Computers and Chemical Engineering 75 (2015) 105–119

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Optimizing scheduling of refinery operations based on piecewise linear models

Xiaoyong Gao [a], Yongheng Jiang [a], Tao Chen [b,*], Dexian Huang [a,c,**]

[a] Department of Automation, Tsinghua University, Beijing 100084, China
[b] Department of Chemical and Process Engineering, University of Surrey, Guildford GU2 7XH, UK
[c] Tsinghua National Laboratory for Information Science and Technology, Beijing 100084, China

ARTICLE INFO

Article history:
Received 11 September 2013
Received in revised form 26 January 2015
Accepted 27 January 2015
Available online 7 February 2015

Keywords:
Optimization
Piecewise linear programming
Piecewise linear representation
Refinery scheduling
Unified simplicial partition

ABSTRACT

Optimizing scheduling is an effective way to improve the profit of refineries; it usually requires accurate models to describe the complex and nonlinear refining processes. However, conventional nonlinear models will result in a complex mixed integer nonlinear programming (MINLP) problem for scheduling. This paper presents a piecewise linear (PWL) modeling approach, which can describe global nonlinearity with locally linear functions, to refinery scheduling. Specifically, a high level canonical PWL representation is adopted to give a simple yet effective partition of the domain of decision variables. Furthermore, a unified partitioning strategy is proposed to model multiple response functions defined on the same domain. Based on the proposed PWL partitioning and modeling strategy, the original MINLP can be replaced by mixed integer linear programming (MILP), which can be readily solved using standard optimization algorithms. The effectiveness of the proposed strategy is demonstrated by a case study originated from a refinery in China.

1. Introduction

Refinery processes crude oils into various products including fuels and chemicals. In the background of global economy, refinery has been confronted with ever increasing challenges, such as intense competition that reduces margin profit, increasing requirement for product quality, strict environmental regulations, frequent fluctuation of crude oils caused by tighter supply, changes in demand for product oils, and so on. To address these challenges, optimal scheduling of refinery has become a necessity. It was estimated that an extra margin of up to 1 US dollar can be achieved per product barrel through better implementation of planning, scheduling and control systems for the gasoline blending process alone (Shobrys and White, 2002).

In the research community, a lot of fruitful results have been obtained and have promoted the development of scheduling optimization methods. Zhang and Zhu (2000) proposed a novel modeling and decomposition strategy for refinery optimization. The general framework for refinery planning and scheduling model were proposed by Pinto et al. (2000), Joly et al. (2002) and Smania and Pinto (2003). They stressed the necessity of considering operating conditions and inflow properties in scheduling models. However, how to model these items remained an open problem. Jia et al. (2003) and Jia and Ierapetriton (2004) proposed a continuous time formulation for refinery scheduling problem and spatially decomposed it into three sub-problems, where fixed yield model is adopted regardless of operation and feed changes. Similar modeling method is adopted by Dogan and Grossmann (2006) and Wu and Ierapetritou (2007). More recently, Shah and Ierapetritou (2011) incorporated logistics into the short term scheduling problem of a large scale refinery, where outlet yields of production units are taken as optimized variable, constrained by predefined bounds. Gao et al. (2014a,b) considered the impact of variations in crude oil on scheduling. Göthe-Lundgren et al. (2002) proposed an multi-fixed yield model in terms of several predefined operating states (each operating state refers to the fixed feed quality and quantity, and fixed unit operating condition), which may not be sufficient to cover the entire scheduling domain. This multi-fixed yield model was adopted in Luo and Rong (2007), which also proposed a hierarchical approach to short-term scheduling and significantly reduced the binary variables in optimization. In addition to these specific

* Corresponding author. Tel.: +44 1483 686593.
** Corresponding author at: Department of Automation, Tsinghua University, Beijing 100084, China. Tel.: +86 10 62784964.
E-mail addresses: t.chen@surrey.ac.uk (T. Chen), huangdx@tsinghua.edu.cn (D. Huang).

http://dx.doi.org/10.1016/j.compchemeng.2015.01.022
0098-1354/

Nomenclature

B_u	batches set for unit u
BLD	blenders set
$HUPUs$	hydro-upgrading processing units set
$INV_{u,t}$	inventory in tank u at the end of time period t
INV_u^{MIN}	lower-bound for the storage capacity of tank u
INV_u^{MAX}	upper-bound for the storage capacity of tank u
IS_u	inflow streams set of unit u
M_u	operation modes set of unit u
o_u	price of the product oil stored in product oil tank u
$OPC_{u,t}$	operating cost of processing unit u during time period t
OS_u	outflow streams set of unit u
P	properties set
P_u	properties set to be considered for unit u
PTK	product oil tanks set
$PRO^*_{s,u,p}$	value of property p for the inflow stream s to unit u at the beginning of scheduling
$PRO_{s',u,t,p}$	value of property p for the outflow stream s' from unit u during time period t
$PRO^{low}_{s',u,mp}$	minimum value of property p for the discharging stream s' of unit u in operation mode m
$PRO^{up}_{s',u,mp}$	maximum value of property p for the discharging stream s' of unit u in operation mode m
$\Delta PRO_{s',u,t,p}$	delta-value of property p for the connecting stream s' from unit u during time period t
$Q^*_{s,u}$	inflow stream s of unit u at the beginning of scheduling
$Q_{s,u,t}$	inflow stream s of unit u during time period t
$Q_{s',u,t}$	outflow stream s' from unit u during time period t
QF_u^{MIN}	lower-bound flow of unit u
QF_u^{MAX}	upper-bound flow of unit u
QI^*_u	inflow of unit u at the beginning of scheduling
$QI_{u,t}$	inflow of unit u during time period t
$QO_{u,t}$	outflow of unit u during time period t
$r^{MIN}_{s,u,m}$	lower bound proportion of charging stream s for blender u in operation mode m
$r^{MAX}_{s,u,m}$	upper bound proportion of charging stream s for blender u in operation mode m
s_u	storage cost of tank u
$T_{u,b}$	scheduling periods set for unit u and batch b
TK	tanks set
TP	scheduling horizon
$YIELD^*_{s',u}$	yield of discharging stream s' of unit u at the beginning of scheduling
$YIELD_{s',u,t}$	yield of discharging stream s' of unit u during time period t
$\Delta YIELD_{s',u,t}$	delta yield of discharging stream s' of unit u during time period t because of operating condition changes
$z_{u,m,t}$	binary variable denoting that unit u is in operation mode m during time period t
U	units set

Subscripts

b	batch
m	operation mode
p	property
s, s'	streams
t	time period
u	units

studies, some excellent reviews have been published in this area (Floudas and Lin, 2004; Bengtsson and Nonås, 2010; Shah et al., 2010; Joly, 2012; Harjunkoski et al., 2014).

Despite the large amount of work in refinery scheduling optimization, very limited attention has been placed on the modeling of the yield and operating cost of refinery units (response variables) as function of decision variables. Due to the process complexity and variability in operation, the yield and cost are highly nonlinear with respect to the decision variables (Li et al., 2005). In the majority of existing studies, the yield and cost (and possibly other process measures) are fixed for each predefined operating mode, i.e. models resembling look-up tables (Göthe-Lundgren et al., 2002; Jia and Ierapetritou, 2004; Luo and Rong, 2007); this strategy do not well represent the real processes. However, if highly complex nonlinear models are used, such as neural networks or Gaussian process regression models (Yan et al., 2011), the subsequent optimization becomes nonlinear and is hard to solve efficiently. In this paper, a piecewise linear (PWL) method is proposed for refinery scheduling because of its global nonlinearity and local linearity. PWL is capable of modeling nonlinearity, and also results in a linear programming problem in scheduling optimization.

In the community of systems engineering, a range of PWL representations have been reported, such as canonical representation of section-wise piecewise linear functions (Chua and Kang, 1977), hinging hyperplanes (Breiman, 1993), generalized piecewise linear functions (Lin et al., 1994), high level canonical piecewise linear functions (Julián et al., 1999), generalized hinging hyperplanes (Wang and Sun, 2005), and adaptive hinging hyperplanes (Xu et al., 2009). Nevertheless, these compact PWL representations often result in a large

number of subregions (Huang et al., 2012), and thus the model structure becomes too complex to be useful in practice. This phenomenon, referred to as "curse of partitions" was recently addressed by using a high level canonical PWL representation (CPWL) (Gao et al., 2014a), which also improved the modeling accuracy of the original simplicial partition strategy (Julián et al., 1999) through allowing adjustable partition intervals.

Our previous work, as reported in (Gao et al., 2014a,b), developed a theoretical CPWL framework to model a single response variable; but how it can be applied to scheduling problem was not explored. Building upon this theoretical framework, the present study applies the CPWL to optimal scheduling of refinery processes. Furthermore, in order to model multiple response variables with the same decision variables, we propose a unified simplicial partitioning strategy so that the same domain partitions are obtained for different responses (referred to as multi-CPWL in this paper). Otherwise if the domain is partitioned separately for each response variable, the combined partitions (required for the models to be used in subsequent schedule optimization) give rise to very complex subregions that cannot be analytically represented. Using such complex CPWL models in scheduling would be computationally intensive, an issue that can be addressed by the proposed multi-CPWL approach in this paper. It should be noted that all PWL methods are approximation to the original non-linear problem, and thus do not guarantee to (and often cannot) find the optimum of the original problem. However, they can be useful engineering solutions, balancing model accuracy and computational complexity. Moreover, based on the proposed multi-CPWL process models, a piecewise linear programming strategy is developed for scheduling optimization, and thus the original MINLP is transformed into an MILP problem which can be solved more efficiently. Computational efficiency is particularly useful in practice, since it allows timely response to short-term variations in demand.

To the best knowledge of the authors, this study is the first to use piecewise linear models for refinery scheduling. Such an approach makes it feasible to accurately model the process, while at the same time maintain reasonable computation time for scheduling optimization. The idea of piecewise linear approximation is closely related to some state-of-the-art global MINLP solvers, for example GloMIQO (Misener and Floudas, 2013), and there are many other commonly adopted MINLP global solvers, such as BARON (Sahinidis, 1996). However, the proposed approach significantly differs from these solvers. Global solvers aim to solve an already-formulated MINLP; the focus is on *solver* and the challenge is due to the model nonlinearity. In contrast, the proposed approach aims to obtain an approximate piecewise linear model for the process first, thus the optimization problem can be formulated as an MILP which can be easily solved to find the global optimum; the focus is on modeling. In addition, most studies of global solvers rely on explicit models, whereas there are no such ready-to-use models in the scheduling problem considered in this paper. As such, modeling needs to be carried out as the first step.

