HAL open science

Online gasoline blending with EPA Complex Model for predicting emissions

Stefan Janaqi, Mériam Chèbre, Guillaume Pitollat

To cite this version:

Stefan Janaqi, Mériam Chèbre, Guillaume Pitollat. Online gasoline blending with EPA Complex Model for predicting emissions. Frontiers of Engineering Management, 2018, 10.15302/J-FEM-2017022 . hal-04083388

HAL Id: hal-04083388

https://imt-mines-ales.hal.science/hal-04083388

Submitted on 27 Apr 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Front. Eng. Manag. 2018, 5(2): 214–226
https://doi.org/10.15302/J-FEM-2017022

RESEARCH ARTICLE

Stefan JANAQI, Mériam CHÈBRE, Guillaume PITOLLAT

Online gasoline blending with EPA Complex Model for predicting emissions

© The Author(s) 2017. Published by Higher Education Press. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0)

Abstract The empirical Complex Model developed by the US Environmental Protection Agency (EPA) is used by refiners to predict the toxic emissions of reformulated gasoline with respect to gasoline properties. The difficulty in implementing this model in the blending process stems from the implicit definition of Complex Model through a series of disjunctions assembled by the EPA in the form of spreadsheets. A major breakthrough in the refinery-based Complex Model implementation occurred in 2008 and 2010 through the use of generalized disjunctive and mixed-integer nonlinear programming (MINLP). Nevertheless, the execution time of these MINLP models remains prohibitively long to control emissions with our online gasoline blender. The first objective of this study is to present a new model that decreases the execution time of our online controller. The second objective is to consider toxic thresholds as hard constraints to be verified and search for blends that verify them. Our approach introduces a new way to write the Complex Model without any binary or integer variables. Sigmoid functions are used herein to approximate step functions until the measurement precision for each blend property is reached. By knowing this level of precision, we are able to propose an extremely good and differentiable approximation of the Complex Model. Next, a differentiable objective function is introduced to penalize emission values higher than the threshold emissions. Our optimization module has been implemented and tested with real data. The execution time never exceeded 1 s, which allows the online regulation of emissions the same way as other traditional properties of blended gasoline.

Keywords emissions, reformulated gasoline, online control, global optimization

Received April 5, 2017; accepted August 21, 2017

Stefan JANAQI (✉)
Ecole des Mines d'Alès, LGI2P, Parc G. Besse, 30035 Nîmes, France
E-mail: stefan.janaqi@mines-ales.fr

Mériam CHÈBRE
TOTAL SA, 24, crs Michelet, 92069 PARIS LA DEFENSE Cedex, France

Guillaume PITOLLAT
3Manufacturing Engineering, BECKMAN COULTER, Georgia Institute of Technology, Marseille, France

1 Introduction

The US Environmental Protection Agency (EPA, 2015) established reformulated gasoline (RFG) guidelines in reducing emissions from gasoline-powered vehicles to satisfy air quality requirements. In 1998, the EPA introduced the Complex Model, which is denoted herein as EPA-CM, to allow refineries calculate the nitrogen oxides (NO_x), volatile organic compounds (VOCs), and toxic air pollutants (TOX) based on 11 gasoline properties. The regulations embedded in the EPA-CM are available in print form (EPA, 2015). The EPA also publishes and updates an Excel spreadsheet (EPA, 2016) to calculate the emission reductions compared to a statutory baseline gasoline. This spreadsheet is selected to give the reference equations of the present study. In its original form, the EPA-CM is difficult to implement within mathematical optimization models. Since 1998, considerable effort has been expended (see Appendix for a bibliographical background) to implement the EPA-CM into refinery operations. The first breakthrough occurred in 2008 (Furman and Androulakis, 2008) and 2010 (Misener et al., 2010), in which EPA-CM was represented as a mixed-integer nonlinear programming (MINLP) using disjunctive programming techniques. As discussed in the next section, the EPA-CM cannot be operated with the original properties of the target gasoline. More specifically, this model utilizes edge target and delta target values derived from the target property values by satisfying a number of "if–then"

conditions, which introduce binary variables that must be managed along with continuous variables to mathematically model the emission functions. The MINLP by Furman and Androulakis (2008) is composed of 78 continuous variables and 21 binary variables. The authors' signal computation times for the optimization solver extend approximately 600 s. The part of the EPA-CM model included in the more general pooling problem treated by Misener et al. (2010) requires 87 continuous variables and only 15 binary variables. The EPA-CM is part of the pooling problem, and its execution time has not been reported accurately. Nevertheless, an EPA-CM computation time of approximately 140 s is estimated in Appendix A.

The current state-of-the-art approaches cannot control the emission in online blending to the same extent as the typical properties of target gasoline. The present work addresses this problem. As mentioned previously, MINLP model computation times remain prohibitively long for online blending processes. Our online controller ANAMEL (Chèbre et al., 2010) actually calls the optimization solver up to 16 times/min (because of the frequency of changes in component properties) to regulate some 40 properties of the target gasoline; thus, the optimal solution is returned by the solver in less than 4 s.

Our approach is completely different from all preceding contributions. We begin by introducing a differentiable approximation of the EPA-CM emission functions. To this end, sigmoid functions are used as an approximate differentiable variant of if–then logical conditions contained in the original model. The accuracy of this approximation is controlled by using the measured precision of gasoline properties. In this way, we avoid the need for binary variables and mixed constraints. Then, instead of imposing (non-convex) constraints on emission values, we minimize an objective function that penalizes emission values exceeding a given threshold. We thus obtain an optimization problem with a differentiable and non-convex objective function with respect to the classical linear constraints on recipe and gasoline properties.

This new control/optimization problem is implemented and tested with real data. The execution time of the global optimizer solver, which is written in MATLAB, is less than 1 s; this good result paves the way to controlling emissions online the same way as the other traditional properties of blended gasoline. This fast model is also suitable for refinery scheduling departments. The international patent (Janaqi et al., 2015) for our process was awarded in July 2015.

This paper is organized as follows. Section 2 provides a bibliographical review. Section 3 describes the NO_x computation and lists useful definitions in the remainder of the article. Section 4 contains the theoretical development of our approach. Section 5 provides some implementation details and results. Finally, Section 6 presents the conclusion.

2 Bibliographical review

RFG fabrication is the subject of the first part of a negotiation between the US legislature and refiners in 1991. This negotiation aimed to decrease toxic emissions (i.e., NO_x, VOC, and TOX) of an RFG with respect to a baseline gasoline. This negotiation defined new standards that were subsequently introduced into the calculation models, namely, EPA Simple Model and EPA-CM (in effect since 1998). The following paragraphs chronologically review the papers discussing the economic, mathematical, and implementation aspects of the calculation models.

In 1994, Nocca et al. (1994) discussed the specifications to be imposed on RFG and the modifications to an upstream trial on a mixture to satisfy the emission constraints imposed by the EPA Simple and Complex Models.

At the same time, Hirshfield and Kolb (1994) demonstrated the benefits of linking EPA-CM with the mathematical models (i.e., linear and nonlinear) used in refineries. They stressed the need to formulate "simplified reduced forms" of EPA-CM that could easily be integrated with mathematical solvers. They did not succeed however to introduce simplified reduced (linear) forms because these forms, as signaled by Treiber (1998), provide highly inaccurate approximations.

In 1995, Trierwiler (1995) asserted that the emission functions cannot be (globally) linearized with an acceptable level of precision sufficient to be integrated into the refinery linear programming. He developed a tangent surface technique to locally linearize the EPA functions. We observed that this technique, although theoretically solid, can only be applied efficiently in the neighborhood of gasolines x whose emission functions are relatively linear. We found some actual cases with emission functions exhibiting strong variations around some gasolines x. The singular Hessian values at these points vary between 10^{-1} and 10^{3}. In such cases, the linear approximation is only valid over an extremely small zone around x; consequently, recipe modification by the solver is nearly zero.

