.743	1.674	2.154	2.333
	Solution time (s)	0.5	1.3	4.1	22.5	6.6
	Relative gap	N/A	N/A	N/A	N/A	N/A

Using ANTIGONE's options, piecewise linear underestimators with logarithmic partitioning scheme are selected to relax the nonconvex bilinear terms. Using default McCormick type convex envelopes can give results much faster, but the solution algorithm takes considerably more time to close the optimality gap within the vicinity of the global optimum. The rate of closing the gap also slows down with time. On the other hand, the solution algorithm with tighter piecewise linear underestimators can take more time to obtain the result, but it can close the optimality gap much faster by the end of the solution time. The optimal solutions obtained with ANTIGONE are later compared with the ones obtained using BARON 19.7.13 and IPOPT 3.11 and they are given in Table 2. The difference between optimal solutions of different solvers are found to be within 0.2%. IPOPT is much faster

than the other two solver to find a solution, however, it does not provide a gap of optimality since it is a local solver. Among the two global solvers, BARON is found to be faster to locate a solution than ANTIGONE but was slower to close the gap in SP and MP studies, this is due to BARON being used with default options, whereas with ANTIGONE we use piecewise underestimators. ANTIGONE is the solver to bound the lower and upper bound solutions more consistently as well as finding the best optimal solution within the one hour time, even though the MP problem can leave an optimality gap. The size of a 8-period problem shown in Table 2 shows that large-scale nonconvex NLPs still pose a challenge to the state-of-the-art global solvers. While, no solver used in this study has a significant advantage over others, since ANTIGONE gives the best results, we will continue to show its results in the following section.

The problem of computational complexity is a common concern with multi-period models. In this study, an 8-period plans considers 8 days of operation horizon, since each period is as long as day. While multi-period planning presents improvements over single-period planning even when the optimality gap is not closed, computational limitations prevent the maximum number of periods that the planning problem can be solved for. For plans that cover longer time horizons, use of representative time periods with varying weights assigned to each time period can be useful. Some clustering methods such as k-means [52, 53] and hierarchical clustering [54, 55] are already used in planning and scheduling problems like capacity expansion. Future work of our group will focus on reducing the number of representative time periods and the current planning model can be easily modified work with representative periods.

3.3.2. Optimal Plans

The multi-period (MP) problem has been solved for 2-, 4-, 6-, and 8-periods. Here we present the results obtained by 8-period problems (MP-8). Single-period (SP) problem is solved for each of the 8 periods separately. Crude oil is the primary input to the refinery process. In the optimization problem, crude oil properties are fixed to those of the actual operation, however inlet crude oil flow rate is left free as a decision variable. Since during 8-period horizon, there are multiple crude blends used, the crude properties change with respect to the period. Time-dependent parameters in this 8-period case study are the crude oil properties, the crude oil supply, and the product demands. In this case study, raw material costs, product prices, and product quality specifications are not changing. Due to confidentiality restrictions, the full details of the actual or optimal plans are not disclosed. Instead a breakdown of the major gross profit contributors is presented in Figure 7.