In this paper, only two decision variables are considered for each processing unit, partly because this is required by the particular refinery under study (and other similar refineries), and partly because of simplicity in presentation. In principle, a recursive construction method could be used to extend the two-dimensional partition to higher dimensions, which may significantly increase the model complexity and computation. In practical refinery scheduling problems, the operation of many processing units can be summarized by a few decision variables (such as desulfurization amount and research octane number in gasoline etherification). High dimensional partitioning strategies will be the topic for future research.

The rest of the paper is organized as follows. Section 2 provides the problem statement. The proposed piecewise linear model is presented in Section 3. In Section 4, the detailed piecewise linear scheduling model and transformation from the original MINLP to the MILP is described. After that, a case study is given in Section 5, using the benchmark Petro-SIM simulation environment, to demonstrate the effectiveness of the proposed methodology. Finally, a brief conclusion is drawn in Section 6.

2. Problem statement

A real world refinery in China is considered in this paper. For strategic reasons, this refinery has been guaranteed a relatively steady supply of crude oils with no significant variation in the chemical/physical properties. In the past, the operation of the primary (mainly distillation including preliminary, atmospheric and vacuum distillation units) and the secondary processing units (referring to heavy oil cracking, such as fluidized catalytic cracking (FCC), hydro-cracking, delayed coking, etc.) has been maintained stable. In this study, the adjustable units for scheduling are the downstream of the primary and secondary operations, including the hydro-upgrading processing units (*HUPUs*) and product oil blenders. Blending directly determines the amount of each product; its operation has immediate impact on revenue (Singh et al., 2000). Moreover, the operation of *HUPUs* significantly influences the yield and quality of the inlet flows of the blenders. Therefore, it is customary to open up *HUPUs* and blenders for scheduling, which also mimics the actual practice in the refinery under investigation. If needed, the proposed modeling methodology can also be generalized for scheduling other processing units. The flow diagram illustrating these adjustable units is shown in Fig. 1.

In more details, the *HUPUs* include a straight run gasoline (SRG) catalytic reformer, a light diesel hydrotreater (1# in the figure), a heavy diesel hydrotreater (2# in the figure), an FCC heavy gasoline hydrodesulfurizer and an FCC light gasoline etherification unit. The material streams, which are from the upstream units and depicted with dash lines, have fixed values and thus cannot be adjusted. In addition, two product oil blenders, one for gasoline and the other for diesel, are considered. Five different grades are derived from the gasoline blender, while three different grades are from the diesel blender. These grades comply with the relevant national standard; the detailed quality specifications can be found in Appendix A. Petro-SIM, a state-of-the-art simulation software of petroleum refinery processes developed by KBC Advanced Technologies (www.kbcat.com) (Mohaddecy et al., 2006), is used as platform for simulating the operations of *HUPUs* in this study. Petro-SIM is a full-featured, graphical process simulator that combines proprietary KBC technology and industry-proven process simulation environment for advanced modeling of hydrocarbon processing facilities (Aylott and Van der Merwe, 2008). Blenders are not included in simulation, since the properties of the product can simply be calculated as a flow-weighted average of the inlet oil streams. Here, Petro-SIM is utilized to generate the operating data for training the CPWL models.

For this case, the main task of scheduling is to determine the operation of processing units, blenders, and associated feed and storage tanks, in order to meet the market demand of product oils. The objective is to maximize the profit subject to the process and operations constraints, and quality specifications. To formulate the optimal scheduling problem, models are needed to represent the yield and operating cost of each processing unit as a function of the operating decision variables. Since the outlet streams of *HUPUs* will go to blenders before released as final products, the outlet properties are key decision variables. In particular, for gasoline *HUPUs* (SRG catalytic reformer, FCC

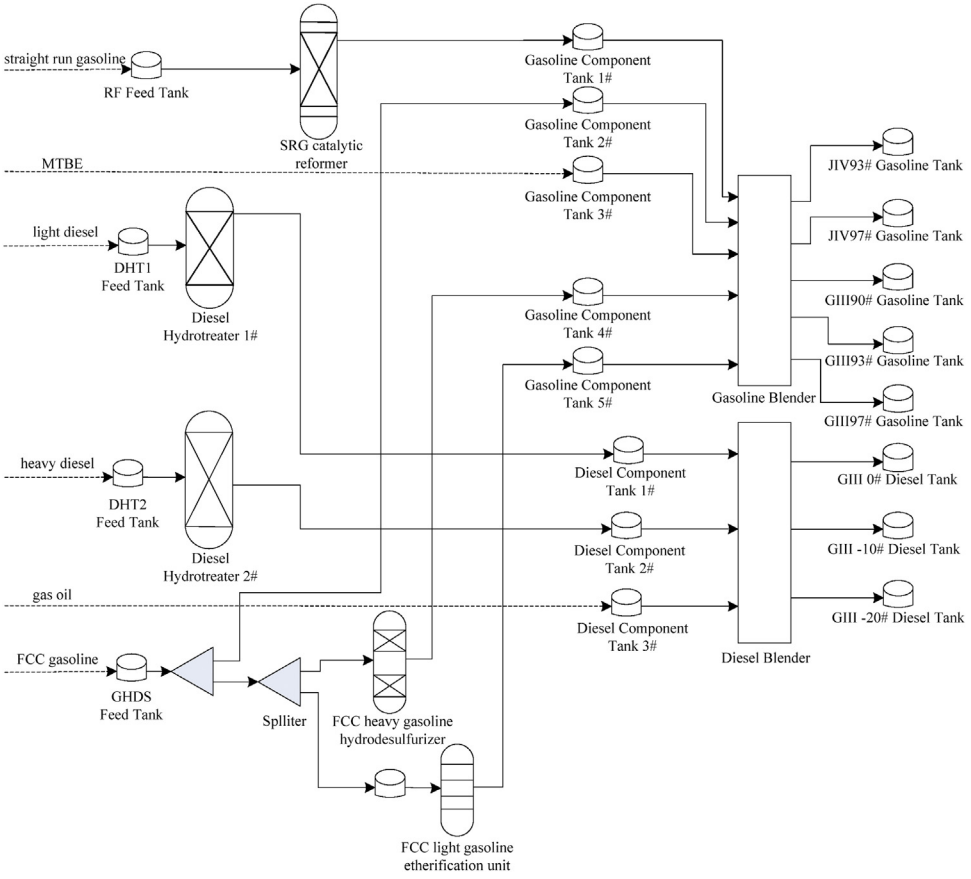

Fig. 1. Flow chart of the refining units subject to scheduling. The dash lines denote the fixed inflows that are solely determined by the upstream units, and the solid lines denote flows that can be adjusted in scheduling.

heavy gasoline hydrodesulfurizer, FCC light gasoline etherification unit), the desulfurization amount and delta research octane number (RON) are the decision variables, whereas for diesel *HUPUs* (the two hydrotreaters), the desulfurization amount and delta freezing point are taken as decision variables. Besides these HUPU-specific decision variables, the inlet and/or outlet mass flows also need to be determined for each unit and blending/feed/storage tank. Operating cost for blending is negligible in comparison with *HUPUs*. For blenders, the outlet properties are linear functions of those of the inlet streams, following the established literature (Luo and Rong, 2007), and thus CPWL representation is not needed. To mimic the actual practice in the plant, only one blender is used for each grade of gasoline and diesel, and dedicated tanks are assigned for each grade. This assumption also simplifies the optimization problem, though removing it to suit more generic cases would be straightforward.

In many reported studies (Göthe-Lundgren, 2002; Jia and Ierapetriton, 2004; Luo and Rong, 2007), the yield and operating cost are usually taken as fixed values under each operating mode. This approach is a very rough approximation to the actual processes, especially for the secondary processing units and *HUPUs*. In this paper, we introduce a more accurate representation using piecewise linear formulation in the next section.

3. Piecewise linear model

This section first discusses how to partition the domain of decision variables into subregions, within each of which a linear regression model can then be developed. On a two dimensional domain, the adopted simplicial partitioning strategy can be easily visualized; however it must be analytically represented so that the models can be used for optimal scheduling. Section 3.1 presents the analytical description of the simplicial partitioning strategy; Section 3.2 presents multi-CPWL representation to approximate multiple functions with the same simplicial partition. The reader is referred to the Nomenclature for symbols used.

3.1. Piecewise linear representation based on simplicial partitions

The concept of simplicial partitions is illustrated by using a two-dimensional case. Suppose that the domain of the function to be fitted is $[0,a] \times [0,b]$, and the number of grids for partitioning is $I \times J$. For now, we assume that the grids are already determined with the boundary values ξ_i for x_1 ($i = 1, 2, \ldots I$) and ζ_j for x_2 ($j = 1, 2, \ldots J$), where x_1 and x_2 are decision variables. The simplicial partition refers to the shaded triangular regions in Fig. 2, denoted $\Omega_{i,j,k}$ ($k = 1, 2, \ldots 8$). Such partition is the result of tradeoff between representation capability and model complexity. It was demonstrated (Lin and Unbehauen, 1992) that simple lattice partition is inadequate in representing the domain, while more advanced methods (e.g. hinging hyperplanes, generalized hinging hyperplanes) lead to a very complex model structure that is difficult for subsequent use.

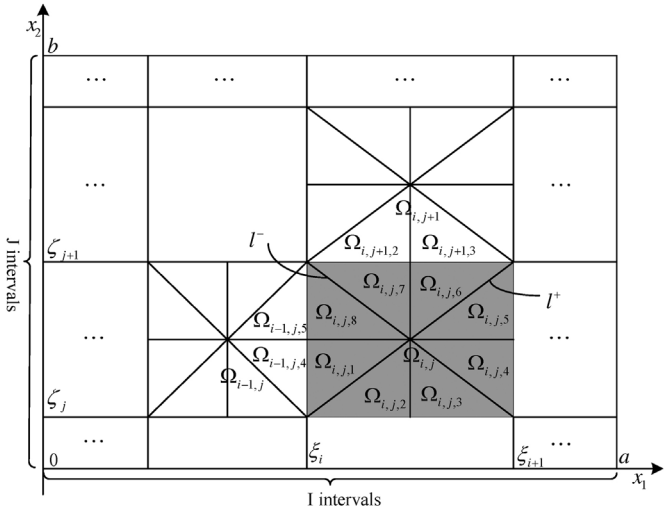

Fig. 2. Depiction of the simplicial partitioning strategy.

