Global Optimization of Large-scale MIQCQPs via Cluster Decomposition: Application to Short–Term Planning of an Integrated Refinery–Petrochemical Complex

Ariel Uribe-Rodriguez\textsuperscript{a,b}, Pedro M. Castro\textsuperscript{c*}, Gonzalo Guillén-Gozálbez\textsuperscript{d} and Benoît Chachuat\textsuperscript{b}

\textsuperscript{a} Colombian Petroleum Institute, ECOPETROL, 681011 Piedecuesta, Colombia.

\textsuperscript{b} Centre for Process Systems Engineering, Department of Chemical Engineering, Imperial College London, South Kensington Campus, SW7 2AZ, London, United Kingdom.

\textsuperscript{c} Centro de Matemática Aplicações Fundamentais e Investigação Operacional, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal.

\textsuperscript{d} Institute for Chemical and Bioengineering, Department of Chemistry and Applied Biosciences, ETH Zürich, Vladimir-Prelog-Weg 1, 8093 Zürich, Switzerland

Abstract

Integrated refinery-petrochemical facilities are complex systems that require advanced decision-support tools for optimal short-term planning of their operations. The problem can be formulated as a mixed-integer quadratically constrained quadratic program (MIQCQP), in which discrete decisions select operating modes for the process units, while the entire process network is represented by input-output relationships based on bilinear expressions describing yields and stream properties, pooling equations, fuels blending indices and cost indicators. We develop a novel decomposition-based

\*Corresponding authors: pmcastro@fc.ul.pt, ariel.uribe@ecopetrol.com.co, gonzalo.guillen.gosalbez@chem.ethz.ch, b.chachuat@imperial.ac.uk
algorithm for deterministic global optimization that divides the network into small clusters according to their functionality. Inside each cluster, we derive a mixed-integer linear programming relaxation based on piecewise McCormick envelopes, dynamically partitioning the variables that belong to the cluster and reducing their domains through optimality-based bound tightening. Results for an industrial case study in Colombia show profit improvements above 10% and significantly reduced optimality gaps compared with the state-of-the-art global optimization solvers BARON and ANTIGONE.

**Keywords:** Refinery; Petrochemicals; Global Optimization; Process Operations; Planning.

1. Introduction

Integrated refining–petrochemical complexes (IRPCs) monetize residual refining streams, adding value to the supply chains of both industries. IRPCs benefit from synergies that lead to reduction of capital investments and operating costs (Ketabchi et al., 2019). This level of process integration and coordination facilitates the production of a wide range of marketable petroleum-based products such as fuels, petrochemicals, waxes, lubes, and industrial diluents. These commodities are required to supply almost all the needs of our modern societies.

Without integration, intermediate refinery streams produced by the Fluid Catalytic Cracking (FCC), Delayed Coking (DC) and Catalytic Reforming (CR) units, such as off-gas, ethane, ethylene, propylene, butylene, naphtha and light cycle oil are burnt, sold as a low-valued product (liquefied petroleum gas, LPG), or blended as gasoline or diesel. Through process integration, ethane from FCC and naphtha produced by DC and FCC units can be used as feedstock for Steam Crackers to produce ethylene. A separation train after CR can also be utilized for recovering benzene-toluene-xylene (BTX) and its derivatives. In essence, the refinery provides olefins and aromatics as raw materials to the petrochemical plant. In return, the petrochemical facility improves gasoline quality adding high-octane components and supplies a fraction of the hydrogen needed by hydrotreating processes (Li et
Moreover, a site-wide heat integration can reduce utility requirements for steam, electricity and water, thereby mitigating environmental impacts (De Oliveira Magalhães, 2009).

Despite these advantages, the design and planning of an IRPC is challenging due to the complexity of the process network and intermediate streams connectivity, the large variety of fuels and petrochemical products, the fulfillment of fuels specifications, and the market fluctuations. Both process synergies and competition for raw materials are important (Al-Qahtani and Elkamel, 2010, 2009; Al-Qahtani, 2009; Leiras et al., 2010). Critical factors that might affect an IRPC’s profitability are environmental regulations on fuels quality, impact of geopolitics on crude oil supply and price, volume and quality of new crude oil discoveries in terms of Sulphur content and API gravity, and reduction of fuels demands resulting from a boom on petrochemical commodities (International Energy Agency, 2019; WEO/IEA, 2016). This inherent complexity and uncertainty calls for advanced tools to assist with the decision-making.

An accurate mathematical representation of refining and petrochemical processes is crucial for these computational tools to provide accurate, quantitative decisions. Large-scale formulations based on Linear Programming (LP) have been developing since the 1950s (Adams and Griffing, 1972; Manne, 1958), with specific LP applications such as gasoline blending and optimal cut-point selection at fractionation units (Charnes et al., 1952; Garvin et al., 1957) highlighting the nonlinear response of the refining processes. Distributive Recursion (DR) (Haverly, 1980, 1979, 1978) and Successive Linear Programming (SLP) (Baker and Lasdon, 1985) enhanced the LP’s performance by minimizing the error between a stream property calculated by a linear expression and its value obtained using a nonlinear equation. DR is unstable for large-scale applications involving multiple refineries and multi-period optimization, even with good initialization and tuning. SLP applies first-order Taylor expansion to linearize the nonlinear equations, with a good performance depending on selecting an appropriate reference point. Advantages of DR and SLP are favorable solution times, easy model updating, flexibility to represent process units complexity and ability to handle large-scale refinery
networks, making them an industry standard for refinery planning (ASPEN Technology Inc, 2010; Bonner & Moore, 1979; Haverly, 2015; Kutz et al., 2014).

In LP/DR/SLP formulations, CDUs and process units are modelled based on fixed yield representations. In the case of CDUs, the laboratory crude oil assay is transformed into a true boiling point (TBP) curve describing how yields and properties change as a function of temperature. The cut-points that predict yields and stream properties in a TBP curve are pre-defined. Other approaches to determine the optimal cut-points, include: swing-cuts, which are based on the TBP distribution (Zhang et al., 2001); micro-cuts (Menezes et al., 2013); empirical correlations (Wenkai et al., 2007); short-cut methods (Alattas et al., 2011); and polynomial approximations (López et al., 2012, 2013).

Likewise, the accuracy of process unit models can be improved by formulating empirical correlations (Li et al., 2005), or using surrogate models based on rigorous simulations (Guerra and Le Roux, 2011a) and polynomial approximations (Alhajri et al., 2008).

The predictions from LP/DR/SLP formulations under/overestimate yields and stream properties, with the errors propagating to the pooling equations, process models, fuels blending and cost functions. Thus, formulating a nonlinear programming (NLP) model for a subset of or the whole refinery can improve the solution accuracy. Moro et al. (1998) increased profit for a real-world application by about $6 MM/year by considering nonlinear models for the heavy diesel hydrotreating and blending. Pinto and Moro (2000) later integrated operational refinery planning and crude oil scheduling. Alattas et al. (2012) incorporated their CDU short-cut model in a mixed-integer nonlinear programming (MINLP) planning model for determining crude oil sequencing, changeovers and processing time. López et al. (2012, 2013) integrated a system of three CDUs and its associated heat exchanger network (HEN) to account for the utilities cost in the profit calculation. Zhang et al. (2015) included the optimization of the utilities network and hydrogen generation. Due to improved model accuracy, López et al. (2012, 2013) reported a profit increase of 13% in comparison with a base line, whereas Zhang et al. (2015) claimed a 24% improvement after using their MINLP planning model.
Although leading to more accurate prediction, nonlinear models also introduce nonconvexities into refinery planning models. To overcome local optimal solutions, (Guerra et al., 2010; Guerra and Le Roux, 2011b) applied multi-start approaches by passing different initial points to the NLP solver, Andrade et al. (2016) presented a heuristic approach based on solving LP relaxations. However, neither of the approaches can guarantee global optimality. A systematic procedure is needed to avoid sacrificing optimality and improve the total profit (Khor and Varvarezos, 2017). Faria and Bagajewicz (2011a, 2011b, 2012) presented a deterministic global optimization (DGO) algorithm composed of a bound contraction method to update variable bounds and a direct relaxation of the bilinear terms using reference functions. Castillo Castillo et al. (2017, 2018) presented DGO algorithms to solve mixed-integer quadratically constrained quadratic programs (MIQCQPs) arising in large-scale refinery planning that performed similarly to the commercial solvers BARON (Sahinidis, 2004) and ANTIGONE (Misener and Floudas, 2014). The approach relies on mixed-integer linear programming (MILP) relaxations, derived from piecewise McCormick envelopes (Castro, 2015; Gounaris et al., 2009; McCormick, 1976) or the multiparametric disaggregation technique (Andrade et al., 2018; Castro, 2016; Kolodziej et al., 2013; Teles et al., 2013), and on the reduction of the variables domain through optimality-based bound tightening (OBBT) (Castro and Grossmann, 2014; Puranik and Sahinidis, 2017).