In 1995, Korotney (1995) conducted a program to test whether some EPA-CM extrapolations might be useful. They showed in several cases that EPA-CM outputs emission values greater than the actual emissions. At first glance, this finding would seem to be extremely good given that if EPA-CM emissions were less than certain thresholds, then the actual emissions would surely comply with the legal requirements. The problem here is that, with such overestimations, the feasible zone narrows and blend cost rises.

During the same period, Jackson and Vittachi (1995) integrated the EPA Simple Emissions Model into the online control; their model only set bounds for TOX and exhausted benzene. Moreover, they estimated the impact

of each component over the two pollutants and introduced this information into the online control. Their work did not integrate the EPA-CM.

Cason (1997) performed a thorough analysis of the influence of emission constraints on the flexibility of gasoline blending. He demonstrated the impossibility of defining easy-to-respect bounds on the blend properties that would imply emissions in compliance with the EPA Standard because a large reduction in blend flexibility implies a needless cost increase. Mathematically speaking, introducing supplementary bounds on gasoline properties dramatically reduces the feasible zone. Cason wrote, "It is doubtful whether a refiner could consistently blend any gasoline with these properties... The industry is effectively stifled by mathematics."

In the same year, Hirshfield and Kolb (1997) released a model for estimating the cost of emission reduction as required by the EPA Simple and Complex Models. They introduced a linear model of a virtual refinery that simulated the blending of five types of RFG under more restrictive emission constraints for Petroleum Administration for Defense Districts I, II, and III. Moreover, they demonstrated a monotonic increase in the cost of emission reduction for the five considered RFGs and proved that the major cost component is accounted for by the cost of refinery operation and the cost of credit required to update the technology in response to the new standards. Nevertheless, ancillary costs (due to more difficult management of basic stocks, tests and traceability, non-compliant blends, safety margins, and operational flexibility) are considered to lie between 12% and 15% of the total blend cost. Online blend control would reduce these ancillary costs to a considerable extent.

Treiber (1998) indicated that the strategies organized to satisfy the EPA-CM constraints were initially aimed at reducing reid vapor pressure (RVP) and later at reducing sulfur content (SUL). The sale of RFG violating the statutory emission thresholds would involve civil or criminal penalties. They also indicated the gasoline properties most heavily influencing emissions (i.e., $\mathrm{VOC}(x) = f(\mathrm{RVP})$, $\mathrm{NO}_x(x) = f(\mathrm{SUL})$, and $\mathrm{TOX}(x) = f(\mathrm{BEN})$). Two optimization strategies were proposed. The first strategy consisted of defining adequate thresholds on RVP, SUL, and benzene content (BEN) and then using a linear program. To counter the interactions between variables and the effect of the other variables on emissions, a wide safety margin was observed. This optimization step resulted in a 1.1% blend cost reduction. The second strategy used a nonlinear optimizer to obtain all the variable interactions and precise emission values. The safety margin was scaled back, and the finer optimization approach lowered cost by 1.8%.

Naman (1999) used linear regression to predict emissions according to gasoline quality. Ideally, a "good" linear approximation of emissions would solve the problem of integrating emission constraints during the programming phase, as well as during the gasoline blending phase. He established two models. The first model was local and resulted in randomly generating 100 blends that followed a normal distribution with a mean equal to a typical gasoline and standard deviation equal to the laboratory measurement error. In other words, he replicated the same gasoline 100 times. The standard deviations of his results were as follows: TOX = 1.162, NO_x= 10.419, and VOC = 33.539. The second model generated gasolines uniformly over a wide min/max interval (closer to reality). In the latter case, the linear approximation errors were as follows: TOX = 8.991, NO_x = 101.314, and VOC = 112.474. These standard deviations are extremely large, thereby reconfirming our opinion regarding the strong nonlinearity of functions used to calculate EPA-CM emissions.

In 2008 (given the jump in time, one could surmise that the problems concerning EPA-CM had been definitively resolved), Furman and Androulakis (2008) tackled the critical problem of satisfying even more restrictive emission constraints while imposing a minimum number of changes in the blending process. They asserted that the current EPA-CM representation is extremely difficult to implement in refining operations or combine with gasoline design models. The main motivation behind their work was to derive a model capable of including the EPA-CM constraints within a mathematical programming formulation. The major difficulty herein lies in the EPA-CM formulation as a series of complicated disjunctive constraints that basically makes its incorporation into algebraic optimization models impossible. At present, a large array of typical "boutique" gasolines are available (up to 14 in number) that meet state and/or local air quality specifications. Such formulations use the technique of writing disjunctive constraints as a series of mixed binary/continuous variables, resulting in an optimization problem with 20 binary variables, 100 continuous variables, and 200 mixed constraints (for a blending problem with 10 components and 12 properties). Their objective function is a normalized distance between the target gasoline x and the baseline gasoline xb (see objective function $F_{FA}(x)$, Eq. (5)). The indicated calculation times are on the order of 10 min in a 2.4 GHz AMD Athlon processor with 2 GB memory.

Misener et al. dedicated a suite of papers to the problem of pooling. Misener et al. (2010) addressed the problem of extended pooling with emission constraints. They employed and improved the technique of Furman and Androulakis (2008) to write the disjunctive constraints as a series of mixed constraints (i.e., binary/continuous), covering all EPA-CM scenarios with the IF List. Nevertheless, among the illustrative examples given, only the first one actually imposes useful reductions on the blended gasoline. This "utility" is deduced with regard to the reference baseline emissions (see Table 4, page 1449 of this reference), which has been copied in Table 1:

As shown in Table 1, only the C1 case is nearly correct

Table 1 Emissions from the paper of Furman and Androulakis

	VOC	NO_x	TOX
*x*b	1466.30	1340	86.34
C1	1200	1300	90
C2	1700	1400	95
C3	1600	1300	95

(except for TOX), whereas the C2 and C3 cases violate the threshold emissions. Computation times vary from a few seconds to 5274 s. Calculating the time ascribed to the integration of EPA-CM is difficult. For all these examples, the branch-and-bound procedure visits 44 nodes, for a total time of 6249 s, thereby amounting to 142 s/node on average. In other words, the CPLEX + MINOS solver required on average 142 s to find the min/max bounds for any given node.

To conclude this bibliographical review, these references can be divided into two major categories, namely, i) economic aspects and ii) mathematical/modeling/implementation aspects. From the first category, the optimization with constraints on emissions exerts significant economic impact on the cost of gasoline. The second category underscores the difficulty of integrating the EPA-CM constraints into the refining operations or combining them with gasoline model design. Even when such an integration step is performed, the calculation time of MINLP models remains impracticable for use on the online control of emissions and blending.

Notably, these contributions do not compare their results because the approaches are probably different and incomparable.

3 EPA-CM and useful notations and equations

A target gasoline x (more specifically, $x \in R^p$ is a vector of p gasoline properties) is blended from a set of components characterized by their properties $B_1,...,B_n \in R^p$. The following subset of these properties is used to calculate emissions in EPA-CM:

OXY = Oxygen content of (any) target fuel, expressed as weight percent;

SUL = Sulfur content of (any) target fuel, expressed in parts per million by weight;

RVP = Reid vapor pressure of the target fuel, expressed in pounds per square inch;

E200 = 200°F distillation fraction of the target fuel, expressed as volume percent;

E300 = 300°F distillation fraction of the target fuel, expressed as volume percent;

ARO = Aromatics content of the target fuel, expressed as volume percent;

BEN = Benzene content of the target fuel, expressed as volume percent;

OLE = Olefin content of the target fuel, expressed as volume percent;

MTB = Methyl tertiary butyl ether content of the target fuel, expressed as weight percent;

ETB = Ethyl tertiary butyl ether content of the target fuel, expressed as weight percent;

TAM = Tertiary amyl methyl ether content of the target fuel, expressed as weight percent;

ETH = Ethanol content of the target fuel, expressed as weight percent.

