Imperial College London

GLOBAL OPTIMISATION OF LARGE-SCALE QUADRATIC PROGRAMS: APPLICATION TO SHORT-TERM PLANNING OF INDUSTRIAL REFINERY-PETROCHEMICAL COMPLEXES

by

Ariel Uribe Rodríguez

Sargent Centre for Process Systems Engineering

Department of Chemical Engineering

Imperial College London

Submitted in part fulfilment of the requirements for the degree of

Doctor of Philosophy of the Imperial College London

And the Diploma of Imperial College London

June 2023

DECLARATION

I herewith declare that this dissertation is the result of my own work and has not been previously submitted for consideration to any other degree or qualification at this or any other university. The work which is not my own has been properly acknowledged.

Ariel Uribe Rodríguez

ABSTRACT

This thesis is driven by an industrial problem arising in the short-term planning of an integrated refinery-petrochemical complex (IRPC) in Colombia. The IRPC of interest is composed of 60 industrial plants and a tank farm for crude mixing and fuel blending consisting of 30 additional units. It considers both domestic and imported crude oil supply, as well as refined product imports such as low sulphur diesel and alkylate. This gives rise to a large-scale mixed-integer quadratically constrained quadratic program (MIQCQP) comprising about 7,000 equality constraints with over 35,000 bilinear terms and 280 binary variables describing operating modes for the process units. Four realistic planning scenarios are recreated to study the performance of the algorithms developed through the thesis and compare them to commercial solvers.

Local solvers such as SBB and DICOPT cannot reliably solve such large-scale MIQCQPs. Usually, it is challenging to even reach a feasible solution with these solvers, and a heuristic procedure is required to initialize the search. On the other hand, global solvers such as ANTIGONE and BARON determine a feasible solution for all the scenarios analysed, but they are unable to close the relaxation gap to less than 40% on average after 10h of CPU runtime. Overall, this industrial-size problem is thus intractable to global optimality in a monolithic way.

The first main contribution of the thesis is a deterministic global optimisation algorithm based on cluster decomposition (CL) that divides the network into groups of process units according to their functionality. The algorithm runs through the sequences of clusters and proceeds by alternating between: (i) the (global) solution of a mixed-integer linear program (MILP),

obtained by relaxing the bilinear terms based on their piecewise McCormick envelopes and a dynamic partition of their variable ranges, in order to determine an upper bound on the maximal profit; and (ii) the local solution of a quadratically-constrained quadratic program (QCQP), after fixing the binary variables and initializing the continuous variables to the relaxed MILP solution point, in order to determine a feasible solution (lower bound on the maximal profit). Applied to the base case scenario, the CL approach reaches a best solution of 2.964 MMUSD/day and a relaxation gap of 7.5%, a remarkable result for such challenging MIQCQP problem. The CL approach also vastly outperforms both ANTIGONE (2.634 MMUSD/day, 32% optimality gap) and BARON (2.687 MMUSD/day, 40% optimality gap).

The second main contribution is a spatial Lagrangean decomposition, which entails decomposing the IRPC short-term planning problem into a collection of smaller subproblems that can be solved independently to determine an upper bound on the maximal profit. One advantage of this strategy is that each sub-problem can be solved to global optimality, potentially providing good initial points for the monolithic problem itself. It furthermore creates a virtual market for trading crude blends and intermediate refined–petrochemical streams and seeks an optimal trade-off in such a market, with the Lagrange multipliers acting as transfer prices. A decomposition over two to four is considered, which matches the crude management, refinery, petrochemical operations, and fuel blending sections of the IRPC. An optimality gap below 4% is achieved in all four scenarios considered, which is a significant improvement over the cluster decomposition algorithm.

ACKNOWLEDGMENTS

I am sincerely grateful to my supervisor Dr Benoit Chachuat for all his support and motivation. I cannot thank you enough Dr Benoit Chachuat for the opportunity and constant encouragement throughout this journey, which formed me beyond the academic perspective. I am also extremely thankful to Dr. Pedro Castro, because without his help and clear guidance, this thesis would not have been possible. I am also thankful to Dr Gonzalo Guillén Gosálbez for his insightful comments and endless patience.

I am deeply gratified to Professors Ignacio Grossmann and Nilay Shah, my external and internal examiners respectively. Thank you for creating the optimal atmosphere for my PhD defence. I really enjoyed the discussion about my research, your constructive comments, and challenging questions.

I would like to thank the Colombian Ministry of Science, Technology and Innovation (MINCIENCIAS), and the Center for Innovation and Technology Colombian Petroleum Institute at ECOPETROL S.A., for the financial support granted to pursue my PhD.

Thanks to Andrés, Daniel, Amjad, Raúl, Naveed, Nathaniel, Phantisa, Ibrahim, Lucian, Yukon, Antonio, Maria, Ana, David and all my dear friends at the Sargent Centre for Process Systems Engineering, who made every day a great time, full of genius discussions and laughs.

It was an immense honour to share my time with such talented people, you are the generation next of PSE stars.

I would also like to thank all those friends for whom time and distance mean nothing: The squad (Edgar Castillo and Edgar Yánez); Newton (Omar Guerra and Andrés Calderón), the Union (Iván Ordónez, Luz Angela Bohórquez), Oscar Neira, Andrés Mantilla, Sandra Montagut, Freddy Avila, Alain Núnez and Victor Mojica. I also express my gratitude to all my close relatives, especially my dear Robert, Nubia, and Ricardo for their permanent support.

Most importantly, I would like to express my endless gratitude to my love Paty, you are all that gives sense to my life. Thanks for bearing very close to me in this stage of my life, during which I worked much harder than usually.

Finally, yet the other half of my heart: my parents, both in heaven, for all their love, hard work, and sacrifice, which I treasure in my heart and have always set an example for me.

DEDICATION

◊

To my beloved mother, I can only imagine you riding on your white horse, no saddle, the wind blowing your angel's hair and smiling. I can only imagine you on heaven watching and taking care of me.

◊

Paty, my dear love, we made it!! This I promised you. You taught me the really meaning of $(\sin x)^2 + (\cos x)^2 = 1$. I dedicate this thesis to you, to your ∞ love.

◊

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Abbreviations

BCS	Base case scenario
BTX	Benzene – Toluene – Xylene
C2	Ethane
C3	Propane
C4	Butane
C4=	Butylene
CDU	Atmospheric crude distillation unit
CB	Crude blend
CM	Crude management
CR	Catalytic reforming
DB	Debutanized naphtha
DBU	Debutanizer columns
DC	Delayed coking
DMO	Demetallized oil
DMOH	Demetallized hydrotreated oil
DR	Distributed recursion
DRS	Demand reduction scenario
FCC	Fluid catalytic cracking
FB	Fuel blending
HCN	Heavy cracked naphtha
HDT	Hydrotreating
HN	Heavy naphtha
HT	Hydrotreating
HVGO	Heavy vacuum gas oil
iC4	Isobutane

idiesel	Intermediate streams to be blended to produced Diesel
ifuel	Intermediate streams to be blended to produced Fuel-oil
ijet	Intermediate streams to be blended to produced Jet
inaphta	Intermediate streams to be blended to produced Naphta
IRPC	Integrated refinery – petrochemical complex
LB	Lower bound on the solution value of **P**
LCN	Light cracked naphtha
LCO	Light cycle oil
LD	Lagrangean decomposition
LDS	Logistic disruption scenario
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
MIQCQP	Mixed-integer quadratically constrained quadratic program
MSS	Multi-start strategy
MMUSD	Million USD
NLP	Nonlinear programming
OBBT	Optimality-based bound tightening
PTQ	Petrochemical
QCQP	Quadratically constrained quadratic program
RB	Refinery and fuel blending
RC	Reduced crude
REF	Refinery
RON	Research octane number
RPB	Refinery petrochemical and fuel blending
RVP	Reid vapor pressure
S	Total number of subproblems in each spatial Lagrangean decomposition scheme
SLP	Successive linear programming

SQP	Sequential quadratic programming
TAN	Total acid number
UB	Upper bound on the solution value of **P**
VDU	Vacuum distillation unit
VN	Virgin naphtha
VR	Vacuum residue
WRPS	Without integration refinery-petrochemical scenario

Sets

BL_m	Set of index pairs (i, j) mapping bilinear terms $x_i x_j$ participating in function f_m
CL	Clusters
DC	Domestic crude oils produced by different oil fields in Colombia
E^{CB}	Crude blends produced by crude management section, linking to REF
Ein^i	All linking streams belonging to subproblem i, outgoing from other subproblems and entering to subproblem i
Ein^i_j	Entering linking streams belonging to subproblem i, outgoing from subproblem j
$Eout^i$	All linking streams belonging to subproblem i, outgoing from subproblem i and entering to other subproblems
$Eout^i_j$	Outgoing linking streams belonging to subproblem i, entering subproblem j
E^{PB}	Linking streams from PTQ to FB
E^{PR}	Linking streams from PTQ to RB
E^{RB}	Linking refined streams from REF to FB
E^{RP}	Linking refined streams from REF to PTQ
IC	Imported crudes fed to the studied Colombian IRPC
MT^i	Set of all incoming and outgoing transportation modes of subproblem i
PE^e	Properties of the intermediate stream e, between two subproblems.
PE^{pr}	Properties of the product pr, outgoing from a subproblem.
PE^{rw}	Properties of the raw material rw, entering to a subproblem.
PI_e	Subset of properties to consider for the stream e, entering a unit
PI_u	Subset of properties to consider for the feed stream of unit u

PL	Pipeline transportation system
$PO_{u,s}$	Subset of properties to consider for outlet stream s of unit u
PR^i	Product streams of subproblem i sold to external market
P^V	Properties blended on a volume base
P^{VF}	Volume fractions of all components in all streams
RW^i	Raw material streams of subproblem i
S^i_{int}	Internal streams of subproblem i
S^i	Set of all streams of subproblem i
SO_u	Subset of outlet streams from unit u
U^{BLD}	Fuel-blending units
U^{CB}	Units that receive or deliver crude blends
U^{CDU}	Atmospheric crude distillation units
U^{CM}	Crude management units
U^{CMX}	Crude mixing tanks
U^{CONV}	Conversion units
U^{CR}	Crude fractionation units
U^{DBU}	Debutanizer columns
$UDBU_u$	Subset of DBUs connected to atmospheric unit u
$UEin^i$	Units belonging to subproblem i and receiving linking streams from other subproblems
$UEin^i_j$	Units belonging to subproblem i and receiving linking streams from subproblems j
$UEout^i$	Units belonging to subproblem i and delivering linking streams to other subproblems
$UEout^i_j$	Units belonging to subproblem i and delivering linking streams to subproblem j.
U^i	Units belonging subproblem i.
$UInt^i$	Internal units belonging to subproblem i different to $UEout^i$, $UEin^i$, U^{PR} and U^{RW}
UO_u	Subset of downstream units connected to unit u
U^{PR}	Subset of units that produce sellable products
UPR_{mt}	Raw-material receiving units served by transportation mode mt

UPR_{mt}^{i}	Subset of all product streams pr, from product unit upr, in subproblem i, being carried through transportation mode mt
UPR_{pr}^{i}	Subset of all units producing product pr, in subproblem i, regardless of the transportation mode used to move it out
U^{PTQ}	Petrochemical units
U^{REF}	Refinery units
U^{RW}	Units receiving raw materials
URW_{mt}^{i}	Subset of all raw material streams rw, to raw material unit urw, in subproblem i, being carried through transportation mode mt
URW_{rw}^{i}	Subset of all units receiving raw material rw, regardless of the transportation mode used to move it in
US_u	Subset of upstream units connected to unit u
U^{VDU}	Vacuum distillation units
$UVDU_u$	Subset of VDUs connected to atmospheric unit u
UW_w	Virtual units of real unit w
V_i	Set of complicating variables of subproblem i
W^{ME}	Real units with mutually exclusive operating modes
X_{ij}	Set of indexes of the complicating variables shared by subproblems i and $j > i$

Indices

e	Intermediate streams between subproblems
mt	Transport modes
p	Stream properties
rw	Raw materials from external market
pr	Final products to external market
s	Process streams
u	Logical process units
w	Real processing units

Parameters

α_v^{ij}	Maximal step-size for linking variables v of the subproblems i and $j > i$
α_u	Selling price for product from unit u. (\$/bbl or \$/ton)
α_{pr}	Selling price for product *pr* (\$/bbl or \$/ton)
β_{rw}	Purchasing cost for raw material rw (\$/bbl, \$/ton)
$\beta_{u,s}$	Purchasing cost for raw material s coming from unit u. (\$/bbl, \$/ton)
γ_{mt}	Transportation cost of transport mode mt. (\$/bbl)
$\psi_{u,p}$	Cost for unit u operating with feedstock quality p (\$/bbl)
λ_j	Relaxation score for the partitioned variable x_j
$\lambda^{\mathrm{L}}, \lambda^{\mathrm{U}}$	Lower and upper bounds for Lagrange multiplier λ
Δ_v^{ij}	Trust-region radius for linking variables v of the subproblems i and $j > i$
ρ_{ij}	Normalized gap between the bilinear term $x_i x_j$ and its relaxation w_{ij}
τ	Threshold for refining the partition size in piecewise McCormick relaxation
$\phi_{u',s,u}$	Cost for unit u operating with feedstock stream s from unit u'. (\$/bbl)
ω_u	Operating cost for unit u. (\$/bbl)
a_{ijm}	Scalar multiplying bilinear term $x_i x_j$ in function f_m
$a_{u,s}$	Coefficient for predicting flow of output stream s from conversion unit u
B_m	Vector multiplying variable x in function f_m
$b_{u,s,p}$	Coefficient for predicting flow of output stream s from unit u based on input property p
C_m	Vector multiplying variable y in function f_m
$c_{u,s,p}$	Coefficient for predicting property p of output stream s from conversion unit u
d_m	Constant term in function f_m
$d_{u,s,p',p}$	Coefficient for predicting property p of output stream s from unit u based on property p'
cap_w^L, cap_w^U	Lower and upper bounds on processing capacity for real unit w (kbbl/day)
N_j	Partition size for variable x_j
N^0, N^U	Initial and maximal partition size in piecewise McCormick relaxation
$ps_{u,s}$	Property of raw-material stream s going into unit u
$prop_{vf,u,s,p}$	Crude assay characterization for the straight-run properties from the CDU

r_{ij}	Gap between the bilinear term $x_i x_j$ and its relaxation w_{ij}
sf_w	Service factor for the real unit w (between 0 and 1)
$x^{\mathrm{L}}, x^{\mathrm{U}}$	Lower and upper bounds for variable x
x_{jn}^{L}, x_{jn}^{U}	Lower and upper bounds for the partitioned variable x_j in the partition n
$yield_{vf,u,s}$	Crude assay yields for the straight-runs from CDU

Binary variables

y	Binary variable indicating the selection of operating conditions
y^i	Binary variable belonging to subproblem i
$y_{w,u}$	Binary variable indicating real unit w is operating in the mode associated to unit u
y_{jn}	Binary variable selecting partition n in the variable x_j

Continuous positive variables

$cost^i$	Cost of subproblem i for buying linking streams from other subproblems
$cost\_lgstc^i$	Cost of subproblem i for transportation of raw materials and products
$cost\_rw^i$	Cost of subproblem i for buying raw materials from external market
$PF_{u,p}$	Value of property p in the input stream to unit u
$PF_{e,p}^i$	Value of property p in stream e, outgoing from or incoming to subproblem i
$PS_{u,s,p}$	Value of property p for the outlet stream s from unit u
QF_e^i	Volume flowrate of stream e, outgoing from or incoming to subproblem i (kbbl/day).
QF_u	Flowrate into unit u (kbbl/day)
$QS_{u,s}$	Flowrate of outlet stream s from unit u (kbbl/day)
QT_{mt}	Total flowrate delivered by transportation mode mt (kbbl/day)
QT_{mt}^i	Total flowrate through transportation mode $mt \in MT^i$
$Q_{u',s,u}$	Flowrate of the outlet stream s from unit u' heading for unit u (kbbl/day)
$revenue^i$	Revenue of subproblem i for selling linking streams to other subproblems
$revenue\_pr^i$	Revenue of subproblem i for selling product streams to external market
$TotalCost$	Total cost of problem **P** for buying raw materials, transportation of raw materials and products, and operational costs.

$totalcost^i$	Total cost of subproblem i for buying raw materials, incoming linking streams, transportation of raw materials and products, and operational costs
$TotalRevenue$	Revenue of problem **P** for selling products of the IRPC
$totalrevenue^i$	Revenue of subproblem i for selling product and outgoing linking streams
v^i	Relaxation of the bilinear terms
w_{ij}	Linearization of the bilinear term $x_i x_j$
x	Flowrates and stream properties
x^i	Flowrates and streams properties belonging to subproblem i
$\hat{x}_{ijn}$	Aggregated variable for x_i present in the bilinear term $x_i x_j$ inside the partition n
$\hat{x}_{jn}$	Value of the partitioned variable x_j inside the partition n
x_v^i	Linking variables v of subproblem i (e.g. $QF_e^i, PF_{e,p}^i$)

Free variables

λ_e^{ij}	Lagrange multiplier for linking flowrate of stream e between subproblems i and j
$\lambda_{e,p}^{ij}$	Lagrange multiplier for linking property p of stream e between subproblems i and j
λ_v^{ij}	Lagrange multiplier for linking variable v between subproblems i and j
η	Auxiliary cost variable in Lagrangean dual problem $\mathbf{DP^K}$
$profit$	Profit obtained by the whole IRPC, maximized by problem **P**
$profit^i$	Profit obtained by subproblem i
z^*	Optimal solution value for **P**
z_λ^{LR}	optimal solution value for the Lagrangean relaxation $\mathbf{LR_\lambda}$
z^R	Optimal solution value for **PR**
z^{PF}	Optimal solution value for **PF**
$z_\lambda^{i,\mathrm{LD}}$	Optimal solution value for the subproblem i corresponding to $\mathbf{LD_\lambda^i}$
$z^{\mathrm{DP},K}$	optimal solution value for the Lagrangean dual problem $\mathbf{DP^K}$

Chapter 1 INTRODUCTION

1.1 Motivation

Integrated operations of petrochemical plants and crude oil refineries are more resilient to fluctuations in the hydrocarbons market compared to independent businesses for petrochemical commodities and fuels production (Al-Qahtani and Elkamel, 2010, 2009, 2008; Jia and Ierapetritou, 2004; Ketabchi et al., 2019; Leiras et al., 2010; Méndez et al., 2006; Nasr et al., 2011). Such integrated refinery-petrochemical complexes (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 by adding high-octane components,

supplying a fraction of the hydrogen needed by hydrotreating processes (Li et al., 2006). 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 presents many challenges due to the complexity of the process network and intermediate streams connectivity, the large variety of fuels and petrochemical products, the fulfilment of fuel specifications, and the market fluctuations. Crude unloading and blending, crude separation trains and conversion processes, fuel blending, and petrochemical production all need to be coordinated for maximizing the overall benefit of an IRPC, instead of optimizing these operations separately (Jia and Ierapetritou, 2004; Méndez et al., 2006; Nasr et al., 2011). 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 fuel 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 fuel demand resulting from a petrochemical commodities boom (International Energy Agency, 2019; WEO/IEA, 2016). This inherent complexity and uncertainty call for advanced tools to assist with the decision-making.

1.2 Research questions and thesis aims

The IRPC short-term planning problem can be formulated as a large-scale mixed-integer quadratically constrained quadratic program (MIQCQP), whereby discrete decisions select operating modes for the process units and the entire process network is represented by input-output relationships based on bilinear expressions describing yields and stream properties, pooling equations, fuel blending indices and cost indicators. A typical instance of such MIQCQP may contain several thousand equality constraints with tens of thousands of bilinear

terms and hundreds of binary variables. Local optimisation solvers fail to solve such large-scale MIQCQPs reliably due to their nonconvex nature, and even determining a feasible solution with these solvers is challenging. State-of-the-art global solvers, on the other hand, can determine feasible solutions of these MIQCQPs, but they struggle to close the optimality gap so these feasible solutions may end up being suboptimal by a large margin. By and large, short-term planning of real-life IRPC would therefore appear to be intractable to global optimality when tackled in a monolithic way.

The main research objective of the thesis is **to devise decomposition algorithms that can exploit an IRPC's physical structure to determine near-optimal solutions within a tractable computational runtime of several hours**.

The specific research aims are as follows:

- To derive the mathematical model for the optimal planning of a real-life IRPC and its associated supply chain and to devise realistic short-term planning scenarios.
- To test the effectiveness of state-of-the-art deterministic global solvers at bracketing the optimal solutions of these IRPC planning scenarios and at identifying good feasible solutions.
- To devise algorithms that decompose the IRPC into clusters or sections to reduce the optimality gap and determine near-globally optimal solutions.
- To analyse in detail the strengths and weaknesses of these decomposition algorithms.

1.3 Thesis outline

Leading on from the current chapter outlining the motivation and broad aims of the thesis, the remaining chapters are structured as follows. Chapter 2 reviews the literature about the main algorithmic strategies developed to tackle short-term IRPC planning problems over the past few

decades, from the onset of linear programming and successive linear programming to modern algorithmic advances for tackling nonconvex and mixed-integer programming models.

The novel algorithmic developments in the thesis are motivated by an industrial problem arising in the short-term planning of an existing IRPC in Colombia. Chapter 3 introduces this IRPC and details the corresponding short-term planning optimisation formulation, as well as four planning scenarios for benchmarking solution approaches. Formulation of relaxation of bilinear and trilinear terms and addition case study data are provided in Appendix B and Appendix C, respectively.

Chapter 4 presents the first main clustering decomposition approach contribution of the thesis, a clustering decomposition approach tailored to global optimisation of large-scale MIQCQPs, which decomposes a large process network into smaller clusters according to their functionality. Then inside each cluster, a MILP relaxation based on piecewise McCormick envelopes is solved, which dynamically partitions the variables that belong to the cluster and their domains are reduced through optimality-based bound tightening. These algorithmic developments and results in are based on the following peer-reviewed open-access journal article:

Uribe-Rodriguez, A., Castro, P.M., Guillén-Gosálbez, G., & Chachuat, B. (2020) Global optimization of large-scale MIQCQPs via cluster decomposition: Application to short-term planning of an integrated refinery-petrochemical complex. Computers & Chemical Engineering, 140, 106883. DOI: https://doi.org/10.1016/j.compchemeng.2020.106883

Further results are presented in Appendix D, Appendix E and Appendix F.

Chapter 5 presents the second main contribution of thesis, a tailored spatial Lagrangean decomposition algorithm, which subdivides an IPRC into two, three or four sections, namely crude management, refining, petrochemical operations, and fuel blending. This decomposition algorithm iterates between the solution of smaller optimisation subproblems for each section in

order to bracket the MIQCQP global solution value and determine good feasible solutions. These algorithmic developments and results build heavily on the following peer-reviewed open-access journal article:

Uribe-Rodriguez, A., Castro, P.M., Guillén-Gosálbez, G., & Chachuat, B. (2023) Assessment of Lagrangean decomposition for short-term planning of integrated refinery-petrochemical operations. Computers & Chemical Engineering, 174, 108229. DOI: https://doi.org/10.1016/j.compchemeng.2023.108229

It is complemented by Appendix G with further details about the subproblem formulations.

Lastly, key thesis conclusions, as well as possible directions for future work, are discussed in Chapter 6. This is complemented with preliminary results on the normalized multiparametric disaggregation technique (NMDT) applied to the clustering decomposition in Appendix H.

Chapter 2 LITERATURE REVIEW

2.1 Historical developments

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 optimisation, even with good initialization and tuning. SLP applies first-order Taylor expansion to linearize the nonlinear equations, with a good performance depending on the selection of an appropriate reference point. Advantages of DR and SLP are favourable 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).

Models of crude distillation units (CDUs) also have a large impact on the predictive capability of the planning model, since CDUs provide all the intermediate streams for further processing in the downstream units and fuel blending. In LP/DR/SLP formulations, CDUs and process units are modelled based on fixed yield representations. As a result, the predictions from

LP/DR/SLP formulations under/overestimate yields and stream properties, with the errors propagating to the pooling equations, process models, fuel blending and cost functions. Thus, formulating a nonlinear programming (NLP) model for a subset of or the whole process network, can improve the solution accuracy by enhancing the prediction of yields and outlet stream properties from refining and petrochemical units.

In the case of CDUs, the laboratory crude oil assay can be 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 (Kelly et al., 2014; Zhang et al., 2001); micro-cuts (Menezes et al., 2013); empirical correlations (Wenkai et al., 2007); surrogate models using polynomial approximations (López et al., 2012, 2013) and short-cut methods (Alattas et al., 2011). This latter approach (Alattas et al., 2011) incorporates the CDU short-cut model in a mixed-integer nonlinear programming (MINLP) planning model for determining crude oil sequencing, changeovers and processing time.

Likewise, the accuracy of process unit models can be improved by formulating empirical correlations for other type of units. Li et al. (2005) integrated the CDU and FCC into a small-scale refinery planning problem. Surrogate models based on rigorous simulations were used by Guerra and Le Roux (2011a) to optimize the same process network. Alhajri et al. (2008) employed polynomial approximation surrogates for the catalytic reformer, FCC, hydrotreating and hydrocracking units, within a more complex process topology. On the other hand, pooling equations representing mixing operations with bilinear and trilinear terms and nonlinear blending rules to compute fuel specifications, improve the predictive capability (Guerra et al., 2010). Siamizade (2019) proposed a MINLP refinery planning model based on commercial empirical correlations for the process units (Baird, 1987) and the Geddes fractionation-index

method for the CDU (Geddes, 1958; Gilbert et al., 1966). The former manages operating conditions such as reaction temperatures and severities as continuous decision variables, whilst the latter uses binary variables to determine whether a component is in the stripping or rectifying section.

Although these modelling frameworks provide an accurate description of the complex network of processes in an IRPC, their formulation leads to large-scale MINLP models that are challenging to solve (Neiro and Pinto, 2004). Nevertheless, researchers have achieved better solutions than those obtained with previous models and claimed significant profit improvements.

Moro et al. (1998) increased profit for a real-world application by about $6 MM/year, in a facility producing three grades of diesel, considering nonlinear models for the heavy diesel hydrotreating and blending, delay coking and fluid catalytic cracking (FCC) units. 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 optimisation 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.

2.2 Main solution methods

Although they enable more accurate predictions, nonlinear models also introduce nonconvexities into refinery planning models. Local search algorithms have long been applied to solve NLP models of industrial-sized refinery planning problems, including successive linear programming (SLP) or successive quadratic programming (SQP) (ASPEN Technology Inc,

2010; Baker and Lasdon, 1985; Bonner & Moore, 1979; Haverly, 2015; Kutz et al., 2014) even finding a feasible solution can prove challenging with this approach.

A multi-start strategy (MSS) by passing different initial points to a fast and robust local NLP solver can sometimes overcome such feasibility and local optimality issues (Guerra et al., 2010; Guerra and Le Roux, 2011b), but it does not compute a dual bound. And it does not offer any mathematical guarantees of reaching a global optimum either. Andrade et al. (2016) applied MSS to bilinear programming models and computed a dual bound by solving a linear programming (LP) relaxation based on the McCormick envelopes (McCormick, 1976), yet no procedure was implemented to refine this dual bound.

Andrade et al. (2016) presented a heuristic approach based on solving LP relaxations. However, this approach does not guarantee global optimality either. 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 optimisation (DGO) algorithm by aggregating the decision variables into three groups: partitioning variables, which are discretized into several intervals and are used to formulate a linear relaxation of the bilinear terms; bound contracted variables are also partitioned but their bounds are updated following an interval elimination procedure relying on the solution of a MILP problem for each variable bound, selecting one variable at a time, and branch and bound variables, which are used to split the problem in two to apply a branch and bound search. The authors claimed that if the selected partitioned variable is different from the bound contracted variable, then the algorithm tends to reach a tighter lower bound, since the algorithm solves a MILP lower bound by partitioning the feasible space of the partitioned and bound contracted variables. However, a heuristic should be applied to classify the decision variables.

Castillo Castillo et al. (2017, 2018) considered a mixed-integer quadratically constrained quadratic program (MIQCQP) formulation of a large-scale refinery planning for profit maximization and solved it to global optimality with a DGO algorithm. This two-stage solution strategy proved to be competitive with the commercial solvers BARON (Sahinidis, 2004) and ANTIGONE (Misener and Floudas, 2014a). In the first stage, a mixed-integer linear programming (MILP) relaxation of the MIQCQP model was derived from piecewise McCormick envelopes (Castro, 2015; Castro et al., 2021; Gounaris et al., 2009; Karuppiah and Grossmann, 2006; McCormick, 1976; Misener et al., 2011; Wicaksono and Karimi, 2008) or from the multiparametric disaggregation technique (Andrade et al., 2018; Castro, 2016; Castro and Grossmann, 2014; Kolodziej et al., 2013; Teles et al., 2013). A quadratically constrained quadratic program (QCQP) model obtained by fixing the binary variables in the MIQCQP model to the values from the solution of the MILP relaxation, was then solved to local optimality in the second stage. In cases where fixing the binaries did not compromise feasibility, this decomposition procedure could find solutions that were close to the global optimum. Such an iterative procedure for reducing the dual gap works by increasing the number of intervals in the partition for one of the variables in every bilinear term. The algorithm also employed optimality-based bound tightening (OBBT) (Castro and Grossmann, 2014; Puranik and Sahinidis, 2017) for reducing the domain of nonlinearly appearing variables. A drawback with this approach was that the first-stage MILP relaxation could already be computationally demanding to solve with just a few intervals in the partition, leading to high dual gaps and poor initial points for the second-stage QCQP model.

(Zhang et al., 2021) proposed a methodology to address refinery planning operations under uncertainty. The framework considers the development of surrogate models for the process units, aiming to obtain low-complexity, accurate and amenable models for process optimisation (Wilson and Sahinidis, 2017), the uncertainty is given by product demands and sale price, which

are represented by interval and interval plus ellipsoidal sets, leading to a nonconvex MINLP problem, where the nonconvexities are present in the bilinear and quadratic terms to represent the process units. In addition, convex nonlinear expressions such as a polynomial function to represent the crude distillation unit and other nonlinearities arising from the uncertainty management are considered in the MINLP problem. Following the same idea of a two-stage strategy, a convex MINLP – NLP approach is applied by relaxing the nonconvex terms applying the enhanced normalized multiparametric technique (Andrade et al., 2018) and leveraging the solution of the convex MINLP problem to ANTIGONE. Moreover, OBBT is performed to the variables presented in the bilinear terms in order to enhance the quality of the relaxed model.

Regarding model integration and coordination of IRPC operations, little research work has employed DGO to date. Li et al. (2016) obtained a mixed-integer quadratically constrained quadratic program (MIQCQP) from the data-driven approach to optimize an IRPC; with second-order polynomial correlations as submodels for various parts of a large-scale nonconvex MINLP model. This model was solved using ANTIGONE (Misener and Floudas, 2014a) and led to a profit improvement of 30–65% compared to current operation. A multiperiod extension to this problem was recently addressed by Demirhan et al. (2020). Zhao et al. (2017) presented an integrated model for fuels and olefins transformation that was solved by Lagrangean decomposition. They reported a profit increase between 14 – 53%, compared to a sequential optimisation process (first the refinery and then the ethylene unit). Siamizade (2019) solved the same model based on commercial empirical correlations and the Geddes fractionation-index method by using the state-of-the-art global optimizer BARON (Sahinidis, 2004), but they omitted to report the optimality gaps at termination.

A hierarchical approach to solve an industrial multiperiod IRPC was proposed by Zhang et al. (2022). The refinery and petrochemical processes are represented by 24 and 15 process units respectively. Crude procurement and unloading operations were not considered in the model

formulation, as well as crude blending, processing one crude at a time. The nonconvexities arises from the yields prediction and blending operations, leading to a large-scale MINLP model. Instead of solving this challenging problem upfront, it is proposed an iterative procedure, first solving one of the subproblems to optimality, then the optimal value of the intermediated streams to be exchange between the refinery and petrochemical is added as equality constraints to the other subproblem. Thus, the second subproblem is solved with extra constraints. If both subproblems converge, the optimal solution of the IRPC is given by the summation of the optimal objective function of each subproblem. Even though this decomposition decreases the complexity of the original MINLP, the authors reported that first solving the refinery and then the petrochemical was successfully applied to short-term planning scenarios. The hierarchical approach fails when the petrochemical is first solved and then the refinery for the same short-term planning instances. Moreover, medium-term planning scenarios did not converge for any sequence refinery – petrochemical or petrochemical – refinery. These findings highlighted that more investigation is required to improve the performance of Lagrange decomposition strategies to solve large-scale IRPC planning problems.

2.3 Conclusions

This literature review reveals that much progress has been made on solving short-term IRPC planning problems over the past few decades, especially in terms of leveraging algorithmic advances for tackling nonconvex and mixed-integer programming models. However, much remains to be done before real-life problems can be solved reliably to guaranteed global optimality in a daily routine. The focus of this thesis, therefore, shall be on the deterministic global optimisation of large-scale MIQCQP formulations. The main case study is inspired by an industrial problem arising in the short-term planning of an existing IRPC in Colombia. The

following Chapter 3 describes the IRPC and presents a formulation of the short-term planning model, before showcasing the performance of state-of-the-art global optimisation solvers through four planning scenarios.