Regarding model integration and coordination of IRPC operations, little work has employed DGO to date. Li et al. (2016) presented a data-driven approach to generate submodels for various parts of a large-scale nonconvex MINLP model. This model was solved using ANTIGONE and led to a profit improvement of 30 – 65% compared to current operation. Zhao et al. (2017) presented an integrated model for fuels and olefins transformation that was solved by Lagrangian decomposition. The authors reported a profit increase between 14 – 53%, compared to a sequential optimization process (first the refinery and then the ethylene unit).
Herein, we present a deterministic global optimization algorithm for large-scale MIQCQP problems that relies on solving a series of lower and upper bounding sub-problems. Our approach, (Section 2), decomposes the problem structure into small clusters, then considers each cluster sequentially. Bilinear terms participating in the constraints of the active cluster or a previous cluster are relaxed with piecewise McCormick envelopes, whereas the other bilinear terms are relaxed using standard McCormick envelopes. The variable ranges are furthermore reduced using OBBT. We apply the clustering approach on a short-term planning problem inspired by an existing IRPC in Colombia (Section 3). Compared to previous studies, crude selection and allocation are more difficult since we consider a wider variety of sources from the market; demand concerns a larger variety of fuels and petrochemicals; and more process units are considered, which results in a larger number of nonconvex terms in the MIQCQP problem. The process units themselves are modeled using lab-scale, pilot-scale, or full industrial-scale data. The result of several scenarios are compared in terms of computational performance and their physical significance is also discussed (Section 4), before concluding the paper (Section 5).

2. Methodology

The short-term planning problem for the IRPC can be cast as the following MIQCQP:

\[
\begin{align*}
z^* := \max f_0(x, y) \\
\text{s.t. } & f_m(x, y) \leq 0 \ \forall m \in \{1, \ldots, M\} \\
& x \in [x^L, x^U] \subseteq \mathbb{R}_+^n, \ y \in \{0,1\}^r
\end{align*}
\]  

(P)

where \( x \) is a \( n \)—dimensional vector of non-negative continuous variables constrained between lower \( x^L \) and upper \( x^U \) bounds, and \( y \) is a \( r \)—dimensional vector of binary variables used to select process operating conditions. The functions \( f_m : \mathbb{R}_+^n \times \mathbb{R}^r \to \mathbb{R} \), with \( m = 0, \ldots, M \), comprising the objective function and the constraints, are quadratic in \( x \) and linear in \( y \): \( f_m(x, y) := \sum_{(i,j) \in BL_m} a_{ijm}x_ix_j + B_m x + C_m y + d_m \). \( BL_m \) is an \((i,j)\)-index set defining the bilinear terms \( x_ix_j \) present in function \( m \), parameters, \( a_{ijm} \) and \( d_m \) are scalars, and \( B_m \) and \( C_m \) are row vectors.
For the solution of problem $P$, we propose a deterministic global optimization algorithm that solves a sequence of upper and lower bounding problems. A relaxed MILP model $PR$ provides an upper bound $UB$ on the optimal value $z^*$ of problem $P$, while a restriction of $P$ with fixed binary variables, named $PF$, computes a lower bound $LB$. Optimality-based bound tightening problems $BC$ are also solved to improve the bounds of variables participating in bilinear terms, in order to make the $PR$ relaxation tighter.

2.1. Lower bounding problem

The lower bounding problem $PF$ shares the objective function and the constraints of $P$ but fixes the binary variables $y$ for the operating modes to the values $\hat{y}$ obtained from the solution of relaxed model $PR$ (see below):

$$
\begin{align*}
    z^{PF} &= \max f_0(x, \hat{y}) \\
    \text{s. t. } f_m(x, \hat{y}) &\leq 0 \forall m \in \{1, \ldots, M\} \\
    x &\in [x^L, x^U] \subseteq \mathbb{R}^n_+, \ y \in \{0,1\}^r
\end{align*}
$$

(PF)

Note that fixing the variables may lead to an infeasible problem. If this is not the case, any feasible solution $z^{PF}$ of $PF$ provides a lower bound $LB$ on the optimal value $z^*$ of $P$.

2.2. Upper bounding problem

Upper bounding problem $PR$ is obtained by substituting bilinear terms $x_i x_j$ in $P$ with auxiliary variables $w_{ij}$, essentially linearizing $f_m(x, y)$ as $f_m^R(x, y, w) := \sum_{(i,j) \in BL_m} a_{ijm} w_{ij} + B_m x + C_m y + d_m$. Additional constraints are added to the problem to relate $w_{ij}$ to the original variables and their lower and upper bounds. This is done through global or piecewise McCormick envelopes. The standard McCormick relaxation, adds 4 linear inequality constraints per bilinear term (McCormick, 1976), whereas the piecewise McCormick relaxation, adds 9 mixed-integer linear constraints. The latter is tighter due to the use of binary variables for partitioning the domain of one of the variables in every bilinear term (Bergamini et al., 2005; Castro, 2015) but may lead to a computationally
intractable MILP problem even for a small number of partitions. In order to improve the computational tractability of the relaxed MILP model PR, the continuous variables $x_j$ are furthermore grouped into $k = 1, \ldots, K$ clusters, where $CL^k \subseteq \{1, \ldots, n\}$.

By only partitioning the continuous variables $x_j$ belonging to a given cluster $CL$, we reduce the number of binary variables $y_{jn}$ added by the piecewise McCormick relaxation in problem PR, where $N_j$ is the chosen number of partitions for $x_j$. The McCormick relaxations are used for the relaxation of bilinear terms located outside the cluster ($j \notin CL$).

\[
\begin{align*}
\max & \quad z^R := \max f^R_0(x, y, w) \\
\text{s.t.} & \quad f_m^R(x, y, w) \leq 0 \quad \forall m \in \{1, \ldots, M\} \\
& \quad w_{ij} \geq \sum_{n=1}^{N_j} (\hat{x}_{ijn} x^n_j + \hat{x}_{ijn} x^n_i - y_{jn} x^n_j x^n_{jn}) \\
& \quad w_{ij} \geq \sum_{n=1}^{N_j} (\hat{x}_{ijn} x^n_j + \hat{x}_{ijn} x^n_i - y_{jn} x^n_i x^n_{jn}) \\
& \quad w_{ij} \leq \sum_{n=1}^{N_j} (\hat{x}_{ijn} x^n_j + \hat{x}_{ijn} x^n_i - y_{jn} x^n_i x^n_{jn}) \\
& \quad w_{ij} \leq \sum_{n=1}^{N_j} (\hat{x}_{ijn} x^n_j + \hat{x}_{ijn} x^n_i - y_{jn} x^n_i x^n_{jn}) \\
& \quad x_i = \sum_{n=1}^{N_j} \hat{x}_{ijn} \forall (i, j) \in BL_m, j \in CL \\
& \quad x_j = \sum_{n=1}^{N_j} \hat{x}_{ijn} \forall j \in CL: (i, j) \in BL_m \\
& \quad \sum_{n=1}^{N_j} y_{jn} = 1 \quad \forall j \in CL: (i, j) \in BL_m \\
& \quad x^n_j y_{jn} \leq \hat{x}_{ijn} \leq x^n_j y_{jn} \forall (i, j) \in BL_m, j \in CL, n \in \{1, \ldots, N_j\} \\
& \quad x^n_{jn} y_{jn} \leq \hat{x}_{ijn} \leq x^n_{jn} y_{jn} \forall j \in CL: (i, j) \in BL_m, n \in \{1, \ldots, N_j\} \\
& \quad w_{ij} \geq x_i x^n_j + x_i^U x_j - x_i^U x^n_j \\
& \quad w_{ij} \geq x_i x^n_j + x_i^U x_j - x_i^U x^n_j \\
& \quad w_{ij} \leq x_i x^n_j + x_i^U x_j - x_i^U x^n_j \\
& \quad w_{ij} \leq x_i x^n_j + x_i^U x_j - x_i^U x^n_j \\
& \quad x \in [x^L, x^U] \subseteq \mathbb{R}_+^n, \ y \in \{0, 1\}^m, \ w \subseteq \mathbb{R}_+^n \\
& \quad y_{jn} \in \{0, 1\} \forall j \in CL: (i, j) \in BL_m, n \in \{1, \ldots, N_j\}
\end{align*}
\]