Other properties such as RON and MON are calculated and controlled; however, the above list is limited to those contributing to the calculation of NO_x, VOC(x) and TOX(x). A recipe is a vector $u \in R^n$, such that u_i, $i = 1,...,n$ is the volume percentage of component B_i in blend x. Recipe u must verify the constraint $\Sigma_i u_i = 1$. The properties of target blend x are a function of the component matrix B $=[B_1,...,B_n]$ and recipe u, with $x = x(B,u)$. Over a short time horizon (e.g., a few minutes, as is the case herein), B can be considered as a constant; consequently, the only control variable is recipe u. The hydraulic and operational constraints actually yield the upper and lower bounds on the recipes:

$$\underline{u} \leqslant u \leqslant \overline{u},\ \Sigma_i u_i = 1. \tag{1}$$

The regulatory constraints on the target gasoline are given as a set of upper and lower bounds on its properties: $\underline{c} \leqslant x = x(u) \leqslant \overline{c}$. Moreover, $x(u)$ should satisfy the EPA regulatory constraints $\underline{x} \leqslant x = x(u) \leqslant \overline{x}$, (see Appendix). The intersection of these constraints yields

$$\max(\underline{c},\ \underline{x}) \leqslant x = x(u) \leqslant \min(\overline{c},\ \overline{x}). \tag{2}$$

In our control process, when gasoline $x(u)$ violates the constraints in Eq. (2) or is not optimal, recipe u is modified to optimize some general form of an objective function: $F(u) = \alpha \cdot \|u - u^0\| + \beta \cdot \|x(u) - x^0\|$. Here, u^0 is a baseline recipe calculated by the scheduling department, and x^0 is an "ideal" gasoline that minimizes the giveaway.

The emission calculation entails an additional step. (This step is given in detail for NO_x in Appendix.) All constraints in Eqs. (1) and (2) are linear, except for the one on RVP. In practice, some transformation of RVP that blends linearly is introduced. The functions used to calculate emissions in EPA-CM (i.e., $NO_x(u)$, VOC(u) and TOX(u) are continuous but non-differentiable and non-convex (see Appendix). As shown in Appendix, these emissions are feasible if they verify constraints:

$$NO_x(x(u)) \leqslant \overline{N}, VOC(x(u)) \leqslant \overline{V}, TOX(x(u)) \leqslant \overline{T}. \tag{3}$$

A recipe u now becomes feasible if it verifies Constraints (1), (2), and (3).

The non-convex and non-differentiable functions $NO_x(x(u))$, VOC($x(u)$) and TOX($x(u)$) serve to input the constraints in Eq. (3); hence, Furman and Androulakis

(2008), Misener et al. (2010), and all preceding authors attested that EPA-CM is extremely difficult to implement for refinery operations or combine with existing models in predicting the quality of blended gasoline. Furman and Androulakis (2008) and Misener et al. (2010) introduced mixed (i.e., binary/continuous) models to integrate EPA-CM into the quality prediction and optimization sequence for blended gasoline. These authors applied the technique of disjunctive constraints (Raman and Grossmann, 1994) and inserted binary variables to manage the "IF List." More specifically, they introduced mixed (i.e., binary/continuous) constraints to obtain the correct values for y (edge target) and z (delta target) of Step 1 (see Appendix). To control the $NO_x(u)$, VOC(u) and TOX(u) emissions, Furman and Androulakis (2008) optimized the following objective function with respect to the constraints in Eqs. (1) and (2):

$$F_{FA}(x) = \Sigma_i \left(\frac{x(i) - x\mathrm{b}(i)}{\overline{x}(i) - \underline{x}(i)} \right)^2. \tag{4}$$

As discussed in Section 5, this optimization tends to find a blend as near as possible to baseline blend xb (see Appendix). However, the optimization may not verify the constraints in Eq. (3).

4 Differential approximation of EPA-CM

We continue with the example of calculating $NO_x(u)$ for a given recipe u as a means of explaining our approach. The VOC and TOX calculations are relatively similar. The calculation of terms found in TOX(u) (i.e., exhausted benzene, non-exhausted benzene, acetaldehyde, formaldehyde, and butadiene) is simple because the delta target z is not utilized.

The disjunctive constraint technique used by Furman and Androulakis (2008) and Misener et al. (2010) is recalled in the next section. This technique is illustrated by providing the equivalent form of the IF List (see Appendix). Afterward, we develop a new way of writing the IF List. Finally, a differential objective function is introduced to manage the emission levels.

4.1 Disjunctive formalism[1)]

Disjunctive programming was first developed in 1974 by Balas (2010). Its MINLP model was built by Raman and Grossmann (1994). Furman and Androulakis (2008) and Misener and Floudas (2009) extensively used the MINLP formalism of disjunctive programing in reformulating EPA-CM as an optimization problem. We examine the following conditions on the IF List:

If x(ARO)$\geqslant$18 and x(ARO)$\leqslant$36.8, then y(ARO) = x(ARO), z(ARO)= 0.

If x(ARO) < 18, then y(ARO) = 18, z(ARO) = x(ARO) − 18.

If x(ARO) > 36.8, then y(ARO) = 36.8, z(ARO) = x(ARO) − 36.8.

The disjunctive constraints for these conditions are as follows (in this case, the x, y, z values correspond to the ARO property):

$$\begin{bmatrix} x < 18 \\ y = 18 \\ z = x - 18 \end{bmatrix} \vee \begin{bmatrix} x > 36.8 \\ y = 36.8 \\ z = x - 36.8 \end{bmatrix} \vee \begin{bmatrix} 18 \leqslant x \leqslant 36.8 \\ y = x \\ z = 0 \end{bmatrix}.$$

These conditions can now be written as a series of mixed linear constraints. The domain of valid properties for the EPA-CM functions is determined by the min/max bounds of x given by the constraints in Eq. (2).

Furman and Androulakis (2008) introduced four binary variables b_0, b_1, b_2, $b_3 \in \{0,1\}$ to translate the disjunctive constraints on ARO. This example will be analyzed herein. The values x, y, z pertain to the ARO property.

General constraints on ARO: Variable b_0

This variable is introduced to yield the edge target y of x(ARO) used in all emission calculations.

$$y - 18 \leqslant (\overline{x} - 18)(1 - b_0), \tag{b0.1}$$

$$18 - y \leqslant (18 - \underline{x})(1 - b_0). \tag{b0.2}$$

We obtain b_0= 1 if and only if x(ARO) < 18. When b_0= 1, the result of constraints in Eqs. (b0.1) and (b0.2) is y = 18. If b_0= 0, then these two constraints simply state that the edge target y lies in the box $[\underline{x}, \overline{x}]$. However, this finding is implied by Eq. (2) because the edge target interval is included in the interval $[\underline{x}, \overline{x}]$ for all the gasoline properties of EPA-CM.