Chapter 3 INTEGRATED REFINERY-PETROCHEMICAL OPERATIONS: CASE STUDY

3.1 Description of integrated refinery-petrochemical complex

The integrated refinery-petrochemical complex (IRPC) under consideration corresponds to one of the main facilities operating in the Colombian refining industry. It is composed of 60 industrial plants, represented by 125 models, and two tank farms, one for crude mixing and the other for fuel blending, consisting of 30 additional units.

This IRPC 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 3.1). 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 fuel blending operations.

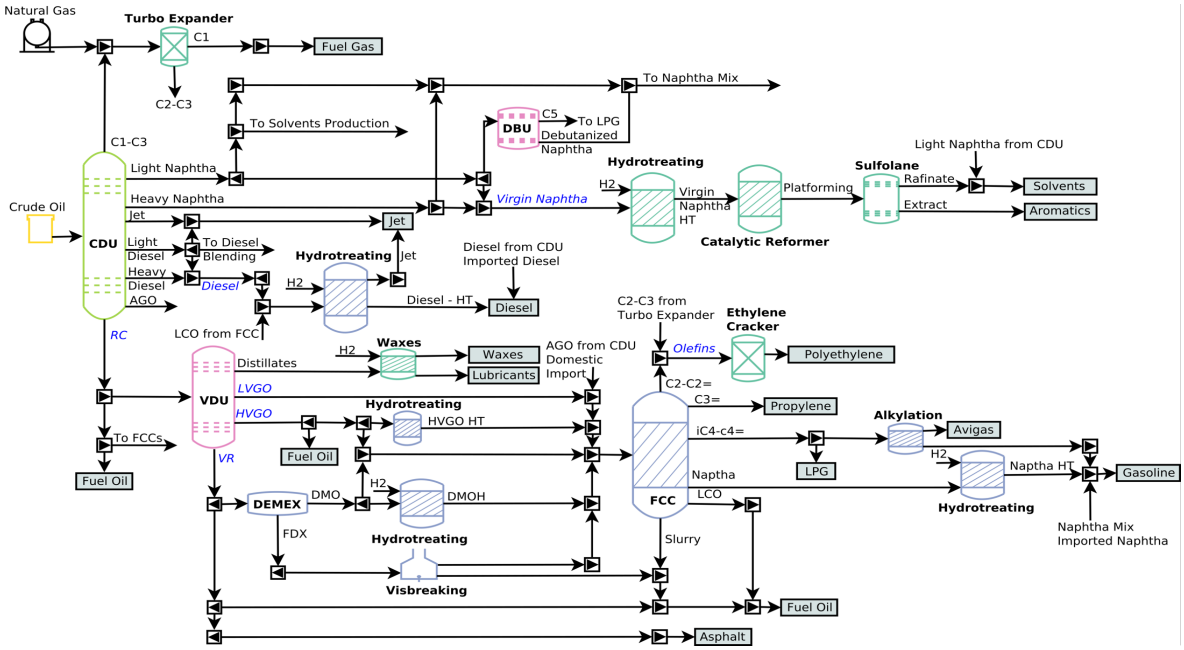


Figure 3.1. Diagram of the full integrated refining-petrochemical complex (IRPC).

The raw material to the IRPC (crude oil and refined products) is supplied by both domestic production and imports. We assume that 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 248 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 fuel blending. Demand is represented by 22 grades of fuels and petrochemical requirements imposed by the domestic and export markets. Meeting the demand for petrochemicals entails the

production of up to 27 commodities, divided into BTX, propylene, polyethylene, specialty solvents and waxes.

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.

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). Thus, the IRPC can be divided into four sections, which can be independent business units: crude management (CM), refinery (REF), petrochemical (PTQ), and fuel blending (FB). A brief description of each section is provided below.

3.1.1 Crude management

Crude management (CM) involves procurement, transportation and blending of crude oils to produce streams with suitable bulk properties, such as sulphur content, API gravity and total acid number (TAN), for feeding into the CDUs (Guyonnet et al., 2009; Oddsdottir et al., 2013; Zhang et al., 2012). Crude oil characterization provides insight into potential economic and operational benefits. The gravity and sulphur content determine the market price, as crudes with low gravity and high sulphur content are cheaper than their high-density and low-sulphur counterparts. Thus, the crude oil management problem involves a trade-off between the cost and quality of the blends.

Accounting for transportation enables a more realistic crude management operation. The logistic involves delivering batches of crude oil by pipelines and multimodal transport from oil fields, transport stations, and import ports to the refineries. As seen in Figure 3.2, supply is

given by domestic production of 17 different types of crude oil ($DC = \{DC1, \dots, DC17\}$) geographically distributed across 8 regions in the country ($R = \{R1, \dots, R8\}$). Domestic crude production ranges between 10 - 100 kbbl/day, for a total national production of 297 kbbl/day. A total of 7 imported crudes ($IC = \{IC1, \dots, IC7\}$), up to a maximum of 15 kbbl/day/crude, complete the market availability. Domestic and imported crudes are delivered in batches through pipelines ($PL = \{PL3, \dots, PL8\}$). Then, at the refinery, the 24 qualities of crude (specifications are shown in Table 3.1) are combined into 9 crude blends ($E^{\mathrm{CB}} = \{CB1, \dots, CB9\}$). For instance, $CB7$ is obtained from domestic crudes $DC1, DC4, DC7 - DC9, DC12, DC16 - DC17$ and imported crudes $IC1 - IC7$. Note that crude blend $CB9$ was excluded from Table 3.1, since as it is produced from crude blends $CB1 - CB7$.

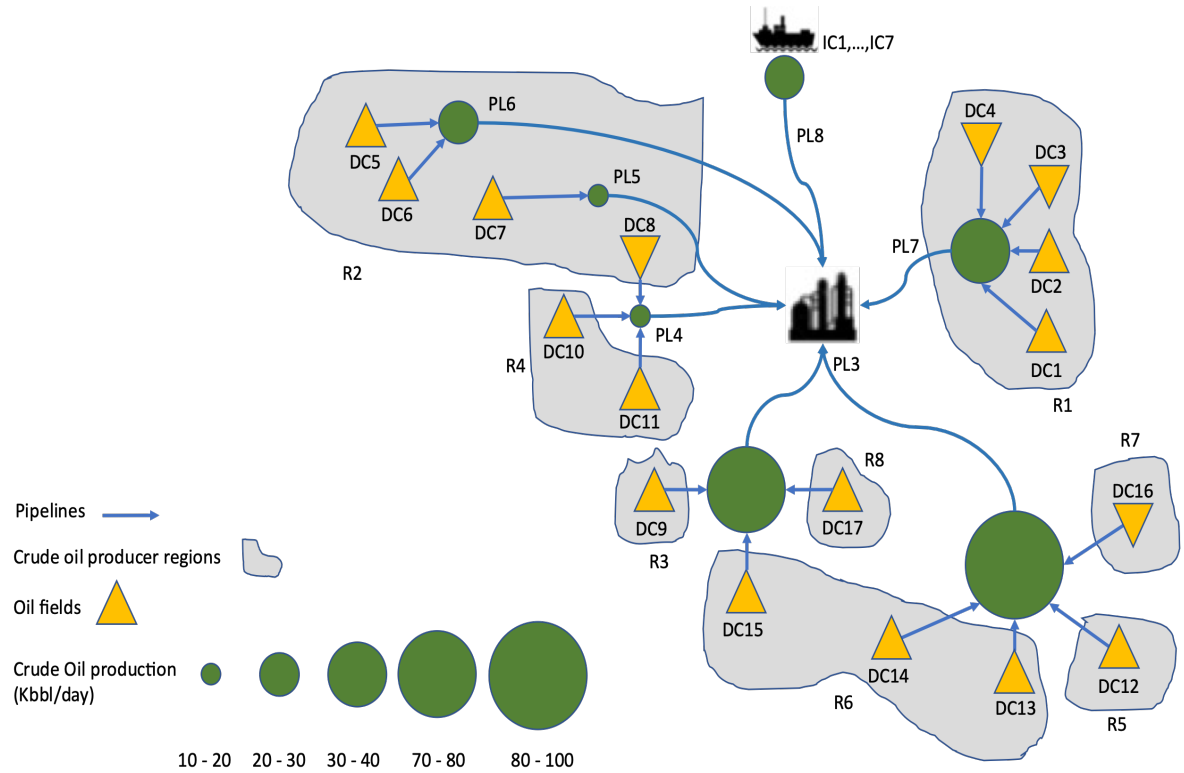


Figure 3.2. Crude oil production and pipeline transportation network.

Table 3.1. Bulk properties and prices of domestic and imported crudes, and their incorporation into crude blends.

Crude Oil	Crude Blends								Bulk properties			Supply	
	CB1	CB2	CB3	CB4	CB5	CB6	CB7	CB8	API	Sulphur (%wt)	TAN (mg KOH/g crude)	Price* (USD/bbl)	Available (kbbl/day)
DC1	x								24	1.218	1.135	39.50	2.1
DC2		x							29	0.515	0.097	49.70	31.1
DC3			x					x	32	0.812	0.168	49.68	3.4
DC4	x								20	1.934	0.528	35.43	2.2.
DC5					x				28	0.642	1.494	45.27	5.6
DC6				x					23	0.929	2.139	41.26	24.2
DC7	x								22	1.008	2.300	40.28	12.6
DC8	x								19	0.957	3.126	38.50	5.8
DC9	x								20	1.129	3.341	34.57	71.7
DC10						x			26	1.223	1.680	42.49	3.3.
DC11						x			23	1.239	2.642	40.13	3.7
DC12	x								19	1.848	0.122	40.86	32.0
DC13							x		44	0.306	0.093	52.00	24.1
DC14								x	45	0.048	0.070	52.11	26.7
DC15						x			24	0.984	0.468	42.78	13.5
DC16	x								18	1.140	0.137	30.77	16.5
DC17	x								20	1.139	2.381	35.08	18.9
IC1	x					x			39	0.156	0.629	52.23	15.0
IC2	x					x			39	0.921	0.060	52.30	15.0
IC3	x					x			29	0.246	0.590	49.28	15.0
IC4	x					x			29	0.690	1.266	49.17	15.0
IC5	x					x			29	0.605	0.470	49.15	15.0
IC6	x					x			40	0.482	0.043	52.37	15.0
IC7	x					x			34	0.158	0.605	50.64	15.0

*Low pre-pandemic crude oil prices scenario

3.1.1.1 Crude allocation

Crude allocation provides the feedstock to the crude distillation units. The network in Figure 3.3 illustrates how the 17 domestic crude oils $(DC1, \dots, DC17)$ are delivered to the refinery via 5 pipelines $(PL3, \dots, PL7)$. A system of 9 mixing tanks $(Tk1, \dots, Tk9)$ are then used to obtain homogeneous crude blends $(CB1, \dots, CB9)$ with given quality properties (specific gravity, sulphur content and total acid number) for further processing by the CDUs.

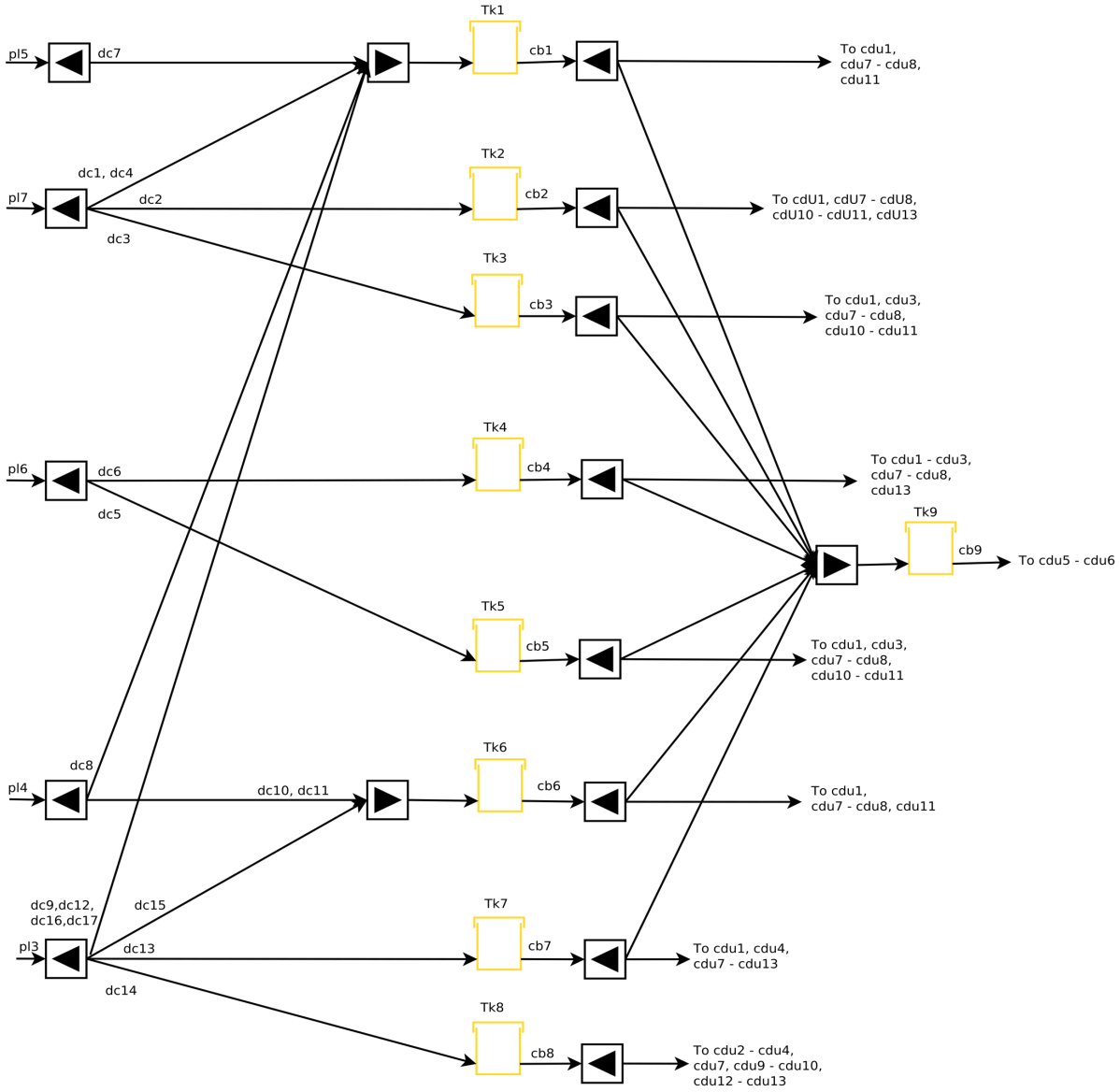


Figure 3.3. Crude allocation network.

3.1.2 Refinery

The refinery (REF) has a total capacity of 248 kbbl/day, distributed over 6 crude distillation units ($RCDU = \{RCDU1, \dots, RCDU6\}$) that can operate in different campaigns during the planning horizon. Each campaign is represented by a logical unit (CDU), leading to 13 such logical CDUs ($CDU = \{CDU1, \dots, CDU13\}$), each described by a specific set of constraints. As an example, $RCDU1$ has a processing capacity of 38 kbbl/day and can operate in 4 campaigns ($CDU1, \dots, CDU4$).

Table 3.2. Set of logical units associated to each distillation column and corresponding crude blends processed by each logical unit.

	RCDU1				RCDU2	RCDU3	RCDU4	RCDU5			RCDU6		
Blend/CDU	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
CB1	x						x	x		x	x		x
CB2	x			x			x	x	x	x	x	x	x
CB3		x	x	x			x		x	x		x	x
CB4	x		x				x	x		x	x		
CB5	x		x				x	x					x
CB6	x						x	x			**x**		
CB7	x						x	x			x		
CB8	x		x				x	x		x	x		
CB9					x	x							
Capacity (kbbl/day)	38				52	27	39	55			37		

The CDUs produce intermediate streams such as light ends (C1 – C3), light and heavy naphtha, jet fuel, light and heavy diesel, atmospheric gas oil and reduced crude, which are either processed by the refining downstream units or routed to fuel blending. Commodities such as alkylate and gasoil ($IR = \{Alkylate, GasOil\}$) can be imported as feedstock to certain refining units. The refinery also produces natural gas, ethane, olefins, virgin naphtha, and raw materials for petrochemical production.

3.1.2.1 Crude fractionation system

The crude fractionation system provides all the intermediate streams for further processing in the refinery, petrochemical and fuel blending units. This separation train comprises 6 units operating in a total of 13 operating modes (logic units $cdu1, \dots, cdu13$). For instance, the first distillation unit has four operating modes ($cdu1, \dots, cdu4$). 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 3.3). For example, $cdu4$ 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 3.1).

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

ADU	VDU1	VDU2	VDU3	VDU4	VDU5	VDU6	RC Pool	DBU1	DBU2
CDU1				x				X	
CDU2				x				X	
CDU3				x				X	
CDU4							x	X	
CDU5	x	x	x		x	x		X	
CDU6					x	x		X	
CDU7	x		x					X	
CDU8	x	x	x					X	X
CDU9	x		x				x	X	X
CDU10	x		x				x	X	X
CDU11	x	x	x						
CDU12	x		x				x	X	
CDU13	x		x				x	X	

As can be seen in Table 3.3, 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 $cdu11$, which goes to gasoline blending or petrochemicals production.

3.1.2.2 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 fuel blending (Figure 3.4).

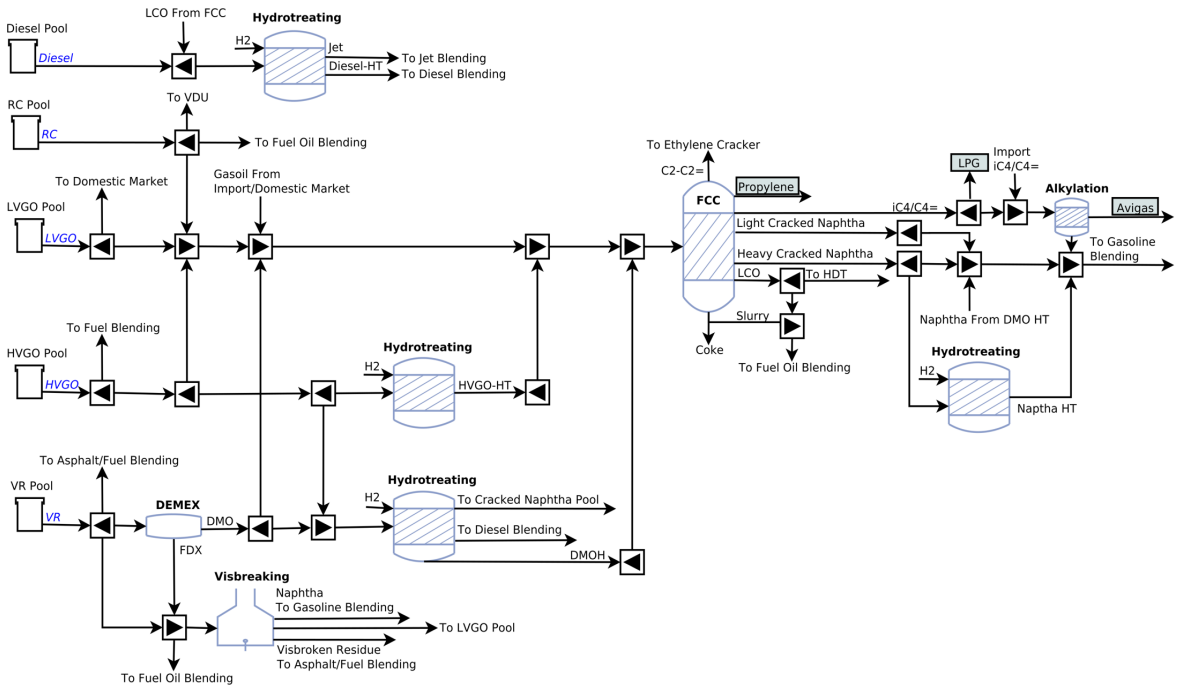


Figure 3.4. Diagram for refinery conversion.

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/fuel blending and for DEMEX, which reduces the metal content. Demetallized 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 demetallized 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.

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.

3.1.3 Petrochemical

The petrochemical (PTQ) plant transforms ethane, olefins, and virgin naphtha to obtain added value products such as polyethylene, propylene, benzene, and toluene. Several intermediate streams can be sold to the refinery for improving gasoline quality and specialty solvents production and supplying hydrogen to hydrotreating units. Relevant properties for these streams include their specific gravity, sulphur content, octane, Reid pressure vapor, and aromatics content.

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 3.5).

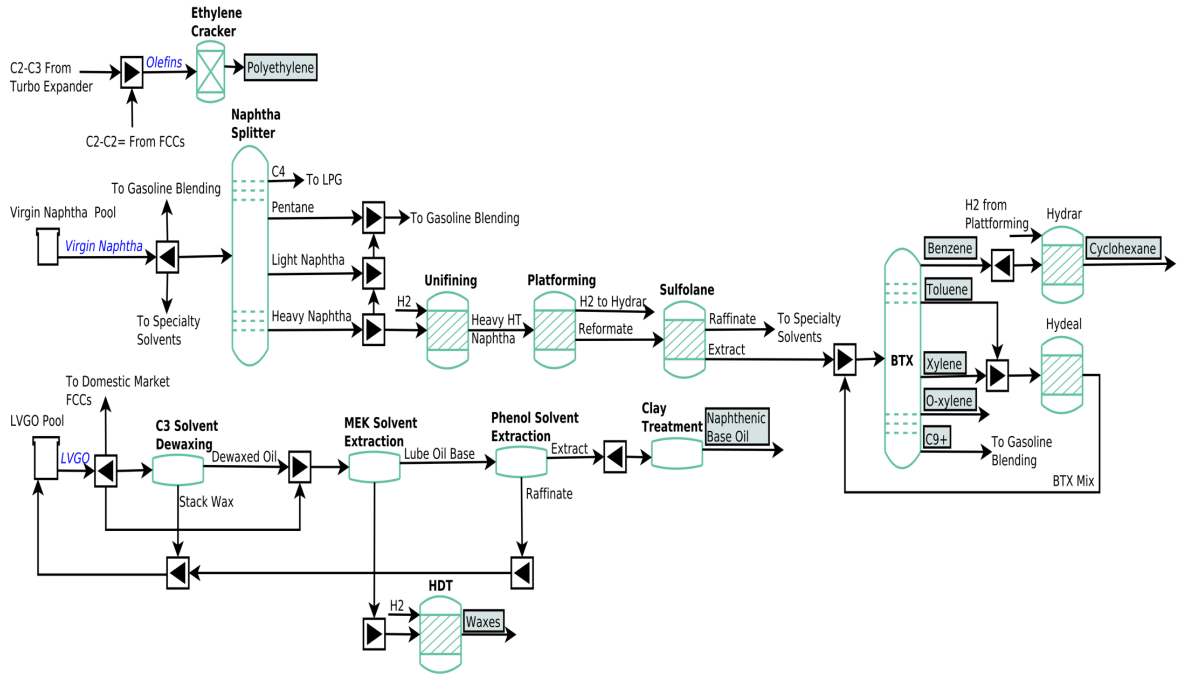


Figure 3.5. Diagram for petrochemical production.

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.

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.

3.1.4 Fuel Blending

To produce fuels (FB) with the required quality, the blenders can buy and receive intermediate process streams from REF and PTQ, complemented with refined products from the domestic and international markets. Overall, 88 refined streams can produce up to 22 grades of fuels (Table 3.4), with different quality specifications for each grade (Table 3.5). 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 3.4. Streams for fuel blending.

Fuel	LPG	Gasoline	Medium distillate	Fuel Oil	Asphalt	Total
Streams to blend	24	23	25	10	6	88
Products grade	3	7	6	4	2	22

Table 3.5. Property/quality specifications of the different fuels.

Fuel/Quality	Specific gravity	Sulphur content	Cetane number	RON	RVP	Viscosity
LPG	x					
Gasoline	x	x		x	x	
Medium distillate	x	x	x			
Fuel Oil	x	x				x
Asphalt	x	x				x

A total of 25 streams are blended into 6 grades of medium distillates, which must comply with quality constraints on specific gravity, sulphur content, and cetane number. Two grades of jet fuel and four grades of diesel are produced as illustrated in Figure 3.6.

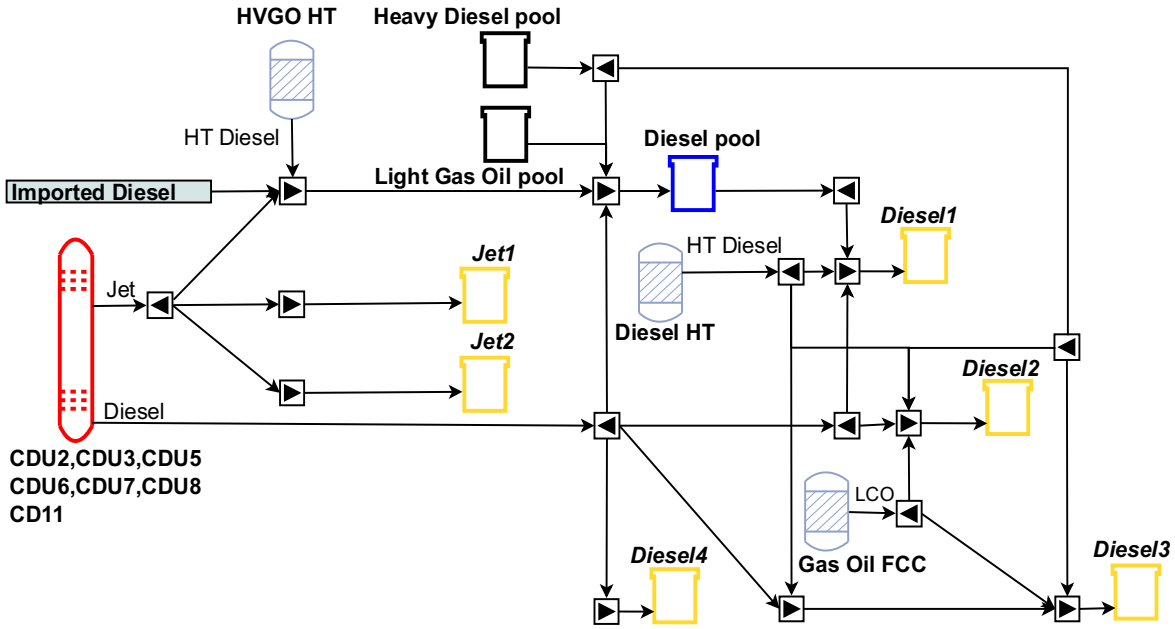


Figure 3.6. Medium distillate blending.

Diesel components from the CDUs provide one grade of diesel (Diesel4), while jet fuel components provide two grades of Jet fuel (Jet1 and Jet2). Jet fuel components are also routed to the diesel pre-mix tank (Diesel pool). This tank also receives imported diesel, and other

refinery streams such as HT diesel from the HVGO HT process, gas oil and heavy diesel pools. A second diesel grade with ultra-low sulphur (Diesel1) is obtained from hydrotreated diesel and diesel pre-mix blends. A fraction of the hydrotreated diesel is blended with heavy diesel from upstream refinery processes and LCO from FCC to produce these two diesel grades (Diesel2 and Diesel3).

Note that steady-state operations are assumed as opposed to multiperiod operation (Demirhan et al., 2020; Mouret et al., 2009; Neiro and Pinto, 2006).

3.2 Short-term planning optimisation formulation

The optimisation 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 fuel blending.
7. The routing for all 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.

An accurate mathematical representation of the Integrated refining and petrochemical complex (IRPC) is crucial for computational tools to provide accurate, quantitative decisions for short-term planning. This industrial-size problem can be formulated as a large-scale 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, fuel blending indices and cost indicators.

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

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

where z^* is the maximum profit, 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 such as high, medium, and low severity in fluid catalytic process.

The functions $f_m: \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$, with $m = 0$ (objective function), and with $m = 1,\dots,M$ (constraints of **P**) are quadratic in x and linear in y:

$$f_m(x,y) := \sum_{(i,j)\in BL_m} a_{ijm} x_i x_j + B_m x + C_m y + d_m,$$

where BL_m is an (i,j)-index set defining the bilinear terms $x_i x_j$ present in function m, parameters a_{ijm} and d_m are scalars, and B_m and C_m are row vectors.

A detailed mathematical formulation of the objective and constraint functions in the MIQCP model (**P**) is given next.

3.2.1 Generic process unit models

The topology representation for the IRPC is based on the modelling framework by Neiro and Pinto (2004). The network superstructure is built from unit models that include a mixer for the input streams to the unit and one splitter for each output stream (Figure 3.7). The topology for our IRPC model comprises 155 different types of units connected by intermediate streams. Recall that the IRPC is composed by 60 industrial plants, which can operate with exclusive or non-exclusive operating modes. Thus, a process unit u is modelled for each operating window of the real plant w (see Section 3.2.5).

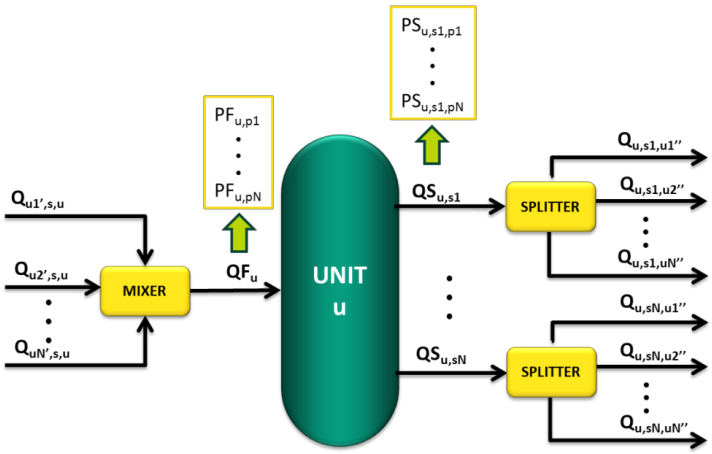


Figure 3.7. Diagram for a petrochemical production unit u.

Inputs to a unit u include the feedstock volume and properties, represented by variables QF_u and $PF_{u,p}$, respectively, and the operational conditions $y_{w,u}$ which can be non-exclusive or mutually exclusive campaigns. These inputs are used to predict the flowrate $QS_{u,s}$ of each outlet stream s from unit u and its corresponding properties $PS_{u,s,p}$. An outlet stream can be routed to other process units, fuel blending or tank farms to be sold, with variable $Q_{u,s,u'}$ giving the volumetric flowrate of stream s from unit u that is fed to unit u'. Next, SO_u denotes the subset of outlet streams from unit u, US_u the subset of upstream processes connected to u, and UO_u the subset of downstream units connected to u.