Note that the lower $x^n_{jn}$ and upper $x^n_j$ bounds for the partitioned variable $x_j$ are calculated by Eq. 1 before solving the relaxed problem PR.

\[
\begin{align*}
x^n_{jn} & := x^n_j + \frac{(x^n_j - x_j)(n-1)}{N_j} \quad \forall j: (i, j) \in BL_m, j \in CL, n \in \{1, \ldots, N_j\} \\
x^n_{jn} & := x^n_j + \frac{(x^n_j - x_j)n}{N_j} 
\end{align*}
\]
Since \( PR \) is a relaxation of \( P \), \( PR \) will be feasible whenever \( P \) is feasible. Since the domain of the variables in \( P \) defines a compact set, its McCormick or RLT relaxation will always return a finite bound. Thus, the solution \( z^R \) of \( PR \) provides a finite upper bound \( UB \) on the optimal value \( z^* \) of \( P \).

2.3. Optimality-based bound tightening (OBBT)

Consider the variable \( x_h \) appearing in a bilinear term of cluster \( CL \). Its lower bound \( x^L_h \) and upper bound \( x^U_h \) can be tightened after solving optimization problem \( BC \).

\[
x^L_h / x^U_h := \min \max x_h \\
\text{s.t. } f^R_m(x, y, w) \leq 0 \forall m \in \{1, \ldots, M\} \\
f^R_0(x, y, w) \geq LB \\
x \in [x^L, x^U] \subseteq \mathbb{R}^n, \quad y \in \{0,1\}^n, \quad w \in W \subseteq \mathbb{R}^m \\
y_{jn} \in \{0,1\} \forall j \in CL: (i, j) \in BL_m, n \in \{1, \ldots, N_j\}
\]

Note that the inclusion of constraint \( f^R_0(x, y, w) \geq LB \) ensures that we only explore regions of the feasible space that can improve the current incumbent. As in \( PR \), \( BC \) linearizes the bilinear terms and relaxes the problem using the McCormick and piecewise relaxations based on the cluster decomposition. To avoid duplicating the thirteen sets of constraints in \( PR \), we refer to the domain of the auxiliary variables resulting from such constraints simply as \( w \in W \). In order to keep a moderate model size for \( BC \), \( N_j \) may be set to a lower value than in \( PR \).

2.4. Relaxation refinement

The tightness of the \( PR \) relaxation can be improved by increasing the number of partitions \( N_j \) for every partitioned variable \( x_j \). Since these variables might have different lower and upper bounds \((x^L_j, x^U_j)\), a uniform increment could increase the size of the relaxation prohibitively (Nagarajan et al., 2019). Instead, we update \( N_j \) dynamically by applying Eqs. (2)-(4) for every partitioned variable \( x_j \) inside the cluster \( CL \):

\[
 r_{ij} = \left( \frac{w_{ij} - x_i^L}{w_{ij}} \right) \forall (i,j) \in BL_m, j \in CL 
\]

\[
 \rho_{ij} = \max_{j \in CL(i)} \min_{(i,j) \in BL_m} r_{ij}^{\text{min}} \forall (i,j) \in BL_m, \forall j \in CL 
\]
\[
\lambda_j = \max_{i:(i,j) \in BL_m} \rho_{ij} \forall j \in CL
\]  

(4)

Here, \( r_{ij} \) represents the relative deviation at the solution point of PR between the exact value of the bilinear term \( x_i x_j \) and its relaxation \( w_{ij} \). This deviation is normalized in the range [0,1] as \( \rho_{ij} \), then used to compute the normalized score \( \lambda_j \). If this score is greater than a user-defined threshold \( \tau \), then the number of partitions \( N_j \) is incremented by \( \Delta \), up to the maximum partition size \( N^U \).

2.5. Deterministic global optimization algorithm

A flowchart of the clustering algorithm for the global optimization of the MIQCQP resulting from the short-term IRPC planning problem is shown in Figure 1. The boxes in red and blue indicate parallel processing for the solution of problems PR, PF and BC. These steps are further detailed below:

Step 1: Initialize parameters for controlling the algorithm performance, including total maximum runtime \( \text{MaxRunTime} \), optimality gap tolerance \( \epsilon \), maximum runtime \( \text{MaxRunTime}_{PRBC} \) and optimality gap \( \epsilon_{PRBC} \) for the solution of PR and BC problems, maximum number of iterations \( \text{MaxNumIter} \), initial \( N^0 \) and maximum number of partitions \( N^U \) for the Piecewise McCormick relaxation, with \( N^0 \leq N^U \). Set upper \( UB = +\infty \) and lower \( LB = -\infty \) bounds for the value \( z^* \) of the optimal solution of \( P \). Initialize the number of iterations: \( \text{Iter} = 0 \).

Step 2: Solve problem \( P \) to feasibility or local optimality. If a feasible solution \( z^* \) is found, set \( LB \leftarrow z^* \); otherwise, keep changing the initial point and solving \( P \) until finding a feasible solution.

Step 3: Solve relaxed problem \( \text{PR} \) to optimality using the standard McCormick relaxation for all bilinear terms (since there are no active clusters at this point). Set \( UB \leftarrow z^R \).

Step 4: Update relative optimality gap \( \epsilon^* = (UB - LB)/UB \)
Step 5. Repeat for each cluster $k = 1, ..., K$

**Step 5.1.** Define the active cluster $CL \leftarrow \bigcup_{l=1}^{k} CL^l$

**Step 5.2.** Initialize the number of partitions for every partitioned variable $x_j$ belonging to the active cluster $CL^l$ to $N_j = N^0$. If $l \geq 2$, keep the number of partitions in sub clusters $CL^1, ..., CL^{l-1}$ at their final values in previous iterations.

**Step 5.3.** Generate a population with up to $PFS$ feasible solutions of $PR$. Select the best solution at termination ($MaxRunTime_{PRBC}$ or $\varepsilon_{PRBC}$), $z^R$. If $z^R < UB$, update $UB \leftarrow z^R$.

**Step 5.4.** Each feasible solution obtained in step 5.3 provides initial values for the continuous variables and the values $\hat{y}$ for fixing the binary variables in $PF$. Solve up to $PFS$ instances of $PF$ and select the best feasible solution $z^{PF}$. If $z^{PF} > LB$, update $LB \leftarrow z^{PF}$.

**Step 5.5.** If $(UB - LB)/UB < \varepsilon^*$, update $\varepsilon^* \leftarrow (UB - LB)/UB$ and go step 5.5.1.

**Step 5.5.1.** If stop condition is reached, then report $LB$, $UB$ and $\varepsilon^*$ and finish. Otherwise, perform bound contraction by solving both instances of problem $BC$ for each variable present in the bilinear terms of the active cluster $CL$.

**Step 5.6.** Increment the number of iterations $Iter \leftarrow Iter + 1$. If stop condition is reached, report $LB$, $UB$ and $\varepsilon^*$ and finish. Otherwise go to step 5.6.1.

**Step 5.6.1.** If $\varepsilon^*$ has decreased, update dynamically the number of partitions $N_j$ for each partitioned variable $x_j \in CL$ (Eqs. 2-4) and go to step 5.3. Otherwise explore the next cluster.

The algorithm stops if any of these conditions are fulfilled: $TimeElapsed > MaxRunTime$, $Iter > MaxNumIter$, $\varepsilon^* \leq \varepsilon$; or if all the clusters are explored. For small problems like the one in section 2.6, convergence to the given tolerance might be achieved without the need to explore all the
clusters. For large-scale problems like the one in section 3, the main focus of this work, the algorithm may stop after exploring all the clusters, even if the gap is not closed to zero and the maximum iteration and run time limits are not reached.