Edge target and delta target constraints on ARO: Variable b_1; Case x(ARO) < 18

These constraints are unique to the calculation of $NO_x(u)$. Here, b_1= 1 if and only if $x \leqslant 18$. This equivalence is maintained by Constraint (b1.1). For b_1= 1, y = 18 is from Constraints (b0.1) and (b0.2). Constraints (b1.2) and (b1.3) result in $z = x$–18. To simplify the notation, we introduce the min/max values for delta target $\underline{z}$, $\overline{z}$.

$$x - 18 \leqslant (\overline{x} - 18)(1 - b_1), \tag{b1.1}$$

$$z - x + 18 \leqslant (\overline{z} - \underline{x} + 18)(1 - b_1), \tag{b1.2}$$

$$x - 18 - z \leqslant (\overline{x} - 18 - \underline{z})(1 - b_1). \tag{b1.3}$$

If b_1= 0, Constraint (b1.1) simply states that $x \leqslant \overline{x}$. This equation is a redundant statement and follows from Eq. (2). Constraints (b1.2) and (b1.3) are simplified to $z - x \leqslant (\overline{z} - \underline{x})$ and $x - z < (\overline{x} - \underline{z})$, respectively. These constraints

1) The equations on binary variables of this section will be enumerated with (b*.*) given that they do not intervene in the rest of the paper.

are a consequence of $\underline{x} \leqslant x \leqslant \overline{x}$ and $\underline{z} \leqslant z \leqslant \overline{z}$ and are thus redundant.

Edge target and delta target constraints on ARO: Variable b_2; Case $18 \leqslant x \leqslant 36.8$

Here, $b_2 = 1$ if and only if $18 \leqslant x \leqslant 36.8$.

$$18 - x \leqslant (18 - \underline{x})(1 - b_2), \tag{b2.1}$$

$$x - 36.8 \leqslant (\overline{x} - 46)(1 - b_2), \tag{b2.2}$$

$$y - x \leqslant (\overline{x} - \underline{x})(1 - b_2), \tag{b2.3}$$

$$x - y \leqslant (\overline{x} - \underline{x})(1 - b_2), \tag{b2.4}$$

$$z \leqslant (\overline{z})(1 - b_2), \tag{b2.5}$$

$$-z \leqslant (-\underline{z})(1 - b_2). \tag{b2.6}$$

If $b_2 = 1$ or equivalently if $18 \leqslant x \leqslant 36.8$, then Constraints (b2.1) and (b2.2) imply that $18 \leqslant x \leqslant 36.8$. Constraints (b2.3) and (b2.4) yield $y = x$. Moreover, Constraint (b2.5) with (b2.6) yields $z = 0$.

Edge target and delta target constraints on ARO: Variable b_3; Case $36.8 < x$

Here, $b_3 = 1$ if and only if $36.8 \leqslant x$. This equivalence is a consequence of Constraints (b3.1) and (b3.2).

$$36.8 - x \leqslant (36.8 - \underline{x})(1 - b_3), \tag{b3.1}$$

$$y - 36.8 \leqslant (\overline{x} - 36.8)(1 - b_3), \tag{b3.2}$$

$$36.8 - y \leqslant (36.8 - \underline{x})(1 - b_3), \tag{b3.3}$$

$$z - x + 36.8 \leqslant (\overline{z} - \underline{x} + 36.8)(1 - b_3), \tag{b3.4}$$

$$x - 36.8 - z \leqslant (\overline{x} - 36.8 - \underline{z})(1 - b_3). \tag{b3.5}$$

If $b_3 = 1$, then Constraint (b3.3) yields $y = 36.8$, and Constraint (b3.4) with (b3.5) gives $z = x - 36.8$. When $b_3 = 0$, these constraints are automatically verified (to be redundant) as a consequence of Eq. (2) and $\underline{z} \leqslant z \leqslant \overline{z}$. The following constraint is then added to limit the number of possible cases to just one:

$$b_1 + b_2 + b_3 \geqslant 1. \tag{b4}$$

Consequently, to ensure the correct case for the edge target and delta target in the $NO_x(u)$ calculation, Furman and Androulakis (2008) introduced the binary variables b_0, b_1, b_2, and b_3, along with 17 mixed constraints: (b0.*), (b1.*), (b2.*), (b3.*), and (b4).

All the disjunctive constraints introduced by Furman and Androulakis (2008) require 20 binary variables, 100 continuous variables, and 200 mixed constraints. Misener et al. (2010) achieved this result using 15 binary variables, 56 continuous variables, and 100 mixed constraints. A common characteristic of these approaches is the large number of redundant constraints. As shown in the ARO example, as their values vary, only a small subset of constraints will be of any utility, whereas the others will be useless for the solver. Our approach does not make use of binary variables and has only a few redundant constraints or none at all, as discussed in the next section.

4.2 Functional form of the IF List

Our approach is based on the simple fact that the binary condition $t \leqslant r$ is equivalent to $S(t, r) = 0$, where $S(t, r)$ is the function

$$S(t,\ r) = \begin{cases} 0 & t \leqslant r \\ 1 & t > r \end{cases}.$$

One implementation of this function might use the sign function:

$$sign(t - r) = \begin{cases} -1 & t \leqslant r \\ 1 & t > r \end{cases},$$

$$S(t,\ r) = 0.5(1 + sign(t - r)). \tag{5}$$

$S(t, r)$ has been used to calculate the correct edge target and delta target values for x(ARO) given in the previous section. We simply set $m = 18$ and $M = 36.8$. Then,

$$y = (1 - S(x,\ m)) \cdot m + S(x,\ m) \cdot (1 - S(x,\ M)) \cdot x + S(x,\ M) \cdot M, \tag{6}$$

$$z = (1 - S(x,\ m)) \cdot (x - m) + S(x,\ M) \cdot (x - M). \tag{7}$$

The equivalence of these expressions with the result of the IF List is straightforward.

Case $x < m$. $S(x,m) = 0$ and $S(x,M) = 0$, which gives $y = m$ and $z = x - m$.

Case $x \geqslant m$ and $x \leqslant M$. $S(x,m) = 1$ and $S(x,M) = 0$, which gives $y = x$ and $z = 0$.

Case $x > M$. $S(x,m) = 1$ and $S(x,M) = 1$, which gives $y = M$ and $z = x - M$.

Consequently, Eqs. (2), (6), and (7) are all equivalent to the set of mixed constraints (b*.*) discussed in Section 4.1. The correct values of y and z can thus be calculated online, which simplifies Step 1 of the emission calculation (see Appendix). Steps 2 and 3 are merely composite functions with Eqs. (6) and (7). Intrinsically, no additional constraints are needed to find the correct edge target and delta target values.

We now introduce a differentiable approximation of $S(t, r)$ via the sigmoid.

$$SC(t,\ r,\ a) = 0.5(1 + \tanh(a(t - r))). \tag{8}$$

The coefficient $a = \dfrac{\mathrm{d}(SC(t,\ r,\ a))}{\mathrm{d}t}$ for $t = r$ can be

selected, such that $|SC(t,\ r,\ a)-S(t,\ r)|<\varepsilon$ for any given ε everywhere except within a small interval $t \in [r-\delta,\ r+\delta]$. We consider the level of precision when measuring any property t of gasoline x in our calculations. Selecting a, such that 2δ is less than the measurement error for any property of the blended target gasoline required for the emission calculation, is in fact straightforward. The deviation in emissions calculated using Eq. (8) compared to those calculated with Eq. (5) is controlled by the level of measurement accuracy of the properties of x.