The balance equation (Eq. 3-1) states that the feed mixer to a process unit u collects all streams from the units u' that are connected to u. This balance does not apply to raw material receiving units $u \in U^{RW}$. The splitter at the outlet of unit u, divides the volumetric flowrate $QS_{u,s}$ over its connected downstream units $u' \in UO_u$, as stated in Eq. (3-2).

$$QF_u = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall u \in U \backslash U^{RW} \tag{3-1}$$

$$QS_{u,s} = \sum_{u' \in UO_u} Q_{u,s,u'} \quad \forall u \in U, s \in SO_u \tag{3-2}$$

The properties p associated with the feedstock to unit u are included in subset PI_u. Eq. (3-3) applies to those properties that are mixed on a volume base ($p \in P^V$). The balance for properties blended on a weight basis, e.g. sulphur content, is given by Eq. (3-4), where variables $PF_{u,SPG}$

and $PS_{u',s,SPG}$ give the specific gravity of the inlet stream to unit u and the outlet stream s from unit u', respectively. Notice that Eq. (3-4) features trilinear terms instead of the bilinear terms in Eq. (3-3).

$$QF_u PF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \ \forall u \in U, p \in PI_u \cap P^V \quad (3\text{-}3)$$

$$QF_u PF_{u,p} PF_{u,SPG} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \ \forall u \in U, p \in PI_u \backslash P^V \quad (3\text{-}4)$$

Note that Eq. (3-4) can be expressed in terms of bilinear terms, defining two auxiliary variables QF_u^{Aux} and $Q_{u',s,u}^{Aux}$ that represent mass flows as indicated in Eqs. (3-5) and (3-6). The mixing rule on a mass basis is then calculated by Eq. (3-7).

$$QF_u^{Aux} = QF_u PF_{u,SPG} \ \forall u \in U \quad (3\text{-}5)$$

$$Q_{u',s,u}^{Aux} = Q_{u',s,u} PS_{u',s,SPG} \ \forall (u', u) \in U, s \in SO_{u'} \quad (3\text{-}6)$$

$$QF_u^{Aux} PF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u}^{Aux} PS_{u',s,p} \ \forall u \in U, p \in PI_u \backslash P^V \quad (3\text{-}7)$$

3.2.2 Crude allocation model

The total flowrate and crude composition are propagated from the crude allocation to the crude distillation processing units. The set P^{VF} stand for the volumetric composition of different crudes going into a crude mixing tank. Then, the sum over all the volumetric compositions $PS_{u,s,p}$ for every outlet stream s from crude mix tank $u \in U^{CMX}$, must be equal to one (Eq. 3-8). The same goes for the volumetric compositions $PF_{u,p}$ of the feedstock of a CDU (Eq. 3-9).

$$\sum_p PS_{u,s,p} = 1 \ \forall u \in U^{CMX}, s \in SO_u, p \in PO_{u,s} \cap P^{VF} \quad (3\text{-}8)$$

$$\sum_p PF_{u,p} = 1 \ \forall u \in U^{CDU}, p \in PI_u \cap P^{VF} \quad (3\text{-}9)$$

3.2.3 Crude fractionation model

The crude fractionation units $u \in U^{CR}$ include atmospheric distillation U^{CDU}, vacuum separation U^{VDU} and debutanizer U^{DBU} units. The subset $UVDU_u$ indicates the connectivity of

the CDU u with VDUs, and $UDBU_u$ the connectivity with the DBUs. The crude assay characterization provides the yields, $yield_{vf,u,s}$ and outlet stream properties, $prop_{vf,u,s,p}$ due to the fractionation of crude vf in unit u. The flowrates and properties of the outlet streams from the CDUs are computed by Eqs. (3-10) - (3-11), where subset $PO_{u,s}$ holds the properties of stream s from unit u.

$$QS_{u,s} = QF_u \sum_{vf \in P^{VF}} yield_{vf,u,s} PF_{u,vf} \quad \forall u \in U^{CDU}, s \in SO_u \tag{3-10}$$

$$QS_{u,s} PS_{u,s,p} = QF_u \sum_{vf \in P^{VF}} prop_{vf,u,s,p} PF_{u,vf} \quad \forall u \in U^{CDU}, s \in SO_u, p \in PO_{u,s} \tag{3-11}$$

The RC and light naphtha (LN) streams from the CDUs can be routed to more than one vacuum/debutanizer column. Because of this, the flowrate $QS_{u,s}$ and properties $PS_{u,s,p}$ need to be determined as a fraction of the total yield from the CDU, leading to Eqs. (3-12) to (3-16), where DB stands for debutanized naphtha. Note that Eqs. (3-13) to (3-16) can be expressed as bilinear terms if the denominator is moved to the left-hand side of the equation.

$$QF_u = \sum_{s \in SO_u} QS_{u,s} \quad \forall u \in U^{VDU} \cup U^{DBU} \tag{3-12}$$

$$QS_{u,s} = QF_u \frac{\sum_{vf \in P^{VF}} yield_{vf,u,s} PF_{u,vf}}{\sum_{u' \in U^{CDU}: u \in UVDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',RC} PF_{u',vf}} \quad \forall u \in U^{VDU}, s \in \{LVGO, HVGO, VR\} \tag{3-13}$$

$$QS_{u,s} = QF_u \frac{\sum_{vf \in P^{VF}} yield_{vf,u,s} PF_{u,vf}}{\sum_{u' \in U^{CDU}: u \in UBDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',LN} PF_{u',vf}} \quad \forall u \in U^{DBU}, s \in \{C5, DB\} \tag{3-14}$$

$$PS_{u,s,p} = \frac{\sum_{vf \in P^{VF}} prop_{vf,u,s} PF_{u,vf}}{\sum_{u' \in U^{CDU}: u \in UVDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',RC} PF_{u',vf}} \quad \forall u \in U^{VDU}, s \in \{LVGO, HVGO, VR\}, p \in PO_{u,s} \tag{3-15}$$

$$PS_{u,s} = \frac{\sum_{vf \in P^{VF}} prop_{vf,u,s} PF_{u,vf}}{\sum_{u' \in U^{CDU}: u \in UBDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',LN} PF_{u',vf}} \quad \forall u \in U^{DBU}, s \in \{C5, DB\}, p \in PO_{u,s} \tag{3-16}$$

In order to consider the operating conditions in the atmospheric or vacuum separation processes, swing-cuts are defined between the straight-runs heavy naphtha – jet, jet – light

atmospheric diesel, light – heavy atmospheric diesel, heavy atmospheric diesel – reduced crude, LVGO – HVGO and HVGO – VR. Because these hypothetical streams and their properties can be partially or totally routed to their light or heavy adjacent cut, they are modelled using the general mass and properties balance Eqs. 3-1 to 3-4.

3.2.4 Conversion unit models

The set U^{CONV} represents the conversion units. On the refinery side, it includes solvent extraction for the vacuum residue; fluid catalytic cracking for the LVGO, HVGO, DMO and DMOH; hydrotreating for the jet, diesel, naphtha, HVGO and DMO; and mid-thermal cracking for the vacuum and bottom residues from the solvent extraction units. On the petrochemical side, it includes straight-run naphtha fractionation; heavy naphtha hydrotreating; naphtha catalytic reforming; solvent extraction of aromatics with sulfolane; the separation column train for the BTX; the ethylene cracker; the specialty solvents units; and the wax production processes.

Conversion units are modelled as simple input – output relationships based on laboratory characterization for their feedstock and yields, pilot-plant runs and industrial test-runs. These expressions include bilinear terms to predict the yields and outlet stream properties. More specifically, the outlet stream flowrate $QS_{u,s}$ is determined by the empirical correlation shown in Eq. (3-17), where $a_{u,s}$ and $b_{u,s,p}$ are regression parameters. In the same way, Eq. (3-18) predicts the outlet stream property $PS_{u,s,p}$, where $c_{u,s,p}$ and $d_{u,s,p',p}$ are regression parameters as well.

$$QS_{u,s} = QF_u(a_{u,s} + \sum_{p \in PI_u} b_{u,s,p} PF_{u,p}) \; \forall u \in U^{CONV}, s \in SO_u \tag{3-17}$$

$$QS_{u,s}PS_{u,s,p} = QF_u\left(c_{u,s,p} + \sum_{p' \in PI_u} d_{u,s,p',p} PF_{u,p'}\right) \; \forall u \in U^{CONV}, s \in SO_u, p \in PO_{u,s} \tag{3-18}$$

Note that if the dependence on the feedstock property $b_{u,s,p}$ is dropped, then Eq. (3-17) becomes linear, representing a fixed-yield approach. And if the dependence on the feedstock property $d_{u,s,p',p}$ is neglected in Eq. (3-18), the approach is a proportional distribution with respect to the feedstock.

3.2.5 Operating modes modelling

Let w represent a real processing unit that has been divided into a set of virtual units UW_w and let W^{ME} include the real units with mutually exclusive modes. For such real units, at most one logical unit u that represents one operating condition needs to be selected through Eq. (3-19), where the binary variable $y_{w,u}$ takes a value of 1 if logical unit u is selected. Note that if the left-side term in Eq. (3-19) is zero, the real unit w is idled. Eq. (3-20) then determines the bounds on the flowrate QF_u. The maximum capacity available for units with inclusive operating modes is given by Eq. (3-21), where cap_w^L/cap_w^U and sf_w represent the minimum/maximum capacity and service factor for the real unit w. sf_w is a value in the range $[0,1]$ indicating the fraction of the process capacity available for the unit w.

$$\sum_{u\in WU_w} y_{w,u} \leq 1 \; \forall w \in W^{ME} \tag{3-19}$$

$$cap_w^L sf_w y_{w,u} \leq QF_u \leq cap_w^U sf_w y_{w,u} \; \forall w \in W^{ME}, u \in UW_w \tag{3-20}$$

$$cap_w^L sf_w \leq \sum_{u\in UW_w} QF_u \leq cap_w^U sf_w \; \forall w \in W \backslash W^{ME} \tag{3-21}$$

3.2.6 Logistic constraints

Logistic constraints ensure a feasible production plan by preventing the model from producing a commodity with no delivery options to markets and from buying a raw material with no available transportation capacity. A transportation mode mt can transport raw material and refined products simultaneously. The total volume delivered QT_{mt} is computed by Eq. (3-

22), where the set UPR_{mt} indicates the relationship between the product unit $u \in U^{PR}$ and the respective transport mode mt, and the set $MT_{u,s}$ indicates the transport mode for the stream s of raw-material unit $u \in U^{RW}$.

$$QT_{mt} = \sum_{u \in U^{PR} \cap UPR_{mt}} QF_u + \sum_{u \in U^{RW}} \sum_{s \in SO_u : mt \in MT_{u,s}} QS_{u,s} \ \forall mt \in MT \quad (3\text{-}22)$$

3.2.7 Variable bounds

Defining tight lower and upper bounds for the decision variables is critical to the good performance of deterministic global optimisation. The bound constraints in Eq. (3-23) define valid ranges for the flowrates and quality specifications of product streams, feed streams and intermediate refined streams, as well as the transportation capacity.

$$\begin{aligned}
&qf_u^L \le QF_u \le qf_u^U \ \forall u \in U \\
&qs_{u,s}^L \le QS_{u,s} \le qs_{u,s}^U \ \forall u \in U, s \in SO_u \\
&q_{u',s,u}^L \le Q_{u',s,u} \le q_{u',s,u}^U \ \forall u \in U \backslash U^{RW}, u' \in US_u, s \in SO_{u'} \\
&qf_u^L \le QF_u \le qf_u^U \ \forall u \in U \\
&qs_{u,s}^L \le QS_{u,s} \le qs_{u,s}^U \ \forall u \in U, s \in SO_u \\
&q_{u',s,u}^L \le Q_{u',s,u} \le q_{u',s,u}^U \ \forall u \in U \backslash U^{RW}, u' \in US_u, s \in SO_{u'} \\
&qt_{mt}^L \le QT_{mt} \le qt_{mt}^U \ \forall mt \in MT \\
&pf_{u,p}^L \le PF_{u,p} \le pf_{u,p}^U \ \forall u \in U, p \in PI_u \\
&ps_{u,s,p}^L \le PS_{u,s,p} \le ps_{u,s,p}^U \ \forall u \in U, s \in SO_u, p \in PO_{u,s}
\end{aligned} \quad (3\text{-}23)$$

3.2.8 Cost function

The profit (Eq. 3-24) consists of the revenue minus the total production costs. The revenue in Eq. (3-25) is obtained from selling the products, which include fuels from blending (LPG, gasoline, jet, diesel and fuels) and commodities that do not require blending (BTX, specialty solvents, waxes, alkylate, polyethylene and propylene). These marketable products are sold for a unit price α_u.

$$Profit := TotalRevenue - TotalCost \quad (3\text{-}24)$$

$$TotalRevenue = \sum_{u \in U^{PR}} \alpha_u \, QF_u \quad (3\text{-}25)$$

The total production cost is calculated in Eq. (3-26), with the supply and logistic costs in Eqs. (3-27) - (3-28). The parameter $\beta_{u,s}$ represents the purchase costs for the raw materials, and γ_{mt} the transportation costs.

$$TotalCost = RawMaterial + Transport + OpEx \tag{3-26}$$

$$RawMaterial = \sum_{u \in U^{RW}} \sum_{s \in SO_u} \beta_{u,s} QS_{u,s} \tag{3-27}$$

$$Transport = \sum_{mt \in MT} \gamma_{mt} QT_{mt} \tag{3-28}$$

The operating cost (OpEx) in Eq. (3-29) comprises three contributions. The first two contributions depend on a unit's feed flowrate and properties, and the third one considers the cost of a pool's flowrate. Thus, the OpEx calculation for a given process unit u in U^{CR} or U^{CONV} can be composed of any combination of these terms, where ω_u, $\psi_{u,p}$ and $\phi_{u',s,u}$ are costing parameters.

$$OpEx = \sum_{u \in U^{CR} \cup U^{CONV}} (\omega_u QF_u + \sum_{p \in PI_u} \psi_{u,p} QF_u PF_{u,p} + \sum_{u' \in US_u} \sum_{s \in SO_{u'}} \phi_{u',s,u} Q_{u',s,u}) \tag{3-29}$$

3.2.9 Summary

The short-term planning problem for the IRPC consists of the maximization of profit as in Eq. (3-24) and Eqs. (3-25) - (3-29), subject to constraints Eqs. (3-1) - (3-22), including bound constraints in Eq. (3-23). Nonconvexity in the form of multilinear terms stems from the constraints concerning the pooling equations to determine intermediate stream quality and fuel blending specifications: Eqs. (3-3) - (3-4), the correlations to predict yields and product properties of the process units: Eqs. (3-11), (3-13) - (3-18) and the operational expenditure function Eq. (3-29). The resulting model has 9,572 continuous and 280 discrete variables, 6,975 equations and 35,104 bilinear terms.

3.3 Definition of planning scenarios

We define four typical planning scenarios to establish a baseline for comparing the performance of the proposed algorithms with state-of-the-art local or global solvers. **P**

3.3.1 Base scenario

The base-case scenario (BCS) considers 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.

3.3.2 Without refinery-petrochemical integration scenario

The second scenario (WRPS) omits the petrochemical processes, by setting the demands for petrochemicals, industrial solvents and waxes to zero.

3.3.3 Logistic disruption scenario

The third scenario (LDS) analyses the impact of a disruption in the domestic crude supply, by halving the capacity of pipeline system PL3, responsible for delivering up to 80% of the crude to the IRPC.

3.3.4 Gasoline demand reduction scenario

The final scenario (DRS) analyses the effect of reducing gasoline demand by 25%, from 183 kbbl/day to 137 kbbl/day. Since the main income in BCS comes the from production of fuels, dominated by gasoline and medium distillate, the IRPC will be forced to shift production towards other commodities.

3.3.5 Input data

We consider a time horizon planning of 30 days. Detailed information regarding raw material availability, costs and specifications, as well as demand and specifications for petrochemicals and fuels are given in Appendix C, Table C. 1 - Table C. 8, while the logistic system is characterized in Table C. 9 - Table C. 10. 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.

3.4 Performance of commercial global optimisation solvers on the four scenarios of planning problem P

Two commercial solvers for deterministic global optimisation were used to solve the IRPC short-term planning problem **P**: The Branch-And-Reduce Optimization Navigator – BARON (Sahinidis, 2004) and the Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations – ANTIGONE (Misener and Floudas, 2014a). ANTIGONE is the result of the evolution and integration of two previous works from the same authors: GloMIQO - Global Mixed-Integer Quadratic Optimizer, which is a computational framework for solving MIQCQPs (Misener and Floudas, 2013), and an algorithm for solving mixed-integer signomial optimisation problems (Misener and Floudas, 2014b), subclasses of MINLPs.

ANTIGONE and BARON both implement a spatial branch-and-bound algorithm at their core. The former reformulates the original MINLP problem to solve by detecting special structures, then ANTIGONE formulates a series of convex relaxations by bisecting the feasible set in order to converge to a global optimal solution. On the other hand, BARON underestimates the objective function and/or enlarges the feasible region to construct the relaxations. Both,

ANTIGONE and BARON rely on local optimisation to find a feasible solution for the original MINLP and solve the MILP relaxations. More details regarding these commercial solvers can be found on the GAMS website (https://www.gams.com/latest/docs/S_MAIN.html).

We select the upper bound (UB), lower bound (LB), optimality gap and CPU runtime as criteria to compare the performance of the global optimisation algorithms developed in this research.

- **Best-possible solution:** This is the best bounding value provided by the relaxation of the original problem. Since the IRPC planning problem is a maximization problem, it represents an Upper Bound (UB) for the global optimal value of the objective function.
- **Best-found solution:** Also known as incumbent solution in the context of mixed-integer programming problems. It is the current best solution found during an algorithmic branch and bound search procedure for problem **P**. For the IRPC planning problem it represents a Lower Bound (LB) for the global optimal value of the objective function.
- **Dual gap also known as relaxation gap:** This is the difference between the best-found solution (LB) and the best possible solution (UB). This term is not presented as the absolute gap, which is the magnitude of the difference, but as the relative gap, which is the absolute gap divided by the best-possible solution (UB). Note that, BARON computes the dual gap dividing the absolute gap by best-found solution.
- **Optimality gap:** This is used to denote difference between best feasible solution found and last upper bound found in the search (typically branch and bound).

The results of solving the MIQCQPs arising from the IRPC planning problem using these state-of-the-art commercial solvers are shown in Table 3.6.

Table 3.6. Commercial deterministic global solvers results.

	ANTIGONE 1.1	BARON V 19.7.13	BARON V 20.10.16
	Base case scenario (BCS)		
LB [MMUSD/day]	2.634	2.687	2.684
UB [MMUSD/day]	3.898	4.250	4.505
Optimality gap [%]	32	37	40
Runtime [h]	10.0*	10.0*	10.0*
	No petrochemical integration scenario (WRPS)		
LB [MMUSD/day]	1.219	1.523	1.574
UB [MMUSD/day]	2.926	3.404	3.536
Optimality gap [%]	58	55	55
Runtime [h]	10.0*	10.0*	10.0*
	Logistic disruption scenario (LDS)		
LB [MMUSD/day]	2.156	2.360	2.473
UB [MMUSD/day]	3.451	3.842	3.981
Optimality gap [%]	38	39	38
Runtime [h]	10.0*	10.0*	10.0*
	Gasoline demand reduction scenario (DRS)		
LB [MMUSD/day]	2.186	2.445	2.478
UB [MMUSD/day]	3.719	4.134	4.214
Optimality gap [%]	41	41	41
Runtime [h]	10.0*	10.0*	10.0*

*Algorithm interrupted after reaching the maximum runtime of 10 CPU hours

We conducted these computational experiments using two versions of GAMS (version 25.1.2 and 33.0), featuring ANTIGONE (v 1.1) and BARON (v 19.7.13 and 20.10.16, respectively). For the BCS scenario, ANTIGONE and BARON both obtained similar results in terms of the LB, with optimality gaps above 32% after 10 h of CPU runtime. Moreover, BARON reached a better LB than ANTIGONE for the WRPS, LDS and DRS scenarios, but the optimality gap is still above 38%. Note that the most recent version of BARON obtained significantly better solutions for all the scenarios except the base case. Overall, ANTIGONE and BARON were able to find a feasible solution (LB), but these figures were not improved after 10 hours of runtime, the quality of the relaxation was poor, leading to large optimality gaps. This behaviour could be due to the wide bounds of the variables participating in bilinear terms, as the

relaxations become weaker with wider bounds, leading to an accumulation of nodes during the branch-and-bound search (Lara et al., 2018). The results in Table 3.6 are furthermore consistent with similar computational studies, such as Li et al. (2016) and Zhao et al. (2017).

3.5 Conclusions

This chapter introduced the IRPC of interest and detailed the corresponding short-term planning optimisation formulation. Four planning scenarios were defined and these scenarios were then used to benchmark the performance of the global solvers ANTIGONE and BARON (Lara et al., 2018). Large optimality gaps between 32–58% were obtained across all the scenarios, which are a perfect illustration of the challenges inherent to solving real-life IRPC planning models to global optimality when tackled in a monolithic way.

In the subsequent chapters, two novel decomposition algorithms for deterministic global optimisation are introduced. The first one, introduced through Chapter 4 subdivides the IRPC into small clusters according to their functionality. Inside each cluster, a MILP relaxation based on piecewise McCormick envelopes is solved, which dynamically partitions the variables that belong to the cluster and their domains are reduced through optimality-based bound tightening. The second one, presented in Chapter 5, is a spatial Lagrangean decomposition algorithm which divides the problem into two, three or four sections, namely crude management, refining, petrochemical operations, and fuel blending.

Chapter 4 CLUSTERING DECOMPOSITION STRATEGY

This chapter presents a heuristic for the deterministic global optimisation of large-scale MIQCQPs such as **P** that relies on solving a series of lower and upper bounding sub-problems. This approach 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 to the short-term planning problem described in Chapter 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. Moreover, the process unit models involve lab-scale, pilot-scale, or full industrial-scale data.

The rest of this chapter is organized as follows: A definition for process units clusters is given in Section 4.1 and the methodology is detailed in Sections 4.2 and 4.3. Applications of this approach to 14 benchmark pooling problems and a large-scale MINLP refinery planning problem, are shown in Sections 4.4 and 4.5, respectively. Section 4.6 presents the results obtained for the real-life IRPC problem, including a complete graphic display of results for the

base case scenario (see Appendix D, Appendix E and Appendix F for other scenarios) and a comparison with static piecewise relaxations and no clustering. Finally, Section 4.7 concludes the chapter.

4.1 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.

The 155 models are divided into 6 clusters, as detailed in Table 4.1.

Table 4.1. Cluster definition (units identified by different colors in Figure 3.1).

Cluster	Process Units	Physical Units	Logical Units
CL^{I}	Logistic (U^{RW}) and Crude Allocation (U^{CMX})	8	8
CL^{II}	Crude Distillation (U^{CDU})	6	13
CL^{III}	Vacuum (U^{VDU}) and Debutanizer Columns (U^{DBU})	8	11
CL^{IV}	Refining (U^{REF})	36	53
CL^{V}	Petrochemical (U^{PTC})	9	48
CL^{VI}	Fuel Blending (U^{BLD})	22	22

4.2 Methodology

For the solution of problem **P**, we propose a deterministic global optimisation 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.

4.2.1 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):

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

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**.

4.2.2 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. To improve the computational tractability of the relaxed MILP model **PR**, the continuous variables x_j are furthermore grouped into $k = 1, .., K$ clusters, where $CL = \{CL^1 \cup CL^2 \cup \dots CL^{k-1} \cup CL^k\}$.

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 ($j \notin CL$).

$$
\begin{aligned}
z^R :=\ & \max f_0^R(x,y,w) && \textbf{(PR)}\\
& \text{s.t. } f_m^R(x,y,w) \le 0\ \forall m \in \{1,\dots,M\}\\
& \left.\begin{aligned}
& w_{ij} \ge \textstyle\sum_{n=1}^{N_j}\left(\hat{x}_{ijn}x_{jn}^L + \hat{x}_{jn}x_i^L - y_{jn}x_i^L x_{jn}^L\right)\\
& w_{ij} \ge \textstyle\sum_{n=1}^{N_j}\left(\hat{x}_{ijn}x_{jn}^U + \hat{x}_{jn}x_i^U - y_{jn}x_i^U x_{jn}^U\right)\\
& w_{ij} \le \textstyle\sum_{n=1}^{N_j}\left(\hat{x}_{ijn}x_{jn}^L + \hat{x}_{jn}x_i^U - y_{jn}x_i^L x_{jn}^U\right)\\
& w_{ij} \le \textstyle\sum_{n=1}^{N_j}\left(\hat{x}_{ijn}x_{jn}^U + \hat{x}_{jn}x_i^L - y_{jn}x_i^U x_{jn}^L\right)
\end{aligned}\right\} \forall (i,j) \in BL_m, j \in CL\\
& x_i = \textstyle\sum_{n=1}^{N_j}\hat{x}_{ijn}\ \forall (i,j) \in BL_m, j \in CL\\
& x_j = \textstyle\sum_{n=1}^{N_j}\hat{x}_{jn}\ \forall j \in CL\colon (i,j) \in BL_m\\
& \textstyle\sum_{n=1}^{N_j} y_{jn} = 1\ \forall j \in CL\colon (i,j) \in BL_m\\
& x_i^L y_{jn} \le \hat{x}_{ijn} \le x_i^U y_{jn} \forall (i,j) \in BL_m, j \in CL, n \in \{1,\dots,N_j\}\\
& x_{jn}^L y_{jn} \le \hat{x}_{jn} \le x_{jn}^U y_{jn} \forall j \in CL\colon (i,j) \in BL_m, n \in \{1,\dots,N_j\}\\
& \left.\begin{aligned}
& w_{ij} \ge x_i x_j^L + x_i^L x_j - x_i^L x_j^L\\
& w_{ij} \ge x_i x_j^U + x_i^U x_j - x_i^U x_j^U\\
& w_{ij} \le x_i x_j^L + x_i^U x_j - x_i^U x_j^L\\
& w_{ij} \le x_i x_j^U + x_i^L x_j - x_i^L x_j^U
\end{aligned}\right\} \forall (i,j) \in BL_m, j \notin CL\\
& x \in [x^L, x^U] \subseteq \mathbb{R}_+^n,\ y \in \{0,1\}^r, w \subseteq \mathbb{R}_+^n\\
& y_{jn} \in \{0,1\}\ \forall j \in CL\colon (i,j) \in BL_m, n \in \{1,\dots,N_j\}
\end{aligned}
$$

Note that the lower x_{jn}^L and upper x_{jn}^U bounds for the partitioned variable x_j are calculated by Eq. (4-1) before solving the relaxed problem **PR**.

$$\left.\begin{aligned} x_{jn}^{L} &:= x_j^{L} + \frac{\left(x_j^{U} - x_j^{L}\right)(n-1)}{N_j} \\ x_{jn}^{U} &:= x_j^{L} + \frac{\left(x_j^{U} - x_j^{L}\right)n}{N_j} \end{aligned}\right\} \forall j: (i,j) \in BL_m, j \in CL, n \in \{1, \dots, N_j\} \tag{4-1}$$

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**.

4.2.3 Optimality-based bound tightening (OBBT)

Consider the variable x_h appearing in a bilinear term of cluster CL. Its lower bound x_h^L and upper bound x_h^U can be tightened after solving optimisation problem **BC**.

$$\begin{aligned} x_h^L / x_h^U \coloneqq \min/\max\, & x_h \\ \text{s.t. } & f_m^R(x,y,w) \le 0 \ \forall m \in \{1, \dots, M\} \\ & f_0^R(x,y,w) \ge LB \\ & x \in [x^L, x^U] \subseteq \mathbb{R}_+^n,\ y \in \{0,1\}^r, w \in W \subseteq \mathbb{R}_+^n \\ & y_{jn} \in \{0,1\}\ \forall j \in CL: (i,j) \in BL_m, n \in \{1, \dots, N_j\} \end{aligned} \tag{BC}$$

Note that the inclusion of constraint $f_0^R(x,y,w) \ge LB$ 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 $w \in W$. To keep a moderate model size for **BC**, N_j may be set to a lower value than in **PR**.

4.2.4 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 $\left(x_j^L, x_j^U\right)$, a uniform increment could increase the size of the relaxation prohibitively (Nagarajan et al., 2019). Instead, we update N_j dynamically by applying Eqs. (4-2) - (4-3) for every partitioned variable x_j inside the cluster CL:

$$r_{ij} = \left|\frac{w_{ij} - x_i x_j}{w_{ij}}\right| \forall (i,j) \in BL_m, j \in CL \tag{4-2}$$

$$\rho_{ij} = \frac{r_{ij} - \min\limits_{j' \in CL:(i,j') \in BL_m} r_{ij'}}{\max\limits_{j' \in CL:(i,j') \in BL_m} r_{ij'} - \min\limits_{j' \in CL:(i,j') \in BL_m} r_{ij'}} \forall (i,j) \in BL_m, \forall j \in CL \tag{4-3}$$

$$\lambda_j = \max_{i:(i,j) \in BL_m} \rho_{ij} \, \forall j \in CL \tag{4-4}$$

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 ρ_{ij}, then used to compute the normalized score λ_j. If this score is greater than a user-defined threshold τ, then the number of partitions N_j is incremented by Δ, up to the maximum partition size N^U.

4.3 Deterministic global optimisation algorithm

A flowchart of the clustering algorithm for the global optimisation of the MIQCQP resulting from the short-term IRPC planning problem is shown in Figure 4.1. 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 $MaxRunTime$, relaxation gap tolerance ε, maximum runtime $MaxRunTime_{PRBC}$ and optimality gap ε_{PRBC} for the solution of **PR** and **BC** problems, maximum number of iterations $MaxNumIter$, 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: $Iter = 0$.

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 **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 relaxation gap $\varepsilon^* = (UB - LB)/UB$

Step 5: Repeat for each cluster $k = 1, \dots, K$

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 $CL^1, \dots CL^{l-1}$ 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 ε_{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 ε^* 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.

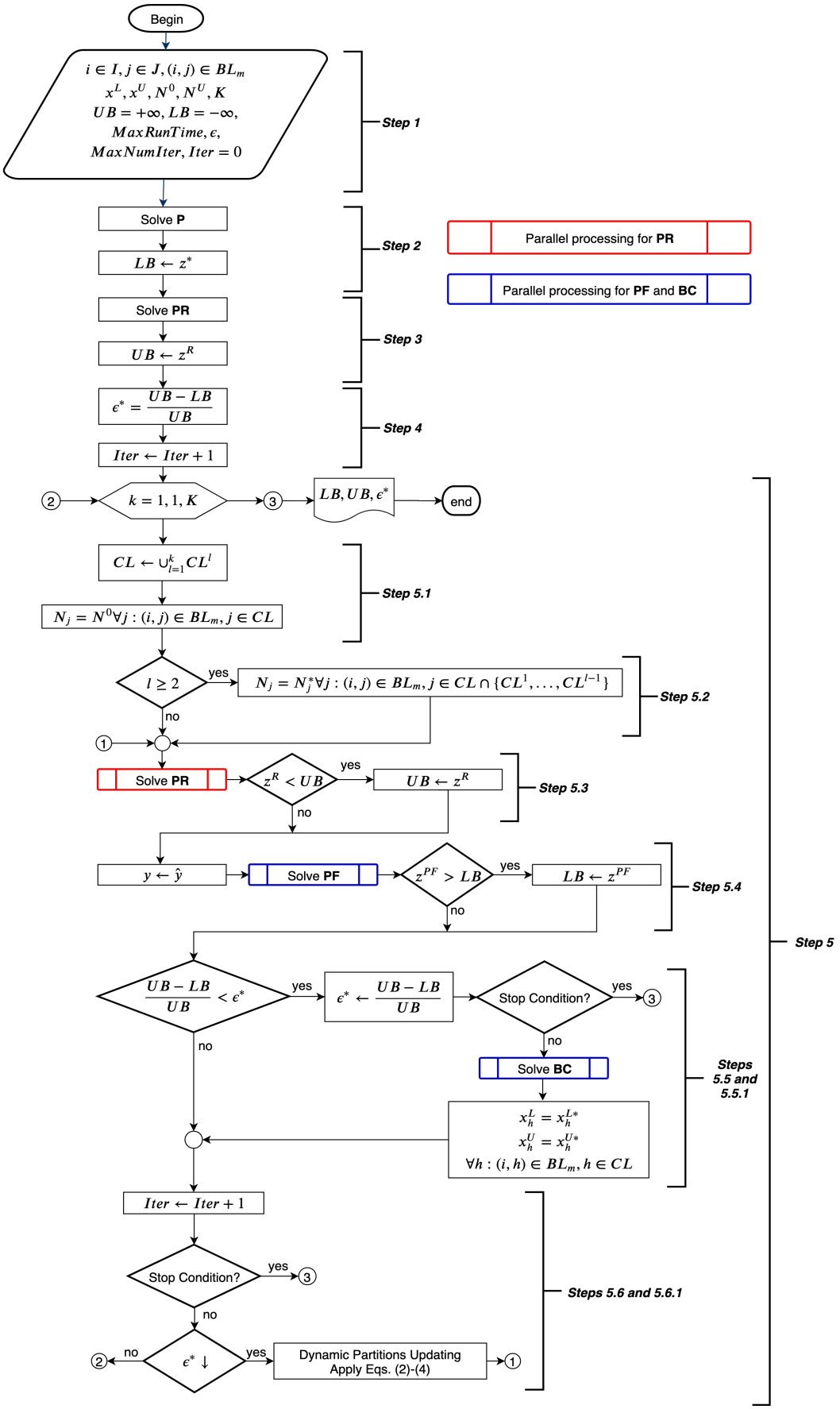


Figure 4.1. Flowchart of the clustering algorithm for global optimisation.

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

Step 5.6.1. If ε^* has decreased, update dynamically the number of partitions N_j for each partitioned variable $x_j \in CL$ (Eqs. 4-2 - 4-4) 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 Sections 4.4 and 4.5, 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 3.1, 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 optimisation 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 *Solnpoolintensity, Solnpoolpop, Solnppolgap* and *Solnpoolcapacity* set to 0, 2, 0.10 and PFS, respectively. In addition, *Paralellmode = 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.

We consider 14 standard pooling problems for illustration of our cluster-based global optimisation algorithm (section 4.4) and also, we compare our methodology against a two-stage MILP-NLP approach to solve a large-scale MINLP refinery planning problem (Castillo Castillo et al., 2017), see section 4.5. The clustering approach was able to reach a global optimal solution in both cases as it is reported in the literature. Moreover, the results of the clustering approach

applied to a much more complex problem, which recreates typical planning decisions in the Colombian hydrocarbon market are presented in section 4.6.

4.4 Clustering approach applied to benchmark pooling problems

We now consider standard pooling problem Bental5, reported by Ben-Tal et al., (1994), for illustration of our cluster-based global optimisation 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 the section 3.2.1, which is like the p-formulation of Haverly (1978). The full set of constraints of problem **P** (including the generic process unit models) is given in sections 3.2.1 - 3.2.7. The process topology in Figure 4.2 shows the input data, together with some of the main constrains and optimal flows obtained using our clustering approach.