Steps 5.3, 5.4 and 5.5.1 of the deterministic global optimization algorithm can be implemented using parallel computing. For example, in step 5.3, we took advantage of the solution pool facility of CPLEX (GAMS Software GmbH, 2012) with parameters Solnpooldensity, Solnpoolpop, Solnpoolgap and Solnpoolcapacity set to 0, 2, 0.10 and $PFS$, respectively. In addition, Parallellmode = 0 and Threads enable CPLEX to process in parallel. Steps 5.4 and 5.5.1 were also implemented in parallel using the GAMS grid computing facility (Bussieck et al., 2009). Thus, the $PFS$ instances of PF and BC problems were solved simultaneously.
Figure 1. Flowchart of the clustering algorithm for global optimization
2.6. Illustrative example

We now consider standard pooling problem Bental5, reported by Ben-Tal et al., (1994), for illustration of our cluster-based global optimization algorithm. This quadratically constrained problem (QCP) was solved to global optimality to yield a maximum of 3500, with known local solutions of 900, 1900, 2700 and 2900 (Adhya et al., 1999; Misener et al., 2011). We use the generic modelling framework of Neiro and Pinto (2004), detailed in section 1 of the Supplementary Material, which is similar to the p-formulation of Haverly (1978). The full set of constraints of problem \( \mathbf{P} \) is given in section 2 of the Supplementary Material. The process topology in Figure 2 shows the input data, together with some of the main constraints and optimal flows obtained using our clustering approach.

![Figure 2. Process Network for the pooling problem.](image)

We define \( K = 8 \) clusters, one for each intermediate pool: \( CL^1 = \{u6\}, \ CL^2 = \{u7\}, \ CL^3 = \{u8\}; \) and product: \( CL^4 = \{u9\}, \ CL^5 = \{u10\}, \ CL^6 = \{u11\}, \ CL^7 = \{u12\} \) and \( CL^8 = \{u13\}. \) Thus, the unit inside the cluster identifies the index of the partitioned flowrate variable \( QF_{u} \) in problem \( \mathbf{PR}. \) The balance for property \( p1 \) at unit \( u6 \) involves 5 bilinear terms, with a total of 80 bilinear terms involved.
in the estimation of properties \( p1 \) and \( p2 \) through the network (see section 2.1 of the Supplementary Material for details).

In the first iteration (steps 2 – 4 of the algorithm), we obtain a lower bound \( LB = 2700 \) and an upper bound \( UB = 3500 \), leading to a relative optimality tolerance \( \varepsilon^* = 0.30 \). In iteration \( Iter = 2 \), we explore cluster \( CL = CL^1 = \{ u6 \} \), partitioning variable \( QF_{u6} \) with \( N_{u6} = 2 \) partitions (step 5). Since neither \( LB \) or \( UB \) change after solving \( PF \) and \( PR \), \( \varepsilon^* \) is not improved, OBBT is not applied and the number of partitions for \( QF_{u6} \) remains the same. We then move to the next cluster \( (CL^2 = \{ u7 \}) \). In \( Iter = 3 \), \( CL = CL^1 \cup CL^2 = \{ u6, u7 \} \), \( QF_{u7} \) is partitioned with \( N_{u7} = 2 \), leading to a \( LB = 2900 \) when solving \( PF \) and to an improved \( \varepsilon^* = 0.21 \) (Step 5.4). Consequently, we perform bound contraction in step 5.5.1 and are able to reduce the domain of variables \( QF_{u6} \) and \( QF_{u7} \) by 50%. In iteration \( Iter = 4 \), we continue exploring the current cluster, by doubling the number of partitions in step 5.6.1, to \( N_{u6} = N_{u7} = 4 \) (check illustration in Figure 3). When solving \( PF \) in step 5.4, we are able to find the global optimal solution, \( LB = 3500 \). Since \( \varepsilon^* \) is now zero, the stopping condition has been reached and so there is no need to consider the other clusters.

Note that that during the search, the clustering approach found two of the previously reported local solutions. The algorithm converged after exploring just two clusters because in the optimal solution of \( PR \) the following conditions held: (i) most of the flowrate variables belonging to the other clusters, \( QF_{u9} - QF_{u13} \), are at their upper bounds and so there is no error when relaxing the bilinear terms where they appear; and (ii) the relaxation errors associated to partitioned variables \( QF_{u6} \) and \( QF_{u7} \), which took values away from the partition boundaries \( (QF_{u6}=209.1 \text{ and } QF_{u7}=150.9; \text{ check Figure 3}) \), and to the remaining flowrate variable, \( QF_{u8} \), do not affect the objective function value. In other words, while the solution of \( PR \) is not feasible in \( P \), it provides a good initialization for the NLP solver to converge to a feasible solution in \( P \) that has the same value of the objective function.

A step-by-step application of the algorithm is shown in section 2.3 of the Supplementary Material.
3. The integrated refinery petrochemical complex (IRPC) of Ecopetrol

The IRPC of interest is composed of a medium conversion refinery producing several grades of gasoline, diesel and fuel oil, and a set of petrochemical processes producing BTX, polyethylene, propylene, waxes and specialty solvents (Figure 4). The synergy between the refinery and petrochemical plants is enabled by the olefins production at the fluidized catalytic cracking (FCC) units, the hydrogen generation for diesel hydrotreating (HDT), and the platforming route for improving octane number in gasoline blending. Competition for raw materials arises from the routing of: (i) naphtha from the atmospheric crude distillation unit (CDU) to either gasoline blending or the production of aromatics, which are petrochemical precursors; (ii) atmospheric gasoil (distillates, LVGO and HVGO streams) from the vacuum distillation unit (VDU) to produce waxes, improve the FCC feedstock quality, or complete the processing capacity of diesel HDT. Coordination of the entire IRPC is thus crucial to exploit this process flexibility and manage feedstock competition between the refinery, petrochemical and fuels blending operations.

The raw material to the IRPC (crude oil and refined products) is supplied by both domestic production and imports. The national petroleum production in Colombia amounts to 297 kbbl/day and is aggregated into 17 types of crude oil distributed over 8 geographical regions (R1-R8). It is complemented with 7 types of imported crude, with a maximum availability of 15 kbbl/day/crude.
The crude assay provides the detailed characterization in terms of yields and properties for each region. The total refining capacity is 240 kbbl/day. Besides crude, refined products such as naphtha, low sulphur diesel and olefins may be imported to complement the feedstock of certain process units or be used for fuels blending. Demand is represented by the fuels and petrochemical requirements imposed by the domestic and export markets. Meeting the demand for petrochemicals entails the production of up to 26 commodities, divided into BTX, propylene, polyethylene, specialty solvents and waxes, and up to 22 grades of fuels.

**Figure 4.** Diagram of the full integrated refining-petrochemical complex (IRPC).

The logistic system comprises multimodal transport for the delivery and reception of commodities from the IRPC to the markets, and from the production wells, local market and importation ports to the IRPC. In particular, 4 river fleet routes (RF1-RF4) and a system of 9 pipelines (PL1-PL9) are available for the exchange of commodities.
3.1. Crude allocation

Crude allocation provides the feedstock to the crude distillation units. The network in Figure 5 illustrates how 17 domestic crude oils \((dc_1, \ldots, dc_{17})\) are delivered to the refinery via 5 pipelines \((pl_3, \ldots, pl_7)\). A system of 9 mixing tanks \((tk_1, \ldots, tk_9)\) are then used to obtain homogeneous crude blends \((cb_1, \ldots, cb_9)\) with given quality properties (specific gravity, sulphur content and total acid number) for further processing by the CDUs.

Figure 5. Crude allocation network.
3.2. Crude fractionation system

The crude fractionation system provides all the intermediate streams for further processing in the refinery, petrochemical and fuels blending units. This separation train comprises 6 units operating in a total of 13 operating modes (logic units cd${u}_{1}$, ..., cd${u}_{13}$). For instance, the first distillation unit has four operating modes (cd${u}_{1}$, ..., cd${u}_{4}$). A process model is developed for each operating mode, which is characterized by crude quality and straight-run cut-points. The feedstock is determined by routing the crude blend to the CDUs (Figure 5). For example, cd${u}_{4}$ is fed by a crude mix composed of cb7 and cb8.