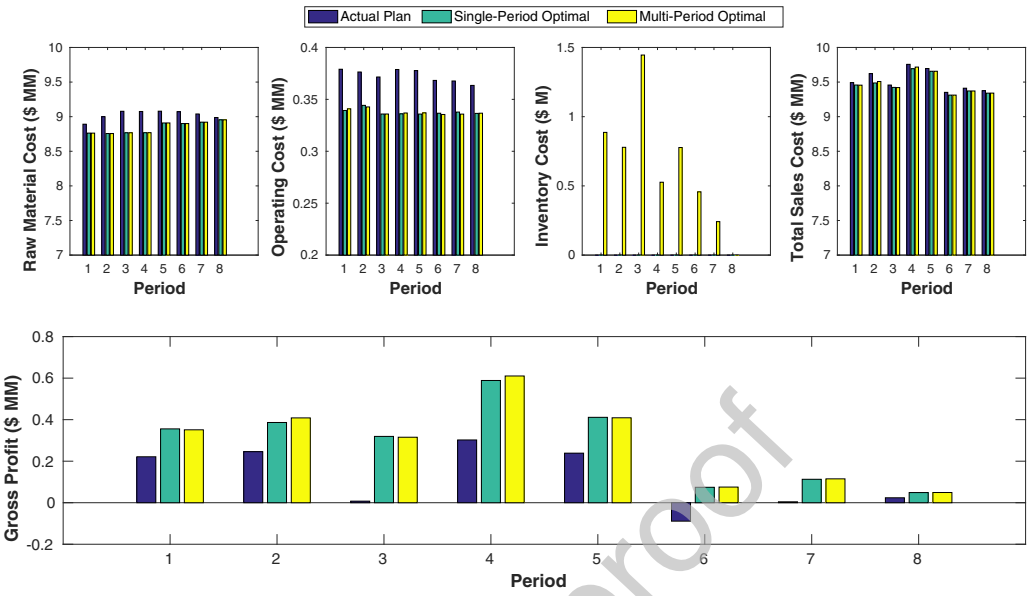

Figure 7: Comparison of actual, single-period, and multi-period optimization results.

The negative gross profit in period 6 actual plan is confirmed by our industrial partner and we are informed that Plant #1 is the oldest and least profitable of the entire refinery complex. Results indicate that both SP and MP optimal plans outperform the actual plan in every period. Optimal plans reduce the operating costs and the amount of raw material purchased while producing the same output as the actual plan by using the full advantage of the mathematical models for the processing units and optimal stream allocation. When MP-8 is compared with actual daily plans, we find the biggest reductions in the operating costs come from the changes in the operation crude distillation (CDU), vacuum distillation (VDU), delayed coker (DCU) and hydrocracker (HCR) units. In the MP optimal plan, CDU operation increases kerosene production slightly while decreasing the light and heavy gasoil production. VDU unit increases the vacuum gasoil production to be sent to HCR, while decreasing the vacuum residue that is sent to the DCU. DCU produces more lighter products, while HCR produces more kerosene and less diesel. This decrease in diesel production is compensated by the increased production of the lube base oil plants (LBO). LBO increases diesel production by using the available unconverted oil. Naphtha (NHT) and gasoil hydrotreating (GHT) units' throughputs do not change significantly, while kerosene mercaptan oxidation units (KMX1 and KMX2) slightly increase their production. Overall, we observe some portion of the diesel production to shift from HCR to LBO. This makes sense, since HCR is more costly to operate for this refinery.

In periods 1, 2, 4, 5 and 8 the optimal plans result in 57-94% improvement in gross profit. Periods 3, 6, and 7 are operational days where actual plan is vastly inferior to optimal plans. Total gross profit for the eight periods considered is $955,300 for the actual plan. SP plan gives $2,297,500 whereas MP plan gives $2,334,200. Improvements in gross profit are 140.5% and 144.3% for SP and MP plans, respectively. Although, the aforementioned changes in

operation in the MP plan are similar to the ones in SP plans, MP plan also has the advantage of the inventory management, that results in the additional 3.8% improvement in gross profit. While the next step is the validation of these results before an implementation, it is beyond the scope of this work. It needs to be investigated in future studies.

4. Conclusions

With this work, we showed an integrated data-driven modeling and global optimization approach to solve multi-period production planning problems. This work achieves (i) automatic generation of nonlinear and sparse data-driven process models where yields and properties of the process models are based on input properties and compositions, (ii) estimation of model parameters using real-plant data, and (iii) global optimization solution strategy of the large-scale nonlinear and multi-period production planning model using commercial solvers. We show that given operational data, accurate nonlinear data-driven input-output models for refinery processing and mixing units can be obtained. Lasso- and elastic net-regularization methods are used to obtain the process models. Obtained models have enough sparsity to be efficiently used in large-scale multi-period nonlinear optimization problems. Optimal production plans can improve the actual operation by allocating the streams more efficiently between units to reduce raw material and operating costs. Multi-period planning approach provides further improvement over single-period planning. While this work specifically focuses on production planning in refinery operations, our integrated modeling and optimization approach can be applied to any production facility.