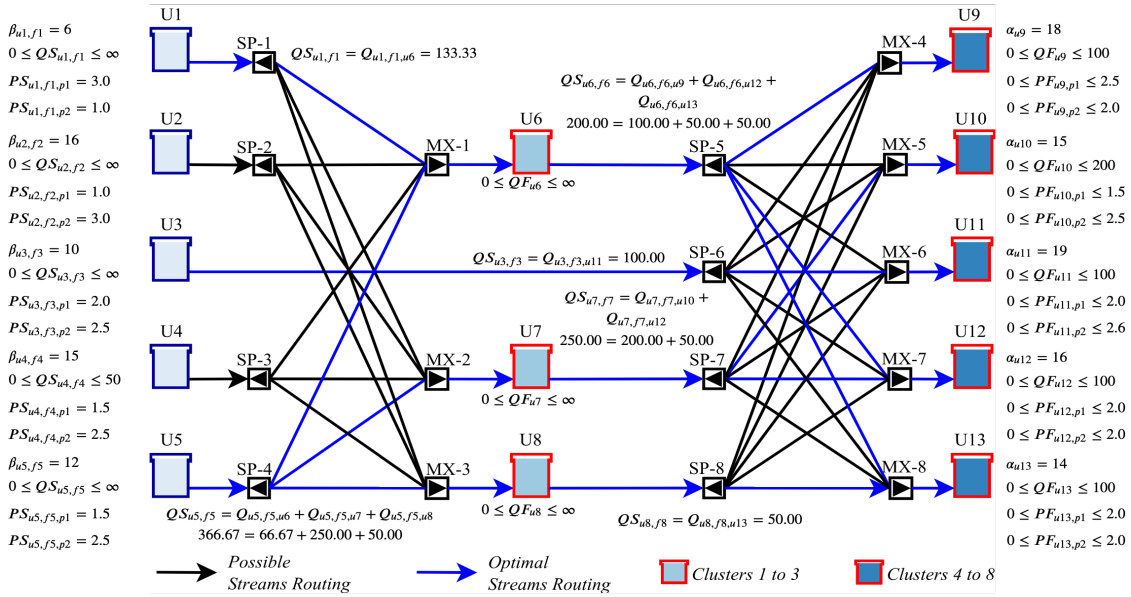


Figure 4.2. Process Network for the pooling problem.

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 **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 the Appendix B 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 relaxation 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**, ε^* 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 can 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 4.3). When solving **PF** in step 5.4, we can find the global optimal solution, $LB = 3500$. Since ε^* 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 and QF_{u7}=150.9; check Figure 4.3), 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.

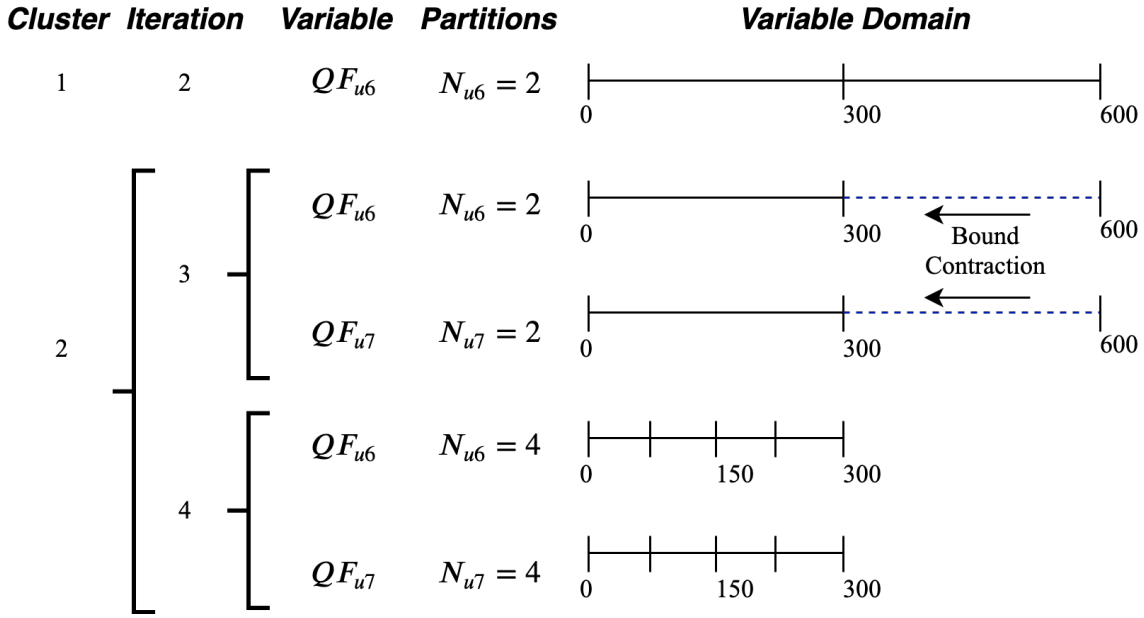


Figure 4.3. Variable partition and bound contraction for the benchmark pooling problem.

The model formulation for this problem is described in the next section, following that, a step-by-step application of the algorithm is shown.

4.4.1 Model formulation

The process network is composed by 5 input streams, 3 intermediate pools, 5 outlet streams, 2 stream properties, 8 splitters, 8 mixers, 13 tanks. The objective function formulated in the section 3.2.8 are explicitly stated as in Eqs. (4-5) - (4-7):

$$z^* := \max f_0(u, f, p) = TotalRevenue - TotalCost \tag{4-5}$$

$$TotalRevenue = 18QF_{u9} + 15QF_{u10} + 19QF_{u11} + 16QF_{u12} + 14QF_{u13} \tag{4-6}$$

$$TotalCost = 6QS_{u1,f1} + 16QS_{u2,f2} + 10QS_{u3,f3} + 15QS_{u4,f4} + 12QS_{u5,f5} \tag{4-7}$$

Subject to:

$$QF_{u6}PF_{u6,p1} = Q_{u1,f1,u6}PS_{u1,f1,p1} + Q_{u2,f2,u6}PS_{u2,f2,p1} + Q_{u4,f4,u6}PS_{u4,f4,p1} + Q_{u5,f5,u6}PS_{u5,f5,p1} \tag{4-8}$$

$$QF_{u7}PF_{u7,p1} = Q_{u1,f1,u7}PS_{u1,f1,p1} + Q_{u2,f2,u7}PS_{u2,f2,p1} + Q_{u4,f4,u7}PS_{u4,f4,p1} + Q_{u5,f5,u7}PS_{u5,f5,p1}$$

$$QF_{u8}PF_{u8,p1} = Q_{u1,f1,u8}PS_{u1,f1,p1} + Q_{u2,f2,u8}PS_{u2,f2,p1} + Q_{u4,f4,u8}PS_{u4,f4,p1} + Q_{u5,f5,u8}PS_{u5,f5,p1}$$

$$QF_{u9}PF_{u9,p1} = Q_{u3,f3,u9}PS_{u3,f3,p1} + Q_{u6,f6,u9}PS_{u6,f6,p1} + Q_{u7,f7,u9}PS_{u7,f7,p1} + Q_{u8,f8,u9}PS_{u8,f8,p1}$$

$$QF_{u10}PF_{u10,p1} = Q_{u3,f3,u10}PS_{u3,f3,p1} + Q_{u6,f6,u10}PS_{u6,f6,p1} + Q_{u7,f7,u10}PS_{u7,f7,p1} + Q_{u8,f8,u10}PS_{u8,f8,p1}$$

$$QF_{u11}PF_{u11,p1} = Q_{u3,f3,u11}PS_{u3,f3,p1} + Q_{u6,f6,u11}PS_{u6,f6,p1} + Q_{u7,f7,u11}PS_{u7,f7,p1} + Q_{u8,f8,u11}PS_{u8,f8,p1}$$

$$QF_{u12}PF_{u12,p1} = Q_{u3,f3,u12}PS_{u3,f3,p1} + Q_{u6,f6,u12}PS_{u6,f6,p1} + Q_{u7,f7,u12}PS_{u7,f7,p1} + Q_{u8,f8,u12}PS_{u8,f8,p1}$$

$$QF_{u13}PF_{u13,p1} = Q_{u3,f3,u13}PS_{u3,f3,p1} + Q_{u6,f6,u13}PS_{u6,f6,p1} + Q_{u7,f7,u13}PS_{u7,f7,p1} + Q_{u8,f8,u13}Ps_{u8,f8,p1}$$

$$QF_{u6}PF_{u6,p2} = Q_{u1,f1,u6}PS_{u1,f1,p2} + Q_{u2,f2,u6}PS_{u2,f2,p2} + Q_{u4,f4,u6}PS_{u4,f4,p2} + Q_{u5,f5,u6}PS_{u5,f5,p2}$$

$$QF_{u7}PF_{u7,p2} = Q_{u1,f1,u7}PS_{u1,f1,p2} + Q_{u2,f2,u7}PS_{u2,f2,p2} + Q_{u4,f4,u7}PS_{u4,f4,p2} + Q_{u5,f5,u7}PS_{u5,f5,p2}$$

$$QF_{u8}PF_{u8,p2} = Q_{u1,f1,u8}PS_{u1,f1,p2} + Q_{u2,f2,u8}PS_{u2,f2,p2} + Q_{u4,f4,u8}PS_{u4,f4,p2} + Q_{u5,f5,u8}PS_{u5,f5,p2}$$

$$QF_{u9}PF_{u9,p2} = Q_{u3,f3,u9}PS_{u3,f3,p2} + Q_{u6,f6,u9}PS_{u6,f6,p2} + Q_{u7,f7,u9}PS_{u7,f7,p2} + Q_{u8,f8,u9}PS_{u8,f8,p2} \quad (4\text{-}9)$$

$$QF_{u10}PF_{u10,p2} = Q_{u3,f3,u10}PS_{u3,f3,p2} + Q_{u6,f6,u10}PS_{u6,f6,p2} + Q_{u7,f7,u10}PS_{u7,f7,p2} + Q_{u8,f8,u10}PS_{u8,f8,p2}$$

$$QF_{u11}PF_{u11,p2} = Q_{u3,f3,u11}PS_{u3,f3,p2} + Q_{u6,f6,u11}PS_{u6,f6,p2} + Q_{u7,f7,u11}PS_{u7,f7,p2} + Q_{u8,f8,u11}PS_{u8,f8,p2}$$

$$QF_{u12}PF_{u12,p2} = Q_{u3,f3,u12}PS_{u3,f3,p2} + Q_{u6,f6,u12}PS_{u6,f6,p2} + Q_{u7,f7,u12}PS_{u7,f7,p2} + Q_{u8,f8,u12}PS_{u8,f8,p2}$$

$$QF_{u13}PF_{u13,p2} = Q_{u3,f3,u13}PS_{u3,f3,p2} + Q_{u6,f6,u13}PS_{u6,f6,p2} + Q_{u7,f7,u13}PS_{u7,f7,p2} + Q_{u8,f8,u13}Ps_{u8,f8,p2}$$

$$\begin{aligned}
QF_{u6} &= Q_{u1,f1,u6} + Q_{u2,f2,u6} + Q_{u4,f4,u6} + Q_{u5,f5,u6} \\
QF_{u7} &= Q_{u1,f1,u7} + Q_{u2,f2,u7} + Q_{u4,f4,u7} + Q_{u5,f5,u7} \\
QF_{u8} &= Q_{u1,f1,u8} + Q_{u2,f2,u8} + Q_{u4,f4,u8} + Q_{u5,f5,u8} \\
QF_{u9} &= Q_{u3,f3,u9} + Q_{u6,f6,u9} + Q_{u7,f7,u9} + Q_{u8,f8,u9} \\
QF_{u10} &= Q_{u3,f3,u10} + Q_{u6,f6,u10} + Q_{u7,f7,u10} + Q_{u8,f8,u10} \\
QF_{u11} &= Q_{u3,f3,u11} + Q_{u6,f6,u11} + Q_{u7,f7,u11} + Q_{u8,f8,u11} \\
QF_{u12} &= Q_{u3,f3,u12} + Q_{u6,f6,u12} + Q_{u7,f7,u12} + Q_{u8,f8,u12} \\
QF_{u13} &= Q_{u3,f3,u13} + Q_{u6,f6,u13} + Q_{u7,f7,u13} + Q_{u8,f8,u13}
\end{aligned} \tag{4-10}$$

$$\begin{aligned}
QS_{u1,f1} &= Q_{u1,f1,u6} + Q_{u1,f1,u7} + Q_{u1,f1,u8} \\
QS_{u2,f2} &= Q_{u2,f2,u6} + Q_{u2,f2,u7} + Q_{u2,f2,u8} \\
QS_{u3,f3} &= Q_{u3,f3,u9} + Q_{u3,f3,u10} + Q_{u3,f3,u11} + Q_{u3,f3,u12} + Q_{u3,f3,u13} \\
QS_{u4,f4} &= Q_{u4,f4,u6} + Q_{u4,f4,u7} + Q_{u4,f4,u8} \\
QS_{u5,f5} &= Q_{u5,f5,u6} + Q_{u5,f5,u7} + Q_{u5,f5,u8} \\
QS_{u6,f6} &= Q_{u6,f6,u9} + Q_{u6,f6,u10} + Q_{u6,f6,u11} + Q_{u6,f6,u12} + Q_{u6,f6,u13} \\
QS_{u7,f7} &= Q_{u7,f7,u9} + Q_{u7,f7,u10} + Q_{u7,f7,u11} + Q_{u7,f7,u12} + Q_{u7,f7,u13} \\
QS_{u8,f8} &= Q_{u8,f8,u9} + Q_{u8,f8,u10} + Q_{u8,f8,u11} + Q_{u8,f8,u12} + Q_{u8,f8,u13}
\end{aligned} \tag{4-11}$$

$$\begin{aligned}
QF_{u6} &= QS_{u6,f6} \\
QF_{u7} &= QS_{u7,f7}
\end{aligned} \tag{4-12}$$

$$QF_{u8} = QS_{u8,f8}$$

$$\begin{aligned}
&PF_{u6,p1} = PS_{u6,f6,p1}\\
&PF_{u6,p2} = PS_{u6,f6,p2}\\
&PF_{u7,p1} = PS_{u7,f7,p1}\\
&PF_{u7,p2} = PS_{u7,f7,p2}\\
&PF_{u8,p1} = PS_{u8,f8,p1}\\
&PF_{u8,p2} = PS_{u8,f8,p2}
\end{aligned} \tag{4-13}$$

$$\begin{aligned}
&0 \le QF_{u6}, QF_{u7}, QF_{u8} \le 600\\
&0 \le QF_{u9}, QF_{u11}, QF_{u12}, QF_{u13} \le 100\\
&0 \le QF_{u8} \le 200\\
&0 \le QS_{u1,f1}, QS_{u2,f2}, QS_{u4,f4,}QS_{u5,f5} \le 1800\\
&0 \le QS_{u3,f3} \le 600\\
&0 \le QS_{u4,f4} \le 50\\
&0 \le Q_{u1,f1,u6}, Q_{u1,f1,u7}, Q_{u1,f1,u8} \le 600\\
&0 \le Q_{u2,f2,u6}, Q_{u2,f2,u7}, Q_{u2,f2,u8} \le 600\\
&0 \le Q_{u3,f3,u9}, Q_{u3,f3,u11}, Q_{u3,f3,u12}, Q_{u3,f3,u13} \le 100\\
&0 \le Q_{u3,f3,u10} \le 200\\
&0 \le Q_{u4,f4,u6}, Q_{u4,f4,u7}, Q_{u4,f4,u8} \le 600\\
&0 \le Q_{u5,f5,u6}, Q_{u5,f5,u7}, Q_{u5,f5,u8} \le 600\\
&0 \le PF_{u6,p1}, PF_{u7,p1}, PF_{u8,p1} \le 3.0
\end{aligned} \tag{4-14}$$

$$0 \leq PF_{u6,p2}, PF_{u7,p2}, PF_{u8,p2} \leq 3.0$$

$$0 \leq PF_{u9,p1} \leq 2.5$$

$$0 \leq PF_{u9,p2} \leq 2.0$$

$$0 \leq PF_{u10,p1} \leq 1.5$$

$$0 \leq PF_{u10,p2} \leq 2.5$$

$$0 \leq PF_{u11,p1} \leq 2.0 \quad (4\text{-}15)$$

$$0 \leq PF_{u11,p2} \leq 2.6$$

$$0 \leq PF_{u12,p1}, PF_{u12,p2}, PF_{u13,p1}, PF_{u13,p2} \leq 2.0$$

$$0 \leq PS_{u6,f6,p1}, PS_{u7,f7,p1}, PS_{u8,f8,p1} \leq 3.0$$

$$0 \leq PS_{u6,f6,p2}, PS_{u7,f7,p2}, PS_{u8,f8,p2} \leq 3.0$$

The properties $p1$ and $p2$ are calculated by the group of equations (4-8) - (4-9). Note that the bilinear terms arise from these equations. The block of equations (4-10) represents the mass balance to estimate the feedstock for the process units $u = \{u6, u7, \dots, u13\}$. The mass balance for the splitters is represented by the set of equations (4-11). Since there is no inventory, equations in (4-12) and (4-13) relate the outlet streams from the tanks with their input flowrate. Variable bounds given in Eqs. (4-14) are defined analysing the process topology (Adhya et al., 1999; Misener et al., 2011). Products quality specifications are defined by Eqs. (4-15). Thus, problem **P** is defined as maximize Eq. (4-5) subject to Eqs. (4-6) - (4-15). Model **P** has 41 equations, 95 continuous variables and 80 bilinear terms.

4.4.2 Definition of set BL

The set BL represents the pair flowrate – property involved in the bilinear products. For each equation $m \in M$ calculating a property $p \in P$, we define the following sets:

$$BL_1 = \{(u6, p1), (u1, f1, p1), (u2, f2, p1), (u4, f4, p1), (u4, f5, p1)\}$$

$$BL_2 = \{(u6, p2), (u1, f1, p2), (u2, f2, p2), (u4, f4, p2), (u5, f5, p2)\}$$

$$BL_3 = \{(u7, p1), (u1, f1, p1), (u2, f2, p1), (u4, f4, p1), (u5, f5, p1)\}$$

$$BL_4 = \{(u7, p2), (u1, f1, p2), (u2, f2, p2), (u4, f4, p2), (u5, f5, p2)\}$$

$$BL_5 = \{(u8, p1), (u1, f1, p1), (u2, f2, p1), (u4, f4, p1), (u5, f5, p1)\}$$

$$BL_6 = \{(u8, p2), (u1, f1, p2), (u2, f2, p2), (u4, f4, p2), (u5, f5, p2)\}$$

$$BL_7 = \{(u9, p1), (u3, f3, p1), (u6, f6, p1), (u7, f7, p1), (u8, f8, p1)\}$$

$$BL_8 = \{(u9, p2), (u3, f3, p2), (u6, f6, p2), (u7, f7, p2), (u8, f8, p2)\}$$

$$BL_9 = \{(u10, p1), (u3, f3, p1), (u6, f6, p1), (u7, f7, p1), (u8, f8, p1)\}$$

$$BL_{10} = \{(u10, p2), (u3, f3, p2), (u6, f6, p2), (u7, f7, p2), (u8, f8, p2)\}$$

$$BL_{11} = \{(u11, p1), (u3, f3, p1), (u6, f6, p1), (u7, f7, p1), (u8, f8, p1)\}$$

$$BL_{12} = \{(u11, p2), (u3, f3, p2), (u6, f6, p2), (u7, f7, p2), (u8, f8, p2)\}$$

$$BL_{13} = \{(u12, p1), (u3, f3, p1), (u6, f6, p1), (u7, f7, p1), (u8, f8, p1)\}$$

$$BL_{14} = \{(u12, p2), (u3, f3, p2), (u6, f6, p2), (u7, f7, p2), (u8, f8, p2)\}$$

$$BL_{15} = \{(u13, p1), (u3, f3, p1), (u6, f6, p1), (u7, f7, p1), (u8, f8, p1)\}$$

$$BL_{16} = \{(u13, p2), (u3, f3, p2), (u6, f6, p2), (u7, f7, p2), (u8, f8, p2)\}$$

$$BL = \bigcup_{m=1}^{m=16} BL_m$$

4.4.3 Step-by-step algorithm

Next, we describe step by step the application of the clustering approach for the illustrative case study.

Step 1: $\varepsilon = 1 * 10^{-3},\ N^0 = 2, N^U = 100.\ UB = +\infty,\ LB = -\infty, Iter = 0.$

Step 2: Solve problem **P,** set $LB \leftarrow 2700$.

Step 3: Solve relaxed problem **PR** to optimality using the standard McCormick relaxation for all bilinear terms, set $UB \leftarrow 3500$.

Step 4: Update $\varepsilon^* \leftarrow \frac{3500-2700}{3500} = 0.30$. Increase number of iterations $ter \leftarrow Iter + 1 = 1$.

For $k = 1$, $Iter = 2$

Step 5.1. Define the active cluster $CL \leftarrow CL^1 = \{u6\}$

Step 5.2. $N_{u6} = N^0 = 2$.

Step 5.3. Solve relaxed problem PR, $z^R \leftarrow 3500$.

If $z^R < UB$, update $UB \leftarrow z^R$.

Step 5.4. Solve problem **PF,** $z^{PF} \leftarrow 2700$.

If $z^{PF} > LB$, update $LB \leftarrow z^{PF}$.

Step 5.5. If $\frac{3500-2700}{3500} = 0.30 < \varepsilon^*$ update $\varepsilon^* \leftarrow \frac{3500-2700}{3500}$ and go to step 5.5.1.

Step 5.5.1. If stop condition is reached, finish. Otherwise, solve BC for each variable present in the bilinear terms of the active cluster CL.

Step 5.6. $Iter \leftarrow Iter + 1 = 2$. If stop condition is reached, finish. Otherwise go to step 5.6.1.

Step 5.6.1 If ε^* improves, update dynamically the number of partitions and go to step 5.3. Otherwise explore the next cluster.

Since neither UB and LB change in steps 5.3 and 5.4, ε^* is not improved. Consequently, we keep the number of partitions $N_{u6} = 2$ and explore the next cluster. Note that the $LB \leftarrow 2700$ corresponds to one of the local optimal reported previously in the literature (Adhya et al., 1999).

For $k = 2$, $Iter = 3$

Step 5.1. Define the active cluster $CL \leftarrow CL^1 \cup CL^2 = \{u6, u7\}$

Step 5.2. $N_{u_7} = N^0 = 2$.

Step 5.3. Solve relaxed problem **PR**, $z^R \leftarrow 3500$.

If $z^R < UB$, update $UB \leftarrow z^R$.

Step 5.4. Solve problem **PF,** $z^{PF} \leftarrow 2900$.

If $z^{PF} > LB$, update $LB \leftarrow 2900$.

Step 5.5. $\varepsilon_{CL} = \frac{3500-2900}{3500} = 0.21$.

If $\frac{3500-2900}{3500} = 0.21 < \varepsilon^*$ update $\varepsilon^* \leftarrow 0.21$ and go to step 5.5.1.

Step 5.5.1. If stop condition is reached, finish. Otherwise, solve **BC** for each variable present in the bilinear terms of the active cluster CL.

$0 \leq QF_{u6} \leq 300$; $0 \leq QF_{u7} \leq 300$.

Step 5.6. $Iter \leftarrow Iter + 1 = 3$. If stop condition is reached, finish. Otherwise go to step 5.6.1.

Step 5.6.1 If ε^* improves, update dynamically the number of partitions: $N_{u6} = N_{u6} + 2 = 4$. $N_{u7} = N_{u7} + 2 = 4$ and go to step 5.3. Otherwise explore the next cluster.

In the step 5.4 the LB is improved leading to $\varepsilon^* = 0.21$. As a consequence, the domain for variables QF_{u6} and QF_{u6} is reduced in step 5.5.1, number of partitions is increased (step 5.6.1) and we continue exploring cluster $k = 2$ at step 5.3. Note that the $LB \leftarrow 2900$ corresponds to one of the local optimal reported by Adhya et. al. (1999).

For $k = 2$, $Iter = 4$

Step 5.3. Solve relaxed problem **PR**, $z^R \leftarrow 3500$.

If $z^R < UB$, update $UB \leftarrow z^R$.

Step 5.4. Solve problem **PF,** $z^{PF} \leftarrow 3500$.

If $z^{PF} > LB$, update $LB \leftarrow 3500$.

Step 5.5. If $\varepsilon_{CL} = \frac{3500-3500}{3500} = 0 < \varepsilon^*$ update $\varepsilon^* \leftarrow 0$ and go to step 5.5.1.

Step 5.5.1. If stop condition is reached, $Iter \leftarrow Iter + 1 = 4$ and finish. Otherwise, solve BC for each variable present in the bilinear terms of the active cluster CL.

In this cluster we reach the global optimal solution. Steps 5.4 and 5.5 lead to $LB \leftarrow 3500$ and $\varepsilon^* \leftarrow 0$. Since the condition $\varepsilon^* \leq \varepsilon$ is true in the step 5.5.1, the algorithm finishes at the iteration $Iter = 4$ in the cluster $k = 2$.

QF_{u6} leaves cluster $k = 1$ with $N_{u6} = 2$ partitions. In the next cluster, $N_{u6} = N_{u7} = 2$. Since ε^* is reduced in the iteration 3, bound contraction is applied reducing by 50% the domain for variables QF_{u6} and QF_{u7}. We continue exploring cluster $k = 2$ with $N_{u6} = N_{u7} = 4$, reaching the relative optimality gap. Note that model PR starts with 2 binary variables at cluster $k = 1$, finishing with 8 binary variables (4 for each partitioned variable) at $k = 2$.

Assuming a maximum number of partitions $N^U = 100$ and partitioning variable QF_u without clustering, the size for model PR fluctuates between 361 equations and 175 continuous variables in case of formulating the global McCormick envelopes and 921 equations, 2775 continuous and 1300 binary variables if the piecewise McCormick are formulated. Table 4.2 shows model size for **PR** applying the clustering approach. Note that maximum size for **PR** has 191 and 8 continuous and binary variables respectively. In the Appendix A, we solve problem **P** using the state-of-the-art solver GUROBI (v 10.0.2) and compare its performance with our clustering approach, ANTIGONE (v. 1.1) and BARON (v. 20.10.16).

Table 4.2. Model size for PR in the illustrative case study.

Iteration	Variables		Equations
	Continuous	Binary	
1	175	0	361
2	179	2	368
3	183	4	375
4	191	8	375

Table 4.3. Computational performance of the clustering approach, ANTIGONE and BARON applied to benchmark pooling problems. Part 1.

Reference	(Haverly, 1980, 1979, 1978)			(Foulds et al., 1992)		
Problem	**Haverly1**	**Haverly2**	**Haverly3**	**Foulds2**	**Foulds3**	**Foulds4**
Pooling Structure						
Input streams	3	3	3	6	11	5
Pools	1	1	1	4	8	2
Output streams	2	2	2	4	16	4
Properties	1	1	1	1	1	6
Bilinear terms	2	2	2	8	128	128
# Local optimal slns	2	3	2	6	4	5
Local optimal	100; *400*	0; 400; *600*	125; *750*	400; 600; 700; 800; 1,000; *1,100*	6.5; 7; 7.5; *8*	6; 6.5; 7; 7.5; *8*
Clustering approach						
Global optimum	400	600	750	1,100	8	8
Relaxation gap	**2.84E-16**	**0.00E+00**	**1.52E-16**	7.68E-02	**8.88E-16**	**8.88E-16**
Runtime [s]	53.83	65.76	76.77	719.85	12.01	14.17
ANTIGONE						
Global optimum	400	600	750	1,100	8	8
Optimality gap	**9.99E-10**	**1.00E-09**	**1.00E-09**	**1.00E-09**	**1.00E-09**	**1.00E-09**
Runtime [s]	3.68	3.96	4.06	5.62	5.92	27.03
BARON						
Global optimum	400	600	750	1,100	8	8
Optimality gap	**1.00E-09**	**9.99E-10**	**1.00E-09**	**1.00E-09**	9.10E-01	**9.99E-10**
Runtime [s]	6.05	5.03	5.34	683.61	3600.00	376.00

We apply the clustering approach to 14 standard pooling problems (Table 4.3 – Table 4.5), considering different pooling structures, leading to nonconvex optimisation problems with a multimodal objective function, which in some cases have between 2 and 13 local optimal solutions (see rows local optimal in Table 4.3 – Table 4.5, with the global optimal solution in italic) and up to 128 bilinear terms, which are present in the pooling equations (see rows bilinear terms in Table 4.3 – Table 4.5). Overall, the three global optimisation approaches were able to reach the global optimal solution (see rows relaxation and optimality gap in Table 4.3 – Table 4.5, with figures in bold showing the problems whose optimality is close to zero). Note that ANTIGONE has the best performance in terms of CPU runtime, but problems Adhya2 and Adhya3 were challenging to solve by ANTIGONE and BARON with a maximum runtime of

3600 CPU s. Moreover, BARON reach the maximum runtime for problems Foulds5 and BenTal5. In contrast, for the same four standard pooling problems, the clustering approach reach the global optimal solution at 1483 CPU time on average.

The results from the standard pooling problems shown that the clustering approach has a similar performance in terms of quality solution given by the optimal solution found, optimality and relaxation gap and CPU runtime compared to ANTIGONE and BARON. In the next section 4.5, we address a large-scale refinery planning problem adapted from literature in order to text our approach solving a more challenging problem.

Table 4.4. Computational performance of the clustering approach, ANTIGONE and BARON applied to benchmark pooling problems. Part 2.

Reference	(Foulds et al., 1992)	(Ben-Tal et al., 1994)		(Adhya et al., 1999)
Problem	**Foulds5**	**BenTal4**	**BenTal5**	**Adhya1**
Pooling Structure				
Input streams	11	4	5	5
Pools	4	1	3	2
Output streams	16	2	5	4
Properties	1	1	2	4
Bilinear terms	64	2	30	20
# local optimal slns	8	2	11	7
Local optimal	4.5; 5; 5.5; 6; 6.5; 7; 7.5; *8*	100; *450*	1000; 1600; 1900; 2000; 2100; 2300; 2500; 2600; 2700; 2900; *3500*	0; 56.67; 63.93; 68.74; 340.93; 509.78; *549.80*
Clustering approach				
Global optimum	8.00	450.00	3,500.00	549.80
Relaxation gap	**7.11E-15**	**8.84E-16**	**5.20E-16**	1.21E-01
Runtime [s]	28.37	135.44	16.85	306.86
ANTIGONE				
Global optimum	8.00	450.00	3,500.00	549.80
Optimality gap	**1.00E-09**	**1.00E-09**	**9.99E-10**	**1.00E-09**
Runtime [s]	7.28	8.10	9.67	167.17
BARON				
Global optimum	8.00	450.00	3,500.00	549.80
Optimality gap	9.20E-01	**1.00E-09**	2.30E-01	**1.00E-09**
Runtime [s]	3,600.00	0.18	3,600.00	270.80

Table 4.5. Computational performance of the clustering approach, ANTIGONE and BARON applied to benchmark pooling problems. Part 3.

Reference		(Adhya et al., 1999)		(Audet et al., 2004)
Problem	**Adhya2**	**Adhya3**	**Adhya4**	**RT2**
Pooling structure				
Input streams	5	8	8	3
Pools	2	3	2	2
Output streams	4	4	5	3
Properties	6	6	4	4
Bilinear terms	20	32	40	18
# local optimal slns	6	13	13	2
Local optimal	0; 33.33; 56.67; 65; 509.78; *549.80*	0; 33.33; 50.74; 56.67; 57.74; 65; 390.87; 412.31; 440.76; 456.67; 500; 509.78; 549.8; 552.85; 559.62; *561.04*	0; 90; 97.5; 105; 202.5; 365; 365.83; 373.33; 470.83; 481.12; 505.61; 544.80; 610.61; *877.65*	3273.95; *4391.83*
Clustering approach				
Global optimum	549.80	561.04	877.65	4,391.83
Relaxation gap	7.10E-02	3.41E-01	**9.69E-03**	1.78E-01
Runtime [s]	802.67	648.22	737.65	392.10
ANTIGONE				
Global optimum	549.80	561.04	877.65	4,391.83
Optimality gap	1.45E-01	4.19E-01	**1.00E-09**	**7.17E-08**
Runtime [s]	3,600.00	3,600.00	48.24	3,600.00
BARON				
Global optimum	549.80	561.04	877.65	4,391.83
Optimality gap	4.00E-01	4.00E-01	**1.00E-09**	**9.99E-10**
Runtime [s]	3,600.00	3,600.00	314.45	300.00

4.5 Application of the clustering approach to a refinery planning benchmarking problem

We applied the clustering approach to a large-scale refinery planning problem (Figure 4.4), which is composed by a crude distillation unit, hydrotreating for naphtha, diesel, gas oil and the bottom from the CDU, gas oil hydrocracking, catalytic reformer and fluid catalytic cracking. The fuels production is given by the blending of intermediated refinery streams to obtain

kerosene, gasoline, diesel and fuel oil. A complete detailed data for the refinery can be found on Castillo Castillo et al. (Castillo Castillo et al., 2017).

For this case study, problem **P** comprises 3,992 equations, 17,116 nonlinear terms, 3,012 continuous and 99 discrete variables.

We define five clusters: CDU (blue colour in Figure 4.4), hydrotreating processes (red colour in Figure 4.4), FCC and catalytic reformer (gray colour in Figure 4.4), hydrocracking (green colour in Figure 4.4) and fuel blending (dark squares in Figure 4.4).