4.3 Gradient of emission functions

We now calculate the gradient of $NO_x(u)$ versus recipe u and gasoline properties $x=x(u)$. This process is a simple calculus exercise by following the chain rule for derivative of composite functions. Nevertheless, we indicate it here for the sake of having a self-contained paper. The gradient calculation for VOC(u) and TOX(u) follows a similar manner.

$$\frac{\partial NO_x(u)}{\partial u_k} = \Sigma_{j=1}^{p} \frac{\partial NO_x(x)}{\partial x_j} \cdot \frac{\partial x_j}{\partial u_k},\ k = 1,\ \ldots,\ n.$$

The partial derivative $\frac{\partial x_j}{\partial u_k}$ for each property x_j as a function of recipe u is required. For the linear blending law properties, this partial derivative is simply the entry $B(j,k)$ of the component matrix. Edge target y and delta target z are both defined as functions of x; consequently,

$$\frac{\partial NO_x(x)}{\partial x_i} = \Sigma_{j=1}^{p} \left(\frac{\partial NO_x(y,\ z)}{\partial y_j} \cdot \frac{\partial y_j}{\partial x_i} + \frac{\partial NO_x(y,\ z)}{\partial z_j} \cdot \frac{\partial z_j}{\partial x_i} \right).$$

Steps 2 and 3 (Appendix) of the calculation of each term of $NO_x(y,\ z) = w_N \cdot N(y) \cdot FN(y,\ z) + w_H \cdot H(y) \cdot FN(y,\ z)$ can now be recovered.

$$\begin{aligned} &\frac{\partial NO_x(y,\ z)}{\partial y_i} \\ =& w_N \cdot \left(\frac{\partial N(y)}{\partial y_j} \cdot FN(y,\ z) + \frac{\partial FN(y,\ z)}{\partial y_j} \cdot N(y) \right) \\ &+ w_H \cdot \left(\frac{\partial H(y)}{\partial y_j} \cdot FH(y,\ z) + \frac{\partial FH(y,\ z)}{\partial y_j} \cdot H(y) \right). \end{aligned}$$

Given the exponential form of $N(y)$ and $H(y)$, we obtain

$$\begin{aligned} &\frac{\partial NO_x(y,\ z)}{\partial y_i} \\ =& w_N \cdot N(y) \cdot \left(\frac{\partial n_1(y)}{\partial y_j} \cdot FN(y,\ z) + \frac{\partial FN(y,\ z)}{\partial y_j} \right) \\ &+ w_H \cdot H(y) \cdot \left(\frac{\partial n_2(y)}{\partial y_j} \cdot FH(y,\ z) + \frac{\partial FH(y,\ z)}{\partial y_j} \right), \end{aligned}$$

$$\begin{aligned} \frac{\partial NO_x(y,\ z)}{\partial z_j} &= w_N \cdot \frac{\partial FN(y,\ z)}{\partial z_j} \cdot N(y) \\ &+ w_H \cdot \frac{\partial FH(y,\ z)}{\partial z_j} \cdot H(y). \end{aligned}$$

For the next step, the partial derivatives on y and z are calculated. To avoid cumbersome notations, the following sections denote the coordinates of vectors y and z by y and z indexed by gasoline properties. For example, $y := y(OXY)$ indicates that y is defined as the $y(OXY)$ value of edge target y.

$$y := y(OXY),\ z := z(OXY).$$

The following numbers are found in (EPA, 2016):

$$\frac{\partial n_1(y)}{\partial y} = 0.0018571,$$

$$\frac{\partial n_2(y)}{\partial y} = -0.00913,$$

$$\begin{aligned} \frac{\partial FN(y,\ z)}{\partial y} &= \frac{\partial FH(y,\ z)}{\partial y} = \frac{\partial FN(y,\ z)}{\partial z} \\ &= \frac{\partial FH(y,\ z)}{\partial z} = 0, \end{aligned}$$

$$y := y(SUL),\ z := z(SUL).$$

We find

$$\frac{\partial n_1(y)}{\partial y} = 0.00069205 - 13.2526 \cdot 10^{-7} \cdot y,$$

$$\frac{\partial n_2(y)}{\partial y} = 0.000252,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = z \cdot (-0.00000133),$$

$$\frac{\partial FH(y,\ z)}{\partial y} = 0,$$

$$\frac{\partial FN(y,\ z)}{\partial z} = (-0.00000133 \cdot y + 0.000692),$$

$$\frac{\partial FH(y,\ z)}{\partial z} = 0.000252,$$

$$y := y(RVP),\ z := z(RVP).$$

We also find

$$\frac{\partial n_1(y)}{\partial y} = 0.0090744,$$

$$\frac{\partial n_2(y)}{\partial y} = -0.013973,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = \frac{\partial FH(y,\ z)}{\partial y} = \frac{\partial FN(y,\ z)}{\partial z} = \frac{\partial FH(y,\ z)}{\partial z} = 0,$$

$$y := y(E200),\ z := z(E200),$$

which yields

$$\frac{\partial n_1(y)}{\partial y} = 0.00093065,$$

$$\frac{\partial n_2(y)}{\partial y} = 0.000931,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = \frac{\partial FH(y,\ z)}{\partial y} = \frac{\partial FN(y,\ z)}{\partial z} = \frac{\partial FH(y,\ z)}{\partial z} = 0,$$

$$y := y(E300),\ z := z(E300),$$

which yields

$$\frac{\partial n_1(y)}{\partial y} = 0.00084596,$$

$$\frac{\partial n_2(y)}{\partial y} = -0.004009,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = \frac{\partial FH(y,\ z)}{\partial y} = \frac{\partial FN(y,\ z)}{\partial z} = \frac{\partial FH(y,\ z)}{\partial z} = 0,$$

$$y := y(ARO),\ z := z(ARO).$$

We now have

$$\frac{\partial n_1(y)}{\partial y} = 0.0083632 - 0.0002381 \cdot y,$$

$$\frac{\partial n_2(y)}{\partial y} = 0.007097005 - 2 \cdot 0.000079951 \cdot y,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = z \cdot (-0.000238),$$

$$\frac{\partial FH(y,\ z)}{\partial y} = z \cdot (-0.0001599),$$

$$\frac{\partial FN(y,\ z)}{\partial z} = (-0.000238 \cdot y + 0.0083632),$$

$$\frac{\partial FH(y,\ z)}{\partial z} = -0.0001599 \cdot y + 0.007097,$$

$$y := y(OLE),\ z := z(OLE),$$

which leaves us with

$$\frac{\partial n_1(y)}{\partial y} = -0.0027735 + 0.00073304 \cdot y,$$

$$\frac{\partial n_2(y)}{\partial y} = -0.0027603 + 0.000732 \cdot y,$$

$$\frac{\partial FN(y,\ z)}{\partial y} = z \cdot (0.000733),$$

$$\frac{\partial FH(y,\ z)}{\partial y} = z \cdot (0.000732),$$

$$\frac{\partial FN(y,\ z)}{\partial z} = 0.000733 \cdot y - 0.002774,$$

$$\frac{\partial FH(y,\ z)}{\partial z} = 0.000730 \cdot y - 0.00276.$$

For any property p different from SUL, OLE, and ARO, we obtain $y(p) = x(p)$ and $z(p) = 0$. For these properties, $\frac{\partial y(p)}{\partial x(p)} = 1$ and $\frac{\partial z(p)}{\partial x(p)} = 0$. For the SUL, OLE, and ARO properties, the calculation of these derivatives requires computing the derivative $SC'(t, r, a)$ of $SC(t, r, a)$. For each of these properties, we assume

$p = \text{SUL}$, $m = 10$, $M = 450$, $a = \text{prec(SUL)}$ (with the SUL precision given by the user);

$p = \text{OLE}$, $m = 3.77$, $M = 19$, $a = \text{prec(OLE)}$ (with the OLE precision given by the user);

$p = \text{ARO}$, $m = 18$, $M = 36.8$, $a = \text{prec(ARO)}$ (with the ARO precision given by the user);and then calculate

$$\begin{aligned}\frac{\partial y_p}{\partial x_p} = & -m \cdot SC'(x_p,\ m,\ a) + M \cdot SC'(x_p,\ M,\ a) \\ & + SC'(x_p,\ m,\ a) \cdot (1 - SC(x_p,\ M)) \cdot x_p \\ & - SC(x_p,\ m) \cdot SC'(x_p,\ M,\ a) \cdot x_p + S(x_p,\ m) \\ & \cdot (1 - SC(x_p,\ M)),\end{aligned}$$

$$\begin{aligned}\frac{\partial z_p}{\partial x_p} = & -(x_p - m) \cdot SC'(x_p,\ m,\ a) + (1 - SC(x_p,\ m)) \\ & + SC(x_2,\ M) + (x_p - M) \cdot SC'(x_p,\ M,\ a).\end{aligned}$$

This expression represents the last link in the chain rule for derivatives. These analytical gradients are extremely easy to calculate as functions of recipe u and properties x. This crucial point serves to control emission variations online the same way as the other traditional gasoline properties.