Credit Author Statement

C. Doga Demirhan: Conceptualization, Methodology, Software, Writing - Original draft preparation. **Fani Boukouvala:** Conceptualization, Methodology **Kyungwon Kim:** Data curation, Validation **Hyeju Song:** Data curation **William Tso:** Writing - Review **Christodoulos A. Floudas:** Conceptualization **Efstratios N. Pistikopoulos:** Conceptualization, Supervision, Writing - Review & Editing

Acknowledgments

The authors would like to dedicate this article to the memory of late Christodoulos A. Floudas. His vision, influence, and guidence initiated this work and motivated the authors to finalize it. His inspiration will always be remembered and missed. The authors also wish to state their gratitude to Hyundai Oilbank Company Ltd. for sharing their valuable plant operation information and their financial support, which provided the resources and inspiration to start working on this project. The authors would also like to thank the financial support from Texas A&M Energy Institute. Doga Demirhan is further thankful to Jianyuan Zhai and Sun Hye Kim for the assistance they provided on data-driven modeling and feature selection techniques.

Nomenclature

The list of all the units, subscripts, superscripts, sets, variables, and parameters are given in this section.

Sets & Indices

u	Unit
s	Stream
p	Property
t	Period
i	Model parameter index 1
j	Model parameter index 2

Subsets

U^{pro}	Processing units
U^{hyp}	Hypothetical units
S_u^{in}	Inlet streams to unit u
S_u^{out}	Outlet streams from unit u
$P_{s,u}^{in}$	Inlet stream properties for stream s to unit u
$P_{s,u}^{out}$	Outlet stream properties for stream s from unit u
UC	Stream connection from stream s of unit u to stream s' of unit u'
$I_{yield,u}$	Set of inputs for yield prediction models of unit u
$I_{prop,u}$	Set of inputs for property prediction models of unit u

Subscripts

in	Inlet
out	Outlet

Processing Units

CDU	Crude distillation unit
GHT	Gasoil hydrotreating unit
$KMX1$	Kerosene mercaptan oxidation unit #1
$KMX2$	Kerosene mercaptan oxidation unit #2
LMX	LPG mercaptan oxidation unit
LER	Light ends recovery unit
NHT	Naphtha hydrotreating unit
PLT	Platforming unit
HMU	Hydrogen manufacturing unit
DCU	Delayed coker unit
HCR	Hydrocracker unit
LBO	Lube base oil unit
VDU	Vacuum distillation unit
$LER\ Mixer$	Mixer for light ends recovery unit
$NHT\ Mixer$	Mixer for naphtha hydrotreating unit
$PLT\ Mixer$	Mixer for platforming unit
$DCU\ Mixer$	Mixer for delayed coker unit
$HCR\ Mixer$	Mixer for hydrocracker unit

Hypothetical Units

$PURC$	Unit for all the purchased raw materials
$SALES$	Unit for all the sold products

Properties

ACN	Acid number
API	API density
ARO	Aromatics content
$BENZ$	Benzene content
$C1$	C1 content
$C2$	C2 content
CEN	Cetane number
$D05$	Temperature at which 5% of the mixture boils
$D10$	Temperature at which 10% of the mixture boils
$D95$	Temperature at which 95% of the mixture boils
Fe	Fe content
FP	Flash point
$H2S$	H_2S content
$NAPH$	Naphthenes content
PP	Pour point
RON	Research octane number
PAR	Paraffins content
RVP	Reed vapor pressure
$SALT$	Salt content
SUL	Sulfur content
$V60$	Viscosity at 60 °C
$V100$	Viscosity at 100 °C