In the following, the task is to represent such partition, given in Fig. 2, using mathematical functions, so that the model can later be used for scheduling.

Firstly, we follow (Julián et al., 1999) to introduce the generating function as follows,

$$\gamma_\psi(f_i, f_j) = \frac{1}{4}\{\psi(\psi(-f_i)+f_j) - \psi(-f_i+\psi(f_j)) + \psi(-f_i) + \psi(f_i) - \psi(-f_i+f_j)\} \tag{1}$$

where $\psi(z)=|z|$. Clearly, the generating function follows that

$$\gamma(f_i, f_j) = \begin{cases} f_i & IF\ 0 \le f_i \le f_j \\ f_j & IF\ 0 \le f_j \le f_i \\ 0 & IF\ f_i < 0 \text{ or } f_j < 0 \end{cases} \tag{2}$$

The generating function can be extended to the k-th "nesting level" following the recursion: $\gamma^k(f_1, \ldots, f_k) = \gamma(f_1, \gamma^{k-1}(f_2, \ldots, f_k))$, and $\gamma^0(f_i) = f_i$, for any function f_i. See (Julián et al., 1999) for detailed properties of the generating function.

Next, we use the introduced generating function to construct a model realizing the designed partition strategy in Fig. 2.

Consider the subregion $\Omega_{i,j} = U_{k=1}^{8}\Omega_{i,j,k}$ within $[\xi_i, \xi_{i+1}] \times [\zeta_j, \zeta_{j+1}]$ and the diagonal lines l^+ and l^-, which can be represented by

$$\begin{aligned} l^+ :\ & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) - (x_2-\zeta_j)(\xi_{i+1}-\xi_i) = 0 \text{ or} \\ & (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) - (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) = 0 \\ l^- :\ & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) - (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) = 0 \text{ or} \\ & (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) - (x_2-\zeta_j)(\xi_{i+1}-\xi_i) = 0 \end{aligned}$$

Clearly, these diagonal lines form the diagonal partition boundary, which can be rewritten as

$$\begin{aligned} l^+ :\ & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) = (x_2-\zeta_j)(\xi_{i+1}-\xi_i) \text{ or} \\ & (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) = (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) \\ l^- :\ & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) = (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) \text{ or} \\ & (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) = (x_2-\zeta_j)(\xi_{i+1}-\xi_i) \end{aligned}$$

Motivated by the partition boundary, the following basis functions are designed:

$$\begin{aligned} \pi_{i,j,1} &= (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) \\ \pi_{i,j,2} &= (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) \\ \pi_{i,j,3} &= (x_2-\zeta_j)(\xi_{i+1}-\xi_i) \\ \pi_{i,j,4} &= (\zeta_{i+1}-x_2)(\xi_{i+1}-\xi_i) \end{aligned} \tag{3}$$

The above basis functions will be later used through the generating function to obtain the analytical representation of the partition. To this end, we follow (Julián et al., 1999) to define a vector $\mathbf{\Lambda_k}$ as the set of functions with the nesting level k, and the entire set is

$$\Lambda = [\mathbf{\Lambda}_0^T, \mathbf{\Lambda}_2^T, \mathbf{\Lambda}_4^T] \tag{4}$$

The construction of the above vector is described next.

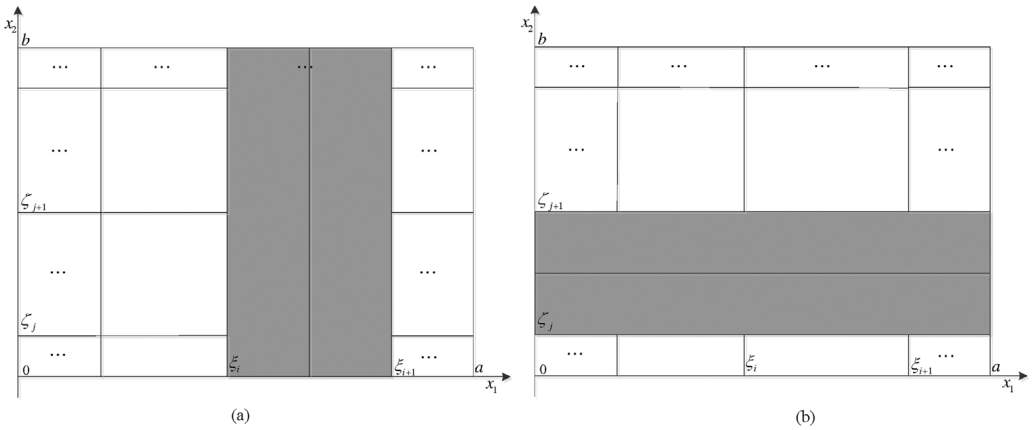

Fig. 3. Partition depiction for the 2nd level function of (a) $\gamma^2(\pi_{r_1,a,1}, \pi_{r_1,a,1})$ if $r_1 = i$; (b) $\gamma^2(\pi_{a,r_2,3}, \pi_{a,r_2,4})$ if $r_2 = j$.

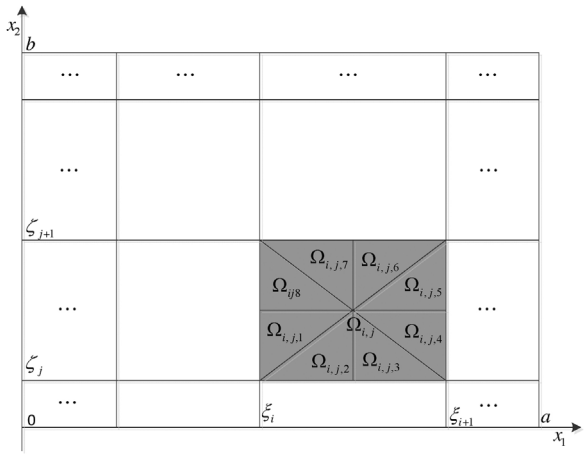

Fig. 4. Partition depiction for 4th level function.

$\mathbf{\Lambda}_0$: The 0-th level functions,

$$\gamma^0(1) = 1,\ \gamma^0(\pi_{i,j,k}) = \pi_{i,j,k}$$

where $i = 1, 2, \ldots I$, $j = 1, 2, \ldots J$ and $k = 1, 2, 3, 4$. The number of 0-th level functions is

$$4\binom{I}{1}\binom{J}{1} + 1.$$

$\mathbf{\Lambda}_2$: The 2nd level functions,

$$\gamma^2(\pi_{r_1,a,1}, \pi_{r_1,a,2}),\ \gamma^2(\pi_{a,r_1,1}, \pi_{a,r_2,2}),$$

where $r_1 = 1, 2, \ldots I$, $r_2 = 1, 2, \ldots J$ and a is an arbitrary integers belonging to [1,min(I, J)]. Geometrically, $\mathbf{\Lambda}_2$ realizes the partition as shown in Fig. 3 if $r_1 = i$ and $r_2 = j$. The detailed explanation of the partitions is given in Appendix B.

$\mathbf{\Lambda}_4$: The 4th level functions,

$$\gamma^4(\pi_{r_1,r_2,1}, \ldots, \pi_{r_1,r_2,4}) = \gamma^2(\gamma^2(\pi_{r_1,r_2,1}, \pi_{r_1,r_2,2}), \gamma^2(\pi_{r_1,r_2,3}, \pi_{r_1,r_2,4}))$$

Taking $r_1 = i$ and $r_2 = j$ as example, the 4th level function divides the region as shown in Fig. 4. The detailed derivation is given in Appendix B.

Given the partition functions, $\mathbf{\Lambda}(x)$, the CPWL model is the following linear regression with regression coefficient vector $\mathbf{c}$,

$$\mathbf{y} = \mathbf{c}^T \mathbf{\Lambda}(x) \tag{5}$$

The CPWL formulation (5) is inherently continuous on the region boundaries; the detailed proof can be found in (Gao et al., 2013). The model is determined by the grid locating ξ_i and ζ_i, the regression coefficients c, and the number of grids I and J. The method to estimate these parameters will be discussed in Section 3.2 with the multi-CPWL model.

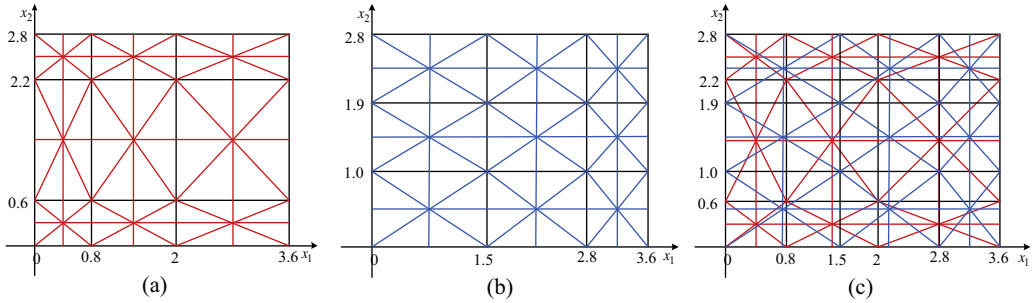

Fig. 5. Domain partitions for (a) model 1, (b) model 2 and (c) the combined domain partitions for models 1 and 2.

3.2. Multi-CPWL model based on a unified simplicial partitioning strategy

The simplicial partitioning method, presented in Section 3.1, forms the basis of CPWL modeling. For a specific process, there may be more than one response variables (also termed "output variables") that need to be modeled, such as yield and operating cost in this study. In principle, it is possible to partition the domain and develop these models separately. In general, the optimal partitions (i.e. the values of ξ_i and ζ_i in the basis functions) would be different for different response variables. When the partitions are combined (so that the models can be used in one scheduling), they give rise to very complex subregions that cannot be analytically represented. Fig. 5 illustrates this issue on a two-dimensional decision domain (x_1 and x_2 here) with two response variables being modeled.