4.4 Objective function to control emissions

We have now defined constraints on recipe u and properties x. We have also defined constraints on emissions in Eq. (3). Given a vector of positive penalties $\pi = [\pi_N, \pi_V, \pi_T]$, the previous constraints in Eq. (3) are satisfied if and only if a recipe u exists, such that

$$F(u, \pi) = \Sigma_{R \in \{N, V, T\}} \pi_R \cdot SG(R, \overline{R}) \cdot (R(u) - \overline{R}) = 0.$$

The idea behind this condition is to minimize the function $F(u, \pi)$ with respect to the linear constraints on recipes in Eq. (1) and properties in Eq. (2). In this manner, we are not changing the online control framework for traditional gasoline properties but merely add a new objective function.

Blend instance

For a given component matrix B, a blend instance is defined by the constraints in Eqs. (1) and (2) and the penalty vector π.

Feasible blend instance

A blend instance is feasible if and only if the optimization problem

$$\begin{cases} \min F(u, \pi) = \Sigma_{R \in \{N, V, T\}} \pi_R \cdot SG(R, \overline{R}) \cdot (R(u) - \overline{R}) \\ \underline{u} \leqslant u \leqslant \overline{u} \\ \max(\underline{x}, \underline{c}) \leqslant x(u) \leqslant \min(\overline{x}, \overline{c}) \end{cases} \quad (9)$$

has a feasible (optimal) solution u_0, such that $F(u_0, \pi) = 0$.

In the following, we denote the cost function as $F_1(u) = c^{\mathrm{T}}u$ and the giveaway function as $F_2(u) = \|x(u) - x^0\|$. The optimization problem in Eq. (9) is central to our online emission control system. The resolution of this problem has been implemented in MATLAB (see Section 5). The general framework for optimization and online emission control is given below. In implementing the instances of this framework, Steps 2 and 3 can be combined.

Online optimization and control process:

Step 1: Define a blend instance

Choose a component matrix B. Define the constraints in Eqs. (1) and (2). Choose the penalty vector $\pi > 0$.

Step 2: Feasibility

Solve the optimization problem in Eq. (9). Let u_0 be its optimal solution.

If $F(u_0, \pi) > 0$ // Emission threshold violated

Change the blend instance; GO TO Step 1;

Otherwise

GO TO Step 3 to identify an optimal recipe.

Step 3: Optimization

3.1. Compute the values of $F_1(u_0)$ and $F_2(u_0)$.

3.2. Find the optimal solution (S^*, T^*, u^*) to the following problem:

$$\begin{cases} \min\ G(S, T, u) = a \cdot S + b \cdot T + w \cdot F(u, \pi) \\ F_1(u) \leqslant S \cdot F_1(u_0) \\ F_2(u) \leqslant T \cdot F_2(u_0) \\ 0 \leqslant S, T \leqslant 1 \\ \underline{u} \leqslant u \leqslant \overline{u} \\ \max(\underline{x}, \underline{c}) \leqslant x \leqslant \min(\overline{x}, \overline{c}) \end{cases} . \quad (10)$$

3.3 **If** $F(u^*, \pi) = 0$

STOP: The optimal recipe has been found;

Otherwise

Find a recipe $u_1 \in [u_0, u^*]$, such that $F(u_1, \pi) = 0$; $u_0 \leftarrow u_1$; GO TO Step 3.1.

In the following section, we present the results from an actual refinery.

5 Results

The quick calculation of the gradient of $F(u, \pi)$ is extremely important for all nonlinear optimization solvers. The optimization problem in Eq. (9) can actually feature up to 80 linear constraints. A standard branch-and-bound global optimizer is used to solve Eq. (9) given that the objective function is non-convex. This branching occurs within the domain of recipe u in verifying $0 \leqslant u \leqslant 1$ and $\Sigma u_i = 1$. In practice, the dimension of u (i.e., the number of components) does not exceed 20 in typical applications. These recipe domain properties make allows the inclusion of a branch-and-bound procedure stop criterion that depends on the diameter of cells created by the branching procedure. Hence, reducing the diameter of the box containing the optimal recipe to a value lying below a given threshold (10^{-8} for our application) is rather easy. All computations are performed on a machine with a Duo CPU, 2 GHz Intel processor, and 3 GB memory. All our optimization instances are run in less than 1 s. These results serve to implement the online emission control strategy into the TOTAL ANAMEL software (ANAMEL is the software used in several TOTAL Group refineries to control blended product quality by means of real-time blend recipe optimization). Preliminary tests with these new emission control features embedded into ANAMEL and approximate values for distillation figures are conducted at the Leuna refinery (Germany) in 2015.

For comparison of results, the optimal values and the emission performance of the following three objective functions are calculated:

(i) Cost function $F_1(u) = c^{\mathrm{T}}u$;

(ii) Giveaway function as the distance $F_2(u) = \|x(u) - x^0\|$ to an ideal gasoline x^0;

(iii) Normalized distance to the baseline gasoline F_{FA} of Furman and Androulakis (2008) in Eq. (4).

We now provide a typical example with data from an American refinery. The EPA-CM parameters are as follows in Table 2.

Table 2 EPA parameters

Season	Region	Gasoline type	NO_x penalty	VOC penalty	TOX penalty
Summer	1	RFG	2	3	4

We then assign arbitrary penalties to illustrate our approach. Table 3 recalls the baseline RFG properties for the given season and region:

The target blend is 57,000 m^3 of RFG. The properties of six components (B_1, ..., B_6) are listed in Table 4.

The two bottom rows list the lower and upper bounds on recipes. The two columns on the right describe the target zone. Table 5 presents the following optimization results for all objective functions implemented: i) F_1: optimal value of the economic function (i.e., blend cost); ii) F_2: optimal value of the objective (ii); iii) F_3: optimal value of the F_{FA} function in Eq. (4) optimized by Furman and Androulakis (2008); and iv) F_4: optimal value of the objective function in Eq. (9) (emission control). The optimal recipes of each optimization are given in columns u_1, u_2, u_3 and u_4. The corresponding RFG gasolines appear in columns x_1, x_2, x_3 and x_4. The emissions of these gasolines are shown in columns E_1, E_2, E_3, and E_4. The cost or giveaway optimizations lead to blends with high emission levels. The optimization of F_3 improves emission levels but still violates the VOC emission threshold. For this blend instance, only the optimization of F_4 yields the gasoline x_4 with emissions E_4, which lie far below the emission thresholds. The computation time for this instance is 0.6 s.

The values in bold indicate the results that satisfy the emissions thresholds.