Continuous Variables

$F(s,u,s',u',t)$	Mass flow rate of stream s from unit u to unit u' as stream s' at period t
$F_{in}(s,u,t)$	Inlet mass flow rate of stream s to unit u at period t
$F_{out}(s,u,t)$	Outlet mass flow rate of stream s from unit u at period t
$Yield(s,u,t)$	Percent yield of outlet stream s from unit u at period t
$P_{in}(p,s,u,t)$	Property p of inlet stream s to unit u at period t
$P_{out}(p,s,u,t)$	Property p of outlet stream s from unit u at period t
$Profit$	Total profit ($\$/day$)
$LossYield(u,t)$	Percent loss yield of unit u at period t
$UnitSlacks(u,t)$	Slacks for sum of yields for unit u at period t
$YieldSlacks(s,u,t)$	Slacks for yield prediction of stream s from unit u at period t
$PropSlacks(p,s,u,t)$	Slacks for property prediction of property p of stream s from unit u at period t
$Inv(s,t)$	Initial inventory level for product s at the beginning of time period t
$Inv^0(s,t)$	Inventory level for product s at time period t after demands are satisfied
$W(s,t)$	Mass flow rate of the waste stream s at period t

Parameters

γ	Weight parameter for the sum of slack variables in the objective function
$\beta_{yield,0}(s,u)$	Parameter for constant term in yield prediction equation of stream s from unit u
$\beta_{yield,i}(s,u)$	Parameter for linear terms in yield prediction equation of stream s from unit u

$\beta_{yield,i,j}(s,u)$	Parameter for bilinear and quadratic term in yield prediction equation of stream s from unit u
$\beta_{prop,0}(p,s,u)$	Parameter for constant term in yield prediction equation of property p stream s from unit u
$\beta_{prop,i}(p,s,u)$	Parameter for linear terms in yield prediction equation of property p of stream s from unit u
$\beta_{prop,i,j}(p,s,u)$	Parameter for bilinear and quadratic term in yield prediction equation of property p of stream s from unit u
$F_{in}^{up}(s,u)$	Upper bound on inlet mass flow rate of stream s to unit u
$F_{in}^{low}(s,u)$	Lower bound on inlet mass flow rate of stream s to unit u
$F_{out}^{up}(s,u)$	Upper bound on inlet mass flow rate of stream s to unit u
$F_{out}^{low}(s,u)$	Lower bound on outlet mass flow rate of stream s from unit u
$Yield^{up}(s,u)$	Upper bound on percent yield of outlet stream s from unit u
$Yield^{low}(s,u)$	Lower bound on percent yield of outlet stream s from unit u
$P_{in}^{up}(p,s,u)$	Upper bound on property p of inlet stream s to unit u
$P_{in}^{low}(p,s,u)$	Lower bound on property p of inlet stream s to unit u
$P_{out}^{up}(p,s,u)$	Upper bound on property p of outlet stream s from unit u
$P_{out}^{low}(p,s,u)$	Lower bound on property p of outlet stream s from unit u
$OperatingCost(u)$	Operational cost of unit u as a function of unit throughput ($/$ton$)
$Price(s,t)$	Price of product stream s ($/$ton$) at period t
$Cost(s,t)$	Cost of raw material stream s ($/$ton$) at period t
$InvCost(s,t)$	Inventory cost of storing product s ($/$ton$) at period t
$CAPmin(u)$	Minimum capacity for unit u
$CAPmax(u)$	Maximum capacity for unit u
$MinDemand(s,t)$	Minimum demand requirement for product s at period t
$MaxDemand(s,t)$	Maximum demand requirement for product s at period t

References

[1] S. L. Janak, C. A. Floudas, J. Kallrath, N. Vormbrock, Production Scheduling of a Large-Scale Industrial Batch Plant. I. Short-Term and Medium-Term Scheduling, Industrial & Engineering Chemistry Research 45 (25) (2006) 8234–8252, doi:10.1021/ie0600588, URL https://doi.org/10.1021/ie0600588.

[2] N. K. Shah, Z. Li, M. G. Ierapetritou, Petroleum Refining Operations: Key Issues, Advances, and Opportunities, Industrial & Engineering Chemistry Research 50 (3) (2011) 1161–1170, doi:10.1021/ie1010004, URL https://doi.org/10.1021/ie1010004.

[3] J. Kallrath, Planning and scheduling in the process industry, OR Spectrum 24 (3) (2002) 219–250, doi:10.1007/s00291-002-0101-7, URL http://dx.doi.org/10.1007/s00291-002-0101-7, cited By 218.