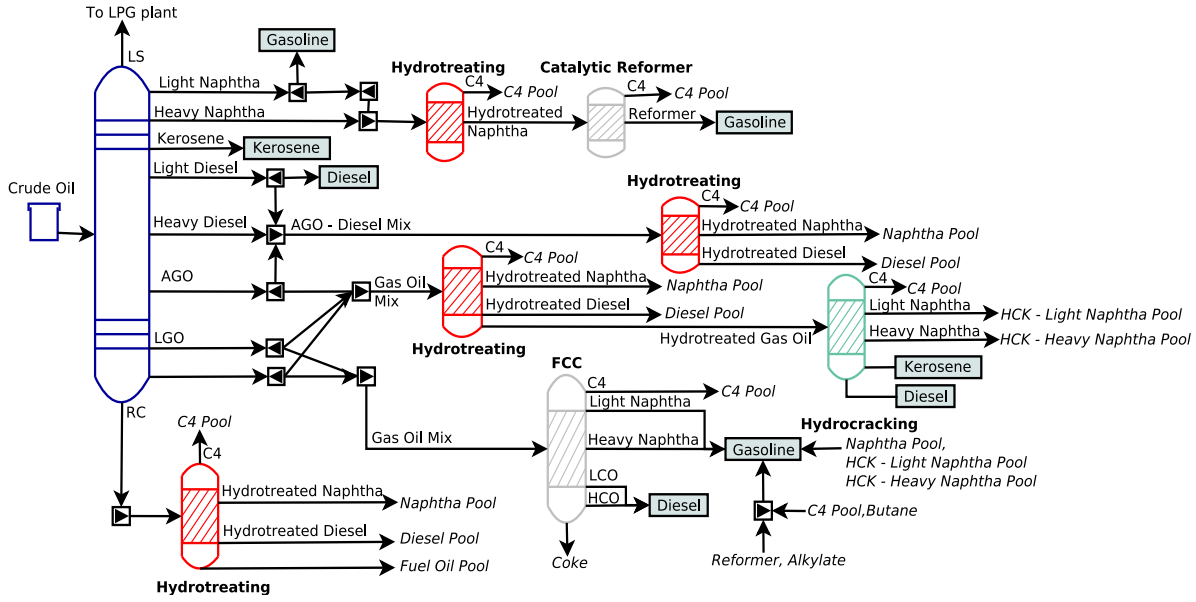


Figure 4.4. Refinery flowsheet, adapted from (Castillo Castillo et al., 2017).

The performance for the clustering approach is shown in the Figure 4.5. Note that between clusters I and II the optimality gap is improved significantly, whilst in clusters III and IV the optimality gap stagnated, and finally exploring the last cluster, which is the fuel blending, our approach reached a near global optimal solution.

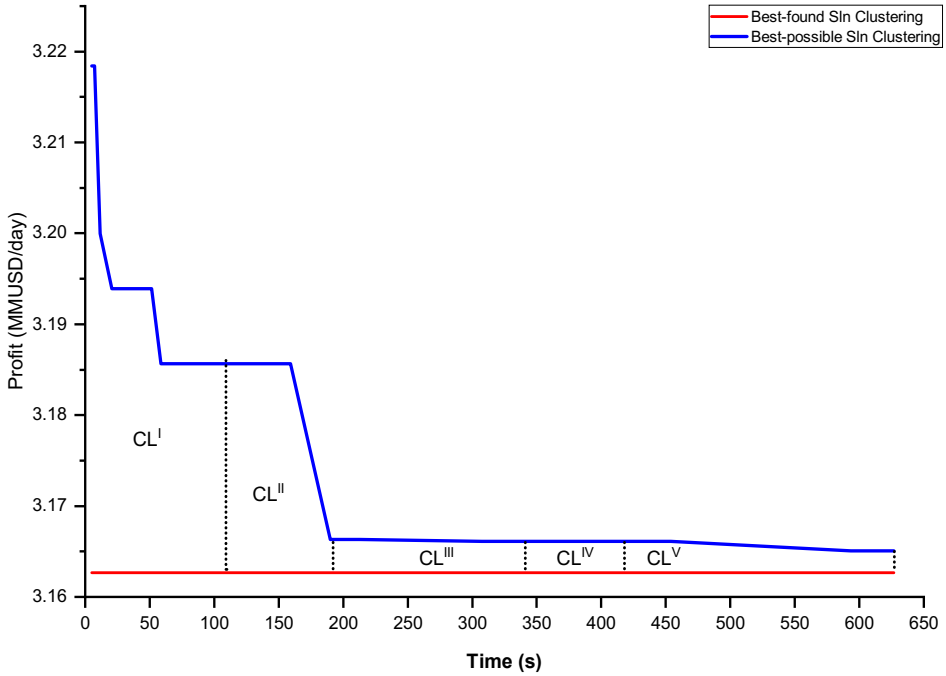


Figure 4.5. Clustering decomposition performance for refinery benchmarking problem.

We compare the results of the clustering decomposition algorithm to the-state-of-the-art global optimisation solvers ANTIGONE and BARON (Table 4.6). Overall, all the solvers were able to reach the same lower bound and optimality/relaxation gaps below 1%, with BARON reporting the lowest of these figures. Since the three approaches have similar performance in terms of quality of the solution, the last criteria to compare is the computational time, where BARON spent less CPU time than the clustering decomposition-based algorithm and ANTIGONE.

Table 4.6. Performance of the clustering approach on a large-scale refinery benchmarking problem.

	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation gap* [%]**	**Runtime [h]**
Clustering approach	3,162.66	3.165.05	0.09	0.17
ANTIGONE	3,162.66	3,163.30	0.02	0.43
BARON	3,162.66	3,162.98	0.01	0.16

*Optimality gap is applied for ANTIGONE and BARON

4.6 Computational results for the IRPC problem

All the models used to address the short-term planning problem of the IRPC 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 $\varepsilon_{PRBC} = 0.01\%$ 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, the optimality and relaxation gaps to ε=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 3.3.1. 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 4.3), 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.

4.6.1 Base scenario

The performance of the three global optimisation algorithms for the base scenario (described in section 3.3.1) is compared in Figure 4.6, representing the progress of the best-found (LB) and best-possible (UB) solutions against computational time. Both **A**NTIGONE (Plan A) and **B**ARON (Plan B) can 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 **C**lustering algorithm (Plan C) starts with the same profit as ANTIGONE, but with a tighter upper bound corresponding to a relaxation gap of 42%. After processing Cluster I, this approach can reduce the relaxation 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 MMUSD/day. During Clusters II and III the gap remains unchanged, then the profit increases to 2.96 MMUSD/day after going through Cluster IV. This value is 11 and 10% greater than the solutions found by ANTIGONE and BARON, respectively.

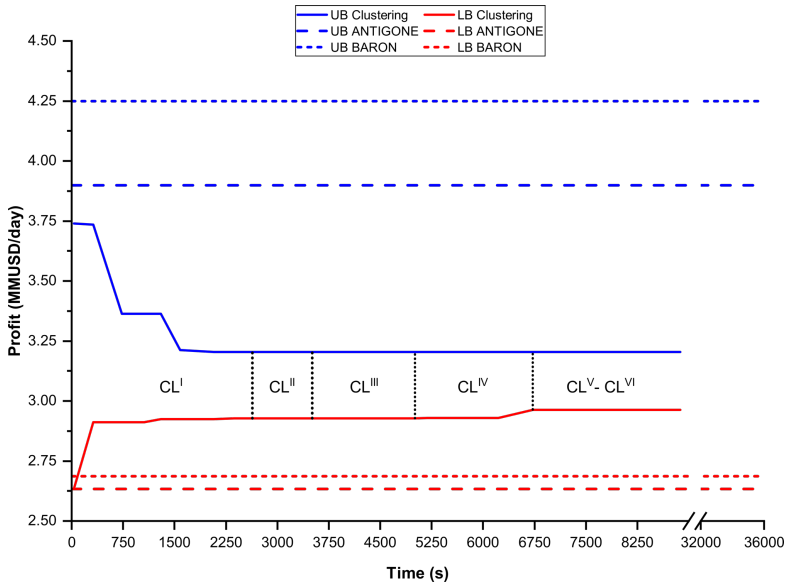


Figure 4.6. Performance of global optimisation algorithms in the base case scenario (BCS).

Figure 4.7 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.

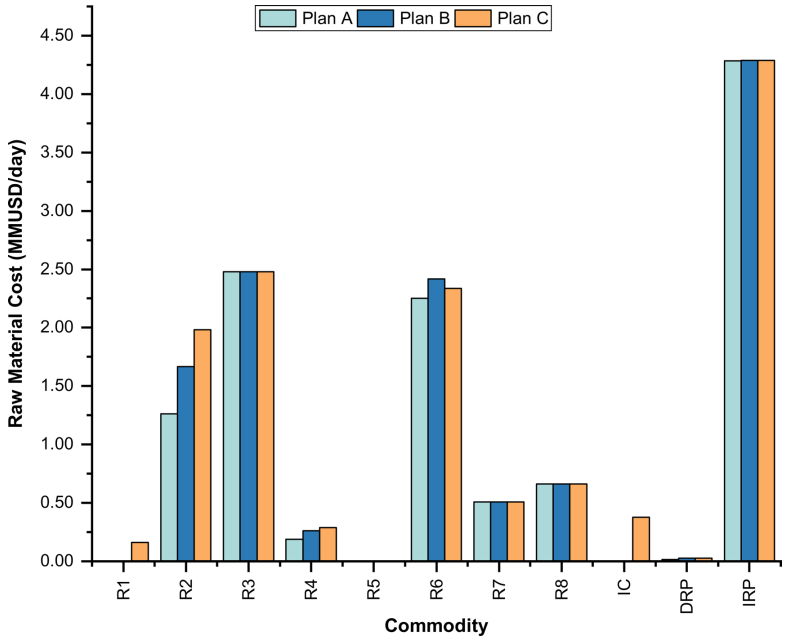


Figure 4.7. Optimized raw material supply in base scenario (Plans A, B and C were generated by ANTIGONE, BARON and the new clustering algorithm, respectively).

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 fuel blending, all three plans recommend purchasing the same amount of refined products.

As seen in Figure 4.8, logistics are another key differentiating factor between the optimized plans. Since the IRPC operates at a higher capacity in Plan C, greater crude supply needs to be delivered by pipelines to the IRPC, 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 4.7 and Figure 3.2). 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.

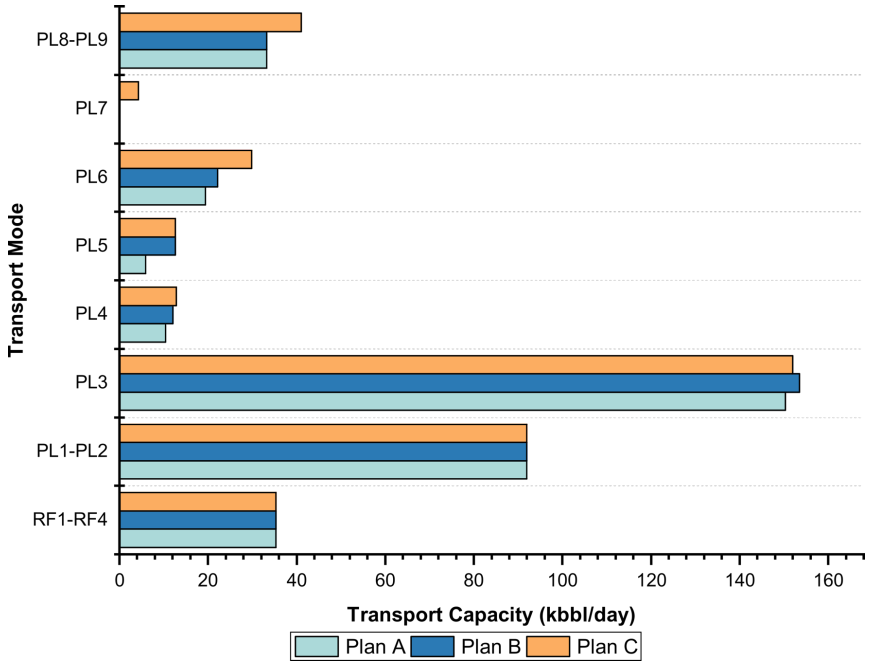


Figure 4.8. Optimized logistics in base scenario.

In terms of revenue, the three plans recommend producing the same volume of petrochemicals, waxes and industrial solvents (

Figure **4.9**).

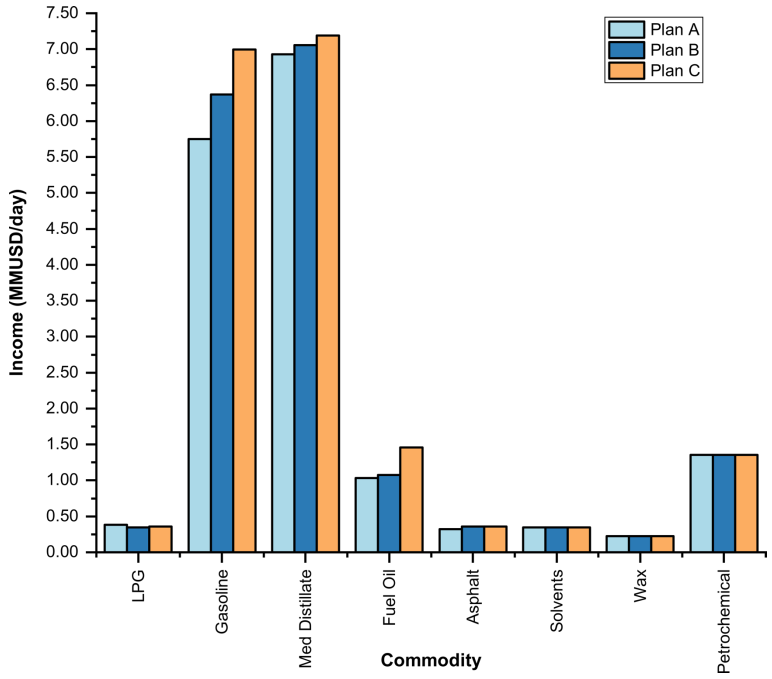


Figure 4.9. Optimized aggregated production income in base scenario.

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 4.10). 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 MMUSD/day is 11% and 6% higher with Plan C, respectively.

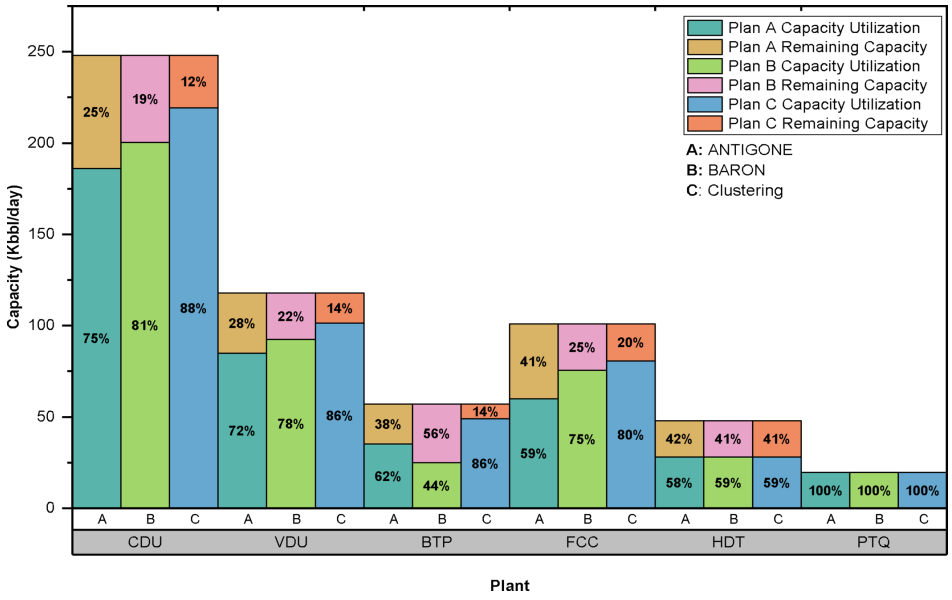


Figure 4.10. Optimized operational capacities in base scenario.

The final comparison concerns the operational capacity (Figure 4.10). 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.

4.6.2 Scenario without petrochemical processes

The scenario without refinery-petrochemical integration (WRPS) 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).

The results for Plan C, computed with the clustering algorithm, present 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 MMUSD/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 MMUSD/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 MMUSD/day.

These profits are 39% and 24% lower than the one from Plan C. The 10% relaxation 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^I, the profit had already increased to 1.97 MMUSD/day, improving slightly in Cluster III

to 2.00 MMUSD/day. The full results of this scenario are shown in Figure D. 1 - Figure D. 5 of the Appendix D.

4.6.3 Logistic disruption scenario

In the scenario of a 50% reduced capacity of pipeline PL3, which is responsible for delivering up to 80% of the crude to the IRPC (LDS), the clustering algorithm predicts a profit of 2.66 MMUSD/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 relaxation gap remains practically unchanged after the first cluster. The clustering approach returns a profit of 2.66 MMUSD/day, which is 19% and 11% better than ANTIGONE and BARON. The best possible solution (upper bound) is 3.05 MMUSD/day, leading to a relaxation gap of 13% after a runtime of 13,250 CPUs. The complete results of this scenario can be found in Figure E. 1 - Figure E. 5 of the Appendix E.

4.6.4 Gasoline demand reduction scenario

In the context of a gasoline demand reduced by 25%, 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 (DRS), the optimal profit after 20,890 CPUs is 2.83 MMUSD/day, with a relaxation gap of 8%. 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 Figure F. 1 – Figure F. 5 of the Appendix F.

4.6.5 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 relaxation gaps between 7% and 13%, which for the scale of this problem is a remarkable result. The results are summarized in Table 4.7.

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 fuel 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.

Table 4.7. Summary of performance of global optimisation solvers for the different scenarios.

	Base case scenario (BCS)		
	Clustering	**ANTIGONE**	**BARON**
Best-found Solution [MMUSD/day]	2.964	2.634	2.687
Best-possible Solution [MMUSD/day]	3.205	3.898	4.505
Relaxation gap* [%]	7.5	32	40
Runtime [h]	5.7	10.0	10.0
	No petrochemical integration scenario (WRPS)		
	Clustering	**ANTIGONE**	**BARON**
Best-found Solution [MMUSD/day]	2.009	1.219	1.574
Best-possible Solution [MMUSD/day]	2.233	2.926	3.536
Relaxation gap* [%]	10	58	58
Runtime [h]	5.8	10.0	10.0
	Logistic disruption scenario (LDS)		
	Clustering	**ANTIGONE**	**BARON**
Best-found Solution [MMUSD/day]	2.664	2.156	2.473
Best-possible Solution [MMUSD/day]	3.050	3.451	3.981
Relaxation gap* [%]	13	37	39
Runtime [h]	3.7	10.0	10.0
	Gasoline demand reduction scenario (DRS)		
	Clustering	**ANTIGONE**	**BARON**
Best-found Solution [MMUSD/day]	2.833	2.186	2.478
Best-possible Solution [MMUSD/day]	3.090	3.719	4.214
Relaxation gap* [%]	8	41	41
Runtime [h]	5.8	10.0	10.0

*Optimality gap is applied to ANTIGONE and BARON

Figure 4.11 shows the impact for the base scenario, where we can see that moving from fuel blending to crude allocation leads to a worse profit, LB (2.928 MMUSD/day), and a higher relaxation 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.

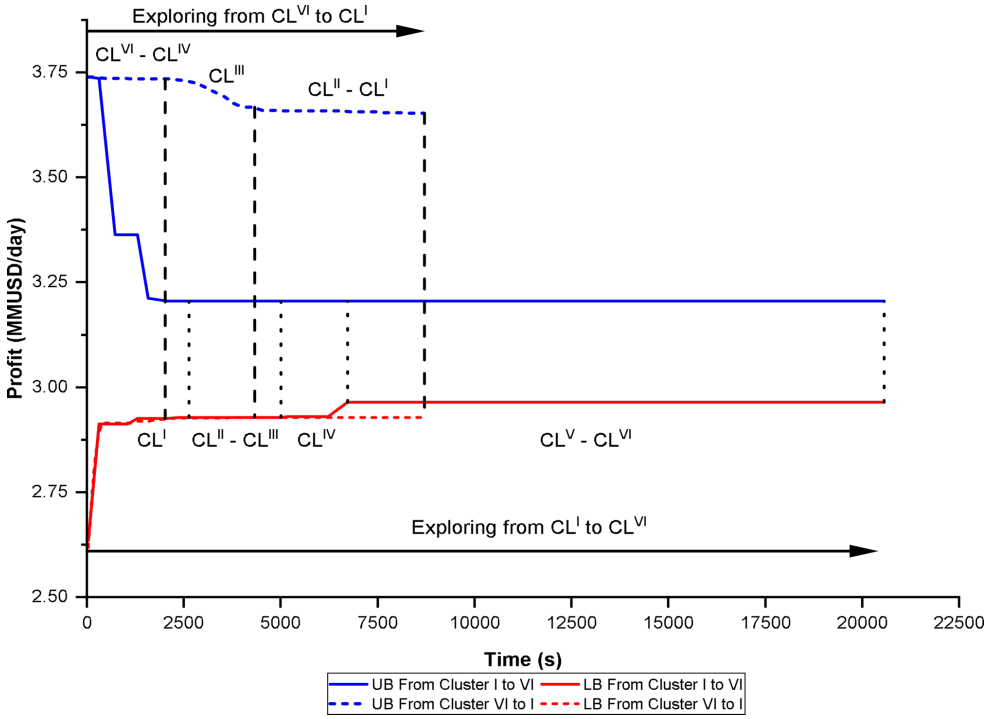


Figure 4.11. Performance of the clustering decomposition in the base case scenario when reversing the order of clusters.

4.6.6 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 280 discrete variables. Table 4.8 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 4.12, we show the performance of an algorithm similar to the one in sections 4.2 and 4.6.1 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 relaxation gaps at termination being roughly 22%. This

behaviour reflects the large size of the **PR** relaxation problems (see Table 4.8). Still, these values are much lower than the gaps obtained with ANTIGONE and BARON.

Table 4.8. Model size for relaxed problem PR as a function of the number of partitions N.

Partitioned variables	N	Variables			Total Equations
		Total	Continuous	Binary	
QF_u	2	19415	18760	655	55339
	4	27915	26216	1699	63319
	8	33575	31180	2395	71855
	12	42807	39716	3091	80947
	16	52595	48808	3787	89483
	20	61827	57344	4483	98019
	24	71059	65880	5179	106555
QF_u, $QS_{u,s}$	4	111081	98354	12727	186971
QF_u, $QS_{u,s}$, $Q_{u',s,u}$	4	269019	236024	32995	427201

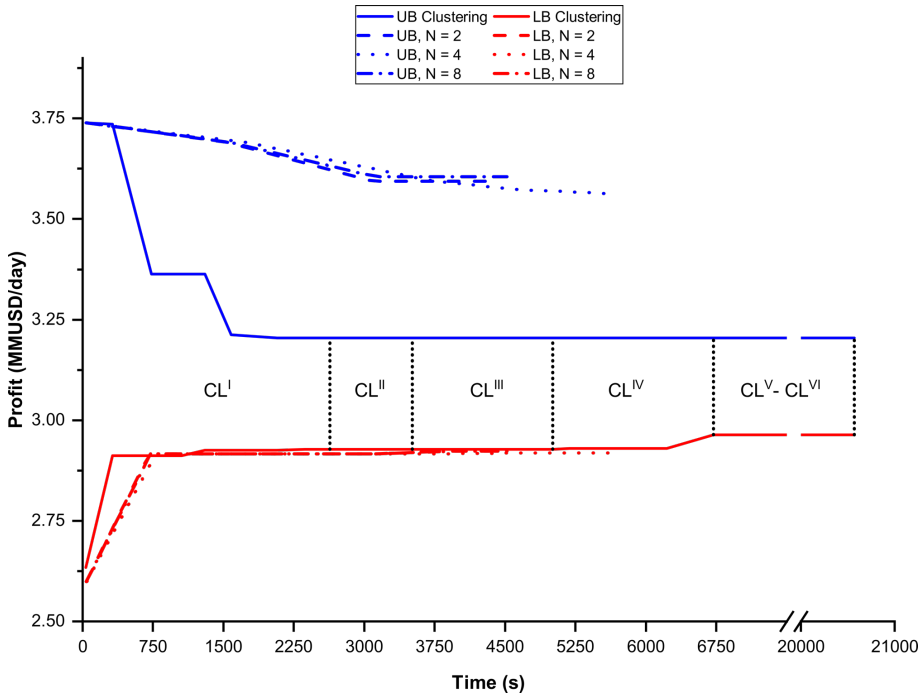


Figure 4.12. Comparison of proposed algorithm to one considering no clustering and static piecewise relaxations (results for different number of partitions N and base scenario).

4.7 Conclusions

This chapter has presented a deterministic global optimisation 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 instances of the pooling problem to global optimality, addressing the global optimisation for a large-scale refinery planning problem from literature 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 fuel 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 relaxation 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.

Even though the clustering approach was able to reach solutions for problem **P** with relaxation gaps between 8% and 14%, we explore in Chapter 5 another approach to decompose such MIQCQP problem, aiming to enhance the process clustering decomposition. Thus, we propose a Lagrangean decomposition-based algorithm which is coupled with our previous algorithm for deterministic global optimisation.

Chapter 5 LAGRANGEAN DECOMPOSITION STRATEGY

In the previous chapter, a deterministic global optimisation algorithm based on clustering decomposition (CL) obtained a better performance than global solvers BARON and ANTIGONE. But still, the relaxation gaps for several scenarios were above 10%.

In this chapter, the same case study of full-scale industrial integrated refinery-petrochemical complex (IRPC) with four scenarios is used to apply other methodology for the optimal short-term planning, based on a spatial Lagrangean decomposition (LD), by dividing the IRPC into multiple sections: crude management, refinery, fuel blending, and petrochemical production; in three decomposition strategies (the four separate IRPC sections, or three or two larger sections resulting first from the aggregation of refinery and fuel blending, and then from merging petrochemical production); so that the large-scale, nonconvex mixed-integer quadratically constrained quadratic program can be tackled to obtain better solutions and/or optimality gaps can be reduced further. This Lagrangean decomposition algorithm is benchmarked against other solution strategies. To the best of our knowledge, this work is the first to apply spatial LD to such a large-scale model and the novelty lies in formulating the decompositions for a full-scale industrial IRPC. This chapter is organized as follows: Section 5.1 focus on the literature background. Section 5.2 details the three decomposition strategies following by the description of the Lagrangean decomposition algorithm, and the coordination between the subproblems,

which involves iterating over the multipliers to generate a solution to the original problem (Section 5.3), then the computational results for the different scenarios and decomposition strategies are shown in Section 5.4. Finally, Section 5.5 concludes the chapter.

5.1 Lagrangean decomposition in refinery planning

Neiro and Pinto (2004) proposed a general framework for modelling petroleum supply chains, where process units, tanks and pipelines are linked through intermediate streams coming from mixers and splitters. This formulation exhibits a block structure, which is amenable to a decomposition technique such as Lagrangean decomposition (LD) (Guignard and Siwhan Kim, 1987). LD replaces the solution of a large optimisation model by a series of smaller subproblems and updates the Lagrange multipliers connecting theses subproblems iteratively. In multi-plant, multiperiod production planning problems, both spatial and temporal decompositions may be developed. The former entails dualizing the mass balances around plants and markets, while the latter dualizes the inventory equations that connect variables in consecutive time periods. Grossmann and co-workers (Jackson and Grossmann, 2003; Terrazas-Moreno et al., 2011) showed that the choice of complicating constraints to dualize can have a significant impact on computational performance and observed that temporal LD tends to provide tighter bounds. Neiro and Pinto (2006) applied temporal LD to solve a multiperiod single-refinery MINLP planning problem under uncertainty. Each realization of the uncertainty was given by a set of discrete scenarios comprising the crude oil procurement costs, product selling prices and demands. Then, a series of subproblems representing the combination of each time period and uncertainty scenario were solved iteratively and the Lagrange multipliers were updated using a sub-gradient method (Fisher, 1981). Zhao et al. (2017) presented a LD approach to solve a multiperiod MINLP planning problem for a petroleum refinery coupled with an ethylene plant, where the refinery sends fuel gas, ethane, propylene, naphtha, atmospheric gas

oil and heavy gas oil to the ethylene plant, which provides hydrogen, residual fuel oil, and pyrolysis gasoline to the refinery in return. The decomposition strategy consisted of duplicating the variables of the material streams connecting the refinery and ethylene plant and dualizing the constraints equating both sets of variables. This resulted in a MILP subproblem for the refinery and a MINLP subproblem for the ethylene plant, which although simpler than the original model remained challenging to solve to global optimality. The Lagrange multipliers were updated following the hybrid approach proposed by Mouret et al. (2011), which combines sub-gradient (Fisher, 1981), cutting planes (Cheney and Goldstein, 1959; Kelley, 1960) and trust-region (Marsten et al., 1975) methods.

Lagrangean decomposition has also been applied to integrate crude oil scheduling operations and refinery planning. Mouret et al. (2011) selected the CDU feedstock as the linking variable between the scheduling operations (MINLP subproblem) and refinery planning (NLP subproblem). For a given crude oil price, each subproblem could be solved independently to global optimality, making the spatial Lagrangean decomposition computationally tractable. Recently, Yang et al. (2020) proposed a multi-scale approach for the integration of a continuous-time MINLP model for crude oil scheduling and a discrete-time NLP for refinery planning, again using the hybrid approaches of Mouret et al. (2011) and Oliveira et al. (2013) to solve the dual subproblems.

5.2 IRPC decomposition

The monolithic model of the IRPC is too complex to be solved to global optimality using state-of-the-art solvers for deterministic global optimisation (Uribe-Rodriguez et al., 2020). Instead, by applying a Lagrangean decomposition-based algorithm, the IRPC can be divided into a few sections or business units that might be solved independently to global optimality. How to best decompose an IRPC is scenario-specific and remains an open question in general.

Therefore, three different decompositions have been investigated in this work. Details about the mathematical formulations for each subproblem are provided in Appendix G; information regarding availability, costs, and specifications for raw-materials and linking-streams, as well as the specifications and demands for the petrochemical and fuel products are provided in Appendix C.

5.2.1 Two sections: CM-RPB

The simplest decomposition entails a split between crude management (**CM**) on the one hand, and a merged section of refinery (**R**EF), petrochemical (**P**TQ) and fuel blending (F**B**) on the other hand, denoted as **RPB**. Solving each of the two subproblems independently creates an imbalance between the flowrates, compositions and bulk properties of the crudes leaving the crude blend tanks and those reaching the CDU charge tanks (Figure 5.1). In effect, the **CM** subproblem seeks to maximize profit by buying cheap crude oil from the market, minimizing the transportation cost to the refinery and selling crude blends at the highest price, regardless of the operational performance of the **RPB** section. Inversely, the **RPB** subproblem seeks to maximize profit by buying enough quantity of good-quality crude blends from **CM** at a cheap price, regardless of the costs incurred on the **CM** section by procuring and delivering the crudes.

Denoting by $e \in E^{\mathrm{CB}} \coloneqq \{\mathrm{CB1}, \ldots, \mathrm{CB8}\}$ the process streams connecting **CM** to **RPB**, the linking variables consist of the flowrate QF_e^i and properties $PF_{e,p}^i$, with $i \in \{\mathrm{CM,RPB}\}$ and index p representing either a bulk property (specific gravity, sulphur content, total acid number) or volumetric composition of the crude blend. The revenue of the **CM** section and cost of the **RPB** section associated to the linking streams can be computed as shown in Eqs. (5-1) - (5-2).

$$\mathrm{revenue}^{\mathrm{CM}} = \sum_{e \in E^{\mathrm{CB}}} \lambda_e^{\mathrm{CM,RPB}} \cdot QF_e^{\mathrm{CM}} + \sum_{e \in E^{\mathrm{CB}}} \sum_{p \in P^{\mathrm{CB}}} \lambda_{e,p}^{\mathrm{CM,RPB}} \cdot PF_{e,p}^{CM} \tag{5-1}$$

$$\mathrm{cost}^{\mathrm{RPB}} = \sum_{e \in E^{\mathrm{CB}}} \lambda_e^{\mathrm{CM,RPB}} \cdot QF_e^{\mathrm{RPB}} + \sum_{e \in E^{\mathrm{CB}}} \sum_{p \in P^{\mathrm{CB}}} \lambda_{e,p}^{\mathrm{CM,RPB}} \cdot PF_{e,p}^{\mathrm{RPB}} \tag{5-2}$$

where the multipliers $\lambda_e^{\mathrm{CM,RPB}}$ and $\lambda_{e,p}^{\mathrm{CM,RPB}}$ act as the marginal prices for the availability and properties of a crude blend, respectively.