Suppose that the domain is divided into 3×3 grids, Fig. 5(a) and (b) illustrates the partitions when the two response variables are modeled separately. Each response is associated with 72 subregions and each subregion has a unique linear function. However, when these two models are used in optimal scheduling, the combined domain partitions in Fig. 5(c) need to be considered to decide which two linear functions should be used. Clearly, because the subregions of different models do not coincide, a lot of intersection subregions emerge and each disjoint subregion represents a unique set of two linear functions. These subregions are too complex to describe analytically. Therefore, in such situation, the point-based search method (Zhu et al., 2010) will have to be used to determine the correct linear functions for a specific point in the decision domain. This point-based search is known to be time-consuming and can only guarantee locally optimal solutions. Moreover, if more than two response variables are to be modeled, the issue will become even worse. Therefore, in this study, we propose a unified simplicial partitioning strategy so that multiple responses are modeled on the same domain partition.

For a specific process, suppose that H response variables need to be modeled by CPWL: $f_h(x_1, x_2)$, $h = 1, 2, \ldots H$. Here, x_1 and x_2 are two independent variables representing operation decisions. Given a unified partition, the multi-CPWL model is formulated as follows

$$\begin{aligned} y_1 &= \boldsymbol{c}_1^T \boldsymbol{\Lambda}(\boldsymbol{x}) \\ y_2 &= \boldsymbol{c}_2^T \boldsymbol{\Lambda}(\boldsymbol{x}) \\ &\vdots \\ y_H &= \boldsymbol{c}_H^T \boldsymbol{\Lambda}(\boldsymbol{x}) \end{aligned} \tag{6}$$

Define $\boldsymbol{y} = [y_1, y_2, \ldots, y_H]^T$, $\mathbf{C} = [c_1^T, c_2^T, \ldots, c_H^T]^T$, then Eq. (6) can be rewritten as

$$\boldsymbol{y} = \boldsymbol{c}^T \boldsymbol{\Lambda}(x) \tag{7}$$

The model parameters to be estimated include the linear regression coefficients $\mathbf{C}$ in Eq. (7) and the nonlinear parameters in the basis function $\boldsymbol{\Lambda}(x)$, i.e. ξ_i and ζ_i in Eq. (3) (note that the basis function contains multiplicative terms of ξ_i's and ζ_i's thus is nonlinear). The nonlinear parameters represent the boundary values based on which the domain is partitioned. Suppose that a set of operation data are obtained from actual plant or specialized process simulation software (e.g. Petro-SIM in this study), noted as $\{\boldsymbol{x}_d, \boldsymbol{y}_d\}$ $d = 1, 2, \ldots D$, where $\boldsymbol{x}_d$ is a vector of decision variables, $\boldsymbol{y}_d$ is the corresponding responses (yield and operating cost in this study), and D is the number of data points. The parameters are estimated by minimizing the following sum of the squared errors,

$$\boldsymbol{E} = \frac{1}{2}\sum_{d=1}^{D} ||\boldsymbol{y}_d - \hat{\boldsymbol{y}}_d||^2 = \frac{1}{2}\sum_{d=1}^{D} ||y_d - \boldsymbol{C}^T \Lambda(\boldsymbol{x}_d)||^2 \tag{8}$$

where $\hat{\boldsymbol{y}}_d = \boldsymbol{C}^T \boldsymbol{\Lambda}(\boldsymbol{x}_d)$ is the predicted response from the piecewise linear model. Since this objective function is optimized by "pooling" all response functions with respect to the partitioning, it guarantees a unified partition across multiple responses.

Due to the presence of both linear and nonlinear model parameters, the above optimization problem is solved by iterating between the following two steps until convergence.

Step 1: Given the value of linear coefficients, calculate the gradient of E with respect to nonlinear parameters ξ_i and ζ_i, where $i = 1, 2, \cdots I, j = 1, 2, \ldots J$. Then, use the steepest descent or the conjugate gradient method to update these nonlinear parameters. The gradients can be derived analytically and are given in Appendix C.

Step 2: Given the value of nonlinear parameters, use the standard least square to update the linear regression coefficients.

The grid numbers I and J are determined by cross-validation (Martens and Dardenne, 1998).

4. Piecewise linear method based scheduling optimization

For refinery, the yield of streams and the operating cost of the processing units are two crucial metrics considered in scheduling problem, and they are modeled using the multi-CPWL method of Section 3.2. The discrete time scheduling problem with the multi-CPWL yield and operating cost models is established as follows, using the state task network (STN) based discrete time scheduling model (Kondili et al., 1993). In comparison with continuous time scheduling model, discrete time representation usually results in simpler optimization problems. For example, Pinto et al. (2000) pointed out that resource constraints under discrete time representation are much easier to handle, while continuous representation will generate a lot of bilinear terms resulting unnecessary nonlinear terms and thus unnecessarily nonconvex programming problems. Further discussions of this issue can be found in (Floudas and Lin, 2004; Pinto et al., 2000; Zhang and Sargent, 1996).

In this case, the decision variables for the *HUPUs* are the inlet mass flow rate, the delta desulfurization amount, and the delta RON (for gasoline) or freezing point (for diesel). These variables are selected because they either reflect the operation of the *HUPUs* (the inlet flow rate), or determine the quality attributes of the outlet streams. Notice that only the two delta properties are used in CPWL modeling. For other units, including tanks, splitters and blenders, the decision variables are the inlet and outlet mass flow rate.

For all units, including *HUPUs*, blenders, tanks and splitters, Eqs. (9) and (10) calculate the mass flows that enter (inflows), and leave (outflows) unit u, respectively:

$$QI_{u,t} = \sum_{s \in IS_u} Q_{s,u,t} \quad \forall u \in U,\ s \in IS_u,\ t \in TP \tag{9}$$

$$QO_{u,t} = \sum_{s' \in OS_u} Q_{s',u,t} \quad \forall u \in U,\ s' \in OS_u,\ t \in TP \tag{10}$$

The *HUPUs* are described by the following equations:

$$Q_{s',u,t} = QI_{u,t} \cdot YIELD_{s',u,t} \quad \forall u \in HUPUs,\ s \in IS_u,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{11}$$

where in Eq. (11), $YIELD_{s',u,t}$ represents the yield of output stream s' of processing unit u during time period t. Clearly, it is bilinear for Eq. (11). To guarantee the linearity, the approximate form is adopted here as shown in Eqs. (11a) and (11b). QI_u^* and $YIELD_{s',u}^*$ represent the initial value of inflow and yield at the beginning of scheduling horizon, respectively:

$$Q_{s',u,t} = QI_u^* \cdot YIELD_{s',u}^* + QI_u^* \cdot \Delta YIELD_{s',u,t}^* + \Delta QI_{u,t} \cdot YIELD_{s',u}^* \quad \forall u \in HUPUs,\ s \in IS_u,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{11a}$$

$$\Delta YIELD_{s',u,t} = PWL_{s',u}^{yield}(\Delta PRO_{s',u,t,p}) \quad \forall u \in HUPUs,\ s \in IS_u,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{11b}$$

Eq. (11b) indicates the CPWL model for delta-yield, $\Delta PRO_{s',u,t,p}$ (i.e. the x in Section 3.2) is the change (delta) of the two decision variables of stream s' from unit u during time period t, constrained by Eq. (14). For gasoline *HUPUs*, $\Delta PRO_{s',u,t,p}$ are desulfurization amount and delta RON, while desulfurization amount and delta freezing point for diesel *HUPUs*. The properties of the outlet streams are expressed in Eq. (12) in terms of the decision variables ($\Delta PRO_{s',u,t,p}$). Eq. (13) corresponds to the CPWL model for operating cost, which is unit specific and obtained by the CPWL model trained from historical operation data (simulated historical data in this paper). Eq. (16) specifies that the minimum and maximum mass capacity must be satisfied for inflows of processing unit u, while the scheduled inflow is formulated in Eq. (15):

$$PRO_{s',u,t,p} = PRO_{s,u,t,p} + \Delta PRO_{s',u,t,p} \quad \forall u \in HUPUs,\ s \in IS_u,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{12}$$

$$OPC_{u,t} = PWL_u^{cost}(\Delta PRO_{s',u,t,p}) \quad \forall u \in HUPUs,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{13}$$

$$\Delta PRO_{s',u,p}^{\mathrm{MIN}} \le \Delta PRO_{s',u,t,p} \le \Delta PRO_{s',u,p}^{\mathrm{MAX}} \ \forall u \in HUPUs\ ,\ s' \in OS_u,\ p \in P_u,\ t \in TP \tag{14}$$

$$QI_{u,t} = QI_u^* + \Delta QI_{u,t} \quad \forall u \in HUPUs,\ t \in TP \tag{15}$$

$$QF_u^{\mathrm{MIN}} \le QI_{u,t} \le QF_u^{\mathrm{MAX}} \quad \forall u \in HUPUs,\ t \in TP \tag{16}$$

For splitters, the mass of all charging streams equals to all discharging streams, as shown in Eq. (17):

$$QI_{u,t} = QO_{u,t} \quad \forall u \in U,\ t \in TP \tag{17}$$

The inventory of all process, feeding and storage tanks at the end of time period t is equal to the inventory at the end of t-1 plus mass flows entering the unit during t, and minus the mass flows leaving the unit during t:

$$INV_{u,t} = INV_{u,t-1} + QI_{u,t} - QO_{u,t} \quad \forall u \in TK,\ t \in TP \tag{18}$$

Eq. (19) relates to the storage capacity constraint for the tanks:

$$INV_u^{\mathrm{MIN}} \le INV_{u,t} \le INV_u^{\mathrm{MAX}} \quad \forall u \in TK,\ t \in TP \tag{19}$$