[4] J. Kallrath, Solving planning and design problems in the process industry using mixed integer and global optimization, Annals of Operations Research 140 (1) (2005) 339–373, doi:10.1007/s10479-005-3976-2, URL http://dx.doi.org/10.1007/s10479-005-3976-2, cited By 47.

[5] J. Li, X. Xiao, F. Boukouvala, C. A. Floudas, B. Zhao, G. Du, X. Su, H. Liu, Data-driven mathematical modeling and global optimization framework for entire petrochemical planning operations, AIChE Journal 62 (9) (2016) 3020–3040, ISSN 1547-5905, doi:10.1002/aic.15220, URL http://dx.doi.org/10.1002/aic.15220.

[6] C. S. Khor, D. Varvarezos, Petroleum refinery optimization, Optimization and Engineering 18 (4) (2017) 943–989, ISSN 1573-2924, doi:10.1007/s11081-016-9338-x, URL https://doi.org/10.1007/s11081-016-9338-x.

[7] R. J. Quann, S. B. Jaffe, Structure-oriented lumping: describing the chemistry of complex hydrocarbon mixtures, Industrial & Engineering Chemistry Research 31 (11) (1992) 2483–2497, doi:10.1021/ie00011a013, URL https://doi.org/10.1021/ie00011a013.

[8] S. B. Jaffe, H. Freund, W. N. Olmstead, Extension of Structure-Oriented Lumping to Vacuum Residua, Industrial & Engineering Chemistry Research 44 (26) (2005) 9840–9852, doi:10.1021/ie058048e, URL https://doi.org/10.1021/ie058048e.

[9] R. J. Quann, Modeling the chemistry of complex petroleum mixtures., Environmental Health Perspectives 106 (suppl 6) (1998) 1441–1448, doi:10.1289/ehp.98106s61441, URL https://ehp.niehs.nih.gov/doi/abs/10.1289/ehp.98106s61441.

[10] J. M. Pinto, M. Joly, L. F. L. Moro, Planning and scheduling models for refinery operations, Computers and Chemical Engineering 24 (9-10) (2000) 2259–2276, ISSN 0098-1354, doi:10.1016/S0098-1354(00)00571-8, URL http://dx.doi.org/10.1016/S0098-1354(00)00571-8.

[11] J. Li, I. A. Karimi, Scheduling Gasoline Blending Operations from Recipe Determination to Shipping Using Unit Slots, Industrial & Engineering Chemistry Research 50 (15) (2011) 9156–9174, doi:10.1021/ie102321b, URL https://doi.org/10.1021/ie102321b.

[12] J. Li, R. Misener, C. A. Floudas, Continuous-time modeling and global optimization approach for scheduling of crude oil operations, AIChE Journal 58 (1) (2012) 205–226, doi:10.1002/aic.12623, URL https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.12623.

[13] W. Li, C.-W. Hui, A. Li, Integrating CDU, FCC and product blending models into refinery planning, Computers and Chemical Engineering 29 (9) (2005) 2010–2028, doi:10.1016/j.compchemeng.2005.05.010, URL http://dx.doi.org/10.1016/j.compchemeng.2005.05.010, cited By 67.

[14] I. Alhajri, A. Elkamel, T. Albahri, P. Douglas, A nonlinear programming model for refinery planning and optimisation with rigorous process models and product quality specifications, International Journal of Oil, Gas and Coal Technology 1 (3) (2008) 283–307.

[15] B. Menezes, J. Kelly, I. Grossmann, Improved swing-cut modeling for planning and scheduling of oil-refinery distillation units, Industrial and Engineering Chemistry Research 52 (51) (2013) 18324–18333, doi:10.1021/ie4025775, URL http://dx.doi.org/10.1021/ie4025775, cited By 19.

[16] M. Tawarmalani, N. Sahinidis, A polyhedral branch-and-cut approach to global optimization, Mathematical Programming 103 (2) (2005) 225–249, doi:10.1007/s10107-005-0581-8, URL http://dx.doi.org/10.1007/s10107-005-0581-8, cited By 367.

[17] R. Misener, C. A. Floudas, ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization

of Nonlinear Equations, Journal of Global Optimization 59 (2-3) (2014) 503–526, ISSN 0925-5001, doi:10.1007/s10898-014-0166-2, URL http://dx.doi.org/10.1007/s10898-014-0166-2.