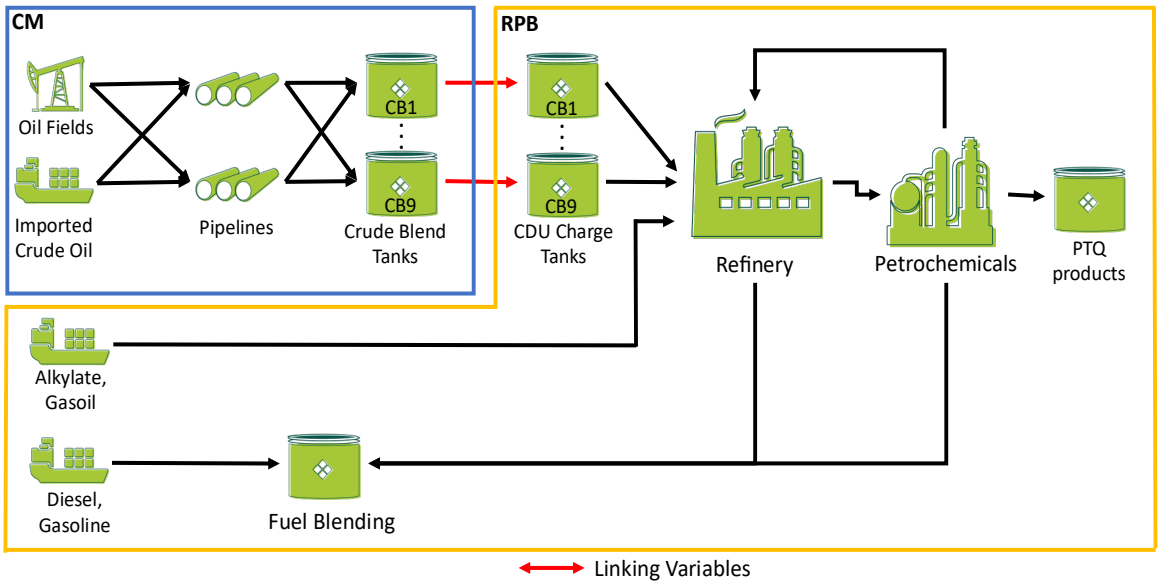


Figure 5.1. Two-level decomposition between CM and RPB sections.

5.2.2 Three sections: CM-RB-PTQ

The next decomposition level entails three subproblems, crude management (**CM**), refinery (**R**EF) merged with fuel blending (F**B**), denoted as **RB**, and petrochemical (**PTQ**). The connecting streams between **CM** and **RB** are identical to those of the previous decomposition (Section 5.2.1 and Figure 5.1). A second bidirectional market is created between **RB**, which sells materials for petrochemical production, and **PTQ**, which provides hydrogen to hydrotreating processes, raffinate for specialty solvent production, and components to improve gasoline quality (Figure 5.2). The intermediate refined streams from **RB** to **PTQ** are $e \in E^{\mathrm{RP}} := \{\mathrm{CH_4}, \mathrm{VirginNaphtha}, \mathrm{Olefins}, \mathrm{Ethylene}\}$ and from **PTQ** to **RB** are $e' \in E^{\mathrm{PR}} := \{\mathrm{H_2}, \mathrm{GasolineComponents}, \mathrm{Raffinate}\}$. Since, **RB** also buys crude blends from **CM**, the profit from **RB** is maximized when buying cheap precursor materials and selling their products at the highest possible price. In the same way, **PTQ** should buy natural gas, ethylene, olefins, and virgin naphtha at the lowest possible cost and it should sell hydrogen, raffinate and gasoline

components at the highest possible price. The commodities QF_e^{RB} and QF_e^{PTQ} are traded at the price $\lambda_e^{\mathrm{RB,PTQ}}$, whereas $QF_{e'}^{\mathrm{RB}}$ and $QF_{e'}^{\mathrm{PTQ}}$ are negotiated at the price $\lambda_{e'}^{\mathrm{PTQ,RB}}$. There are also the penalty costs $\lambda_{e,p}^{\mathrm{RB,PTQ}}$ and $\lambda_{e',p'}^{\mathrm{PTQ,RB}}$ associated with the commodity qualities $PF_{e,p}^{\mathrm{RB}}$, $PF_{e,p}^{\mathrm{PTQ}}$, $PF_{e',p'}^{\mathrm{RB}}$ and $PF_{e',p'}^{\mathrm{PTQ}}$ for either the refinery properties $p \in P^{\mathrm{RP}}$ or the petrochemical properties $p' \in P^{\mathrm{PR}}$. Since the exchange between **RB** and **PTQ** is bidirectional and recalling that **RB** also trades crude blends with **CM**, the revenues and costs can be computed as follows in Eqs. (5-3) - (5-9):

$$\mathrm{revenue}^{\mathrm{CM}} = \sum_{e\in E^{\mathrm{CB}}} \lambda_e^{\mathrm{CM,RB}} \cdot QF_e^{\mathrm{CM}} + \sum_{e\in E^{\mathrm{CB}}} \sum_{p\in P^{\mathrm{CB}}} \lambda_{e,p}^{\mathrm{CM,RB}} \cdot PF_{e,p}^{\mathrm{CM}} \tag{5-3}$$

$$\mathrm{revenue}^{\mathrm{RB}} = \sum_{e\in E^{\mathrm{RP}}} \lambda_e^{\mathrm{RB,PTQ}} \cdot QF_e^{\mathrm{RB}} + \sum_{e\in E^{\mathrm{RP}}} \sum_{p\in P^{\mathrm{RP}}} \lambda_{e,p}^{\mathrm{RB,PTQ}} \cdot PF_{e,p}^{\mathrm{RB}} \tag{5-4}$$

$$\mathrm{cost}^{\mathrm{PTQ}} = \sum_{e\in E^{\mathrm{RP}}} \lambda_e^{\mathrm{RB,PTQ}} \cdot QF_e^{\mathrm{PTQ}} + \sum_{e\in E^{\mathrm{RP}}} \sum_{p\in P^{RP}} \lambda_{e,p}^{\mathrm{RB,PTQ}} \cdot \mathrm{PF}_{e,p}^{\mathrm{PTQ}} \tag{5-5}$$

$$\mathrm{cost}^{\mathrm{RB}} = \mathrm{cost}^{\mathrm{CM,RB}} + \mathrm{cost}^{\mathrm{PTQ,RB}} \tag{5-6}$$

$$\mathrm{cost}^{\mathrm{CM,RB}} = \sum_{e\in E^{\mathrm{CB}}} \lambda_e^{\mathrm{CM,RB}} \cdot QF_e^{\mathrm{RB}} + \sum_{e\in E^{\mathrm{CB}}} \sum_{p\in P^{\mathrm{CB}}} \lambda_{e,p}^{\mathrm{CM,RB}} \cdot PF_{e,p}^{\mathrm{RB}} \tag{5-7}$$

$$\mathrm{cost}^{\mathrm{PTQ,RB}} = \sum_{e'\in E^{\mathrm{PR}}} \lambda_{e'}^{\mathrm{PTQ,RB}} \cdot QF_{e'}^{\mathrm{RB}} + \sum_{e'\in E^{\mathrm{PR}}} \sum_{p'\in E^{\mathrm{PR}}} \lambda_{e,p'}^{\mathrm{PTQ,RB}} \cdot PF_{e,p'}^{\mathrm{RB}} \tag{5-8}$$

$$\mathrm{revenue}^{\mathrm{PTQ}} = \sum_{e\prime\in E^{\mathrm{PR}}} \lambda_{e'}^{\mathrm{PTQ,RB}} \cdot QF_{e'}^{\mathrm{PTQ}} + \sum_{e'\in E^{\mathrm{PR}}} \sum_{p'\in E^{\mathrm{RP}}} \lambda_{e,p'}^{\mathrm{PTQ,RB}} \cdot PF_{e,p'}^{\mathrm{PTQ}} \tag{5-9}$$

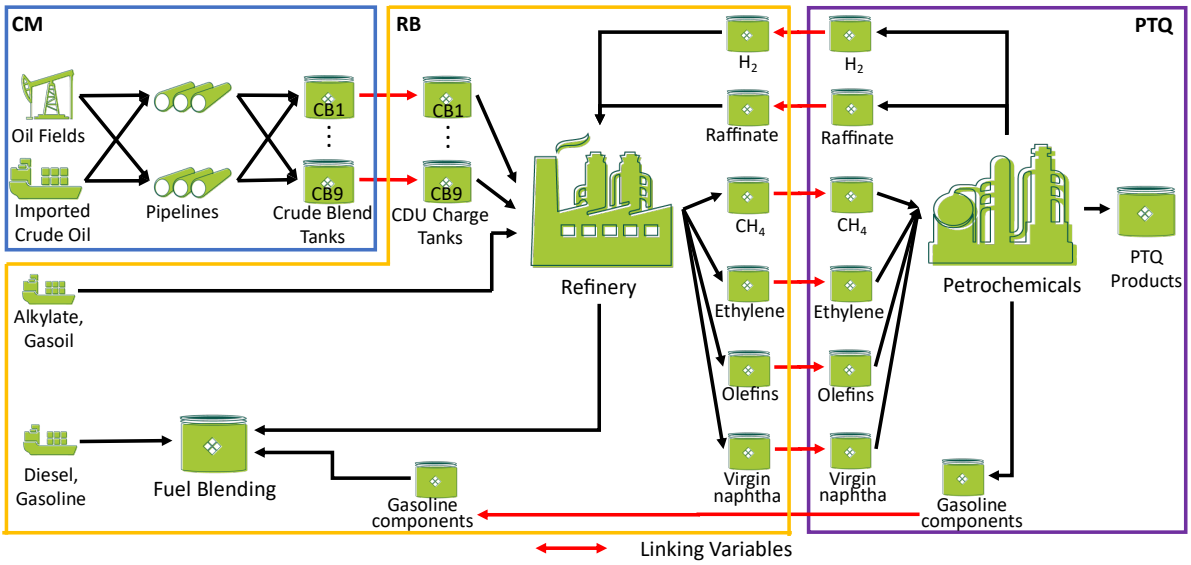


Figure 5.2. Three-level decomposition between CM, RB and PTQ sections.

5.2.3 Four sections: CM-REF-PTQ-FB

The final decomposition level additionally separates fuel blending (**FB**) from refining (**REF**), leading to four sections. **REF** can provide intermediate streams to **FB**, such as naphtha (inaphtha), jet (ijet), diesel (idiesel), and fuel (ifuel) generated from crude and vacuum distillation columns, catalytic processes such as hydrotreating and fluid catalytic cracking, thermal processes (e.g., visbreaking), and solvent extraction processes such as deasphalting, among others. The intermediate refined streams between **REF** and **FB** are $e \in E^{\mathrm{RB}} := \{\text{inaphtha, ijet, idiesel, ifuel}\}$ with properties $p \in P^{\mathrm{RB}} := \{\text{SPG, viscosity, sulphur, RON, MON, cetane}\}$.

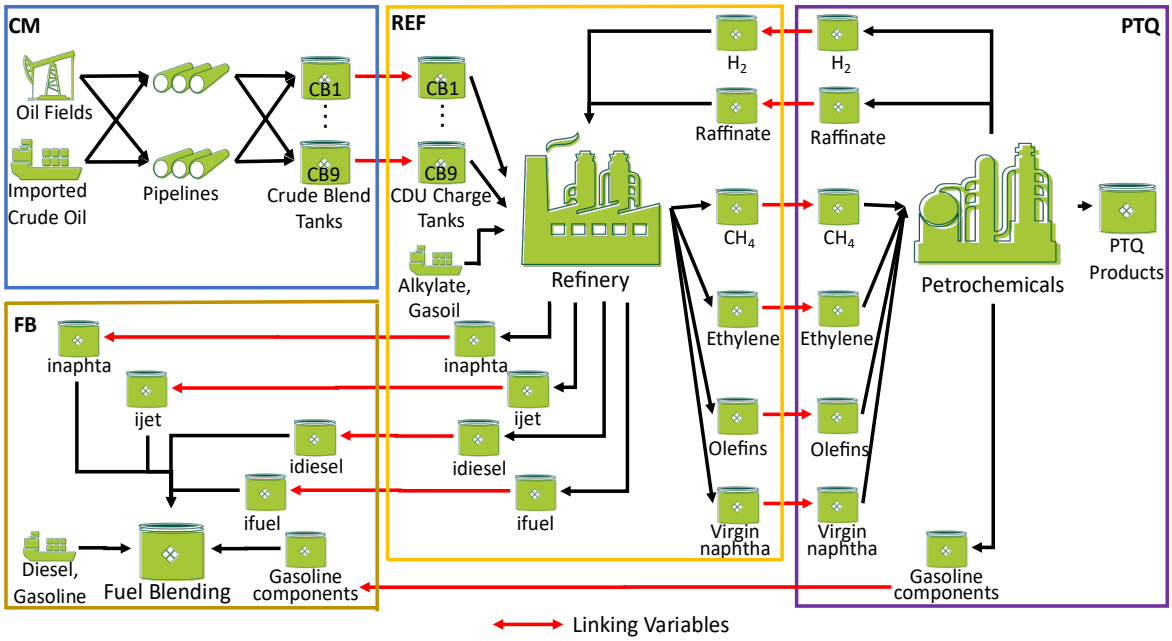


Figure 5.3. Four-level decomposition between CM, REF, PTQ and FB sections.

The commodities QF_e^{REF} and QF_e^{FB} are traded at the prices $\lambda_e^{\mathrm{REF,FB}}$, with penalty costs $\lambda_{e,p}^{\mathrm{REF,FB}}$ associated with the commodity qualities $PF_{e,p}^{\mathrm{REF}}$ and $PF_{e,p}^{\mathrm{FB}}$. In addition, **FB** receives gasoline components $(e' \in E^{\mathrm{PB}} := \{\text{GasolineComponents}\}, p' \in P^{\mathrm{PB}} := \{\text{SPG,sulphur,RON,MON}\})$ from **PTQ** (Figure 5.3). The commodities $QF_{e'}^{\mathrm{PTQ}}$ and $QF_{e'}^{\mathrm{FB}}$ are traded at the prices $\lambda_{e'}^{\mathrm{PTQ,FB}}$, with penalty costs $\lambda_{e,p'}^{\mathrm{PTQ,FB}}$ associated with the commodity qualities $PF_{e,p'}^{\mathrm{PTQ}}$ and $PF_{e,p'}^{\mathrm{FB}}$.

The profit out of the **PTQ** section can be maximized by buying cheap natural gas, ethylene, olefins, and virgin naphtha from the refinery, while selling hydrogen, raffinate and gasoline components at a high price. The **FB** section receives gasoline components from **PTQ** and intermediate refined streams from **REF**. It can also operate as an import terminal, satisfying fuel demand regardless of **REF** and **PTQ** operations. Optimizing each section separately for given prices λ_e^{ij} of the traded commodities between sections $i, j \in \{\text{CM,REF,PTQ,FB}\}$, with $i \neq j$, creates an imbalance between the flowrates QF_e^i and QF_e^j and the properties $PF_{e,p}^i$ and $PF_{e,p}^j$ as illustrated with the yellow circles in Figure 5.4.

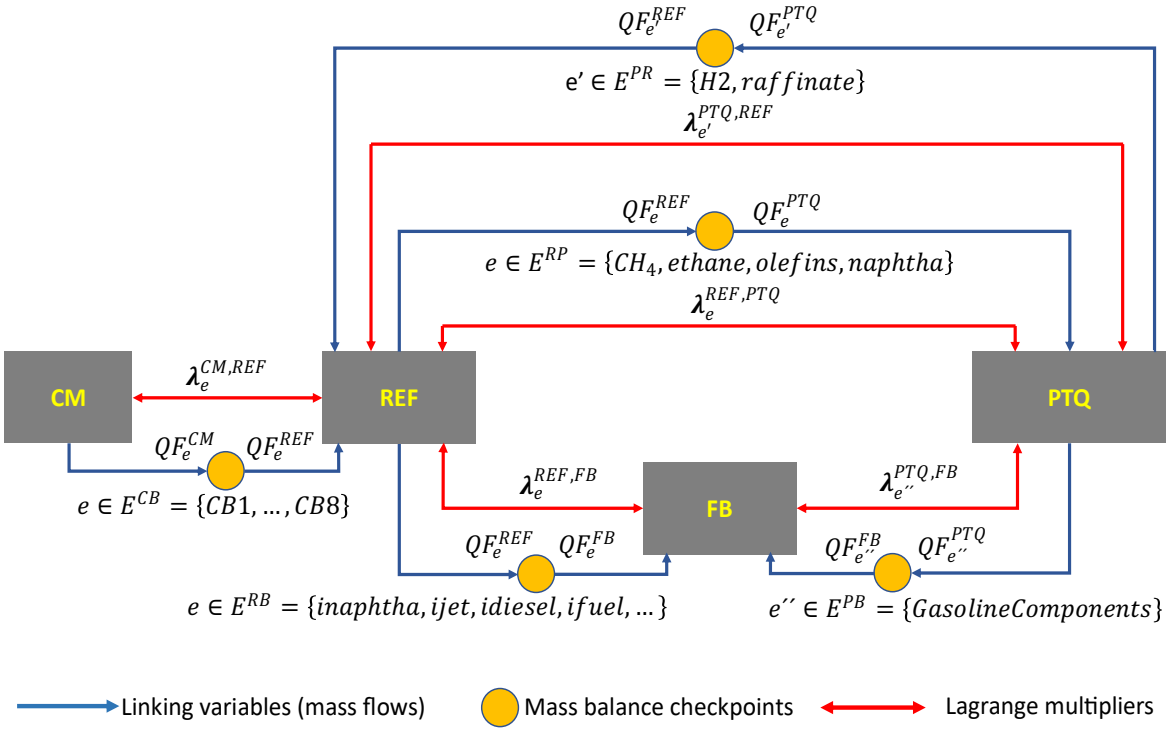


Figure 5.4. Imbalances between the flows and properties from different sections in the four-level decomposition.

REF trades intermediated streams with **CM**, **PTQ** and **FB**, thereby making a profit by selling intermediated streams to **PTQ** ($\text{revenue}^{\text{REF,PTQ}}$) and **FB** ($\text{revenue}^{\text{REF,FB}}$). On the other hand, **REF** supports the costs of any crude blends traded with **CM** ($\text{cost}^{\text{CM,REF}}$) and of any

intermediate streams received from **PTQ** ($\text{cost}^{\text{PTQ,REF}}$). The revenue for **CM** consists of the crude blends sold to **REF**.

$$\text{revenue}^{\text{REF}} = \text{revenue}^{\text{REF,PTQ}} + \text{revenue}^{\text{REF,FB}} \tag{5-10}$$

$$\text{revenue}^{\text{REF,PTQ}} = \sum_{e\in E^{\text{RP}}} \lambda_e^{\text{REF,PTQ}} \cdot QF_e^{\text{REF}} + \sum_{e\in E^{\text{RP}}} \sum_{p\in P^{\text{RP}}} \lambda_{e,p}^{\text{REF,PTQ}} \cdot PF_{e,p}^{\text{REF}} \tag{5-11}$$

$$\text{revenue}^{\text{REF,FB}} = \sum_{e\in E^{\text{RB}}} \lambda_e^{\text{REF,FB}} \cdot QF_e^{\text{REF}} + \sum_{e\in E^{\text{RB}}} \sum_{p\in P^{\text{RB}}} \lambda_{e,p}^{\text{REF,FB}} \cdot PF_{e,p}^{\text{REF}} \tag{5-12}$$

$$\text{cost}^{\text{REF}} = \text{cost}^{\text{CM,REF}} + \text{cost}^{\text{PTQ,REF}} \tag{5-13}$$

$$\text{cost}^{\text{CM,REF}} = \sum_{e\in E^{\text{CB}}} \lambda_e^{\text{CM,REF}} \cdot QF_e^{\text{REF}} + \sum_{e\in E^{\text{CB}}} \sum_{p\in P^{\text{CB}}} \lambda_{e,p}^{\text{CM,REF}} \cdot PF_{e,p}^{\text{REF}} \tag{5-14}$$

$$\text{cost}^{\text{PTQ,REF}} = \sum_{e'\in E^{\text{PR}}} \lambda_{e'}^{\text{PTQ,REF}} \cdot QF_{e'}^{\text{REF}} + \sum_{e'\in E^{\text{PR}}} \sum_{p'\in P^{\text{PR}}} \lambda_{e,p'}^{\text{PTQ,REF}} \cdot PF_{e,p'}^{\text{REF}} \tag{5-15}$$

$$\text{revenue}^{\text{CM}} = \sum_{e\in E^{\text{CB}}} \lambda_e^{\text{CM,REF}} \cdot QF_e^{\text{CM}} + \sum_{e\in E^{\text{CB}}} \sum_{p\in P^{\text{CB}}} \lambda_{e,p}^{\text{CM,REF}} \cdot PF_{e,p}^{\text{CM}} \tag{5-16}$$

PTQ exchanges materials with **REF** and **FB**. The former entails a bidirectional trading between **PTQ** and **REF**, while **PTQ** sells components for the gasoline blending to **FB** in the latter. Thus, the revenue for **PTQ** results from trading gasoline components with **FB (**$\text{revenue}^{\text{PTQ,FB}}$) and from selling other streams to **REF (**$\text{revenue}^{\text{PTQ,REF}}$). On the other hand, the cost of **PTQ** (cost^{PTQ}) is incurred by the procurement of intermediate product streams from **REF**.

$$\text{revenue}^{\text{PTQ}} = \text{revenue}^{\text{PTQ,REF}} + \text{revenue}^{\text{PTQ,FB}} \tag{5-17}$$

$$\text{revenue}^{\text{PTQ,REF}} = \sum_{e'\in E^{\text{PR}}} \lambda_{e'}^{\text{PTQ,REF}} \cdot QF_{e'}^{\text{PTQ}} + \sum_{e'\in E^{\text{PR}}} \sum_{p'\in P^{\text{PR}}} \lambda_{e,p'}^{\text{PTQ,REF}} \cdot PF_{e,p'}^{\text{PTQ}} \tag{5-18}$$

$$\text{revenue}^{\text{PTQ,FB}} = \sum_{e''\in E^{\text{PB}}} \lambda_{e''}^{\text{PTQ,FB}} \cdot QF_{e''}^{\text{PTQ}} + \sum_{e''\in E^{\text{PB}}} \sum_{p''\in P^{\text{PB}}} \lambda_{e',p''}^{\text{PTQ,FB}} \cdot PF_{e',p''}^{\text{PTQ}} \tag{5-19}$$

$$\text{cost}^{\text{PTQ}} = \sum_{e\in E^{\text{RP}}} \lambda_e^{\text{REF,PTQ}} \cdot QF_e^{\text{PTQ}} + \sum_{e\in E^{\text{RP}}} \sum_{p\in P^{\text{RP}}} \lambda_{e,p}^{\text{REF,PTQ}} \cdot PF_{e,p}^{\text{PTQ}} \tag{5-20}$$

Finally, **FB** buys streams for fuel blending from both **REF** ($\text{cost}^{\text{REF,FB}}$) and **PTQ** ($\text{cost}^{\text{PTQ,FB}}$).

$$\text{cost}^{\text{FB}} = \text{cost}^{\text{REF,FB}} + \text{cost}^{\text{PTQ,FB}} \tag{5-21}$$

$$\text{cost}^{\text{REF,FB}} = \sum_{e\in E^{\text{RB}}} \lambda_e^{\text{REF,FB}} \cdot QF_e^{\text{FB}} + \sum_{e\in E^{\text{RB}}} \sum_{p\in P^{\text{RB}}} \lambda_{e,p}^{\text{REF,FB}} \cdot PF_{e,p}^{\text{FB}} \quad (5\text{-}22)$$

$$\text{cost}^{\text{PTQ,FB}} = \sum_{e''\in E^{\text{PB}}} \lambda_{e''}^{\text{PTQ,FB}} \cdot QF_{e''}^{\text{FB}} + \sum_{e''\in E^{\text{PB}}} \sum_{p''\in P^{\text{PB}}} \lambda_{e',p''}^{\text{PTQ,FB}} \cdot PF_{e',p''}^{\text{FB}} \quad (5\text{-}23)$$

5.3 Methodology

Problem **P** is formulated in this chapter, with a change in the notation, letting i and j to be indices referring to the subproblems. Then, short-term MIQCQP planning problem for the IRPC is now formulated as follows:

$$\begin{aligned} z^*: &= \max f_0(x,y) \\ &\text{s.t. } f_m(x,y) \le 0 \ \forall m \in \{1,\dots,M\} \\ &\quad x \in [x^L, x^U] \subseteq \mathbb{R}_+^p,\ y \in \{0,1\}^q \end{aligned} \quad \textbf{(P)}$$

Where z^* is the maximum profit, x is a p −dimensional vector of non-negative continuous variables constrained between lower x^L and upper x^U bounds, and y is a q −dimensional vector of binary variables used to select process operating conditions.

The functions f_m: $\mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$, with $m = 0$ (objective function), and with $m = 1,\dots,M$ (constraints of **P**) are quadratic in x and linear in y:

$$f_m(x,y) \coloneqq \sum_{(r,s)\in BL_m} a_{rsm} x_r x_s + B_m x + C_m y + d_m$$

Where BL_m is an (r,s)-index set defining the bilinear terms $x_r x_s$ present in function m, parameters a_{rsm} and d_m are scalars, and B_m and C_m are row vectors.

5.3.1 Lagrangean decomposition and relaxation

The reformulation **P'** of problem **P** for a set of $S > 1$ subproblems entails duplicating the continuous variables describing the connecting streams between sections and assigning them to different sets of constraints (Grossmann, 2021; Guignard and Siwhan Kim, 1987). A few of these duplicated variables are displayed next to the mass balance checkpoints represented as yellow circles in Figure 5.4 (e.g., the flowrates QF_e^{REF} and QF_e^{FB} of every stream e linking the

refinery and fuel blending section). Formally, problem **P'** is made equivalent to **P** by adding the constraint $x_v^i = x_v^j \; \forall i, j > i, v \in X_{ij}$, where X_{ij} is the index set of the complicating variables (flowrates and properties) of all streams linking subproblems i and j.

$$
\begin{aligned}
z^* \coloneqq \max \; & \textstyle\sum_{i=1}^{S} f_0^i\left(x^i, y^i\right) \\
\text{s.t. } & f_{m_i}^i\left(x^i, y^i\right) \leq 0 \; \forall i \in \{1, \dots, S\}, m_i \in \{1, \dots, M\} \\
& x_v^i - x_v^j = 0 \; \forall i, j \in \{1, \dots, S\}, i < j, v \in X_{ij} \\
& x \in [x^{\mathrm{L}}, x^{\mathrm{U}}] \subseteq \mathbb{R}_+^{p'}, y \in \{0,1\}^q
\end{aligned}
\qquad \textbf{(P')}
$$

Each of the M constraints is allocated to a given subproblem, and the objective function is summing the objective terms of all the subproblems, with x^i and y^i denoting the vectors of continuous and binary variables that participate in subproblem i, respectively. The problem reformulation **P'** makes it possible to apply a solution strategy based on Lagrangean decomposition. In particular, a Lagrangean relaxation (Guignard, 2003; Guignard and Siwhan Kim, 1987) $\mathbf{LR}_{\boldsymbol{\lambda}}$ of problem **P'** can be obtained by transferring into the objective function the complicating constraints $x_v^i = x_v^j$ multiplied by their Lagrange multipliers λ_v^{ij}, which can either take positive or negative values:

$$
\begin{aligned}
z_\lambda^{\mathrm{LR}} \coloneqq \max \; & \textstyle\sum_{i=1}^{S} f_0^i\left(x^i, y^i\right) + \sum_{i=1}^{S-1} \sum_{j=i+1}^{S} \sum_{v \in X_{ij}} \lambda_v^{ij}\left(x_v^i - x_v^j\right) \\
\text{s.t. } & f_{m_i}^i\left(x^i, y^i\right) \leq 0 \; \forall i \in \{1, \dots, S\}, m_i \in \{1, \dots, M\} \\
& x \in [x^{\mathrm{L}}, x^{\mathrm{U}}] \subseteq \mathbb{R}_+^{p'}, y \in \{0,1\}^q
\end{aligned}
\qquad \textbf{(LR}_{\boldsymbol{\lambda}}\textbf{)}
$$

For fixed values of the multipliers λ_v^{ij}, problem $\mathbf{LR}_{\boldsymbol{\lambda}}$ can be decomposed into S parametric optimisation problems, which are solved independently from one another:

$$
\begin{aligned}
z_\lambda^{i,\mathrm{LD}} \coloneqq \max \; & f_0^i\left(x^i, y^i\right) + \textstyle\sum_{j=i+1}^{S} \sum_{v \in X_{ij}} \lambda_v^{ij} x_v^i - \sum_{j=1}^{i-1} \sum_{v \in X_{ji}} \lambda_v^{ji} x_v^i \\
\text{s.t. } & f_{m_i}^i\left(x^i, y^i\right) \leq 0 \; \forall m_i \in \{1, \dots, M\} \\
& x \in [x^{\mathrm{L}}, x^{\mathrm{U}}] \subseteq \mathbb{R}_+^{p'}, y \in \{0,1\}^q
\end{aligned}
\qquad \textbf{(LD}_{\boldsymbol{\lambda}}^{\mathbf{i}}\textbf{)}
$$

Therefore, $z_\lambda^{\mathrm{LD}} \coloneqq \sum_{i=1}^{S} z_\lambda^{i,\mathrm{LD}}$ provides an upper bound on the optimal value z^* of **P**.

5.3.2 Dual problem

A standard practice is to solve a Lagrangean dual problem for determining values of the multipliers λ_v^{ij} that minimize the upper bound z_λ^{LD} from Lagrangean relaxation (Grossmann, 2021). Grossmann and co-workers (Mouret et al., 2011; Oliveira et al., 2013; Yang et al., 2020) developed a hybrid method for updating the Lagrange multipliers by combining a subgradient method (Held et al., 1974; Held and Karp, 1971) with a cutting plane approach (Cheney and Goldstein, 1959), trust-region method (Marsten et al., 1975) and volume algorithm (Barahona and Anbil, 2000). Specifically, at a given iteration $K > 0$, the following LP is solved to update the Lagrange multipliers $\lambda_v^{ij,K}$ that feed into subproblem $\mathbf{LD_\lambda^i}$ at the next iteration $K+1$:

$$\begin{aligned} z_K^{\mathrm{DP}} :=\ & \min \eta \\ & \text{s.t. } \eta \ \geq\ \bar{f}_k\left(\lambda_v^{ij,k}\right) \forall k \in \{1,\dots,K\} \\ & \left|\lambda_v^{ij,K} - \lambda_v^{ij,K-1}\right| \ \leq\ \Delta_v^{ij}, \lambda_v^{ij,K} \in [\lambda^{\mathrm{L}}, \lambda^{\mathrm{U}}] \ \forall i,j \in \{1,\dots,S\}, i<j, v \in X_{ij} \end{aligned} \qquad (\mathbf{DP^K})$$

where the main decision variables are the Lagrange multipliers $\lambda_v^{ij,K}$ within the range $[\lambda^{\mathrm{L}}, \lambda^{\mathrm{U}}]$; $\lambda_v^{ij,K-1}$ are the values of the Lagrangean multipliers at the previous iteration $K-1$; and the augmented objective function $\bar{f}_k$ is given by:

$$\bar{f}_k\left(\lambda_v^{ij,k}\right) \ :=\ \sum_{i=1}^{S}\left[f_0^i\left(x^{i,k}, y^{i,k}\right) + \sum_{j=i+1}^{S}\sum_{v\in X_{ij}} \lambda_v^{ij,k} x_v^{i,k} - \sum_{j=1}^{i-1}\sum_{v\in X_{ji}} \lambda_v^{ji,k} x_v^{i,k}\right] \qquad (5\text{-}24)$$

with $x^{i,k}$ and $y^{i,k}$ taking the optimal solution of subproblem $\mathbf{LD_\lambda^i}$ at iteration k.

In practice, the variability of z_λ^{LD} between iterations may be reduced by adjusting the trust-region radius Δ_v^{ij} of the Lagrange multipliers $\lambda_v^{ij,K}$ around $\lambda_v^{ij,K-1}$, before solving $\mathbf{DP^K}$ (Barahona and Anbil, 2000; Oliveira et al., 2013). The procedure used herein consists of determining an average deviation between the optimal values $x_v^{i,K}$ of the complicating variables in subproblem $\mathbf{LD_\lambda^i}$ at iteration K and the best feasible solution x_v^{i,K^*} of problem **P** up to iteration K, scaling the step-size α_v^{ij} between [0,1] (Eq. 5-25), and finally obtaining the trust-region

radius Δ_v^{ij} (Eq. 5-26). The larger the deviation of the linking variables, the greater the corresponding trust-region radius.

$$\alpha_v^{ij} := \frac{\left(\left|x_v^{i,K^*} - x_v^{i,K}\right| + \left|x_v^{j,K^*} - x_v^{j,K}\right|\right)}{\sum_{\substack{(i',j')=1 \\ i'<j'}}^{S}\left(\left|x_v^{i',K^*} - x_v^{i',K}\right| + \left|x_v^{j',K^*} - x_v^{j',K}\right|\right)} \tag{5-25}$$

$$\Delta_v^{ij} := \alpha_v^{ij}\left|\frac{\mathrm{UB}-\mathrm{LB}}{x_v^{i,K} - x_v^{j,K}}\right| \tag{5-26}$$

where UB is the tightest upper bound z_λ^{LD} found at iteration K from the Lagrangean relaxation of **P'**; and LB is the tightest lower bound at iteration K, as explained in the following subsection.