For blenders, the operation mode needs to be considered, corresponding to which grade product oil is being produced. A binary variable, $z_{u,m,t}$, is used to indicate whether blender unit u is in operation mode m at time t. Eq. (20) constrains the product quality, i.e. the sum of feeding mass flows times the corresponding quality properties, within the specified range. The handling of bilinearity in Eq. (20) is similar to Eq. (11). Under treatment, Eq. (20) is replaced by Eqs. (20a) and (20b). The plant also enforces restrictions on the proportion of blending components, as given in Eq. (21). Eq. (22) specifies that up to one operation mode can be selected for each blender during time period t. The product oils demands are satisfied in Eq. (23):

$$QI_{u,t} \sum_{m \in M_u} z_{u,m,t} \cdot PRO_{s',u,m,p}^{low} \le \sum_{s \in IS_u} Q_{s,u,t} \cdot PRO_{s,u,t,p} \le QI_{u,t} \sum_{m \in M_u} z_{u,m,t} \cdot PRO_{s',u,m.p}^{up} \in BLD,\ t \in TP,\ p \in P,\ s' \in OS_u \tag{20}$$

$$QI_{u,t}\sum_{m\in M_u} z_{u,m,t}\cdot PRO^{low}_{s',u,m,p} \le \sum_{s\in IS_u} Q^{*}_{s,u}\cdot PRO^{*}_{s,u,t} + \Delta Q_{s,u,t}\cdot PRO^{*}_{s,u,p} + \Delta PRO_{s,u,t,p} Q \le QI_{u,t}\sum_{m\in M_u} z_{u,m,t}\cdot PRO^{up}_{s',u,m,t} \tag{20a}$$

$$Q_{s,u,t} = Q^{*}_{s,u} + \Delta Q_{s,u,t} \quad \forall u\in BLD,\ t\in TP \tag{20b}$$

$$r^{\mathrm{MIN}}_{s,u,m}\cdot\sum_{s\in IS_u} Q_{s,u,t} \le Q_{s,u,t} \le r^{\mathrm{MAX}}_{s,u,m}\cdot\sum_{s\in IS_u} Q_{s,u,t} \quad \forall u\in BLD,\ t\in TP \tag{21}$$

$$\sum_{m\in M_u} z_{u,m,t} \le 1 \quad \forall u\in BLD,\ t\in TP \tag{22}$$

$$\sum_{t\in T_{u,b}} QO_{u,t} \ge DD^{low}_{u,b} \quad \forall u\in PTK,\ b\in B_u \tag{23}$$

The overall objective function is to maximize the economic profit, including three items: product oils revenues, inventory costs and processing unit running costs, respectively.

$$\max\sum_{t\in TP}\left(\sum_{u\in PTK} o_u\cdot QO_{u,t} - \sum_{u\in TK} s_u\cdot INV_{u,t} - \sum_{u\in HUPUs} OPC_{u,t}\right) \tag{24}$$

The first two items of the objective function Eq. (24) are linear with respect to the decision variables; while the third is nonlinear, representing the operating cost of *HUPUs* as approximated by the CPWL models. (The CPWL models approximating the yield appear in constraints.) For convenience, the objective function (24) is rewritten as

$$\mathbf{max}_{\mathbf{x}}\sum_{t\in TP}\left(f_L(\mathbf{x}_{L_{obj,t}}) - \sum_{u\in HUPUs} PWL^{cost}_{u}(\Delta PRO_{s',u,t,p})\right) \tag{25}$$

where $f_L(\mathbf{x}_{L_{obj,t}})$ contains the linear terms and $\mathbf{x}_{L_{obj,t}}$ denotes the linear decision variables appeared in the objective function, i.e. $QO_{u,t}$ and $INV_{u,t}$.

In summary, the scheduling optimization problem can be formulated as

$$\begin{aligned}
&\mathbf{max}_{\mathbf{x}}\sum_{t\in TP}\left(f_L(\mathbf{x}_{L_{obj,t}}) - \sum_{u\in HUPUs} PWL^{cost}_{u}(\Delta PRO_{s',u,t,p})\right)\\
&\mathbf{s.t}\ \Delta YIELD_{s',u,t} - PWL^{yield}_{s',u}(\Delta PRO_{s',u,t,p}) = 0 \quad \forall u\in HUPUs,\ s'\in OS_u\\
&g_v(\boldsymbol{x}_{L_{eq},t}) = 0 \quad v = 1,2,\ldots,V\\
&h_w(\mathbf{x}_{L_{neq},t}) \le 0 \quad w = 1,2,\cdots,W
\end{aligned} \tag{26}$$

where $\mathbf{x}_{L_{eq},t}$ denotes the decision variables appeared in linear equality constraints, such as $Q_{s,u',u,t}$ in Eq. (9); $\mathbf{x}_{L_{eq},t}$ denotes the decision variables appeared in linear inequality constraints, such as $QI_{u,t}$ in Eq. (16). In more detail, the PWL model for the yield is $PWL^{yield}_{s',u}(\Delta PRO_{s',u,t,p}) = \boldsymbol{c}^T\boldsymbol{\Lambda}(\Delta PRO_{s',u,t,p})$, where c is the regression coefficient vector as in Eq. (5), and $\boldsymbol{\Lambda}(\Delta PRO_{s',u,t,p})$ is the piecewise linear function of the decision variables $\Delta PRO_{s',u,t,p}$.

Based on the aforementioned partition strategy and the specially designed model structure, the original mixed integer nonlinear programming problem summarized in (26) can be explicitly expressed by the finitely countable mixed integer linear program as follows

$$\begin{aligned}
&\mathbf{max}_{\mathbf{x}}\sum_{t\in TP}\left(f_L(\mathbf{x}_{L_{obj,t}}) - \sum_{u\in HUPUs}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{8}\tilde{z}_{u,i,j,k}\cdot L^{cost}_{u,i,j,k}(\Delta PRO_{s',u,t,p})\right)\\
&\mathbf{s.t.}\\
&\Delta YIELD_{s',u,t} - \sum_{u\in HUPUs}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{8}\tilde{z}_{u,i,j,k}\cdot L^{yield}_{s',u,i,j,k}(\Delta PRO_{s',u,t,p})\\
&\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{8}\tilde{z}_{u,i,j,k} = 1 \quad \forall u\in HUPUs\\
&\forall u\in HUPUs,\ s'\in OS_u\\
&g_v(\boldsymbol{x}_{L_{eq},t}) = 0 \quad v = 1,2,\ldots,V\\
&h_w(\boldsymbol{x}_{L_{neq},t}) \le 0 \quad w = 1,2,\ldots,W
\end{aligned} \tag{27}$$

where $L^{cost}_{u,i,j,k}(\Delta PRO_{s',u,t,p})$ denotes the corresponding linear function for $PWL^{cost}_{u}(\Delta PRO_{s',u,t,p})$ in the sub-region $\Omega_{i,j,k}$ and the detailed formulation is given in Eq. (28), similar to $L^{yield}_{s',u,i,j,k}(\Delta PRO_{s',u,t,p})$. $\tilde{z}_{u,i,j,k}$ is a binary variable indicating whether sub-region $\Omega_{i,j,k}$ is selected for piecewise linear functions $PWL^{cost}_{u}(\Delta PRO_{s',u,t,p})$ and $PWL^{yield}_{u}(\Delta PRO_{s',u,t,p})$. For convenience of description, denote $x_{u,1}$ and $x_{v,2}$ to represent the selected decision variables $\Delta PRO_{s',u,t,p}$ (Two delta properties are selected as decision variables from property set in this case study, i.e.

Table 1
Grid size of the CPWL and multi-CPWL models for *HUPUs*.

	SRG gasoline reformer	Diesel hydrotreater1#	Diesel hydrotreater2#	FCC heavy gasoline hydrodesulfurizer	FCC light gasoline etherification
Multi-CPWL	3 × 3	2 × 3	2 × 3	1 × 3	2 × 3
CPWL					
Yield	2 × 2	2 × 2	1 × 3	2 × 2	2 × 3
Operating cost	2 × 3	3 × 2	2 × 2	2 × 3	2 × 3

Table 2
Comparisons of two modeling methods in terms of RSSE.

Unit	CPWL		Multi-CPWL	
	Yield	Operating cost	Yield	Operating cost
Catalytic reformer	0.0739	0.0843	0.0873	0.0921
Diesel hydrotreater	0.0862	0.0801	0.0903	0.0872
Gasoline hydrodesulphurization	0.0735	0.0869	0.0798	0.0903
FCC gasoline etherification	0.0720	0.0797	0.0731	0.0815

desulfurization amount and delta RON for gasoline *HUPUs*). In formula (26), $\bar{x}_{u,1}^{i,j}$ and $\underline{x}_{u,1}^{i,j}$ represent the upper and lower bound of subregion $\Omega_{i,j}$ along the coordinate axis $\boldsymbol{x}_{u,1}$ respectively, and that is similar to $\bar{x}_{u,2}^{i,j}$, $\underline{x}_{u,2}^{i,j}$.

$$L_{u,i,j,k}^{cost}(\boldsymbol{x}_u)=\boldsymbol{c}_{u,0}\Lambda_{u,0}+\begin{cases} c_{u,2,i}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a}) & k=1\\ c_{u,2,i}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a}) & k=2\\ c_{u,2,i}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a}) & k=3\\ c_{u,2,i}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(x_{u,2}-\underline{x}_{u,2}^{i,a})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a}) & k=4\\ c_{u,2,i}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a}) & k=5\\ c_{u,2,i}(\bar{x}_{u,1}^{i,a}-x_{u,1})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a}) & k=6\\ c_{u,2,i}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a}) & k=7\\ c_{u,2,i}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a})+c_{u,2,1+j}(\bar{x}_{u,2}^{i,a}-x_{u,2})(\bar{x}_{u,1}^{i,a}-\underline{x}_{u,1}^{i,a})+c_{u,4,i,j}(x_{u,1}-\underline{x}_{u,1}^{i,a})(\bar{x}_{u,2}^{i,a}-\underline{x}_{u,2}^{i,a}) & k=8 \end{cases} \quad (28)$$

By transformation, the optimization problem (27) is a typical MILP. Various mature MILP solvers can be used for optimization and LINGO is used in this paper.

In summary, the overall model-based scheduling approach starts from collecting historical plant data (simulated plant data in this paper), including the decision variables and the corresponding yield and operating cost. These data are used to train two types of CPWL models, one for the yield (i.e. $PWL_{s',u}^{yield}$, specific to an outlet stream s' of a particular unit u) and another for the cost (i.e. PWL_{u}^{cost}, specific to a unit u). Finally, the CPWL models are plugged into the schedule optimization problem, and due to the linearity of CPWL in sub-regions, the original MINLP problem is converted to an MILP problem.