[18] C. Tsay, R. C. Pattison, M. R. Piana, M. Baldea, A survey of optimal process design capabilities and practices in the chemical and petrochemical industries, Comput. & Chem. Eng. 112 (2018) 180 – 189.

[19] C. Cuiwen, G. Xingsheng, X. Zhong, A data-driven rolling-horizon online scheduling model for diesel production of a real-world refinery, AIChE Journal 59 (4) (2013) 1160–1174, doi:10.1002/aic.13895, URL http://dx.doi.org/10.1002/aic.13895, cited By 10.

[20] S. M. S. Neiro, J. M. Pinto, Multiperiod optimization for production planning of petroleum refineries, Chemical Engineering Communications 192 (1-3) (2005) 62–88, doi:10.1080/00986440590473155, URL http://dx.doi.org/10.1080/00986440590473155, cited By 0.

[21] L. d. P. A. Sales, F. M. T. d. Luna, B. d. A. Prata, An integrated optimization and simulation model for refinery planning including external loads and product evaluation, Brazilian Journal of Chemical Engineering 35 (1) (2018) 199–215.

[22] S. M. Neiro, J. M. Pinto, A general modeling framework for the operational planning of petroleum supply chains, Computers & Chemical Engineering 28 (6) (2004) 871 – 896, ISSN 0098-1354, doi:https://doi.org/10.1016/j.compchemeng.2003.09.018, URL http://www.sciencedirect.com/science/article/pii/S0098135403002308, fOCAPO 2003 Special issue.

[23] Y. Yang, P. I. Barton, Refinery Optimization Integrated with a Nonlinear Crude Distillation Unit Model, IFAC-PapersOnLine 48 (8) (2015) 205 – 210, ISSN 2405-8963, doi:https://doi.org/10.1016/j.ifacol.2015.08.182, URL http://www.sciencedirect.com/science/article/pii/S2405896315010496, 9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015.

[24] P. Castillo Castillo, P. M. Castro, V. Mahalec, Global Optimization Algorithm for Large-Scale Refinery Planning Models with Bilinear Terms, Industrial & Engineering Chemistry Research 56 (2) (2017) 530–548, doi:10.1021/acs.iecr.6b01350, URL https://doi.org/10.1021/acs.iecr.6b01350.

[25] C. D. Demirhan, W. W. Tso, G. S. Ogumerem, E. N. Pistikopoulos, Energy systems engineering - a guided tour, BMC Chemical Engineering 1 (11), doi:10.1186/s42480-019-0009-5.

[26] R. Jin, W. Chen, T. Simpson, Comparative studies of metamodelling techniques under multiple modelling criteria, Structural and Multidisciplinary Optimization 23 (1) (2001) 1–13, doi:10.1007/s00158-001-0160-4, URL https://doi.org/10.1007/s00158-001-0160-4.

[27] T. Simpson, J. Poplinski, P. N. Koch, J. Allen, Metamodels for Computer-based Engineering Design: Survey and recommendations, Engineering with Computers 17 (2) (2001) 129–150, doi:10.1007/PL00007198, URL https://doi.org/10.1007/PL00007198.

[28] B. Beykal, F. Boukouvala, C. A. Floudas, N. Sorek, H. Zalavadia, E. Gildin, Global optimization of grey-box computational systems using surrogate functions and application to highly constrained oil-field operations, Computers & Chemical Engineering 114 (2018) 99 – 110, ISSN 0098-1354, doi:https://doi.org/10.1016/j.compchemeng.2018.01.005, fOCAPO/CPC 2017.

[29] R. Baliban, J. Elia, C. Floudas, Toward novel hybrid biomass, coal, and natural gas processes for satisfying current transportation fuel demands, 1: Process alternatives, gasification modeling, process simulation, and economic analysis, Industrial and Engineering Chemistry Research 49 (16) (2010) 7343–7370, doi:10.1021/ie100063y, URL http://dx.doi.org/10.1021/ie100063y, cited By 83.