5.3.3 Lower bounding problem

The original problem **P** features binary variables and bilinear terms between continuous variables in the objective function and constraints. The classical approach to determining a lower bound on the optimal solution value of **P** entails fixing the values of the binary variables and solving the resulting NLP subproblem to local optimality with a suitable initialization. Any feasible solution of this NLP subproblem provides a lower bound z^{PF} on **P**. In practice, one may set the binary variables and initialize the continuous variables at the solution point of a MILP relaxation of **P**, constructed for instance from piecewise-linear relaxations of the bilinear terms. This procedure has been successfully applied to a variety of scheduling and planning problems dealing with petroleum refineries (Castillo et al., 2017; Castro, 2016; Mouret et al., 2011; Uribe-Rodriguez et al., 2020; Zhang et al., 2022, 2021). Nevertheless, if the bounds of the variables involved in the bilinear terms are wide, the MILP relaxation is weak, providing poor initial points to the NLP and affecting the performance of such two stage MILP – NLP approach.

To formulate a stronger convex relaxation, the decomposable structure of **P** into S subproblems can be exploited. It can be assumed that $z_\lambda^{i,\mathrm{LD}^*}$ is the global optimal solution of each

subproblem i, which can provide valid cuts into the feasible region of the MILP relaxation of **P**. These constraints are function of the Lagrange multipliers and the terms participating in the relaxed objective function $f_0^R(x, y, \lambda)$. Thus, the constraints $z_\lambda^{\mathrm{i,LD^*,k}} \geq f_0^R(x, y, \lambda)$ can enhance the quality of the MILP relaxation by appending a cut determined by each subproblem i (Grossmann and Karuppiah, 2008). It is applied herein by taking advantage of the Lagrangean dual problem solution to strengthen the piecewise-linear relaxations. Incidentally, the solution value z^{R} of any such MILP relaxation also provides an upper bound on **P**.

5.3.4 Lagrangean decomposition-based algorithm

The main steps of the Lagrangean decomposition algorithm for solving the MIQCQP problem **P** are summarized below:

Step 1: Specify the tuning parameters, including total maximal runtime (TotalMaxRunTime), maximal runtime (MaxRunTime) for solving each subproblem, relative optimality tolerance ϵ for the Lagrangean decomposition algorithm, relative optimality tolerance ϵ^{rel} for each Lagrangean relaxation subproblem, and maximum number of iterations $K^{\max}$. Set the lower bound $\mathrm{LB} = -\infty$, the upper bound $\mathrm{UB} = +\infty$, the initial values for the Lagrange multipliers $\lambda_v^{ij,0} = 0$, and the iteration counter $K = 1$.

Step 2: Search for a feasible solution z^* (primal bound) of problem **P**. If successful, set $\mathrm{LB} \leftarrow z^*$.

Step 3: Solve the S subproblems $\mathbf{LD_\lambda^i}$ with the current Lagrange multipliers $\lambda_v^{ij,K}$ to global optimality with relative tolerance ϵ^{rel} and maximal runtime MaxRunTime. Set each $z_\lambda^{i,\mathrm{LD}}$ to the best-possible solution (dual bound) of $\mathbf{LD_\lambda^i}$ at termination. If $z_\lambda^{\mathrm{LD}} = \sum_{i=1}^{S} z_\lambda^{i,\mathrm{LD}} < \mathrm{UB}$, update $\mathrm{UB} \leftarrow z_\lambda^{\mathrm{LD}}$.

Step 4: Append a new cut from the solution $z_{\lambda}^{\mathrm{i,LD}^*}$ of the Lagrangean dual problems $\mathbf{LD}_{\boldsymbol{\lambda}}^{\mathbf{i}}$ to the MILP relaxation of problem **P** and solve it by passing the optimal values for the continuous variables x^i and the discrete variables y^i from the S subproblems $\mathbf{LD}_{\boldsymbol{\lambda}}^{\mathbf{i}}$ as hint (choosing values from one of the subproblems for the duplicated variables). If $z^{\mathrm{R}} < \mathrm{UB}$, update $\mathrm{UB} \leftarrow z^{\mathrm{R}}$.

Step 5: Solve problem **P** to local optimality, by fixing the binary variables and initializing the continuous variables at the solution of the MILP relaxation in Step 4. If successful and $z^{\mathrm{PF}} > \mathrm{LB}$, update $\mathrm{LB} \leftarrow z^{\mathrm{PF}}$.

Step 6: If $(\mathrm{UB} - \mathrm{LB})/\mathrm{UB} \leq \epsilon$, terminate.

Step 7: Update the trust-region radius Δ_v^{ij} using Eqs. (5-25) - (5-26).

Step 8: Solve problem $\mathbf{DP}^{\mathbf{K}}$ to determine the next Lagrange multipliers $\lambda_v^{ij,K}$.

Step 9: If TotalMaxRunTime is exceeded or $K = K^{\max}$, terminate. Otherwise, set $K = K + 1$ and return to Step 3.

The Lagrangean decomposition algorithm was implemented in the modeling environment GAMS v33.2, setting a relative optimality tolerance $\epsilon = 0.05$ and allowing for a maximal runtime TotalMaxRunTime of 36,000 seconds. Step 2 of the algorithm relies on the local solver DICOPT (Viswanathan and Grossmann, 1990) to find a feasible solution to problem **P**. In step 3, the S subproblems are solved with either of the global solvers ANTIGONE v1.1 or BARON v20.10.16, or using the process clustering decomposition approach by Uribe-Rodriguez et al. (2020), with an optimality gap $\epsilon^{\mathrm{rel}} = 0.1$ and a maximal runtime MaxRunTime of 1,000 seconds. In step 4, the MILP relaxations are solved using CPLEX v12.8 running in parallel deterministic mode, with a relative tolerance of 10^{-4}; the initialization values for the subset of complicating variables x^i are taken from the last Lagrangean relaxation subproblem $\mathbf{LD}_{\boldsymbol{\lambda}}^{\mathbf{i}}$ solved. In step 5, the NLP solver used to determine feasible solutions to problem **P** is CONOPT 3 v.

3.17L with a feasibility tolerance for triangular equations of $2E10^{-8}$ and an optimality tolerance of $1E10^{-7}$.

All the computations were conducted on a 64-bit desktop virtual azure machine with an Intel Xeon platinum 8272 CL CPU @2.60 GHz, 16 cores, 32 logical processors, with 64 GB of RAM, running Windows 7.

5.4 Computational results

The performance of the Lagrangean decomposition algorithm to solve the short-term planning problem of an IRPC is assessed on four different scenarios and benchmarked against the process clustering decomposition approach (CL) presented in Chapter 4, and the commercial deterministic global solvers BARON and ANTIGONE.

5.4.1 Comparison of Lagrangean decomposition with other solution strategies

For each scenario, Table 5.1 summarizes the performance of the Lagrangean decomposition (LD) algorithm (Section 5.3.4), for the two- (**CM-RPB**), three- (**CM-RB-PTQ**) and four-section (**CM-REF-PTQ-FB**) decompositions that were described in section 5.2. Results for BARON, ANTIGONE and the clustering decomposition approach (Uribe-Rodriguez et al., 2020) using either 2 (CL2: crude management, **RPB**) or 6 clusters (CL6: crude management, crude distillation, vacuum and debutanizer, refining, petrochemical production, fuel blending) are also reported.

Table 5.1. Results from spatial Lagrangean decomposition algorithm with two sections (**CM-RPB**), three sections (**CM-RB-PTQ**) and four sections (**CM-REF-PTQ-FB**), compared with the commercial deterministic global solvers BARON and ANTIGONE and with the process clustering decomposition approach by Uribe-Rodriguez et al. (2020) with two clusters (CL2) and six clusters (CL6).

	Base case scenario (BCS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap* [%]**	**Runtime [h]**
CM-RPB	2,911	2,982	2.4%	0.19
CM-RB-PTQ	2,953	3,181	7.2%	10.03
CM-REF-PTQ-FB	2,711	3,379	18.8%	10.00
ANTIGONE	2,634	3,898	32.4%	10.00
BARON	2,684	4,505	40.4%	10.00
CL2	2,924	3,458	15.4%	1.35
CL6	2,964	3,205	7.5%	5.70
	Without refinery-petrochemical integration scenario (WRPS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap* [%]**	**Runtime [h]**
CM-RPB	1,943	2,029	4.2%	9.40
CM-RB-PTQ	2,006	2,022	0.8%	0.72
CM-REF-PTQ-FB	1,757	2,615	32.8%	10.00
ANTIGONE	1,219	2,926	58.3%	10.00
BARON	1,574	3,536	55.5%	10.00
CL2	1,970	2,310	14.7%	2.46
CL6	2,009	2,233	10.0%	5.84
	Logistic disruption scenario (LDS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap* [%]**	**Runtime [h]**
CM-RPB	2,637	2,668	1.2%	0.42
CM-RB-PTQ	2,661	2,814	5.4%	10.08
CM-REF-PTQ-FB	2,457	2,848	13.7%	10.00
ANTIGONE	2,156	3,451	37.5%	10.00
BARON	2,473	3,981	37.9%	10.00
CL2	2,625	3,220	18.5%	0.86
CL6	2,664	3,050	12.7%	3.68
	Demand reduction scenario (DRS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap* [%]**	**Runtime [h]**
CM-RPB	2,801	2,998	6.6%	10.12
CM-RB-PTQ	2,804	2,908	3.6%	5.84
CM-REF-PTQ-FB	2,464	2,961	16.8%	10.00
ANTIGONE	2,186	3,719	41.2%	10.00
BARON	2,478	4,214	41.2%	10.00
CL2	2,820	3,133	10.0%	2.54
CL6	2,833	3,090	8.3%	5.80

*Optimality gap is applied to ANTIGONE and BARON

The main benefits afforded by Lagrangean decomposition come from considerably tighter dual bounds (UB) compared to the other algorithms, irrespective of the scenario considered. This translates into much smaller relaxation gaps at termination: relaxation gaps between 0.8–7.2% with either two or three sections, compared to relaxation gaps between 10–18.5% with CL2 and 7.5–12.7% with CL6. In particular, the ability to guarantee a feasible solution around 1% of the global optimum for the WRPS and LDS scenarios is a remarkable result for such large-scale problems. The two- and three-section decompositions are also found to outperform the one using four sections, for reasons that will be discussed later on.

By contrast, none of the LD schemes can improve on the best feasible solutions (LB) found by the six-cluster decomposition (CL6). The best-found solutions from LD with two sections (**CM-RPB**) are comparable to those computed by the cluster decomposition with two clusters (CL2, maximal difference around 1% across all scenarios). Similarly, the best-found solutions from LD with 3 sections (**CM-RB-PTQ**) are comparable to those from CL6 (maximal difference around 1% across). This comparison also reveals that the MILP relaxation from the cluster decomposition algorithm can provide better starting points to the local NLP solver than its LD counterpart (step 4), thus suggesting that the spatial Lagrangean decomposition would benefit from a more effective search for high quality feasible solutions.

The performance of local solvers such as SBB and DICOPT for solving problem P is discussed in the Appendix G, Table G. 1. Overall, these solvers are unreliable to solve such large-scale MIQCQP problem, since they might require a good initial point to converge. Usually, is not straightforward to reach a feasible solution, and a heuristic procedure might be required to initialize the operating conditions (binary variables y) and flowrates and stream properties (continuous variables x).

5.4.2 Analysis of Lagrangean decomposition strategies

The trade-off in decomposing the large-scale MIQCQP into S subproblems ($\mathbf{LD}_{\boldsymbol{\lambda}}^{\mathbf{i}}$) is that as the subproblems (either QCQP or MIQCQP) become smaller in size, they can usually be solved to global optimality more efficiently, but this comes at the cost of more iterations in the LD algorithms since a greater number of Lagrange multipliers need to be updated simultaneously. Here, the two-section decomposition (Figure 5.1) involves a total of 57 Lagrange multipliers (8 flowrates, 24 bulk properties, and 25 compositions); the disaggregation of the petrochemical plant from the refinery and fuel blending in the three-section decomposition (Figure 5.2) adds 20 Lagrange multipliers (6 flowrates and 14 properties), bringing the total to 77; and the disaggregation of the fuel blending from the refinery in the four-section decomposition (Figure 5.3) adds another 239 Lagrange multipliers (88 flowrates and 151 properties), leading to a grand total of 316. The range of the Lagrange multipliers are shown in Table 5.2. Recall that the Lagrange multipliers can either take positive or negative values, since the complicating constraints are equality equations.

Table 5.2. Lagrange multipliers range.

Description	Range [min,max]	Subproblems applied		
		CM-RPB	CM-RB-PTQ	CM-REF-PTQ-FB
Crude blend	[-60,60] USD/bbl	x	x	x
H_2, ethylene, olefins	[-1000,1000] USD/t		x	x
CH_4	[-5000,5000] USD/MMSCFD		x	x
Raffinate, Virgin Naphtha	[-100,100] USD/bbl		x	x
Intermediated Refinery streams	[-100,100] USD/bbl			x
Streams properties	[-0.1,0.1] USD/property	x	x	x

The ability to solve the corresponding Lagrangean relaxation subproblems to global optimality within the 1,000 seconds time-limit on the current platform is summarized in Table 5.3. The QCQP subproblem for the **CM** section can be globally optimized using BARON or ANTIGONE, and so can the QCQP subproblem for the **FB** section and the MIQCQP subproblem for **PTQ**. By contrast, the large-scale MIQCQP subproblem for **REF** alone in the four-section decomposition remains intractable within the set time-limit, and those for the aggregated **RPB** and **RB** in the two- and three-section decompositions are intractable as well. In order to increase the likelihood of finding a global optimum for the challenging MIQCQPs, a cluster-decomposition approach was applied as part of the LD algorithm, using five clusters (crude distillation, vacuum and debutanizer, refining, petrochemical production, fuel blending) in the two-section case, four clusters (crude distillation, vacuum and debutanizer, refining, fuel blending) in the three-section case, and three clusters (crude distillation, vacuum and debutanizer, refinery) in the four-section case; see Uribe-Rodriguez et al., 2020 for further details about these clusters.

Table 5.3. Subproblems for the spatial Lagrangean decomposition.

S	Subproblems	Model type	Solved to global optimality? *	# Clusters in CL approach
2	**CM**	QCQP	Yes	–
	RPB	MIQCQP	No	5
3	**CM**	QCQP	Yes	–
	RB	MIQCQP	No	4
	PTQ	MIQCQP	Yes	–
4	**CM**	QCQP	Yes	–
	REF	MIQCQP	No	3
	FB	QCQP	Yes	–
	PTQ	MIQCQP	Yes	–

*By ANTIGONE and BARON

The performance of a Lagrangean decomposition with four sections (**CM-REF-PTQ-FB**) is illustrated on Figure 5.5 (left plot) for the base-case scenario (BCS). While the best-found

solution (LB) is about constant throughout the iterations, the dual bound (LD) from the Lagrangean decomposition and relaxation presents spurious variations. This is due to large variations in some of the Lagrange multiplier values, some even taking negative values (cf. right plot of Figure 5.5), despite adapting the trust-region radius prior to solving the dual problem ($\mathbf{DP^K}$), see the Lagrange multipliers in Table 5.2.

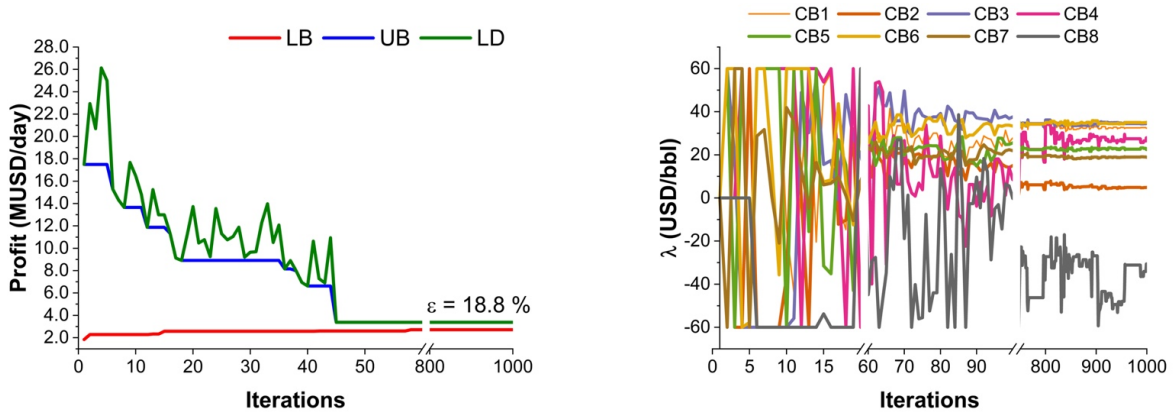


Figure 5.5. Performance of the Lagrangean decomposition with four sections (CM-REF-FB-PTQ) in the base-case scenario (BCS) up to a maximal runtime of 36,000 seconds (left) and corresponding evolution of the Lagrange multipliers for the crude blend flowrates (right).

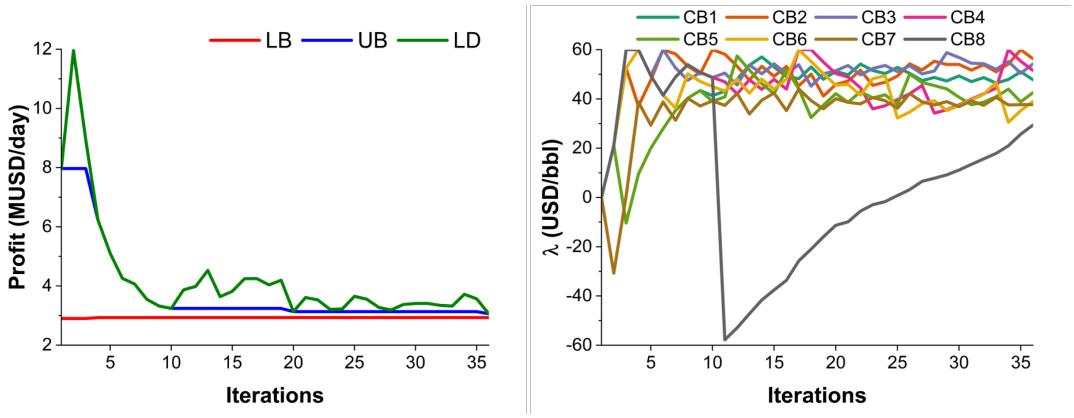


Figure 5.6. Performance of the Lagrangean decomposition with two sections (CM-RPB) in the base-case scenario (BCS) up to a maximal runtime of 36,000 seconds (left) and corresponding evolution of the Lagrange multipliers for the crude blend flowrates (right).

Overall, the presence of over 300 linking variables between all four sections requires many iterations for the dual bound to progress, which overwhelms the benefit of having smaller, more

tractable subproblems in the Lagrangean relaxation. Still, as discussed in section 4.6.5, the final dual bound (UB) is still considerably tighter than that from commercial solvers for all four scenarios (recall Table 5.1). From a practical viewpoint, these variations highlight the challenges of finding an optimal coordination among the four sections: **FB** seeks to meet the fuel demands with cheaper high-quality materials traded with **REF**, while buying as little as possible from **PTQ**; **CM** seeks to sell expensive or low-quality crude blends to **REF**; **PTQ** seeks to buy cheap natural gas, virgin naphtha, olefins and ethylene from **REF**; while **REF** acts as an adversary that seeks to maximize its revenue from selling intermediates to **FB** and **PTQ** and procuring crude blends from **CM**. Competing against **CM**, **FB** and **PTQ** makes it difficult for **REF** to raise its profit, thereby operating at the lowest level of charge (100 kbbl/day) .

The coordination of either two (**CM-RPB**) and three (**CM-RB-PTQ**) sections in the Lagrangean decomposition also results in large variations of the dual bound (LD) and the Lagrange multipliers associated with the linking variables. However, in all the scenarios (BCS & LDS with 2 sections, WPRS & DRS with 3 sections, cf. Table 5.1) the Lagrangean decomposition terminates upon reaching the 5% relaxation tolerance after a few dozen iterations. This success is attributed to the much smaller number of multipliers compared to the four-section decomposition.

For illustration, the performance of a Lagrangean decomposition with two sections is presented on Figure 5.6 (left plot) for the base-case scenario. Recall that the Lagrange multipliers may be interpreted as transfer prices between the crude management (**CM**) and the integrated refinery-petrochemical complex (**RPB**). Important findings while searching for an optimal compromise are the following:

- At iteration 1 with all the multipliers set to zero, **CM** and **RPB** are essentially uncoordinated, and their mismatch does not incur any penalty on the other section. **CM** chooses to only provide 100 kbbl/day (the minimal flow) of the crude blend CB7

(top-left plot of Figure 5.7), which is comprised of 72% of domestic crude DC9, 16% of DC16, and 12% of DC17. It uses all available DC16 and DC17, which are the cheapest domestic crudes, neglecting importing crudes as they are more expensive (cf. Table 3.1), to achieve a minimal loss of 3.46 MMUSD/day. As a result, the crude oils are heavy, sour and acid, leading to a poor-quality crude blend (20 API, 1.13 %wt. sulphur content, 2.7 TAN) that fails to comply with CDU specifications. By contrast, **RPB** chooses to process all possible crude blends in the basket, for a total refinery capacity of 203 kbbl/day (top-right plot of Figure 5.7) and achieves a maximal profit of 11.43 MMUSD/day. CB6 - CB7 comprise a large amount of high-quality domestic and imported crudes, which are compliant with the CDUs maximum limits of 1.2 %wt. and 2.0 TAN. This strategy is expected insofar as there is no premium for processing these higher-quality crude blends. Accordingly, the initial dual bound (UB) at iteration 1 is highly conservative.

- The Lagrange multiplier values are very volatile during the first few iterations, where a fast reduction in the dual bound (UB) is observed. Following this initial phase, the multipliers of the crude blends CB1-CB7 stabilize between 30–60 USD/bbl, the lowest value corresponding to the medium crude CB7 and the highest value to the light crude CB2. The Lagrange multiplier for CB8 is by far the most volatile, remaining negative between iterations 11–26, mainly due to the low fraction of this blend in the crude basket.
- At the final iteration 36, **CM** procures 202 kbbl/day of a basket of medium crude blends (26 API, 0.89 %wt. sulphur and 1.87 TAN) that already meet the quality specifications of the CDUs (cf. bottom-left plot of Figure 5.7). Meanwhile **RPB** has a significantly lower throughput of 127 kbbl/day compared to iteration 1, consisting of a medium crude blend with 29 API, 0.81 %wt sulphur and 0.67 TAN, which is of

better quality than the crude blend provided by **CM** (cf. bottom-right plot of Figure 5.7). With all the Lagrange multipliers – that is, trading prices – now being positive (cf. top section of Table 5.4), **CM** makes a profit of 0.73 MMUSD/day, while the profit of **RPB** decreases to 2.25 MMUSD/day (cf. middle & bottom sections of Table 5.4) , which is within 0.6% of the best strategy found for BCS (cf. Table 5.1).

Table 5.4. Update of Lagrange multipliers, profit and throughput for CM and RPB at iterations 1 and 36.

	Lagrange Multiplier Iteration	
Crude Blend	1	36
CB1	–	47.11
CB2	–	55.56
CB3	–	53.50
CB4	–	50.86
CB5	–	41.45
CB6	–	39.70
CB7	–	38.41
CB8	–	30.12
	Profit (MMUSD/day) Iteration	
Problem	1	36
CM	-3.456	0.732
RPB	11.425	2.249
	Throughput (kbbl/day) Iteration	
Problem	1	36
CM	100	202
RPB	203	127

Overall, these results establish that Lagrangean decomposition is effective at tightly bracketing the global solution value of large-scale IRPC planning problems. They also suggest a large potential for reducing the number of iterations through improving the Lagrange multiplier update in the dual problem.

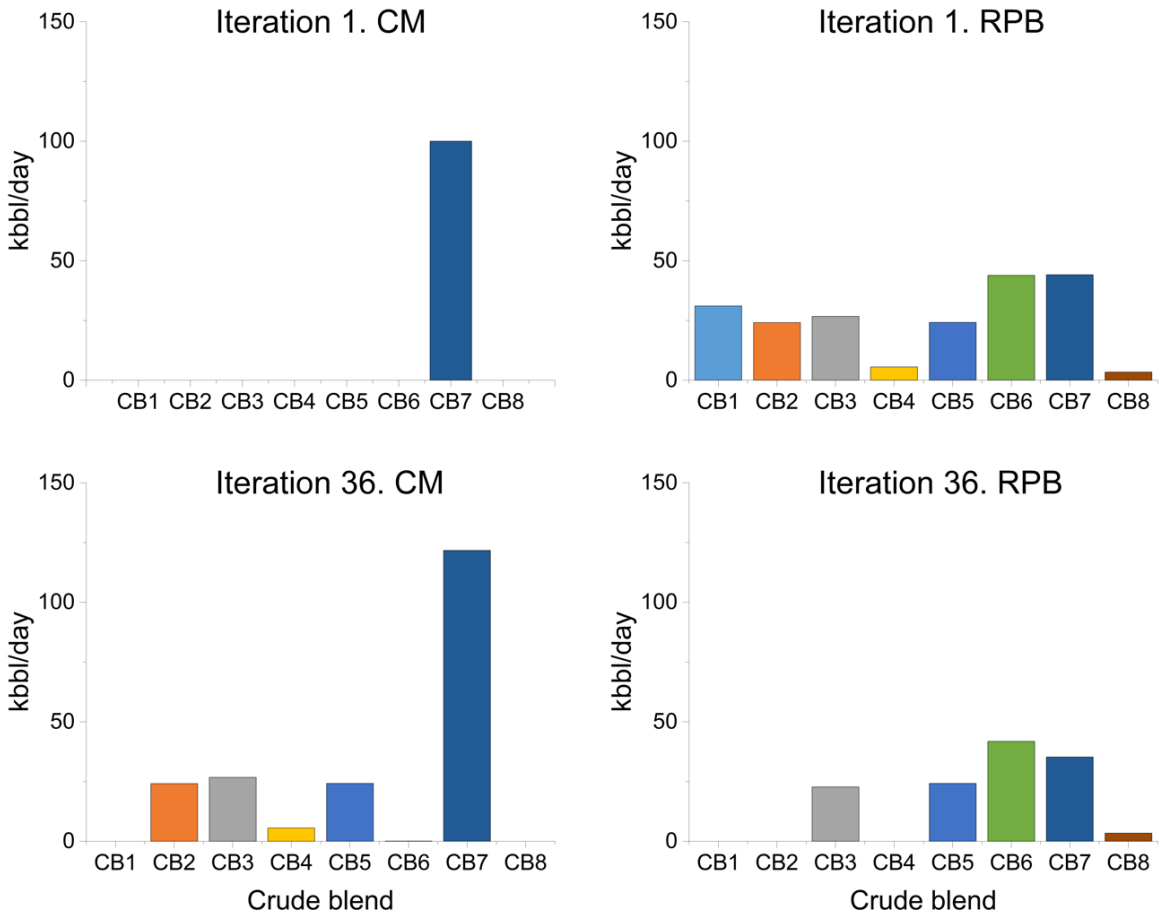


Figure 5.7. Crude blend flowrates provided by CM (left) and processes by RPB (right) at iterations 1 (top) and 36 (bottom) of the Lagrangean decomposition algorithm in the base-case scenario (BCS).

5.5 Conclusions

Through this chapter, a spatial Lagrangean decomposition approach has been investigated to globally optimize large-scale MIQCQP problems arising in short-term planning of integrated refinery-petrochemical complexes. Such problems have not yet been addressed in their full complexity in the literature, remaining intractable to generic global optimisation solvers.

To obtain more manageable QCQP and MIQCQP subproblems, different Lagrangean decomposition strategies have been formulated, which subdivide the IRPC into two, three or four sections. Such Lagrangean decompositions are akin to creating a virtual market for trading the crude blends and other intermediate refined-petrochemical streams between the different sections. The marginal prices associated with the flows and properties of these connecting

streams correspond to the Lagrange multipliers in the decomposition, thus enabling a clear interpretation of the results.

A comparison on an IRPC arising from the Colombian petroleum industry for four real-life scenarios has shown that Lagrangean decomposition could reach relaxation gaps between 0.8–7.2% with either two or three sections, even guaranteeing a near optimal solution (around 1% optimality) in two scenarios. This level of performance is unprecedented and a significant improvement over cluster-decomposition algorithms that rely on piecewise-linear relaxations. A trade-off could also be identified between the number of sections and the number of iterations required by the Lagrangean decomposition algorithm, which causes the four-section decomposition to be outperformed by its two- and three-section counterparts.

Chapter 6 CONCLUSIONS AND RECOMMENDATIONS

6.1 Key conclusions and contributions made

This thesis tackles the problem of short-term planning of an industrial-scale IRPC. This case study is more challenging and realistic than previous studies for the following reasons:

- A large variety of crude oils are considered, characterized by different volumes, qualities, and costs.
- The supply chain for crude oil and refined products is managed considering pipeline and river fleet transportation depending on the geographic region where the crude oil is produced, the import ports and final customer localization. These crudes can be blended to meet CDUs volume and quality specifications.
- Demands are defined on a large variety of fuel and petrochemical products, including five different grades of gasoline.
- The process network presents a high connectivity between units and intermediate streams. For instance, virgin naphtha can either be routed to gasoline blending, be a petrochemical feedstock, or be sold as an intermediate refined product.
- Process units can operate in exclusive or non-exclusive campaigns. Specifically, the FCC units are constrained to a single operating mode during the whole planning horizon (to be decided by the optimisation), while the CDUs can alternate between the maximization of

medium distillates, paraffins or lubes, which represent different campaigns. These are handled as discrete decisions in the optimisation model.

All these features led to a large-scale mixed-integer quadratically constrained quadratic program (MIQCQP) model with many nonconvex terms, while the entire process network is represented by input-output relationships based on bilinear expressions describing yields and stream properties, pooling equations, fuel blending indices and cost indicators.

Whilst global optimizers of large-scale MIQCQP problems may require prohibitive computational time to obtain a feasible solution, local search may lead to poor solutions. For instance, for the four short-term planning scenarios, ANTIGONE and BARON reached optimality gaps of 42.25% and 43.25% on average respectively, at a maximum runtime of 10 hours (Table 3.6). Moreover, both ANTIGONE and BARON were unable to improve their best solution found at the first iteration of the algorithm, remaining unchanged until the maximum CPU time is reached. On the other hand, heuristic decomposition techniques based on the understanding of the physical system may provide good quality solutions within a reasonable computational expense.

Two novel decomposition-based algorithms for deterministic global optimisation were developed in this thesis:

The first one 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. We applied this approach to the same four planning scenarios, highlighting that the clustering approach found higher profit values with relaxation gaps between 7% and 13%, which for the scale of this problem is a remarkable result and a significant improvement over the global solvers ANTIGONE and BARON. (See Table 4.7).

Aiming to improve the performance of the clustering decomposition, we propose the second global optimisation approach, which is a spatial Lagrangean decomposition-based algorithm which divides the problem into a collection of two, three or four subproblems, organizing differently the four physical section of the complex: crude management, refining, petrochemical operations, and fuel blending. Even though this disaggregation, the refinery subproblem is still challenging to solve using ANTIGONE and BARON. Thus, we adapted the clustering approach to solve the refinery subproblem, whilst the other subproblems can be solved to global optimality either by ANTIGONE or BARON. Thus, by applying the Lagrangean decomposition with a modified version of the clustering approach we were able to reach relaxation gaps between 0.8–7.2% with either two or three sections, even guaranteeing a near optimal solution (around 1% optimality) in two scenarios. This level of performance is unprecedented and a significant improvement over cluster-decomposition algorithms that rely on piecewise-linear relaxations.

We observed that the performance of the spatial Lagrangrean decomposition proposed in this work to address the short-term planning problem for the IRPC relies on the coordination of the Lagrange multipliers, which depending on the decomposition strategy can vary from 57 to 316, comprising flowrates and process streams quality and composition. The more Lagrange multipliers to determine in the solution of the dual problem ($\mathbf{DP^K}$), the more challenging is the coordination between the subproblems. We propose to reduce the variability on the upper bound $z_{\lambda}^{\mathrm{LD}}$ between iterations by adjusting the trust-region radius Δ_{v}^{ij} of the Lagrange multipliers $\lambda_{v}^{ij,K}$ around $\lambda_{v}^{ij,K-1}$, before solving $\mathbf{DP^K}$. Thus, the larger the deviation of the linking variables, the greater the corresponding trust-region radius. Other challenge arising from the Lagrangean decomposition approach is the solution of each subproblem (business unit) to global optimality. For instance, the crude management (CM), petrochemical operations (PTQ) and fuel blending (FB) can be solved to global optimality by ANTIGONE and BARON. But the refinery (RFB)

is still a challenging problem to address by these commercial solvers, leading to poor solutions at expenses of a prohibitive computational expenses. This finding is crucial, because if each subproblem cannot be solved to global optimality, it might compromise the performance of the Lagrangean decomposition.

We also observed that a subproblem derived from the Lagrangean decomposition can be composed by at least one aggregation of process clusters. For example, there is a one business unit-to-one cluster relationship for the CM, PTQ and FB subproblems, whilst subproblem RFB is composed by six clusters. Since ANTIGONE and BARON are not able to solve RFB, we apply the clustering approach to solve the refinery subproblem, leveraging the solution of the other subproblems to ANTIGONE or BARON. Thus, the performance of the spatial Lagrangean decomposition is enhanced using the clustering approach to address the most challenging subproblem (RFB).