The pool of light components produced at the CDUs, such as methane, ethane and propane, is sent to the turbo expander unit to separate methane from the C2-C3 blend (Figure 4). Methane is mixed with natural gas from the production wells and routed to the fuel gas network. The C2-C3 blend is mixed with olefins coming from the FCCs to provide the feedstock for the ethane cracker. A fraction of the naphtha produced at the CDUs provides the feedstock for aromatics and specialty solvents production. The rest is used for gasoline blending or upgraded into debutanized gasoline at the DBU units. The straight-run jet is sent directly to blending. The straight-run light diesel can be routed to the medium distillate blending while the heavy diesel is sent to hydrotreating.

**Table 1.** Routing of reduced crude from atmospheric to vacuum distillation units and RC pool (x) and of virgin naphtha from atmospheric to debutanizer columns (X).

<table>
<thead>
<tr>
<th>ADU</th>
<th>VDU1</th>
<th>VDU2</th>
<th>VDU3</th>
<th>VDU4</th>
<th>VDU5</th>
<th>VDU6</th>
<th>RC Pool</th>
<th>DBU1</th>
<th>DBU2</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDU1</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CDU2</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>CDU3</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>CDU4</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>CDU5</td>
<td>x</td>
<td></td>
<td>x</td>
<td>X</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>CDU7</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU8</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU9</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU10</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU11</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU12</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>CDU13</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
As can be seen in Table 1, the reduced crude (RC) from the CDU bottoms is distributed into 6 VDUs that produce the feedstock to the FCC units and provide additional components for fuel blending. The VDUs separate the RC into distillates, light vacuum gasoil (LVGO), heavy vacuum gasoil (HVGO) and vacuum residue (VR). The straight-runs light and heavy naphtha from the CDUs are routed to gasoline blending, petrochemicals production or distributed into two DBUs. The only exception is the naphtha mix from \textit{cdu11}, which goes to gasoline blending or petrochemicals production.

3.3. Refinery conversion units

The conversion units receive the diesel, RC, LVGO, HVGO and VR pools coming from the crude fractionation system to produce intermediate streams for fuels blending (Figure 6). The diesel is hydrotreated to generate a low sulphur component for blending. The RC pool is a feedstock component for the FCC units. The LVGO is a commodity that can be sold or processed at the catalytic crackers. In case of low production of gasoil, it can be imported to complete the FCC feedstock. The HVGO pool can be routed to fuel blending or hydrotreated into a low sulphur intermediate stream for further processing by the FCC units. Finally, the VR pool can be split, providing streams for asphalt/fuels blending or sent to DEMEX for reducing its metal content. Demetalized oil (DMO) is the most valuable product from DEMEX and can be sent to the FCC or HDT. In the HDT unit, it is transformed into demetalized hydrotreated oil (DMOH), an intermediate refined stream with low metal and sulphur content. The bottom stream from DEMEX can be routed to fuel oil blending or upgraded at the visbreaking unit to produce naphtha for gasoline blending, gasoil for FCC processing and a residue for fuel blending.

The feedstock to the FCC units consists of a mix of LVGO, HVGO, RC, DMO and DMOH. These units produce olefins for further processing at the ethylene cracker, propylene, a mix of isobutane-butylene (iC4/C4=), light cracked naphtha (LCN) and heavy cracked naphtha (HCN), light cycle oil (LCO), slurry and coke. The iC4/C4= blend is sent to the alkylation unit to produce alkylate and other
components for gasoline. All the LCN and a fraction of the HCN produced is sent to gasoline blending. The rest of the HCN is hydrotreated. The LCO is recycled to be hydrotreated with the diesel or can be used with the slurry for fuel oil blending. The coke is a by-product that is sold to domestic use.

**Figure 6.** Diagram for refinery conversion.

There are four hydrotreating units. One unit processes a mix of DMO and HVGO and can operate at four different severities. The other three units transform diesel, HVGO and HCN into low sulphur streams and operate in a single mode. Solvent extraction at the DEMEX unit can operate at four different levels of solvent concentration. Thermal cracking is represented by two different process configurations, one with four and the other with three operating modes. The FCC process is represented by four different conversion technologies. FCC1 and FCC2 can operate in four different modes, while there are three modes for FCC3 and FCC4.
3.4. Petrochemical processes

The virgin naphtha (VN) pool from the CDUs, a fraction of the LVGO from the VDUs, and the olefins from the FCC processes comprise the feedstock for the petrochemical, industrial solvents and waxes production (Figure 7). The olefins are blended with an ethane-propane stream to generate polyethylene at the ethylene cracker, to be sold on the domestic market. Note that if the olefins are not upgraded to polyethylene, they may be mixed with light gases to produce LPG or sent to the fuel gas network.

![Diagram for Petrochemical Production](image)

**Figure 7.** Diagram for petrochemical production.

The VN is fractionated in a separation train to obtain butane (C4), pentane, light and heavy naphtha. The C4 stream is routed to the LPG pool, while the pentane, light naphtha and a fraction of heavy naphtha are sent to gasoline blending. The heavy naphtha (HN) is the raw material to the BTX unit, after going through a series of steps. The first one is desulphurization. The second one is platforming, where hydrogen is obtained as by-product for the hydrar unit. The aromatics mix is separated at the sulfolane unit into two streams, the high-purity extract and the raffinate that is a feedstock for producing specialty solvents. The extract is then sent to the BTX separation train to obtain benzene,
toluene, xylene, o-xylene and heavy aromatics for the domestic market. At the hydrar unit, a fraction of benzene is converted into cyclohexane, which is also a marketable product. The aromatics production is completed at the hydeal unit, which converts a toluene-xylene stream into a mix that is recycled to the BTX fractionation units.

The LVGO is a valuable refined stream that can be sold on the domestic market or processed at the FCC or other units. In particular, the LVGO obtained from the fractionation of naphthenic crude oil is suitable to produce waxes and naphthenic base oils. However, several stages of solvent extraction are required to remove heavy components, contaminants and aromatics. More specifically, propane removes metals from the LVGO, the methyl ethyl ketone (MEK) precipitates a stream rich in wax content that is hydrotreated to obtain several grades of waxes, and the phenol removes aromatics to obtain lubricant base oils.

Overall, the petrochemicals, polyethylene, specialty solvents and waxes production might represent a small portion of the refinery margin but, without integration of the olefins, all the virgin naphtha would be blended into gasoline and the naphthenic LVGO would be degraded into lower value commodities, thereby sacrificing profitability.

3.5. Fuels blending

The blenders receive intermediate process streams from upstream units and some refined products bought on the domestic or international market, to produce fuels with the required quality. In total, 88 refined streams can produce up to 22 grades of fuels (Table 2), with different quality specifications for each grade (Table 3). For example, gasoline must comply with quality constraints on the specific gravity, Reid vapor pressure (RVP), research octane number (RON), and sulphur content. Property estimation is thus required for all 88 streams.

Table 2. Streams for fuels blending.

<table>
<thead>
<tr>
<th>Fuel</th>
<th>LPG</th>
<th>Gasoline</th>
<th>Medium distillate</th>
<th>Fuel Oil</th>
<th>Asphalt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Streams to blend</td>
<td>24</td>
<td>23</td>
<td>25</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td>Products grade</td>
<td>3</td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 3. Property/quality specifications of the different fuels.

<table>
<thead>
<tr>
<th>Fuel/Quality</th>
<th>Specific gravity</th>
<th>Sulphur content</th>
<th>Cetane number</th>
<th>RON</th>
<th>RVP</th>
<th>Viscosity</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPG</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gasoline</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Medium distillate</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fuel Oil</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Asphalt</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

3.6. Process clustering

Crude oil refinery operations can be classified into crude oil unloading and blending, unit operations that include separation and reaction processes, products blending and delivery (Jia and Ierapetritou, 2004; Méndez et al., 2006). We extend this classification to the IRPC with chemicals production, following the workflow to decompose the process network into small clusters of process units with similar functionality. For instance, crude selection is made before crude allocation, which determines the CDU feed streams in terms of flowrate and quality and hence their operation. Crude fractionation then determines the performance of downstream processes such as petrochemical production and the conversion of intermediate refined streams. Finally, fuels production and their quality specifications depend on the flowrate and properties of the intermediate refined streams.

Overall, the IRPC is composed of 60 industrial plants which are represented by about 125 models. Moreover, crude mixing and fuels blending is done in a tank farm, modelled as 30 additional units. The 155 models are divided into 6 clusters, as detailed in Table 4.