The comparison of execution times is performed with respect to the approach of Furman and Androulakis (2008). We do not have data to implement the pooling problem of Misener and Floudas (2009).

6 Conclusions

A differentiable approximation of EPA-CM is developed in this study. This approximation is based on a computation chain that offers ease of implementation on online emission control. Only a new objective function should be added to our general control framework for traditional gasoline properties. Our model has been implemented as a global optimization problem of a differentiable objective function with respect to linear constraints. This approach is

Table 3 Values for baseline gasoline

OXY	SUL	RVP	E200	E300	ARO	OLE	BEN	MTB	ETB	ETH	TAM
0.00	339.00	8.70	41.00	83.00	32.00	9.20	1.53	0.00	0.00	0.00	0.00

Table 4 Component properties and bounds for a real blend

Prop	B_1	B_2	B_3	B_4	B_5	B_6		
MON	52.3	80.4	79.4	109.8	86.7	92.8	82.1	90
ROAD	51.9	84.4	83.4	114.9	88.15	97.9	87.1	100
RVP	2.99	13.85	15.46	0.64	153.19	1	12.21	15.37
V/L	182.07	134.4	136.4	216.24	−24.7	234.3	120	150
T90	289	348.4	365.4	285	34	372	100	371
EP	317	444.4	420.4	285	35	466	100	427
DI	1253.5	1027.4	1128.4	1557	−56	1887	100	1230
E200	16.6	64.4	48.4	0	100	0	0	100
E300	94.7	80.4	84.6	0.9	100	0	0	100
OLE	3.1	21.4	30.4	0.1	0	1.7	20	30
ARO	8.9	22.4	23.4	99.9	0	91.6	30	40
BEN	0.7	0.4	0.44	0	0	0.4	0	3.7
SUL	0	49.4	32.4	0	8	4	0	80
OXY	0	0	0	0	0	0	0	0.1
$\underline{u}$ [%vol]	0	10	20	0	0	2		
$\overline{u}$ [%vol]	21	45	60	100	6	20		

Table 5 Optimization results for four objective functions

	Optimal values			
	F_1	F_2	F_3	F_4
	0.3922	2.3609	1.1487	0.1766
Bases	Optimal recipes			
	u_1	u_2	u_3	u_4
B_1	0.00%	0.00%	3.32%	2.80%
B_2	0.00%	0.00%	54.65%	65.05%
B_3	87.25%	79.04%	26.94%	19.67%
B_4	11.57%	6.61%	13.02%	12.48%
B_5	1.18%	1.83%	2.08%	0.00%
B_6	0.00%	12.52%	0.00%	0.00%
Baseline	Emissions			
	E_1	E_2	E_3	E_4
NO_x: 1340.0	1392.62	1362.22	1314.79	1313.76
VOC: 1466.3	1479.45	1610.30	1480.56	1218.20
TOX: 86.34	70.06	73.00	65.69	65.01
Execution time/s	0.04	0.06	145	0.62
Properties	Optimal gasolines			
	x_1	x_2	x_3	x_4
MON	82.10	83.00	83.22	83.16
ROAD	87.10	87.38	87.10	87.10
RVP	15.37	15.19	15.10	12.21
V/L	142.56	149.15	141.79	146.34
T90	352.19	354.85	336.22	342.17
EP	400.19	410.11	404.45	416.22
DI	1164.01	1230.00	1108.54	1119.69
E200	43.41	40.09	50.86	51.88
E300	75.10	68.76	72.07	71.70
OLEF	30.00	26.54	24.25	20.00
AROM	31.97	36.56	31.84	31.89
BENZ	0.38	0.40	0.36	0.37
SULF	28.36	26.26	35.89	38.51
OXY	0.00	0.00	0.00	0.00

relatively straightforward in refinery operations. The execution time of our model is several orders of magnitude faster than the MINLP models developed by Misener and Floudas (2009) and Furman and Androulakis (2008). Our typical actual example has demonstrated the simplicity of our model for emission computations. This model is now patented (Janaqi et al., 2015) and implemented using our ANAMEL online controller.

Acknowledgements This work has been completed with the financial support from TOTAL Refining and Chemicals. Special thanks are addressed to Edith Di Crescenzo, Antoine Milochau, and Jean-Philippe Grimaldi.

Appendix. Description of NO_x calculation in EPA-CM

Depending on the season (i.e., summer and winter), region (i.e., 1 and 2), and gasoline type (i.e., RFG or conventional CFG), the baseline gasoline is selected (denoted by *xb* in this paper). The min/max bounds of target gasoline *x* are shown in Table A1 (EPA, 2015).

Table A1 Baseline gasoline and min/max bounds of the target gasoline in EPA-CM

	*x*b	*x*b	RFG		CFG	
	Summer	Winter	$\underline{x}$	$\overline{x}$	$\underline{x}$	$\overline{x}$
OXY	0.0	0.0	0.0	4.0	0.0	4.0
SUL	339	338	0.0	500.0	0.0	1000.0
RVP	8.7	11.5	6.4	10.0	6.4	11.0
E200	41.0	50.0	30.0	70.0	30.0	70.0
E300	83.0	83.0	70.0	100.0	70.0	100.0
ARO	32.0	26.4	0.0	50.0	0.0	55.0
OLE	9.2	11.9	0.0	25.0	0.0	30.0
BEN	1.53	1.64	0.0	2.0	0.0	4.9

Moreover, we denote *x*(OXY), *x*(SUL) as the values of properties OXY and SUL, etc. of gasoline *x*. The value of OXY is thus OXY = ETB + MTB + ETH + TAM. If target gasoline *x* violates the above min/max bounds, then *x* is declared to be non-conforming and its emissions will not be calculated. The EPA-CM model uses the normal or high emitter coefficients, as listed in Table A2 (EPA, 2015).

Table A2 Coefficients used in the EPA-CM model

	VOC + TOX	NO_x
Normal emitters (w_N)	0.444	0.738
High emitters (w_H)	0.556	0.262

Only the data from Phase 2, which have been in effect since 2000, are presented in Table A3 (EPA, 2015), providing reference emissions with respect to season and region.

For example, during summer in Region 2, the emissions of a target gasoline $x(u)$ must satisfy the following inequalities:

Table A3 Reference emissions

	Summer		Winter	
	Region 1	Region 2	Region 1	Region 2
NO_x	1340.0	1340.0	1540.0	1540.0
VOC	1466.3	1399.1	1341.0	1341.0
TOX	86.34	85.61	120.55	120.55

$$\mathrm{NO}_x(x(u)) \leqslant 134.0; \mathrm{VOC}(x(u)) \leqslant 1399.1;$$

$$\mathrm{TCX}(x(u)) \leqslant 85.61.$$

$\mathrm{NO}_x(u)$ is a function of the properties OXY, SUL, RVP, E200, E300, ARO, and OLE of $x(u)$. To calculate $\mathrm{NO}_x(u)$, two vectors are generated, namely, y (edge target) and z (delta target; EPA, 2015), the values of which depend on the season and the value intervals of E300, SUL, OLE, and ARO (see Table A4).

Table A4 Values whose emission slope is not continuous

Property of x	Min	Max
SUL	10.0	450.0
OLE	3.77	19.0
ARO	18.0	36.8

Several steps are now needed to calculate $\mathrm{NO}_x(u)$.

Step 1: Transformation of target gasoline $x = x(u)$ to y (edge target) and z (delta target). Initialize $y = x$ and $z = 0$.