5. Results

In this case, all *HUPUs*, including the SRG catalytic reformer, two diesel hydrotreaters, FCC heavy gasoline hydrodesulfurizer, and FCC light gasoline etherification unit, need to be modeled using the multi-CPWL method. For each HUPU unit, 1500 data samples were generated from Petro-SIM simulation. During simulation, the operation is varied in such a way that the practical situation is emulated, so that the data provide good coverage of the entire operating space. The data were divided into training (1000 samples) and unseen validation (500 samples) sets to evaluate the modeling accuracy. For comparison, both CPWL and multi-CPWL models were developed for each unit, and the number of grids were determined by using five-fold cross-validation (Martens and Dardenne, 1998). During cross-validation, the candidate grid sizes for each axis are from 1 to 5; this range was determined by experience with preliminary runs of CPWL modeling. The determined grid size is listed in Table 1. It appears that for both modeling methods, a small number of grids are sufficient to model the processes.

The modeling accuracy, in terms of relative sum of squared errors (RSSE) (Xu et al., 2009), is compared in Table 2. It is not surprising that in all cases, the multi-CPWL models are less accurate than multiple CPWL models developed separately, because multi-CPWL has an additional constraint that all response functions have the same partition (i.e. less degrees of freedom than the CPWL models). However, the difference in RSSE is small. Table 3 compares the two partitioning strategies in terms of the number of subregions obtained for the processing units. Clearly, CPWL generates far more subregions that multi-CPWL does, and these large number of subregions are very difficult to be formulated analytically. More importantly, the complex partitions resulting from the CPWL approach will create significant difficulty in subsequent optimization; this will be discussed next.

The scheduling problem is of short term with a horizon of five days, because of the frequent change of oil demand for the investigated refinery. The time interval of discrete decision period is set to 4 h. The future demands are determined by order analysis and monthly plan, provided in the Gantt chart in Fig. 6. According to the actual operation of refinery, demand data for each batch should be met before the

Table 3
Comparison of partitioning strategies (multi-CPWL and CPWL) in term of the number of subregions obtained for the processing units.

Method	Catalytic reformer	Diesel hydrotreater	Gasoline hydrodesulphurization	FCC gasoline etherification
Multi-CPWL	72	48	24	48
CPWL	464	382	344	426

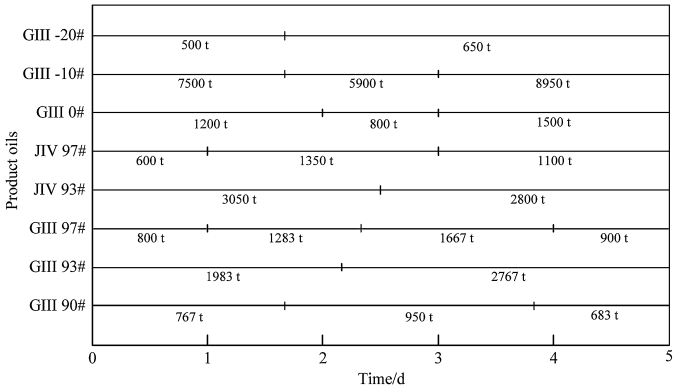

Fig. 6. Future demand of product oils.

end of the batch time, which is formulated as Eq. (23). Here a "batch" refers to a time period in which the demand for each grade of the product oils is constant.

Optimization is implemented with LINGO 11 on a platform of Intel® Corel™ 2 Duo CPU 2.93 GHz with 1.96GB RAM. The multi-CPWL based model involves 4980 continuous variables, 1392 binary variables and 8060 constraints, while the CPWL model has 4980 continuous variables, 240 binary variables and 8060 constraints. Comparing with the multi-CPWL based model, though there is much less binary variables for CPWL based model, it is more time-consuming and computationally difficult due a MINLP problem is solved. The results are summarized in Table 4 for comparing the use of the two modeling methods for scheduling. In CPWL, functions are separately approximated and the resultant partitions, when combined, cannot be analytical expressed. Therefore, the original MINLP problem cannot be explicitly converted to MILP. Optimization using the CPWL approach relies on the point search algorithm (Zhu et al., 2010). In more detail, an initial point is randomly selected in the domain of decision variables, and an MILP can be constructed at the neighborhood of this initial point to obtain a locally optimal solution. Then, small perturbation is applied to the current local optimum to arrive at another point, around which another MILP can be constructed and solved. This procedure is repeated until a certain convergence criterion is met. As shown in Table 4, this CPWL method, together with the point search algorithm for optimization, consumes far more computation time than the multi-CPWL method. In addition, it appears that the slightly better accuracy of the CPWL method has been offset by its inability to find the global optimum. In contrast, the multi-CPWL approach allows the problem to be formulated as an MILP, and the global optimum was found. As a result, the obtained profit from the CPWL method is 182,335 Chinese Yuan (or 4.2%) less than that from multi-CPWL.

To further illustrate the results, the Gantt chart for the scheduled gasoline blender's operation is shown in Fig. 7. We can verify that this schedule is able to meet the demand, and at any particular time only one gasoline grade is being produced.

The schedules for the gasoline *HUPUs*, using either multi-CPWL or CPWL models, are shown in Fig. 8. It appears that the two models results in schedules that have similar trend. The demand for premium and high grade gasoline (e.g. JIV93#, JIV97# and GIII 97#) first

Table 4
Optimization results comparisons.

Model	CPU time/s	Computed profit/Yuan
Multi-CPWL	2631	4,365,271
CPWL (Zhu et al., 2010)	44,710	4,182,936

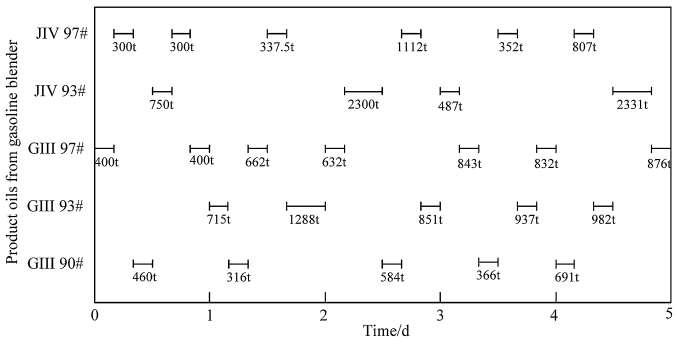

Fig. 7. Gantt chart for gasoline blender's operations.

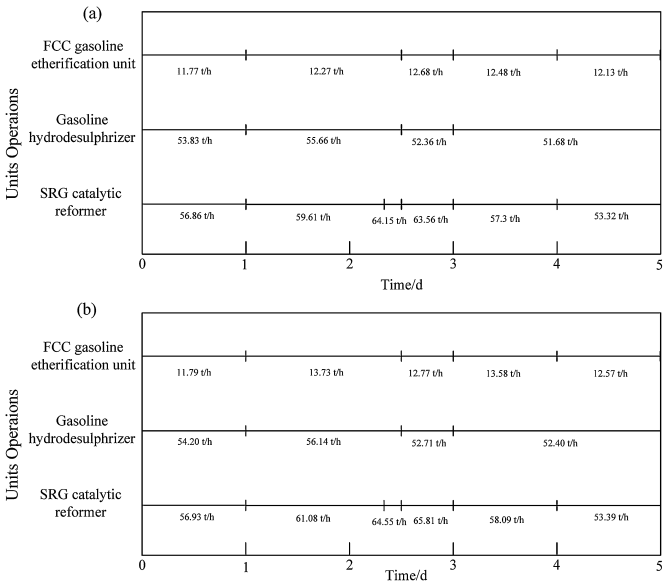

Fig. 8. Gantt chart for gasoline *HUPUs* based on (a) multi-CPWL model, and, (b) CPWL model.

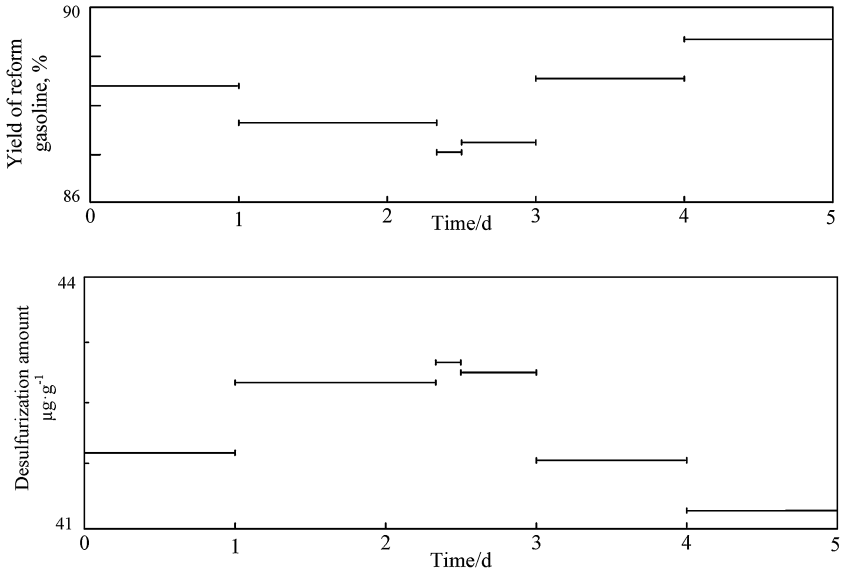

Fig. 9. Yield and desulfurization amount of SRG reformer, obtained by using the multi-CPWL scheduling model.

increases and then decreases over time. In addition, etherification and catalytic reforming are two important processing steps to produce gasoline, which after blending gives premium and high grade product. Therefore, the schedule dictates that the charging amount for FCC gasoline etherification and SRG catalytic reformer should also increase first, and then decrease to match the blending demand. Moreover, the FCC heavy gasoline hydrodesulfurizer produces lower-sulfur oil needed for premium gasoline, and thus its schedule also exhibits the same pattern. However, a close inspection reveals that the multi-CPWL model requires less hydro-upgrading (less production rate for all three units) than CPWL does, and thus the schedule obtained using multi-CPWL is less costly.

The developed multi-CPWL models also enable fine adjustment of *HUPUs*, which is difficult with the conventional approach where the yield and cost are fixed for a pre-defined operation mode. For example, the schedule for the SRG catalytic reformer is depicted in Fig. 9. To satisfy the change of demand for premium and high grade gasoline, the SRG catalytic reformer first tries to increase the desulphurization amount (higher quality oil and lower yield), and then reduce desulphurization when demand for low-grade fuel is high. Such fine tuning of the operation provides more flexibility, and thus opportunities to improve the profit while meeting the demand.