[30] O. Onel, A. Niziolek, H. Butcher, B. Wilhite, C. Floudas, Multi-scale approaches for gas-to-liquids process intensification: CFD modeling, process synthesis, and global optimization, Computers and Chemical Engineering doi:10.1016/j.compchemeng.2017.01.016, URL http://dx.doi.org/10.1016/j.compchemeng.2017.01.016, cited By 0; Article in Press.

[31] C. D. Demirhan, W. W. Tso, J. B. Powell, E. N. Pistikopoulos, Sustainable ammonia production through process synthesis and global optimization, AIChE Journal 65 (7) (2019) e16498, doi:10.1002/aic.16498, URL https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.16498.

[32] A. Bhosekar, M. Ierapetritou, Advances in surrogate based modeling, feasibility analysis, and optimization: A review, Computers & Chemical Engineering 108 (2018) 250–267.

[33] S. H. Kim, F. Boukouvala, Surrogate-Based Optimization for Mixed-Integer Nonlinear Problems, Computers & Chemical Engineering (2020) 106847ISSN 0098-1354, doi:https://doi.org/10.1016/j.compchemeng.2020.106847, URL http://www.sciencedirect.com/science/article/pii/S0098135419306970.

[34] A. M. Alattas, I. E. Grossmann, I. Palou-Rivera, Integration of nonlinear crude distillation unit models in refinery planning optimization, Industrial and Engineering Chemistry Research 50 (11) (2011) 6860–6870, doi:10.1021/ie200151e, URL http://dx.doi.org/10.1021/ie200151e, cited By 30.

[35] M. R. Siamizade, Global Optimization of Refinery-wide Production Planning with Highly Nonlinear Unit Models, Industrial & Engineering Chemistry Research 58 (24) (2019) 10437–10454, doi:10.1021/acs.iecr.9b00887, URL https://doi.org/10.1021/acs.iecr.9b00887.

[36] V. Mahalec, Y. Sanchez, Inferential monitoring and optimization of crude separation units via hybrid models, Computers and Chemical Engineering 45 (2012) 15–26, doi:10.1016/j.compchemeng.2012.05.012, URL http://dx.doi.org/10.1016/j.compchemeng.2012.05.012, cited By 14.

[37] G. Fu, Y. Sanchez, V. Mahalec, Hybrid model for optimization of crude oil distillation units, AIChE Journal 62 (4) (2016) 1065–1078, doi:10.1002/aic.15086, URL http://dx.doi.org/10.1002/aic.15086, cited By 0.

[38] A. Mitsos, N. Asprion, C. A. Floudas, M. Bortz, M. Baldea, D. Bonvin, A. Caspari, P. Schaefer, Challenges in process optimization for new feedstocks and energy sources, Comput. Chem. Eng. 113 (2018) 209–221.

[39] T. E. Swaty, Consider over-the-fence product stream swapping to raise profitability, Hydrocarbon Processing 81 (3) (2002) 37–42, ISSN 00188190.

[40] K. Al-Qahtani, A. Elkamel, Multisite Refinery and Petrochemical Network Design: Optimal Integration and Coordination, Industrial & Engineering Chemistry Research 48 (2) (2009) 814–826, doi:10.1021/ie801001q, URL http://dx.doi.org/10.1021/ie801001q.

[41] S. A. Imtiaz, S. L. Shah, Treatment of missing values in process data analysis, The Canadian Journal of Chemical Engineering 86 (5) (2008) 838–858, doi:10.1002/cjce.20099, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cjce.20099.

[42] A. Folch-Fortuny, F. Arteaga, A. Ferrer, PCA model building with missing data: New proposals and a comparative study, Chemometrics and Intelligent Laboratory Systems 146 (2015) 77 – 88, ISSN 0169-7439, doi:https://doi.org/10.1016/j.chemolab.2015.05.006, URL http://www.sciencedirect.com/science/article/pii/S0169743915001197.

[43] S. Xu, B. Lu, M. Baldea, T. F. Edgar, W. Wojsznis, T. Blevins, M. Nixon, Data cleaning in the process industries., Reviews in Chemical Engineering 31 (5) (2015) 453 – 490, ISSN 01678299, doi:10.1515/revce-2015-0022, URL http://dx.doi.org/10.1515/revce-2015-0022.