IF List
If Season = Winter, **then** y(RVP)= 8.7.
If x(E300) > 95 **then** y(E300)= 95.
If x(SUL) < 10 **then** y(SUL)= 10, z(SUL)= x(SUL)–10.
If x(SUL) > 450, **then** y(SUL)= 450, z(SUL)= x(SUL)–450.
If x(OLE) < 3.77 **then** y(OLE)= 3.77, z(OLE)= x(OLE)–3.77.
If x(OLE) > 19 **then** y(OLE)= 19, z(OLE)= x(OLE)–19.
If x(ARO) < 10 **then** y(ARO)= 10, z(ARO)=–8.
If x(ARO) < 18 **then** y(ARO)= 18, z(ARO)= x(ARO)–18.
If x(ARO) > 36.8 **then** y(ARO)= 36.8, z(ARO)= x(ARO)–36.8.

This list of if–then conditions, which are labeled as IF List in this paper, produces disruptions in the $\mathrm{NO}_x(u)$ slopes. This function is therefore differentiable everywhere, except at the disruption points given by the previous table and the IF List.

Step 2: Calculation of intermediate values

$$\begin{aligned} n_1(y) &= (0.0018571 \cdot y(\mathrm{OXY})) + (0.0006921 \cdot y(\mathrm{SUL})) \\ &+ (0.0090744 \cdot y(\mathrm{RVP})) + (0.0009310 \cdot y(\mathrm{E200})) \\ &+ (0.0008460 \cdot y(\mathrm{E300})) + (0.0083632 \cdot y(\mathrm{ARO})) \\ &+ (-0.002774 \cdot y(\mathrm{OLE})) + (-6.63 \cdot 10^{-7} \cdot y(\mathrm{SUL})^2) \\ &+ (-0.000119 \cdot y(\mathrm{ARO})^2) + (-0.0003665 \cdot y(\mathrm{OLE})^2), \end{aligned}$$

$$\begin{aligned} n_2(y) &= (-0.00913 \cdot y(\mathrm{OXY})) + (0.000252 \cdot y(\mathrm{SUL})) \\ &+ (-0.01397 \cdot y(\mathrm{RVP})) + (0.0009311 \cdot y(\mathrm{E200})) \\ &+ (-0.00401 \cdot y(\mathrm{E300})) + (0.007097 \cdot y(\mathrm{ARO})) \\ &+ (-0.00276 \cdot y(\mathrm{OLE})) + (0.0003665 \cdot y(\mathrm{OLE})^2) \\ &+ (-7.995 \cdot 10^{-5} \cdot y(\mathrm{ARO})^2). \end{aligned}$$

Moreover, one calculates $n_1(x\mathrm{b})$ and $n_2(x\mathrm{b})$ for the baseline xb . These functions are clearly nonlinear and non-convex. The non-convexity (and non-concavity) is due to the fact that the coefficients of quadratic terms of n_1 and n_2 are not all positive (or all negative). Then, we calculate

$$N(y) = \mathrm{e}^{n_1(y) - n_1(xb)} \text{ and } H(y) = \mathrm{e}^{n_2(y) - n_2(xb)},$$

$$\begin{aligned} FN(y, z) &= 1 + z(\mathrm{SUL}) \cdot (-0.00000133 \cdot y(\mathrm{SUL}) \\ &+ 0.000692) + z(\mathrm{ARO}) \cdot (-0.000238 \cdot y(\mathrm{ARO}) \\ &+ 0.0083632) + z(\mathrm{OLE}) \\ &\cdot (0.000733 \cdot y(\mathrm{OLE}) - 0.002774), \end{aligned}$$

$$\begin{aligned} FH(y, z) &= 1 + 0.000252 \cdot z(\mathrm{SUL}) \\ &+ z(\mathrm{ARO}) \cdot (-0.0001599 \cdot y(\mathrm{ARO}) + 0.007097) \\ &+ z(\mathrm{OLE}) \cdot (0.000732 \cdot y(\mathrm{OLE}) - 0.00276). \end{aligned}$$

Step 3: Select the coefficients w_N and w_H from Table 2 and calculate $\mathrm{NO}_x(u)$ as follows:

$$\begin{aligned} \mathrm{NO}_x(u) &= \mathrm{NO}_x(x(u)) = \mathrm{NO}_x(y, z) \\ &= w_N \cdot N(y) FN(y, z) + w_H \cdot H(y) FN(y, z). \end{aligned}$$

References

Balas E (2010). Disjunctive programming. In: 50 Years of Integer Programming 1958–2008: From the Early Years to the State-of-the-Art, 283–340

Cason W W (1997). The illusory flexibility of the complex model: a graphical analysis of the implications of the complex model for refinery blending flexibility. In: NPRA Annual Meeting. San Antonio, Texas: NPRA Annual Meeting, 16–18

Chèbre M, Creff Y, Petit N (2010). Feedback control and optimization for the production of commercial fuels by blending. Journal of Process Control, 20(4): 441–451

EPA (2015). Complex emission model. http://www.laws9.com/cfr/text/40/80/45

EPA (2016). Template spreadsheet for calculating emissions from combustion plants. http://www.epa.ie/pubs/reports/air/airemissions/epatemplatespreadsheetforcalculatingemissionsfromcombustion-plantsxls.html

Furman K C, Androulakis I P (2008). A novel MINLP-based representation of the original complex model for predicting gasoline emissions. Computers & Chemical Engineering, 32(12): 2857–2876

Hirshfield D S, Kolb J A (1994). Minimize the cost of producing reformulated gasoline by integrating EPA's Complex Model into refinery linear programming (LP) models. NPRA Annual Meeting, 64–67

Hirshfield D S, Kolb J A (1997). The economics of gasoline reformulation: refining economics, emissions standards, and the Complex Model. In: NPRA Annual Meeting, San Antonio, USA

Jackson J, Vittachi K (1995). CITGO Corpus Christi refinery gasoline blender upgrade. In: NPRA Annual Meeting, San Francisco, USA

Janaqi S, Chebre M, Pitollat G (2015). Method and device for monitoring induced properties of a mixture of components, in particular emission properties, 69. http://www.sumobrain.com/patents/wipo/Method-device-monitoring-induced-properties/WO2015097305A1.html

Korotney D J (1995). Reformulated gasoline effects on exhaust emissions: Phase III; investigation on the effects of sulfur, olefins, volatility, and aromatics and the interactions between olefins and volatility or sulfur. SAE Special Publications, 179–186

Misener R, Floudas C A (2009). Advances for the pooling problem: modeling, global optimization, and computational studies Survey. Applied and Computational Mathematics, 8(1): 3–22

Misener R, Gounaris C E, Floudas C A (2010). Mathematical modeling and global optimization of large-scale extended pooling problems with the (EPA) complex emissions constraints. Computers & Chemical Engineering, 34(9): 1432–1456

Naman B T (1999). Linear models help refiners develop RFG [(reformulated gasoline)] recipes. Oil & Gas Journal, 97(8): 64–66

Nocca J L, Forestiere A, Cosyns J (1994). Diversify process strategies for reformulated gasoline. Fuel Reformulation, 4(5): 18–22

Raman R, Grossmann I E (1994). Modelling and computational techniques for logic based integer programming. Computers & Chemical Engineering, 18(7): 563–578

Treiber S (1998). RFG [(reformulated gasoline)]: the challenge to conventional blending technology. In: Gulf Publishing, eds. Hydrocarbon Processing 2nd International Conference on Process Optimization. Houston: Hydrocarbon Processing 2nd International Conference on Process Optimization, 101–103

Trierwiler L D (1995). Representing the simple and complex models in linear programming with respect to reformulated gasoline. In: American Energy Week '95 “Pipelines, Terminals & Storage, and Reformulated Fuels” Conference, Houston, USA