6. Concluding remarks

In this paper, a new piecewise linear modeling method was proposed to facilitate the optimal scheduling of refineries. Being able to model global nonlinearity with local linear regressions, the model is well suited to adequately approximating the process, while also keeping the computation manageable when solving the optimization problem. In particular, to deal with the curse of partitions for modeling multiple functions, a unified simplicial partitioning strategy was developed. This strategy and the piecewise linear formulation allows the original MINLP to be converted to an equivalent MILP, which can be efficiently solved. The proposed methodology was demonstrated through a

case study, which was simulated using Petro-SIM to mimic a real refinery in China. It should be noted that although the simplified MILP and the obtained schedule are approximation of the original MINLP, the proposed method is practically useful because, first, the approximate model has satisfactory accuracy (Table 2) and thus the optimized schedule gives significant improvement in profit (Table 4). Moreover, the proposed PWL method is only used for intermediate processing units (i.e. hydro-upgrading processing units (*HUPUs*) in this paper) for the purpose of process optimization. The outlets of these *HUPUs* are fed to blenders to produce the end product. In practice, the blenders are usually equipped with on-line analyzers to ensure the quality of end product oil. Follow on work to extend the methodology for high dimensional decision space is currently undergoing.

Acknowledgements

This research was supported by the National Natural Science Foundation of China (Nos. 21276137, 61273039). Xiaoyong Gao's visit to the University of Surrey was funded by the Santander Universities through a Santander Postgraduate Research Award.

Appendix A. Quality specification of the gasoline and diesel products

Tables A1 and A2

Appendix B. The derivation of the partitions in Figs. 3 and 4 using the generating function

First, we show how the 2nd level function, $\gamma^2(\pi_{i,a,1}, \pi_{i,a,2})$, represents the partition in Fig. 3(a). The definition of the function provides the following expression:

$$\gamma^2(\pi_{i,a,1}, \pi_{i,a,2}) = \gamma(\pi_{i,a,1}, \pi_{i,a,2}) = \begin{cases} \pi_{i,a,1} & \text{if } 0 \le \pi_{i,a,1} \le \pi_{i,a,2} \\ \pi_{i,a,2} & \text{if } 0 \le \pi_{i,a,2} \le \pi_{i,a,1} \\ 0 & \text{if } \pi_{i,a,1} < 0 \text{ or } \pi_{i,a,2} < 0 \end{cases}$$

Table A1
Detailed quality specification of different grade gasoline products.

Items		GIII90#	GIII93#	GIII97#	JIV93#	JIV97#
Antiknock quality						
RON	Not less than	90	93	97	93	97
(RON + MON)/2	Not less than	85	88	Report	88	Report
Lead content/(g/L)	Not more than		0.005		0.005	
Iron content/(g/L)	Not more than		0.01		0.01	
Manganese content/(g/L)	Not more than		0.016		0.006	
Sulfur content/mg/kg	Not more than		50		50	
Benzene (by volume)/%	Not more than		1.0		1.0	
Aromatics (by volume)/%	Not more than		40			
Aromatics + olefins/%	Not more than				60	
Olefin (by volume)/%	Not more than		28		25	
Oxygen content (by mass)/%	Not more than		2.7		2.7	
Reid vapor pressure/kPa						
1st November–30th April			42–85		≤88	
1st May–31th October			40–68		≤65	
Distillation range						
10% point/°C	Not more than		70		70	
50% point/°C	Not more than		120		120	
90% point/°C	Not more than		190		190	
End point/°C	Not more than		205		205	
Residue (by volume)/%	Not more than		2		2	

Table A2
Detailed quality specification of different grade diesel products.

Items		GIII 0#	GIII −10#	GIII −20#
Sulfur content (by mass)/% not more than	Not more than		0.035	
Acidity grade (mgKOH/100 mL)	Not more than		7	
Carbon residue (by mass)/%	Not more than		0.3	
Freezing point/°C	Not more than	0	−10	−20
Viscosity (20 °C)/(mm^2/s)		3.0–8.0		2.5–8.0
Flash point (closed cup)/°C	Not less than		55	
Flammability				
Cetane number	Not less than		45	
Cetane index	Not less than		43	
Distillation range				
50% point/°C	Not more than		300	
90% point/°C	Not more than		355	
95% point/°C	Not more than		365	

which suggests that $\gamma^2(\pi_{i,a,1}, \pi_{i,a,2})$ is non-zero only when $\pi_{i,a,1} > 0$ and $\pi_{i,a,2} > 0$. According to Eq. (3), $\pi_{i,a,1} - (x_1 - \xi_i)(\zeta_{a+1} - \zeta_a)$, $\pi_{i,a,2} - (\xi_{i+1} - x_1)(\zeta_{a+1} - \zeta_a)$. Therefore, for $\gamma^2(\pi_{i,a,1}, \pi_{i,a,2})$ to be non-zero, x_1 must satisfy the condition: $\xi_i < x_1 < \xi_{i+1}$, which gives the rectangular area as depicted in Fig. 3(a). Following the same procedure would give the result for Fig. 3(b).

Second, we follow the similar procedure to derive the partition in Fig. 4. Let $r_1 = i$ and $r_2 = j$, then,

$$\gamma^4(\pi_{i,j,1}, \ldots, \pi_{i,j,4}) = \gamma^2(\gamma^2(\pi_{i,j,1}, \pi_{i,j,2}), \gamma^2(\pi_{i,j,3}, \pi_{i,j,4})) = \begin{cases} \gamma^2(\pi_{i,j,1}, \pi_{i,j,2}) & \text{if } 0 \le \gamma^2(\pi_{i,j,1}, \pi_{i,j,2}) \le \gamma^2(\pi_{i,j,3}, \pi_{i,j,4}) \\ \gamma^2(\pi_{i,j,3}, \pi_{i,j,4}) & \text{if } 0 \le \gamma^2(\pi_{i,j,3}, \pi_{i,j,4}) \le \gamma^2(\pi_{i,j,1}, \pi_{i,j,2}) \\ 0 & \text{if } \gamma^2(\pi_{i,j,3}, \pi_{i,j,4}) < 0 \text{ or } \gamma^2(\pi_{i,j,1}, \pi_{i,j,2}) < 0 \end{cases}$$

This implies that $\gamma^4(\pi_{i,j,1}, \ldots, \pi_{i,j,4})$ is non-zero within $[\xi_i, \xi_{i+1}] \times [\zeta_j, \zeta_{j+1}]$ for a specific set of i and j, which is depicted as the shaded rectangular area in Fig. 4. In addition, $\gamma^2(\pi_{i,j,1}, \pi_{i,j,2})$ divides the grey area into two parts with the boundary line $x_1 = \xi_i + \xi_{i+1}/2$, with left side being $\pi_{i,j,1}$ and right side $\pi_{i,j,2}$. Similarly, $\gamma^2(\pi_{i,j,3}, \pi_{i,j,4})$ divides the grey area into two parts with the boundary line $x_2 = \zeta_j + \zeta_{j+1}/2$, with the lower side being $\pi_{i,j,3}$ and the up side $\pi_{i,j,4}$. Up to now, four rectangular sub-regions have been generated.

Furthermore, if a point (x_1, x_2), lies in the lower-left rectangular sub-region, then $\gamma^2(\pi_{i,j,1}, \pi_{i,j,2}) = \pi_{i,j,1}$, and $\gamma^2(\pi_{i,j,3}, \pi_{i,j,4}) = \pi_{i,j,3}$; thus

$$\gamma^4(\pi_{i,j,1}, \cdots \pi_{i,j,4}) = \gamma^2(\gamma^2(\pi_{i,j,1}, \pi_{i,j,2}), \gamma^2(\pi_{i,j,3}, \pi_{i,j,4})) = \gamma^2(\pi_{i,j,1}, \pi_{i,j,3}) = \begin{cases} \pi_{i,j,1} & \text{if } 0 \le \pi_{i,j,1} \le \pi_{i,j,3} \\ \pi_{i,j,3} & \text{if } 0 \le \pi_{i,j,3} \le \pi_{i,j,1} \\ 0 & \text{if } \pi_{i,j,3} < 0 \text{ or } \pi_{i,j,1} < 0 \end{cases}$$

This results in a crossing division line, $\pi_{i,j,1} - \pi_{i,j,3} = 0$, which again divides the lower-left rectangular sub-region into two triangular sub-regions. The same procedure can be followed to derive the other six triangular sub-regions, resulting in the partition shown in Fig. 4.

Appendix C. The gradients used for training the multi-CPWL model

The optimization objective can be rewritten as the following

$$E = \frac{1}{2}\sum_{k=1}^{K}\left(y_k - \left(\begin{array}{l} c_{0,1} + \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{t=1}^{4} c_{0,(i-1)I+(j+1)J+t+1}\pi_{i,j,t}(x_k) \\ +\sum_{i=1}^{I} c_{2,i}\min(\max(\pi_{i,a,1}(x_k), 0), \max(\pi_{i,a,2}(x_k), 0)) \\ +\sum_{j=1}^{J} c_{2,I+j}\min(\max(\pi_{a,j,3}(x_k), 0), \max(\pi_{a,j,4}(x_k), 0)) \\ +\sum_{i=1}^{I}\sum_{j=1}^{J} c_{4,i,j}\min\left(\begin{array}{l}\min(\max(\pi_{i,j,1}(x_k), 0), \max(\pi_{i,j,2}(x_k), 0)), \\ \min(\max(\pi_{i,j,3}(x_k), 0), \max(\pi_{i,j,k}(x_k), 0))\end{array}\right)\end{array}\right)\right) = \frac{1}{2}\sum_{k=1}^{K} e_k^2 \tag{B.1}$$