Table 4. Cluster definition (units identified by different colors in Figure 4)

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Process Units</th>
<th>Physical Units</th>
<th>Logical Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CL^I$</td>
<td>Logistic $(U^{RW})$ and Crude Allocation $(U^{CMX})$</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>$CL^II$</td>
<td>Crude Distillation $(U^{CDU})$</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>$CL^III$</td>
<td>Vacuum $(U^{VDU})$ and Debutanizer Columns $(U^{DBU})$</td>
<td>8</td>
<td>11</td>
</tr>
<tr>
<td>$CL^IV$</td>
<td>Refining $(U^{REF})$</td>
<td>36</td>
<td>53</td>
</tr>
<tr>
<td>$CL^V$</td>
<td>Petrochemical $(U^{PTC})$</td>
<td>9</td>
<td>48</td>
</tr>
<tr>
<td>$CL^VI$</td>
<td>Fuels Blending $(U^{BLD})$</td>
<td>22</td>
<td>22</td>
</tr>
</tbody>
</table>
3.7. Optimization problem statement

The optimization problem for the short-term planning of an IRPC can be stated as follows:

*Given:*

1. The set of process units and connecting streams in the IRPC
2. The capacities, operating conditions, feedstock properties and operating costs of the process units
3. The volume availability, cost and assay characterization of domestic and imported crude oil
4. The volume, cost and physical properties of domestic and imported refined products
5. The set of multi-modal transportation routes to deliver raw material or refined products from the oil wells/import-ports/domestic-market to the refinery/export-ports/domestic-market, characterized in terms of capacity and cost
6. Prices, market demands (domestic and export) and product specifications for LPG, jet, gasoline, diesel, fuel oil, asphalt, petrochemical products, waxes and industrial solvents

*Determine:*

1. The crude basket composition in terms of domestic and imported crude oil
2. The crude mix to be processed at each CDU
3. The cut-points to predict yields and properties at the CDUs, VDUs and DBUs
4. The feedstock volume, composition and operating conditions of each process unit
5. The yields and properties of the outlet streams of each process unit
6. The volume of refined products from domestic and imported markets needed to complement feedstock or fuels blending
7. The routing for all of the intermediate refined streams
8. The volume and specifications of each marketable commodity
**In order to:**

1. Maximize the profit, determined as the revenue from selling the fuels and petrochemical products, minus the costs of purchasing crude oil and refined products, minus the logistic costs for the delivery/reception of commodities, minus the operational expenditure.

2. Subject to meeting product demand in terms of volume and quality, satisfying operating conditions for each process unit, and respecting logistics.

A complete mathematical formulation of the IRPC short-term planning problem is provided in the Supplementary Material. Detailed information regarding raw material availability, costs and specifications as well demand and specifications for petrochemical and fuels products are given in Tables S1 to S8, while the logistic system is characterized in Table S9 and S10. The crude oil assays, and the parameters used for predicting yields, stream properties, operational cost coefficients and transportation fees were obtained from lab-scale, pilot-scale, or full industrial-scale test-runs. Such information is not disclosed for confidentiality reasons.

The nonconvexities arise from the operational expenditure calculation in the objective function, the pooling equations to determine intermediate stream quality and fuels blending specifications, and the correlations to predict yields and product properties of the process units. The results of the clustering approach applied to several scenarios, recreating typical planning decisions in the Colombian hydrocarbon market are presented next.

4. **Computational results**

All the models used to address the short-term planning problem of the IRPC of Ecopetrol were implemented in GAMS 25.1.2. To solve the MILP relaxations, we have used CPLEX 12.8 running in parallel deterministic mode and using up to 8 threads, a relative tolerance of $\epsilon_{PRBC} = 0.01\%$ and a maximum time of $MaxRunTime_{PRBC} = 2,000$ CPU seconds. The NLP subproblems were solved using the local solver CONOPT. In all the scenarios, the stopping condition of the clustering algorithm was to explore all the clusters. To compare with the proposed clustering approach, the
MIQCQP problem has also been solved by commercial solvers BARON 18.5.8 and ANTIGONE 1.1, after setting the maximum computational time to 10 CPU hours and the optimality gap to $\epsilon=1\%$. We also tested DICOPT 2 and KNITRO 11.1.1, but these solvers were unable to return a feasible solution for the base scenario described in section 4.1. All computations were conducted on a 64-bit desktop computer with 8 Intel i7-6700 (3.4 GHz) processors, 16 GB of RAM, and running Windows 7.

In order to obtain an initial feasible solution in step 2 of the clustering decomposition approach (section 2.5), problem $P$ is solved using a MINLP local optimizer. To initialize the binary variables $y$, we select typical operating conditions. The continuous variables, representing flowrates and stream properties, are set to one of their bounds $(x^L, x^U)$, which are obtained from the unit’s design conditions. If no feasible solution is identified, another set of operating conditions is provided as initial point.

### 4.1. Base scenario

We start by considering a base scenario, with the hydrocarbon market requirements for LPG, gasoline, medium distillate, fuel oil and asphalt set to 15, 183, 149, 80 and 7.2 kbbl/day, respectively. The combined demand for liquid petrochemicals, industrial solvents, waxes and propylene is 13.90 kbbl/day, while the polyethylene demand is 0.96 kton/day. The crude basket can be composed of both domestic and imported crudes.

The performance of the three global optimization algorithms for the base scenario is compared in Figure 8, representing the progress of the best-found (LB) and best-possible (UB) solutions against computational time. Both ANTIGONE (Plan A) and BARON (Plan B) are able to identify a feasible solution to the problem rather quickly, but no significant improvement is observed in either bounds until the maximum runtime (10 CPU hours). In contrast, the clustering algorithm (Plan C) starts with the same profit as ANTIGONE, but with a tighter upper bound corresponding to an optimality gap of 42%. After processing Cluster I, this approach can reduce the optimality gap to 8%. This is accomplished by applying OBBT for reducing the variable ranges and by increasing the number of
partitions in $CL^I$. In the process, the profit jumped from $2.63$ to $2.92$ MM/day. During Clusters II and III the gap remains unchanged, then the profit increases to $2.96$ MM/day after going through Cluster IV. This value is $11$ and $10\%$ greater than the solutions found by ANTIGONE and BARON, respectively.

**Figure 8.** Performance of global optimization algorithms in the base scenario.

**Figure 9.** Optimized raw material supply in base scenario (Plans A, B and C were generated by ANTIGONE, BARON and the new clustering algorithm, respectively).
Figure 9 illustrates the purchases of domestic (per region) and import (IC) crude oil, as well as of refined products from domestic (DRP) and import markets (IRP). The corresponding total refining capacity in Plan C is 219 kbbl/day, comprising 211 kbbl/day of domestic crude. In contrast, Plans A and B recommend processing 186 and 200 kbbl/day of domestic crude without petroleum import.

The main differences in the domestic crude purchase arise in regions R1, R2, R4 and R6. Plan C selects crude from R1, unlike Plans A and B. The largest deviation is for region R2, which accounts for 15% of the total crude cost. To complete the feedstock of process units and contribute to fuels blending, all three plans recommend purchasing the same amount of refined products.

As seen in Figure 10, logistics are another key differentiating factor between the optimized plans. Since the IRPC operates at a higher capacity in Plan C, more crude needs to be delivered by pipeline, particularly by PL6 (10.3 and 7.7 kbbl/day more), PL7 and PL8-PL9. In the case of PL7, it corresponds to the 4.3 kbbl/day purchases from region R1 (check Figure 9). The other significant difference involves the bidirectional system PL8-PL9. The three plans recommend delivering fuel oil from the plant to the export port. However, the gap of 7.8 kbbl/day between Plan C and both Plans A and B, is due to the purchases of imported crude oil (IC), which are delivered in the other direction.

Figure 10. Optimized logistics in base scenario.
In terms of revenue, the three plans recommend producing the same volume of petrochemicals, waxes and industrial solvents (Figure 11). This is because the process units related with the production of these commodities are set to operate at their maximal capacity (see PTQ columns in Figure 12). There are also small discrepancies in the production of LPG and asphalt, with Plan C producing more gasoline, medium distillate and fuel oil than Plan A (production differences of 20.6, 4.7 and 13.9 kbbl/day) and B (differences of 10.7, 2.6 and 12.5 kbbl/day). Consequently, the total income of $18.29 MM/day is 11% and 6% higher with Plan C, respectively.

Figure 11. Optimized aggregated production income in base scenario.