[44] K. A. Severson, M. C. Molaro, R. D. Braatz, Principal Component Analysis of Process Datasets with Missing Values, Processes 5 (3), doi:10.3390/pr5030038, URL http://www.mdpi.com/2227-9717/5/3/38.

[45] A. Folch-Fortuny, F. Arteaga, A. Ferrer, Missing Data Imputation Toolbox for MATLAB, Chemometrics and Intelligent Laboratory Systems 154 (2016) 93 – 100, ISSN 0169-7439, doi:https://doi.org/10.1016/j.chemolab.2016.03.019, URL http://www.sciencedirect.com/science/article/pii/S0169743916300557.

[46] R. H. Myers, D. C. Montogomery, C. M. Anderson-Cook, Response Surface Methodology - Process and Product Optimization Using Designed Experiments, John Wiley & Sons, Hoboken, NJ, 2012.

[47] R. Misener, C. A. Floudas, Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations, Mathematical Programming 136 (1) (2012) 155–182.

[48] H. Xiang, Y. Li, H. Liao, C. Li, An Adaptive Surrogate Model Based on Support Vector Regression and Its Application to the Optimization of Railway Wind Barriers, Struct. Multidiscip. Optim. 55 (2) (2017) 701–713, ISSN 1615-147X, doi:10.1007/s00158-016-1528-9, URL https://doi.org/10.1007/s00158-016-1528-9.

[49] J. Zhai, F. Boukouvala, Nonlinear variable selection algorithms for surrogate modeling, AIChE Journal 65 (8), doi:10.1002/aic.16601, URL https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.16601.

[50] T. Hastie, R. Tibshirani, M. Wainwright, Statistical Learning with Sparsity: The Lasso and Generalizations, Chapman & Hall/CRC, ISBN 1498712169, 2015.

[51] A. Alattas, I. Grossmann, I. Palou-Rivera, Refinery production planning: Multiperiod MINLP with nonlinear CDU model, Industrial and Engineering Chemistry Research 51 (39) (2012) 12852–12861, doi:10.1021/ie3002638, URL http://dx.doi.org/10.1021/ie3002638, cited By 8.

[52] J. H. Merrick, On representation of temporal variability in electricity capacity planningmodels, Energy Economics 59 (2016) 261 – 274, ISSN 0140-9883, doi:https://doi.org/10.1016/j.eneco.2016.08.001, URL http://www.sciencedirect.com/science/article/pii/S0140988316302018.

[53] L. Kotzur, P. Markewitz, M. Robinius, D. Stolten, Impact of different time series aggregation methods on optimal energy system design, Renewable Energy 117 (2018) 474 – 487, ISSN 0960-1481, doi:https://doi.org/10.1016/j.renene.2017.10.017, URL http://www.sciencedirect.com/science/article/pii/S0960148117309783.

[54] S. Pineda, J. M. Morales, Chronological Time-Period Clustering for Optimal Capacity Expansion Planning With Storage, IEEE Transactions on Power Systems 33 (6) (2018) 7162–7170, doi:10.1109/TPWRS.2018.2842093.

[55] W. W. Tso, C. D. Demirhan, C. F. Heuberger, J. B. Powell, E. N. Pistikopoulos, A hierarchical clustering decomposition algorithm for optimizing renewable power systems with storage, Applied Energy 270 (2020) 115190, ISSN 0306-2619, doi:https://doi.org/10.1016/j.apenergy.2020.115190, URL http://www.sciencedirect.com/science/article/pii/S0306261920307029.

Credit Author Statement

C. Doga Demirhan: Conceptualization, Methodology, Software, Writing - Original draft preparation.
Fani Boukouvala: Conceptualization, Methodology
Kyung-won Kim: Data curation, Validation
Hyeju Song: Data curation
William Tso: Writing - Review
Christodoulos A. Floudas: Conceptualization
Efstratios N. Pistikopoulos: Conceptualization, Supervision, Writing - Review & Editing

Declaration of Competing Interests

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