The negative gradient of Eq. (B.1) can be obtained as follows:

$$\begin{aligned}\frac{\partial E}{\partial \xi_i} = &-\sum_{k=1}^{K} e_k \sum_{i=1}^{I}\sum_{j=1}^{J} c_{0,(i-1)I+(j-1)J+2}(\zeta_{j+1} - \zeta_j) + c_{0,(i-1)I+(j-1)J+4}(x_2 - \zeta_j) + c_{0,(i-1)I+(j-1)J+5}(\zeta_{j+1} - x_2) \\ &-\sum_{k=1}^{K} e_k \sum_{i=1}^{I} c_{2,i}\left(\begin{cases} \varsigma_t - \varsigma_{t+1} & (x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) > 0 \wedge (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) > 0 \\ & \wedge (x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) - (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) > 0 \\ 0 & \text{else} \end{cases}\right) \\ &-\sum_{k=1}^{K} e_k \sum_{i=1}^{I}\sum_{j=1}^{J} c_{4,i,j}\left(\begin{cases} \zeta_j - \zeta_{j+1} & \begin{array}{l}(x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) > 0 \wedge (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) > 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_j) > 0 \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) > 0 \\ \wedge (x_1 - \xi_j)(\zeta_{j+1} - \zeta_j) - (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) \le 0 \\ \wedge (x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) - (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) \le 0 \\ \wedge (x_1 - \xi_j)(\zeta_{j+1} - \zeta_j) - (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) \le 0 \end{array} \\ -x_2 + \zeta_j & \begin{array}{l}(x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) > 0 \wedge (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) > 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) > 0 \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) > 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) - (x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) \le 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) - (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) \le 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) - (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) \le 0 \end{array} \\ x_2 - \zeta_{j+1} & \begin{array}{l}(x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) > 0 \wedge (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) > 0 \\ \wedge (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) > 0 \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) > 0 \\ \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) - (x_1 - \xi_i)(\zeta_{j+1} - \zeta_j) \le 0 \\ \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) - (\xi_{i+1} - x_1)(\zeta_{j+1} - \zeta_j) \le 0 \\ \wedge (\zeta_{j+1} - x_2)(\xi_{i+1} - \xi_i) - (x_2 - \zeta_j)(\xi_{i+1} - \xi_i) \le 0 \end{array} \\ 0 & \text{else} \end{cases}\right)\end{aligned}$$

$$\frac{\partial E}{\partial \zeta_i} = -\sum_{k=1}^{K} e_k \sum_{i=1}^{I}\sum_{j=1}^{J} c_{0,(i-1)I+(j-1)J+2}(x_1-\xi_i) + c_{0,(i-1)I+(j-1)J+3}(\xi_{i+1}-x_1) + c_{0,(i-1)I+(j-1)J+4}(\xi_{i+1}-\xi_i)$$
$$-\sum_{k=1}^{K} e_k \sum_{j=1}^{j} c_{2,I+j} \left(\begin{cases} \xi_t - \xi_{t+1} & (x_2-\zeta_j)(\xi_{t+1}-\xi_t) > 0 \wedge (\zeta_{j+1}-x_2)(\xi_{t+1}-\xi_j) > 0 \\ & \wedge (x_2-\zeta_j)(\xi_{t+1}-\xi_t) - (\zeta_{j+1}-x_2)(\xi_{t+1}-\xi_j) > 0 \\ 0 & \text{else} \end{cases} \right)$$
$$-\sum_{k=1}^{K} e_k \sum_{i=1}^{I}\sum_{j=1}^{J} c_{4,i,j} \left(\begin{cases} \xi_i - \xi_{i+1} & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) > 0 \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) > 0 \\ & \wedge (x_2-\zeta_j)(\xi_{i+1}-\xi_i) > 0 \wedge (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) > 0 \\ & \wedge (x_2-\zeta_j)(\xi_{i+1}-\xi_i) - (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) \le 0 \\ & \wedge (x_2-\zeta_i)(\xi_{i+1}-\xi_i) - (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) \le 0 \\ & \wedge (x_2-\zeta_j)(\xi_{i+1}-\xi_i) - (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) \le 0 \\ -x_2+\zeta_j & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) > 0 \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) > 0 \\ & \wedge (x_2-\zeta_j)(\xi_{i+1}-\xi_i) > 0 \wedge (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) > 0 \\ & \wedge (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) - (x_2-\zeta_j)(\zeta_{j+1}-\zeta_j) \le 0 \\ & \wedge (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) - (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) \le 0 \\ & \wedge (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) - (x_2-\zeta_j)(\xi_{i+1}-\xi_i) \le 0 \\ x_2-\zeta_{j+1} & (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) > 0 \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) > 0 \\ & \wedge (x_2-\zeta_j)(\xi_{i+1}-\xi_i) > 0 \wedge (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) > 0 \\ & \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) - (x_1-\xi_i)(\zeta_{j+1}-\zeta_j) \le 0 \\ & \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) - (\zeta_{j+1}-x_2)(\xi_{i+1}-\xi_i) \le 0 \\ & \wedge (\xi_{i+1}-x_1)(\zeta_{j+1}-\zeta_j) - (x_2-\zeta_j)(\xi_{i+1}-\xi_i) \le 0 \\ 0 & \text{else} \end{cases} \right)$$

where $\wedge$ is the logic AND operator.

u, $Q_{s,u',u,t}$.

References

Aylott M, Van der Merwe B. Petro-SIM simulator and cape-open: experiences and successes. In: The 2008 annual meeting; 2008.
Bengtsson J, Nonås SL. Refinery planning and scheduling: an overview. Energy Nat Resour Environ Econ 2010:115–30.
Breiman L. Hinging hyperplanes for regression, classification and function approximation. IEEE Trans Inf Theory 1993;39:999–1013.
Chua L, Kang S. Section-wise piecewise linear functions: canonical representation, properties, and applications. Proc IEEE 1977;65:915–29.
Dogan M, Grossmann I. A decomposition method for the simultaneous planning and scheduling of single-stage continuous multiproduct plants. Ind Eng Chem Res 2006;45(1):299–315 [Comput Chem Eng 28(11), 2109–29].
Floudas C, Lin X. Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. Comput Chem Eng 2004;28(11):2109–29.
Gao X, Jiang Y, Huang D, Xiong Z. A novel high level canonical piecewise linear model based on the simplicial partition and its application. ISA Trans 2014a;53:1420–6.
Gao X, Shang C, Jiang Y, Chen T, Huang D. Refinery scheduling with varying crude: a deep belief network classification and multimodel approach. AIChE J 2014b;60:2525–32.
Göthe-Lundgren M, Lundgren J, Persson J. An optimization model for refinery production scheduling. Int J Prod Econ 2002;78(3):255–70.
Harjunkoski I, Maravelias CT, Bongers P, Castro PM, Engell S, Grossmann IE, et al. Scope for industrial applications of production scheduling modes and solution methods. Comput Chem Eng 2014;62:161–93.
Huang X, Mu X, Wang S. Continuous piecewise linear identification with moderate number of subregions. In: 16th IFAC symposium on system identification; 2012. p. 535–40.
Jia Z, Ierapetriton M. Efficient short-term scheduling of refinery operations based on a continuous time formulation. Comput Chem Eng 2004;28:1001–19.
Jia Z, Ierapetriton M, Kelly JD. Refinery short-term scheduling using continuous time formulation: crude-oil operations. Ind Eng Chem Res 2003;42(13):3085–97.
Joly M. Refinery production planning and scheduling: the refining core business. Braz J Chem Eng 2012;29(2):371–84.
Joly M, Moro LFL, Pinto JM. Planning and scheduling for petroleum refineries using mathematical programming. Braz J Chem Eng 2002;19(2):207–28.
Julián P, Desages A, Agamennoni O. High level canonical piecewise linear representation using a simplicial partition. IEEE Trans Circuits Syst I: Fundam Theory Appl 1999;46:463–80.
Kondili E, Pantelides CC, Sargent RWH. A general algorithm for short-term scheduling of batch operations-I. MILP formulation. Comput Chem Eng 1993;17:211–27.
Li W, Hui C, Li A. Integrating CDU, FCC and product blending models into refinery planning. Comput Chem Eng 2005;29:2010–28.
Lin J-N, Unbehauen R. Canonical piecewise-linear approximations. IEEE Trans Circuits Syst I: Fundam Theory Appl 1992;39(8):697–9.
Lin J, Xu H, Unbehauen R. A generalization of canonical piecewise-linear functions. IEEE Trans Circuits Syst I: Fundam Theory Appl 1994;41:345–7.
Luo C, Rong G. Hierarchical approach for short-term scheduling in refineries. Ind Eng Chem Res 2007;46:3656–68.
Martens H, Dardenne P. Validation and verification of regression in small data sets. Chemom Intell Lab Syst 1998;44(1–2):99–121.
Misener R, Floudas CA. GloNIQO: global mixed-integer quadratic optimizer. J Glob Optim 2013;57:3–50.
Mohaddecy SRS, Zahedi S, Sadighi S, Bonyad H. Reactor modeling and simulation of catalytic reforming process. Pet Coal 2006;48:28–35.
Pinto JM, Joly M, Moro LFL. Planning and scheduling models for refinery operations. Comput Chem Eng 2000;24:2259–76.
Sahinidis VV. BARON: a general purpose global optimization software package. J Glob Optim 1996;8:201–5.
Shah NK, Ierapetritou MG. Short-term scheduling of a large-scale oil-refinery operations: incorporating logistics details. AIChE J 2011;57:1570–84.
Shah NK, Li ZK, Ierapetritou MG. Petroleum refining operations: key issues, advances, and opportunities. Ind Eng Chem Res 2010;50:1161–70.
Shobrys DE, White DC. Planning, scheduling and control systems: why cannot they work together. Comput Chem Eng 2002;26:149–60.
Singh A, Forbes JF, Vermeer PJ, Woo SS. Model-based real-time optimization of automotive gasoline blending operations. J Process Control 2000;10:43–58.
Smania P, Pinto JM. Mixed integer nonlinear programming techniques for the short term scheduling of oil refineries. Comput Aided Chem Eng 2003;15:1038–43.
Wang S, Sun X. Generalization of hinging hyperplanes. IEEE Trans Inf Theory 2005;12:4425–31.
Wu D, Ierapetritou M. Hierarchical approach for production planning and scheduling under uncertainty. Chem Eng Process 2007;46(11):1129–40.
Xu J, Huang X, Wang S. Adaptive hinging hyperplanes and its applications in dynamic system identification. Automatica 2009;45:2325–32.
Yan W, Hu S, Yang Y, Gao F, Chen T. Bayesian migration of Gaussian process regression for rapid process modeling and optimization. Chem Eng J 2011;166:1095–103.
Zhang X, Sargent RWH. The optimal operation of mixed production facilities – general formulation and some approaches for the solution. Comput Chem Eng 1996;20:897–904.
Zhang N, Zhu XX. A novel modeling and decomposition strategy for overall refinery optimization. Comput Chem Eng 2000;24:1543–8.
Zhu Y, Lv W, Gao X, Jiang Y, Huang D. Optimization on distillation via piecewise linear approximation. In: 8th IEEE international conference on control and automation; 2010.