The final comparison concerns the operational capacity (Figure 12). Our new clustering approach recommends processing a total of 219 kbbl/day of crude oil across all six CDUs. In addition, 101 kbbl/day of atmospheric bottom residue produced at the CDUs are sent to all six VDUs. The processing of the bottom of barrel (BTP), which includes two technologies of visbreaking and one of DEMEX, accounts for 49 kbbl/day of reduced crude. The FCC capacity is about 80 kbbl/day, which is distributed into four technologies that can process a combined feedstock composed of reduced crude, LVGO-HVGO, DMO and DMOH. An alkylation capacity of 6 kbbl/day (not shown in the graph) provides a high-octane component to improve gasoline quality specifications. The naphtha, diesel and gasoil hydrotreating (HDT) combine to a processing capacity of 28 kbbl/day, while the
petrochemical units (PTQ) operate at 20 kbbl/day. Both Plans A and B present a similar capacity in the HDT and PTQ units.

![Graph showing operational capacities](image)

**Figure 12.** Optimized operational capacities in base scenario.

### 4.2. Scenario without petrochemical processes

The second scenario omits the petrochemical processes, by setting the demands for petrochemicals, industrial solvents and waxes to zero, and by removing Cluster V from the network topology (48 logic process units). Plan C, computed with the clustering algorithm, presents a total refining capacity of 201 kbbl/day, decreasing crude oil supply by 18 kbbl/day compared to the base scenario. The FCC throughput is 77 kbbl/day (4 kbbl/day less), leading to a 0.75 kbbl/day decrease in gasoline production, while the production of the other fuels remains similar. Another main difference is that the olefins are now sent to the fuel gas network and all of the naphtha virgin is routed to gasoline blending. Consequently, the profit from fuels production only is $2.00 MM/day, a 32% decrease from the base scenario.

Plans A and B, computed with the commercial solvers, recommend refining less than 180 kbbl/day, which would be the lowest historic capacity for this IRPC. Plan A processes 111 kbbl/day of crude oil, shuts down three of the six CDUs, and operates the FCC at 46% of its design capacity, for a profit of $1.2 MM/day. Plan B reduces the refining capacity to 163 kbbl/day, shuts down one CDU, and operates the FCC at a throughput of 57 kbbl/day, for a profit of $1.5 MM/day.
These profits are 39% and 24% lower than the one from Plan C. The 11% optimality gap from the clustering algorithm is also much lower. The search started with a feasible solution similar to the one obtained by ANTIGONE but worse than that found by BARON. However, at the end of $CL^1$, the profit had already increased to $1.97 \text{ MM/day}$, improving slightly in Cluster III to $2.00 \text{ MM/day}$.

The full results of this scenario are shown in Figures S2-S6 of the Supplementary Material.

4.3. **Logistic disruption scenario**

To analyze the impact of a disruption in the domestic crude supply, the capacity of pipeline PL3, which is responsible for delivering up to 80% of the crude to the IRPC, was reduced by half.

The clustering algorithm predicts a profit of $2.66 \text{ MM/day}$, which here again is 19% and 11% better than the plans computed with ANTIGONE and BARON, respectively. Since PL3 is responsible for the transportation of the crude oil produced in regions R3 and R6, their supply represents the main deviation in the crude basket selection. It is compensated by increasing the crude supply from region R1, through pipeline PL7, and by importing the maximum volume that can be delivered by the logistics system (PL8-PL9). The plans from the commercial solvers do not purchase crude from region R1. ANTIGONE suggests importing half of the crude oil, while BARON also maximizes imports. The effect of the disruption on commodities is a reduction in the volume of gasoline and medium distillate. The main income difference is due to gasoline production, with 72 kbbl/day throughput from the FCCs, compared to 49.4 kbbl/day (ANTIGONE) and 60.4 kbbl/day (BARON).

The insight from the comparison of the performance profiles for the logistic disruption scenario is rather similar. Nevertheless, the optimality gap remains practically unchanged after the first cluster. The clustering approach returns a profit of $2.66 \text{ MM/day}$, which is 19% and 11% better than ANTIGONE and BARON. The best possible solution (upper bound) is $3.09 \text{ MM/day}$, leading to an optimality gap of 14% after a runtime of 13,250 CPUs. The complete results of this scenario can be found in Figures S7-S11 of the Supplementary Material.
4.4. Gasoline demand reduction scenario

The final scenario analyzes the effect of reducing gasoline demand by 25%. Since the main income comes from fuels production, dominated by gasoline and medium distillate, the IRPC will be forced to shift production towards other commodities. The three plans advise against importing crude and recommend reducing the crude supply from region R6, affecting the operation of pipelines PL3, PL8 and PL9. The total crude oil throughput is 204 kbbl/day in Plan C compared with 140 and 178 kbbl/day in Plans A and B, respectively. All plans recommend increasing medium distillate production, by reducing the FCC throughput to 72 (Plan C), 50 (Plan A) and 61 kbbl/day (Plan B). Any LVGO that is not processed by the FCC is sold to the domestic market.

For the scenario of gasoline demand reduction, the optimal profit after 20,890 CPUs is $2.83 MM/day, with an optimality gap of 9%. This value is 26% and 18% better than for ANTIGONE and BARON, respectively. As in previous cases, Cluster I provides the largest improvement. Nevertheless, the best-found solution improves slightly after clusters II and IV. The full results are presented in Figures S12–S16 of the Supplementary Material.

4.5. Summary of the results for the different scenarios

Overall, the proposed algorithm based on cluster decomposition significantly outperforms the commercial solvers ANTIGONE and BARON for the short-term planning of the IRPC. We found higher profit values with optimality gaps between 8% and 14%, which for the scale of this problem is a remarkable result. The results are summarized in Table 5.

The cluster decomposition enhances the chance of identifying good-quality solutions. We show for the industrial-size case study that exploring the process network from crude oil allocation to fuels blending leads to a gradual increase in size, i.e. in the number of units in the cluster, number of bilinear terms and number of partitions. The advantage of keeping the size small is that whenever the maximum run time limits prevent problems PR and BC to be solved to optimality, better bounds are obtained. If, however, the order is reversed, PR and BC increase their size more quickly as the clusters
are explored, overall becoming less effective. Figure 13 shows the impact for the base scenario, were we can see that moving from fuels blending to crude allocation leads to a worse profit, LB= $ 2.928 MM/day, and a higher optimality gap (21%). Systematic procedures are thus needed not only to group the variables into clusters, but also decide on the best possible order for the clusters.

Table 5. Summary of performance of global optimization solvers for the different scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Clustering</th>
<th>ANTIGONE</th>
<th>BARON</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base case scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best-found Solution [MMUSD-day]</td>
<td>2.964</td>
<td>2.634</td>
<td>2.687</td>
</tr>
<tr>
<td>Best-possible Solution [MMUSD/day]</td>
<td>3.205</td>
<td>3.898</td>
<td>4.250</td>
</tr>
<tr>
<td>Optimality Gap [%]</td>
<td>8</td>
<td>48</td>
<td>58</td>
</tr>
<tr>
<td>Runtime [h]</td>
<td>5.7</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td><strong>No petrochemical integration scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best-found Solution [MMUSD-day]</td>
<td>2.009</td>
<td>1.219</td>
<td>1.523</td>
</tr>
<tr>
<td>Best-possible Solution [MMUSD/day]</td>
<td>2.233</td>
<td>2.926</td>
<td>3.404</td>
</tr>
<tr>
<td>Optimality Gap [%]</td>
<td>11</td>
<td>140</td>
<td>124</td>
</tr>
<tr>
<td>Runtime [h]</td>
<td>5.8</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td><strong>Logistic disruption scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best-found Solution [MMUSD-day]</td>
<td>2.664</td>
<td>2.156</td>
<td>2.360</td>
</tr>
<tr>
<td>Best-possible Solution [MMUSD/day]</td>
<td>3.050</td>
<td>3.451</td>
<td>3.842</td>
</tr>
<tr>
<td>Optimality Gap [%]</td>
<td>14</td>
<td>60</td>
<td>63</td>
</tr>
<tr>
<td>Runtime [h]</td>
<td>3.7</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td><strong>Gasoline demand reduction scenario</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best-found Solution [MMUSD-day]</td>
<td>2.833</td>
<td>2.186</td>
<td>2.445</td>
</tr>
<tr>
<td>Best-possible Solution [MMUSD/day]</td>
<td>3.090</td>
<td>3.719</td>
<td>4.134</td>
</tr>
<tr>
<td>Optimality Gap [%]</td>
<td>9</td>
<td>70</td>
<td>69</td>
</tr>
<tr>
<td>Runtime [h]</td>
<td>5.8</td>
<td>10.0</td>
<td>10.0</td>
</tr>
</tbody>
</table>
Figure 13. Performance of the clustering decomposition in the base case scenario when reversing the order of clusters.

4.6. Comparison to algorithm with static piecewise relaxations and no clustering

For the base scenario, problem $P$ comprises 6,975 equations, 35,104 nonlinear terms, 9,592 continuous and 279 discrete variables. Table 6 illustrates the size of the relaxed model $PR$ without clusters as a function of the number of partitions $N$, for different choices of partitioned flowrate variables. As can be seen, if we decide to partition all the flowrate variables appearing in bilinear terms, even for a small partitioning size of $N = 4$, the total number of binary variables is in the tens of thousands, which might compromise computational performance.

In Figure 14, we show the performance of an algorithm similar to the one in section 2 but using a constant number of partitions in the piecewise relaxation and no clustering. We can see that it takes twice as long to find a high-quality solution early in the search, and we end up with a solution with 1.4% worse profit. Notice that the performance is very similar for different values of $N$, with the optimality gaps at termination being roughly 22%. This behavior reflects the large size of the $PR$ relaxation problems (see Table 6). Still, these values are much lower than the gaps obtained with ANTIGONE and BARON.
Table 6. Model size for relaxed problem PR as a function of the number of partitions N.

<table>
<thead>
<tr>
<th>Partitioned variables</th>
<th>N</th>
<th>Variables</th>
<th></th>
<th>Total Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td>Continuous</td>
<td>Binary</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>19415</td>
<td>18760</td>
<td>655</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>27915</td>
<td>26216</td>
<td>1699</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>33575</td>
<td>31180</td>
<td>2395</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>42807</td>
<td>39716</td>
<td>3091</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>52595</td>
<td>48808</td>
<td>3787</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>61827</td>
<td>57344</td>
<td>4483</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>71059</td>
<td>65880</td>
<td>5179</td>
</tr>
</tbody>
</table>

\[ QF_u, QS_u, S \]

\[ QF_u, QS_u, S, Q_{u', S, u} \]

|        | 4   | 111081   | 98354  | 12727 | 186971         |
|        | 4   | 269019   | 236024 | 32995 | 427201         |

Figure 14. Comparison of proposed algorithm to one considering no clustering and static piecewise relaxations (results for different number of partitions N and base scenario).

5. Conclusions

This paper has presented a deterministic global optimization algorithm for large-scale MIQCQPs. The novel aspect has been to divide the bilinear terms in clusters. A piecewise McCormick relaxation is performed for the terms belonging to the active cluster while the standard McCormick relaxation is used for the others. Other key features involve optimality-based bound tightening and dynamic partitioning of the variables whose domain is reduced. The advantage is that the size of the MILP
relaxation is increased gradually, as a function of the number of terms in the cluster, instead of generating one large MILP relaxation for the entire MIQCQP, which may lead to low-quality solutions, prohibitively high computational times and/or large optimality gaps.

The clustering-based algorithm was shown capable of solving a QCP benchmark instance of the pooling problem to global optimality and of tackling a short-term planning problem of an existing integrated refinery-petrochemical facility. In the latter problem, binary variables select the optimal operating mode of conversion processes over a set of alternatives, while the bilinear terms appear in the cost function, intermediate pooling and fuel blending equations, and in the expressions to predict the yields and stream properties for each process unit, which are based on laboratory data, pilot plant and industrial test-runs. The decomposition of the process topology into clusters was made according to the functionality of the process units and the workflow, from crude selection to fuels blending.

Besides the topology complexity of the real-life IRPC, logistic considerations for the reception and delivery of commodities as well as the high dimensionality of crude basket selection and product allocation make it challenging to find high-quality solutions to the planning problem within a reasonable optimality gap. Results for a few scenarios have shown that our clustering algorithm, as well as commercial solvers ANTIGONE and BARON, were able to compute solutions that can be implemented by the plant operators. However, there were large differences among the best-found plans with our algorithm outperforming the others, not only in terms of profit but also in optimality gap and computational runtime.

Future work will focus on testing alternative relaxations, such as the normalized multiparametric relaxation technique, and on evaluating the performance in other MIQCQPs problems, such as water management, generalized pooling and multiperiod blending.

Acknowledgments

The authors would like to acknowledge the financial support from the Colombian Science Council (COLCIENCIAS) and the Colombian Petroleum Company (Ecopetrol S.A.). Pedro Castro also
acknowledges support from Fundação para a Ciência e Tecnologia through projects CEECIND/00730/2017 and UID/MAT/04561/2019.

Acronyms

- **BTX** Benzene – Toluene - Xylene
- **C2** Ethane
- **C3** Propane
- **C4** Butane
- **C4=** Butylene
- **CDU** Atmospheric crude distillation unit
- **CR** Catalytic reforming
- **DBU** Debutanizer columns
- **DC** Delayed coking
- **DMO** Demetalized oil
- **DMOH** Demetalized hydrotreated oil
- **DR** Distributed recursion
- **FCC** Fluid catalytic cracking
- **HCN** Heavy cracked naphtha
- **HDT** Hydrotreating
- **HN** Heavy naphtha
- **HVGO** Heavy vacuum gas oil
- **iC4** Isobutane
- **IRPC** Integrated refinery – petrochemical complex
- **LB** Lower bound on the solution value of $P$
- **LCN** Light cracked naphtha
- **LCO** Light cycle oil
- **LN** Light naphtha
- **LP** Linear programming
- **LVGO** Light vacuum gas oil
- **MEK** Methyl ethyl ketone
- **MILP** Mixed-integer linear programming
- **MINLP** Mixed-integer nonlinear programming
- **NLP** Nonlinear programming
- **OBBT** Optimality-based bound tightening
- **PTQ** Petrochemical
- **RC** Reduced crude
- **RON** Research octane number
- **RVP** Reid vapor pressure
- **SLP** Successive linear programming
- **UB** Lower bound on the solution value of $P$
- **VDU** Vacuum distillation unit
- **VN** Virgin naphtha
- **VR** Vacuum residue
Nomenclature

Sets

\(BL_m\) = set of index pairs \((i, j)\) mapping bilinear terms \(x_i x_j\) participating in function \(f_m\)

\(CL\) = clusters

Parameters

\(N^0, N^U\) = initial and maximal partition size in piecewise McCormick relaxation

\(x^L_n, x^U_n\) = lower and upper bounds for the partitioned variable \(x_j\) in the partition \(n\)

\(x^L, x^U\) = lower and upper bounds for variable \(x\)

\(a_{ijm}\) = scalar multiplying bilinear term \(x_i x_j\) in function \(f_m\)

\(B_m\) = vector multiplying variable \(x\) in function \(f_m\)

\(C_m\) = vector multiplying variable \(y\) in function \(f_m\)

\(d_m\) = constant term in function \(f_m\)

\(r_{ij}\) = gap between the bilinear term \(x_i x_j\) and its relaxation \(w_{ij}\)

\(\rho_{ij}\) = normalized gap between the bilinear term \(x_i x_j\) and its relaxation \(w_{ij}\)

\(\lambda_j\) = relaxation score for the partitioned variable \(x_j\)

\(N_j\) = partition size for variable \(x_j\)

\(\tau\) = threshold for refining the partition size in piecewise McCormick relaxation

Binary variables

\(y\) = binary variable indicating the selection of operating conditions

\(y_{jn}\) = binary variable selecting partition \(n\) in the variable \(x_j\)

Continuous variables

\(x\) = flowrates and stream properties

\(w_{ij}\) = linearization of the bilinear term \(x_i x_j\)

\(\hat{x}_{jn}\) = value of the partitioned variable \(x_j\) inside the partition \(n\)

\(\hat{x}_{ijn}\) = aggregated variable for \(x_i\) present in the bilinear term \(x_i x_j\) inside the partition \(n\)

Free variables

\(z^\ast\) = optimal solution value for \(P\)

\(z^R\) = value for \(PR\)

\(z^{PF}\) = value for \(PF\)

References


ASPEN Technology Inc, 2010. ASPEN P.I.M.S. System Reference (v7.2.).