A comparison on the same four real-life scenarios previously addressed by the clustering approach has shown that Lagrangean decomposition could reach relaxation gaps between 0.8–7.2% with either two or three sections, even guaranteeing a near optimal solution (around 1% gap) in two scenarios. This level of performance is unprecedented and highlights that the clustering approach can be integrated into other decomposition-based methodologies (e.g., Lagrangean decomposition). However, a trade-off could also be identified between the number of sections and the number of iterations required by the Lagrangean decomposition algorithm, which causes the four-section decomposition to be outperformed by its two- and three-section counterparts.

6.2 Further research

6.2.1 Process clustering

Future work could 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. Other research topic would be enabling parallel computing mechanisms to run several instances of MILP and QCQP models. These features could improve the computational performance of the proposed framework and, it will enhance the optimality-based bound contraction methods.

In terms of the algorithm implementation, it could be worth exploring other mathematical programming languages such as Pyomo (Hart et al., 2017) and JuMP (Dunning et al., 2017) instead of GAMS. One advantage of these open-source platforms based on python and Julia respectively, is the possibility of integration with state-of-art research which could be beneficial for the performance of the clustering approach. Since problem **P** has about 35,000 bilinear terms, approaches such as SUSPECT (Ceccon et al., 2020) could improve the performance of the clustering decomposition approach by working as a pre-processing stage to the algorithm. According to the authors, OBBT, bounds propagation, monotonicity and convexity detection can also be applied to enhance the performance of commercial solvers for global optimisation. Note that in this work we already have identified all the non-convex terms.

Other possible research direction might be the integration of the clustering approach within a branch and cut global optimisation solvers such as GALINI (Ceccon et al., 2021; Ceccon and Misener, 2022), which is building up on the basis of Pyomo's MINLP formulations. Moreover, other building blocks for the development of global optimisation solvers based on branch and bound such as SUSPECT and CORAMIN ("Coramin: A collection of tools (classes, functions, etc.) for developing MINLP algorithms," n.d.) are already implemented in Pyomo. Thus,

GALINI could be extended to solve problem **P** in combination to the clustering approach. Following this idea, EAGO (Easy Advanced Global Optimization) which is a branch and bound open-source global optimisation package developed in JuMP (Wilhelm and Stuber, 2022) could also be extended to solve problem **P**.

The Adaptive, multivariate partitioning (AMP) algorithm (Kannan et al., 2022; Nagarajan et al., 2019) implemented in JuMP has in common with the clustering approach that both are based on a two stage MILP – NLP strategy to solve the original MINLP problem, but they perform OBBT and manage the number of partitions for the piecewise McCormick relaxations in a different way. The authors argued that using an adaptive and non-uniform bi-variated partitioning scheme, the size of the MILP relaxation can be scaled-up to tackle large-scale MINLP problems. If so, this AMP feature might improve the performance of the clustering approach.

6.2.2 Advances on normalized multiparametric disaggregation technique

We formulate model **PR'** considering the normalized multiparametric disaggregation technique (NMDT) (Castillo Castillo et al., 2017, 2018; Castillo et al., 2017; Castro, 2016) instead the piecewise McCormick to relax the bilinear terms present in problem **P**. Thus, for the bilinear expressions belonging to the cluster $cl \in CL$ we formulate a NMDT relaxation, otherwise, we apply the standard McCormick envelope. Consequently, we solve **PR'** instead **PR** in the steps 3 and 5.3 of the clustering decomposition algorithm (see Section 4.3). In the following formulation, the precision level $p = \{-2, -1\}$ and $K = \{1,2, \dots 10\}$.

The piecewise McCormick envelopes (PCM) and normalized multiparametric disaggregation technique are comparable between them for relaxing a bilinear term for a small number of partitions. In particular, for $N = 2$ in PCM, NMDT should be set up with $K = 2, p = -1$, and for $N = 4$ in PCM, NMDT should be set up with $K = 4, p = -2$ (Castro, 2016). However, the

size of problem **PR** even for a small number of partitions based on PCM relaxations might lead to a large-scale MILP problem, which can be challenging to solve (see Table 4.7).

$$
\begin{aligned}
z^R := \ & \max f_0^R(x,y,w) && \textbf{(PR')}\\
& \text{s.t. } f_m^R(x,y,w) \le 0 \ \forall m \in \{1,\dots,M\}\\
& \left.\begin{aligned} & x_j = x_j^L + \lambda_j\left(x_j^U - x_j^L\right)\\ & \lambda_j = \sum_{l=p}^{-1}\sum_{k=0}^{K-1} K^l \cdot k \cdot z_{jkl} + \Delta\lambda_j\\ & 0 \le \Delta\lambda_j \le K^p \end{aligned}\right\} \ \forall j \in CL\\
& \left.\begin{aligned} & \omega_{ij} = x_i x_j^L + v_{ij}\left(x_j^U - x_j^L\right)\\ & v_{ij} = \sum_{l=p}^{-1}\sum_{k=0}^{K-1} K^l \cdot k \cdot \hat{x}_{ijkl} + \Delta v_{ij}\\ & x_i^L \cdot \Delta\lambda_j \le \Delta v_{ij} \le x_i^U \cdot \Delta\lambda_j\\ & \Delta v_{ij} \le (x_i - x_i^L)\cdot K^p + x_i^L \cdot \Delta\lambda_j\\ & \Delta v_{ij} \ge \left(x_i - x_i^U\right)\cdot K^p + x_i^U \cdot \Delta\lambda_j \end{aligned}\right\} \forall (i,j) \in BL_m, (i,j) \in CL\\
& x_i = \sum_{k=0}^{K-1} \hat{x}_{ijkl} \ \ \forall (i,j) \in BL_m, (i,j) \in CL, l \in \{p,\dots,-1\}\\
& \sum_{k=0}^{K-1} z_{jkl} = 1 \ \ \forall j \in DS, j \in CL, l \in \{p,\dots,-1\}\\
& z_{jkl} x_i^L \le \hat{x}_{ijkl} \le z_{jkl} x_i^U \ \ \forall (i,j) \in BL_m, (i,j) \in CL, k \in \{0,\dots,K-1\}, l \in \{p,\dots,-1\}\\
& \left.\begin{aligned} & w_{ij} \ge x_i x_j^L + x_i^L x_j - x_i^L x_j^L\\ & w_{ij} \ge x_i x_j^U + x_i^U x_j - x_i^U x_j^U\\ & w_{ij} \le x_i x_j^L + x_i^U x_j - x_i^U x_j^L\\ & w_{ij} \le x_i x_j^U + x_i^L x_j - x_i^L x_j^U \end{aligned}\right\} \forall (i,j) \in BL_m, j \notin CL\\
& x \in [x^L, x^U] \subseteq \mathbb{R}_+^n, \ y \in \{0,1\}^r, z \in \{0,1\}^r, w \subseteq \mathbb{R}_+^n\\
& \lambda, v, \hat{x}, \Delta\lambda, \Delta v, \omega \subseteq \mathbb{R}_+^n
\end{aligned}
$$

We apply the NMDT at the steps 3 and 5.3 of the clustering approach, for the BCS scenario, with $K^{max} = 10, p = -1$. The Figure 6.1 shows the performance for NMDT, PCM and commercial solvers ANTIGONE and BARON for the base case scenario. NMDT and PCM relaxations both were set up with $K = N = 2$, and they can be increased every time the optimality gap is improved (step 5.6.1 of the clustering decomposition algorithm). Note that NMDT started with a worse UB compared to the clustering approach and ANTIGONE. However, after 750 and until 1,500 seconds of CPU time, the UB obtained from solving problem **PR'** is the best so far. From 1,600 seconds of computational time, the UB reported by solving problem **PR** is lower than the reached out by the solution of problem **PR'**. In terms of

LB, **PR'** provided the best figure at the beginning of the algorithm, but it is quickly improved by CL applying PCM to solve **PR**. At 2600 s of CPU time, the UB is the same solving **PR** and **PR'**. Indeed, until 6500 seconds of runtime, the CL decomposition solving **PR'** provided a better lower bound, which is about 2.972 MMUSD/day compared to 2.964. thus, for the BCS scenario, the relaxation of bilinear terms using NMDT in the clustering decomposition approach proved a better LB, with similar optimality gaps and less runtime than applying PCM.

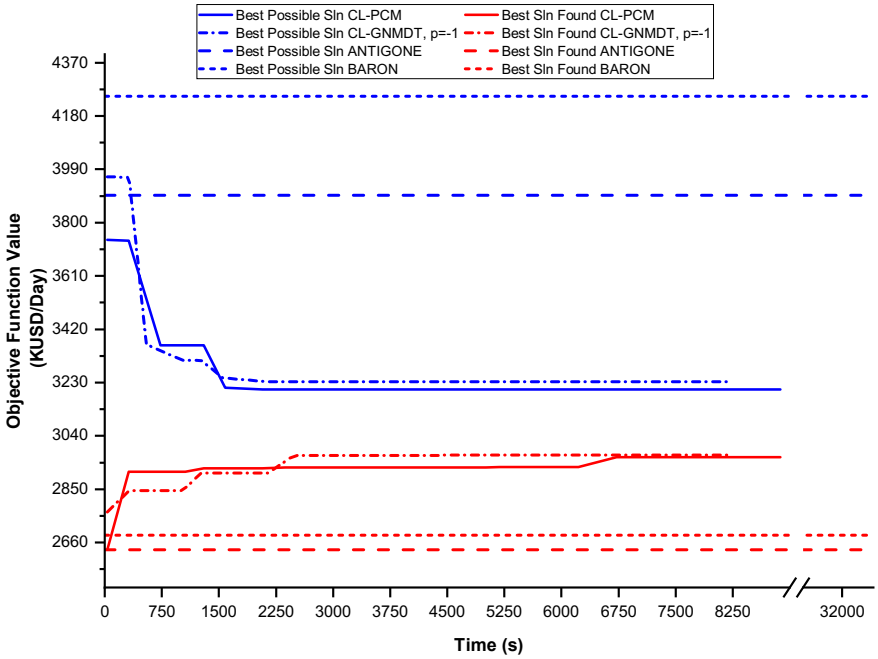


Figure 6.1. Clustering approach performance for NMDT applied to BCS.

The clustering approach using NMDT to solve problem **PR'** reported better LB for the BCS and WRPS scenarios but this relaxation led to a worse upper bound. In general, the relaxation gap was increased for all the scenarios, where WRPS reported an relaxation gap of 23.5%. moreover, for LDS and DRS scenarios, both relaxations led to the same lower bound. Overall, it might be worth to explore how the performance of the NMDT relaxation could be improved, since it reached out a better or at least the same lower bound as the PCM but with less computational expenses (Table 6.1). A detailed description of the performance for the WRPS, LDS and DRS scenarios is shown in the Appendix H.

Table 6.1. Performance of the NMDT relaxation in the clustering approach,

	LB [kUSD/day]	UB [kUSD/day]	Relaxation Gap [%]	Runtime [h]
Base case scenario (BCS)				
CL with PCM	2,964	3,205	7.5%	5.70
CL with NMDT	2,972	3,233	8.1%	3.70
Without refinery-petrochemical integration scenario (WRPS)				
CL with PCM	2,009	2,333	10.0%	5.84
CL with NMDT	2,027	2,650	23.5%	4.88
Logistic disruption scenario (LDS)				
CL with PCM	2,664	3,050	12.7%	3.68
CL with NMDT	2,662	3,217	17.3%	3.90
Demand reduction scenario (DRS)				
CL with PCM	2,833	3,090	8.3%	5.80
CL with NMDT	2,834	3,174	10.7%	6.59

6.3 Lagrangean decomposition assessment

Future work on the Lagrangean decomposition algorithm should focus on improving the dual problem formulation, to handle a large number of Lagrange multipliers within a reasonable number of iterations (i.e., scaling the Lagrange multipliers, see Table 5.2). The algorithm would also benefit from a more effective search for feasibility or locally optimal solutions during the iterations. On the application side, a future research direction entails the integration of refinery-petrochemical short-term planning with crude oil scheduling operations, another challenging problem for which effective global optimisation algorithms need to be developed.

Since the clustering decomposition approach and the Lagrangean-based decomposition algorithm both rely on the solution of MILP relaxations, it would be interesting to use GUROBI ("GUROBI Optimization LLC," 2023) to solve such large-scale models instead CPLEX. Moreover, recent versions of GUROBI can solve MIQCP problems to global optimality. This feature would be explored to solve problem **P** and compare our results now considering

GUROBI. This part of the investigation was not covered in this thesis mainly for confidentiality issues of the model and due to internal protocols regarding the use of trial software at Ecopetrol.

Other interesting research topics to address would be the potential improvement on the IRPC modelling, for example adding uncertainties through stochastic programming or applying robust optimisation or considering environmental constraints using multi-objective optimisation. These new modelling features applied to the IRPC optimisation would require the development of novel solution methods for such challenging problems.

REFERENCES

Adams, F.G., Griffing, J.., 1972. An Economic-Linear Programming Model of the U. S. Petroleum Refining Industry. J. Am. Stat. Assoc. 67, 542–551. https://doi.org/10.2307/1907384

Adhya, N., Tawarmalani, M., Sahinidis, N. V., 1999. A Lagrangian approach to the pooling problem. Ind. Eng. Chem. Res. 38, 1956–1972. https://doi.org/10.1021/ie980666q

Al-Qahtani, K., Elkamel, A., 2010. Robust planning of multisite refinery networks: Optimization under uncertainty. Comput. Chem. Eng. 34, 985–995. https://doi.org/10.1016/j.compchemeng.2010.02.032

Al-Qahtani, K., Elkamel, A., 2009. Multisite refinery and petrochemical network design: Optimal integration and coordination. Ind. Eng. Chem. Res. 48, 814–826. https://doi.org/10.1021/ie801001q

Al-Qahtani, K., Elkamel, A., 2008. Multisite facility network integration design and coordination: An application to the refining industry. Comput. Chem. Eng. 32, 2189–2202. https://doi.org/10.1016/j.compchemeng.2007.10.017

Al-Qahtani, K.Y., 2009. Petroleum Refining and Petrochemical Industry Integration and Coordination under Uncertainty. Ph.D. Thesis. Waterloo University, Canada.

Alattas, A.M., Grossmann, I.E., Palou-Rivera, I., 2011. Integration of nonlinear crude distillation unit models in refinery planning optimization. Ind. Eng. Chem. Res. 50, 6860–6870. https://doi.org/10.1021/ie200151e

Alhajri, I., Elkamel, a., Albahri, T., Douglas, P.L., 2008. A nonlinear programming model for refinery planning and optimisation with rigorous process models and product quality specifications. Int. J. Oil, Gas Coal Technol. 1, 283. https://doi.org/10.1504/IJOGCT.2008.019846

Andrade, T., Oliveira, F., Hamacher, S., Eberhard, A., 2018. Enhancing the normalized

multiparametric disaggregation technique for mixed-integer quadratic programming. J. Glob. Optim. 73, 701–722. https://doi.org/10.1007/s10898-018-0728-9

Andrade, T., Ribas, G., Oliveira, F., 2016. A Strategy Based on Convex Relaxation for Solving the Oil Refinery Operations Planning Problem. Ind. Eng. Chem. Res. 55, 144–155. https://doi.org/10.1021/acs.iecr.5b01132

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

Audet, C., Brimberg, J., Hansen, P., Le Digabel, S., Mladenović, N., 2004. Pooling problem: Alternate formulations and solution methods. Manage. Sci. 50, 761–776. https://doi.org/10.1287/mnsc.1030.0207

Baird, C.T., 1987. Petroleum refining process correlations. Inc, HPI Consultants, Houston.

Baker, T.E., Lasdon, L.S., 1985. Successive Linear Programming at Exxon. Manage. Sci. 31, 264–274. https://doi.org/10.1287/mnsc.31.3.264

Barahona, F., Anbil, R., 2000. The volume algorithm: Producing primal solutions with a subgradient method. Math. Program. Ser. B 87, 385–399. https://doi.org/10.1007/s101070050002

Ben-Tal, A., Eiger, G., Gershovitz, V., 1994. Global minimization by reducing the duality gap. Math. Program. 63, 193–212. https://doi.org/10.1007/BF01582066

Bergamini, M.L., Aguirre, P., Grossmann, I., 2005. Logic-based outer approximation for globally optimal synthesis of process networks. Comput. Chem. Eng. 29, 1914–1933. https://doi.org/10.1016/j.compchemeng.2005.04.003

Bonner & Moore, 1979. RPMS (Refinery and Petrochemical System): A System Description. Bonner & Moore Management Science, Houston.

Bussieck, M.R., Ferris, M.C., Meeraus, A., 2009. Grid-enabled optimization with GAMS. INFORMS J. Comput. 21, 349–362. https://doi.org/10.1287/ijoc.1090.0340

Castillo Castillo, P., Castro, P.M., Mahalec, V., 2017. Global optimization algorithm for large-scale refinery planning models with bilinear terms. Ind. Eng. Chem. Res. 56, 530–548. https://doi.org/10.1021/acs.iecr.6b01350

Castillo Castillo, P.A., Castro, P.M., Mahalec, V., 2018. Global optimization of MIQCPs with dynamic piecewise relaxations. J. Glob. Optim. 71, 691–716. https://doi.org/10.1007/s10898-018-0612-7

Castillo, P.A.C., Castro, P.M., Mahalec, V., 2017. Global Optimization of Nonlinear Blend-Scheduling Problems. Engineering 3, 188–201. https://doi.org/10.1016/J.ENG.2017.02.005

Castro, P.M., 2016. Normalized multiparametric disaggregation: an efficient relaxation for mixed-integer bilinear problems. J. Glob. Optim. 64, 765–784. https://doi.org/10.1007/s10898-015-0342-z

Castro, P.M., 2015. Tightening piecewise McCormick relaxations for bilinear problems. Comput. Chem. Eng. 72, 300–311. https://doi.org/10.1016/j.compchemeng.2014.03.025

Castro, P.M., Grossmann, I.E., 2014. Optimality-based bound contraction with multiparametric disaggregation for the global optimization of mixed-integer bilinear problems. J. Glob. Optim. 59, 277–306. https://doi.org/10.1007/s10898-014-0162-6

Castro, P.M., Liao, Q., Liang, Y., 2021. Comparison of mixed-integer relaxations with linear and logarithmic partitioning schemes for quadratically constrained problems, Optimization and Engineering. Springer US. https://doi.org/10.1007/s11081-021-09603-5

Ceccon, F., Baltean-Lugojan, R., Bynum, M., Li, C., Misener, R., 2021. GALINI: An extensible mixed-integer quadratically-constrained optimization solver. Optim. Online.

Ceccon, F., Misener, R., 2022. Solving the pooling problem at scale with extensible solver GALINI. Comput. Chem. Eng. 159, 107660. https://doi.org/10.1016/j.compchemeng.2022.107660

Ceccon, F., Siirola, J.D., Misener, R., 2020. SUSPECT: MINLP special structure detector for Pyomo. Optim. Lett. 14, 801–814. https://doi.org/10.1007/s11590-019-01396-y

Charnes, A., Cooper, W.W., Mellon, B., 1952. Blending Aviation Gasolines - A study in Programming Interdependent Activities in an Integrated Oil Company. Econometrica 20, 135–159.

Cheney, E.W., Goldstein, A.A., 1959. Newton's method for convex programming and Tchebycheff approximation. Numer. Math. 1, 253–268. https://doi.org/10.1007/BF01386389

Coramin: A collection of tools (classes, functions, etc.) for developing MINLP algorithms [WWW Document], n.d. URL https://github.com/Coramin/Coramin

De Oliveira Magalhães, M.V., 2009. Integrating refining to petrochemical. Comput. Aided Chem. Eng. 27, 107–112. https://doi.org/10.1016/S1570-7946(09)70238-X

Demirhan, C.D., Boukouvala, F., Kim, K., Song, H., Tso, W.W., Floudas, C.A., Pistikopoulos, E.N., 2020. An integrated data-driven modeling & global optimization approach for multi-period nonlinear production planning problems. Comput. Chem. Eng. 141, 107007. https://doi.org/10.1016/j.compchemeng.2020.107007

Dunning, I., Huchette, J., Lubin, M., 2017. JuMP: A modeling language for mathematical optimization. SIAM Rev. 59, 295–320. https://doi.org/10.1137/15M1020575

Faria, D.C., Bagajewicz, M.J., 2012. A new approach for global optimization of a class of MINLP problems with applications to water management and pooling problems. AIChE J. 58, 2320–2335. https://doi.org/10.1002/aic.12754

Faria, D.C., Bagajewicz, M.J., 2011a. Novel bound contraction procedure for global optimization of bilinear MINLP problems with applications to water management problems. Comput. Chem. Eng. 35, 446–455. https://doi.org/10.1016/j.compchemeng.2010.04.010

Faria, D.C., Bagajewicz, M.J., 2011b. Global Optimization of Water Management Problems Using Linear Relaxation and Bound Contraction Methods. Ind. Eng. Chem. Res. 50, 3738–3753. https://doi.org/10.1021/ie101206c

Fisher, M.L., 1981. The lagrangian relaxation method for solving integer programming problems. Manage. Sci. 27, 1–18. https://doi.org/https://doi.org/10.1287/mnsc.27.1.1

Foulds, L.R., Haugland, D., Jörnsten, K., 1992. A bilinear approach to the pooling problem. Optimization 24, 165–180. https://doi.org/10.1080/02331939208843786

GAMS Software GmbH, 2012. CPLEX 12, in: GAMS - The Solver Manuals. GAMS development corporation, pp. 169–215.

Garvin, W.W., Crandall, H.W., John, J.B., Spellman, R.A., 1957. Applications of Linear Programming in the Oil Industry. Manage. Sci. 3, 407–430.

Geddes, R.L., 1958. A general index of fractional distillation power for hydrocarbon mixtures. AIChE J. 4, 389–392. https://doi.org/10.1002/aic.690040403

Gilbert, R.J.H., Leather, J., Ellis, J.F.G., 1966. The application of the Geddes fractionation index to crude distillation units. AIChE J. 12, 432–437. https://doi.org/10.1002/aic.690120309

Gounaris, C.E., Misener, R., Floudas, C.A., 2009. Computational comparison of piecewise-linear relaxations for pooling problems. Ind. Eng. Chem. Res. 48, 5742–5766.

https://doi.org/10.1021/ie8016048

Grossmann, I.E., 2021. Advanced Optimization for Process Systems Engineering. Cambridge University Press. https://doi.org/10.1017/9781108917834

Grossmann, I.E., Karuppiah, R., 2008. A Lagrangean based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures. J. Glob. Optim. 41, 163–186. https://doi.org/10.1007/s10898-007-9203-8

Guerra, O.J., Le Roux, G.A.C., 2011a. Improvements in petroleum refinery planning: 1. Formulation of process models. Ind. Eng. Chem. Res. 50, 13403–13418. https://doi.org/10.1021/ie200303m

Guerra, O.J., Le Roux, G.A.C., 2011b. Improvements in Petroleum Refinery Planning: 2. Case Studies. Ind. Eng. Chem. Res. 50, 13419–13426. https://doi.org/10.1021/ie200304v

Guerra, O.J., Uribe-Rodriguez, A., Montagut, S.M., Duarte, L.A., Angarita, J.D., 2010. A Solution Strategy for Large-Scale Nonlinear Petroleum Refinery Planning Models, in: AIChE Annual Meeting. Salt Lake city.

Guignard, M., 2003. Lagrangean relaxation. TOP 11, 151–200. https://doi.org/10.1007/BF02579036

Guignard, M., Siwhan Kim, 1987. Lagrangean decomposition: A model yielding stronger Lagrangean bounds. Math. Program. 39, 215–228.

GUROBI Optimization LLC [WWW Document], 2023. URL https://www.gurobi.com/

Guyonnet, P., Grant, F.H., Bagajewicz, M.J., 2009. Integrated Model for Refinery Planning, Oil Procuring, and Product Distribution. Ind. Eng. Chem. Res. 48, 463–482. https://doi.org/10.1021/ie701712z

Hart, W.E., Laird, C.D., Watson, J.-P., Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., Siirola, J.D., 2017. Springer Optimization and Its Applications 67 Pyomo-Optimization Modeling in Python Second Edition.

Haverly, C.A., 1980. Recursion Model Behavior: More Studies. ACM SIGMAP Bull 28, 39–41.

Haverly, C.A., 1979. Behavior of Recursion Model - More Studies. ACM SIGMAP Bull 26, 22–28.

Haverly, C.A., 1978. Studies of the Behavior of Recursion for the Pooling Problem. ACM

SIGMAP Bull 25, 22–28.

Haverly, S., 2015. Generalized Refining Transportation Marketing Planning System - GRTMPS [WWW Document]. URL https://www.haverly.com/grtmps

Held, M., Karp, R.M., 1971. The traveling-salesman problem and minimum spanning trees: Part II. Math. Program. 1, 6–25. https://doi.org/10.1007/BF01584070

Held, M., Wolfe, P., Crowder, H.P., 1974. Validation of subgradient optimization. Math. Program. 6, 62–88. https://doi.org/10.1007/BF01580223

International Energy Agency, 2019. Oil 2019 Analysis and forecasts to 2024.

Jackson, J.R., Grossmann, I.E., 2003. Temporal decomposition scheme for nonlinear multisite production planning and distribution models. Ind. Eng. Chem. Res. 42, 3045–3055. https://doi.org/10.1021/ie030070p

Jia, Z., Ierapetritou, M., 2004. Efficient short-term scheduling of refinery operations based on a continuous time formulation. Comput. Chem. Eng. 28, 1001–1019. https://doi.org/10.1016/j.compchemeng.2003.09.007

Kannan, R., Nagarajan, H., Deka, D., 2022. Learning to Accelerate Partitioning Algorithms for the Global Optimization of Nonconvex Quadratically-Constrained Quadratic Programs 1–26.

Karuppiah, R., Grossmann, I.E., 2006. Global optimization for the synthesis of integrated water systems in chemical processes. Comput. Chem. Eng. 30, 650–673. https://doi.org/10.1016/j.compchemeng.2005.11.005

Kelley, J.E., 1960. The cutting-plane method for solving convex programs. J. Soc. Ind. Appl. Math. 8, 703–712.

Kelly, J.D., Menezes, B.C., Grossmann, I.E., 2014. Distillation blending and cutpoint temperature optimization using monotonic interpolation. Ind. Eng. Chem. Res. 53, 15146–15156. https://doi.org/10.1021/ie502306x

Ketabchi, E., Mechleri, E., Arellano-Garcia, H., 2019. Increasing operational efficiency through the integration of an oil refinery and an ethylene production plant. Chem. Eng. Res. Des. 152, 85–94. https://doi.org/10.1016/j.cherd.2019.09.028

Khor, C.S., Varvarezos, D., 2017. Petroleum refinery optimization. Optim. Eng. 18, 943–989. https://doi.org/10.1007/s11081-016-9338-x

Kolodziej, S., Castro, P.M., Grossmann, I.E., 2013. Global optimization of bilinear programs with a multiparametric disaggregation technique. J. Glob. Optim. 57, 1039–1063. https://doi.org/10.1007/s10898-012-0022-1

Kutz, T., Davis, M., Creek, R., Kenaston, N., Stenstrom, C., Connor, M., 2014. Optimizing Chevron's Refineries. Interfaces (Providence). 44, 39–54. https://doi.org/10.1287/inte.2013.0727

Lara, C.L., Trespalacios, F., Grossmann, I.E., 2018. Global optimization algorithm for capacitated multi-facility continuous location-allocation problems. J. Glob. Optim. https://doi.org/10.1007/s10898-018-0621-6

Leiras, A., Elkamel, A., Hamacher, S., 2010. Strategic planning of integrated multirefinery networks: A robust optimization approach based on the degree of conservatism. Ind. Eng. Chem. Res. 49, 9970–9977. https://doi.org/10.1021/ie100919z

Li, C., He, X., Chen, Bingzhen, Chen, Bo, Gong, Z., Quan, L., 2006. Integrative optimization of refining and petrochemical plants. Comput. Aided Chem. Eng. 21, 2039–2044. https://doi.org/10.1016/S1570-7946(06)80348-2

Li, J., Xiao, X., Boukouvala, F., Floudas, C.A., Zhao, B., Du, G., Su, X., Liu, H., 2016. Data-driven mathematical modeling and global optimization framework for entire petrochemical planning operations. AIChE J. 62, 3020–3040. https://doi.org/10.1002/aic.15220

Li, W., Hui, C.W., Li, A., 2005. Integrating CDU, FCC and product blending models into refinery planning. Comput. Chem. Eng. 29, 2010–2028. https://doi.org/10.1016/j.compchemeng.2005.05.010

López, D.C., Hoyos, L.J., Mahecha, C.A., Arellano-Garcia, H., Wozny, G., 2013. Optimization model of crude oil distillation units for optimal crude oil blending and operating conditions. Ind. Eng. Chem. Res. 52, 12993–13005. https://doi.org/10.1021/ie4000344

López, D.C., Hoyos, L.J., Uribe, A., Chaparro, S., Arellano-Garcia, H., Wozny, G., 2012. Improvement of Crude Oil Refinery Gross Margin using a NLP Model of a Crude Distillation Unit System. Comput. Aided Chem. Eng. 30, 987–991. https://doi.org/10.1016/B978-0-444-59520-1.50056-7

Manne, A.S., 1958. A Linear Programming Model of the U . S . Petroleum Refining Industry. Econometrica 26, 67–106.

Marsten, R.E., Hogan, W.W., Blankenship, J.W., 1975. Boxstep Method for Large-Scale Optimization. Oper. Res. 23, 389–405. https://doi.org/10.1287/opre.23.3.389

McCormick, G.P., 1976. Computability of global solutions to factorable nonconvex programs: Part I - Convex underestimating problems. Math. Program. 10, 147–175. https://doi.org/10.1007/BF01580665

Méndez, C.A., Grossmann, I.E., Harjunkoski, I., Kaboré, P., 2006. A simultaneous optimization approach for off-line blending and scheduling of oil-refinery operations. Comput. Chem. Eng. 30, 614–634. https://doi.org/10.1016/j.compchemeng.2005.11.004

Menezes, B.C., Kelly, J.D., Grossmann, I.E., 2013. Improved swing-cut modeling for planning and scheduling of oil-refinery distillation units. Ind. Eng. Chem. Res. 52, 18324–18333. https://doi.org/10.1021/ie4025775

Misener, R., Floudas, C.A., 2014a. ANTIGONE: Algorithms for coNTinuous / Integer Global Optimization of Nonlinear Equations. J. Glob. Optim. 59, 503–526. https://doi.org/10.1007/s10898-014-0166-2

Misener, R., Floudas, C.A., 2014b. A Framework for Globally Optimizing Mixed-Integer Signomial Programs. J. Optim. Theory Appl. 161, 905–932. https://doi.org/10.1007/s10957-013-0396-3

Misener, R., Floudas, C.A., 2013. GloMIQO: Global mixed-integer quadratic optimizer. J. Glob. Optim. 57, 3–50. https://doi.org/10.1007/s10898-012-9874-7

Misener, R., Thompson, J.P., Floudas, C.A., 2011. Apogee: Global optimization of standard, generalized, and extended pooling problems via linear and logarithmic partitioning schemes. Comput. Chem. Eng. 35, 876–892. https://doi.org/10.1016/j.compchemeng.2011.01.026

Moro, L.F.L., Zanin, A.C., Pinto, J.M., 1998. A Planning Model for Refinery Diesel Production. Comput. Chem. Eng. 22, S1039–S1042. https://doi.org/10.1016/S0098-1354(98)00209-9

Mouret, S., Grossmann, I.E., Pestiaux, P., 2011. A new Lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling. Comput. Chem. Eng. 35, 2750–2766. https://doi.org/10.1016/j.compchemeng.2011.03.026

Mouret, S., Grossmann, I.E., Pestiaux, P., 2009. A novel priority-slot based continuous-time formulation for crude-oil scheduling problems. Ind. Eng. Chem. Res. 48, 8515–8528. https://doi.org/10.1021/ie8019592

Nagarajan, H., Lu, M., Wang, S., Bent, R., Sundar, K., 2019. An adaptive, multivariate partitioning algorithm for global optimization of nonconvex programs. J. Glob. Optim. https://doi.org/10.1007/s10898-018-00734-1

Nasr, M.R.J., Sahebdelfar, S., Ravanchi, M., Beshelli, M., 2011. Integration of Petrochemical and Refinery Plants as an Approach to Compete in Hydrocarbon Market [WWW Document]. URL https://www.researchgate.net/publication/268430340_Integration_of_Petrochemical_and_Refinery_Plants_as_an_Approach_to_Compete_in_Hydrocarbon_Market

Neiro, S.M.S., Pinto, J.M., 2006. Lagrangean decomposition applied to multiperiod planning of petroleum refineries under uncertainty. Lat. Am. Appl. Res. 36, 213–220.

Neiro, S.M.S., Pinto, J.M., 2004. A general modeling framework for the operational planning of petroleum supply chains. Comput. Chem. Eng. 28, 871–896. https://doi.org/10.1016/j.compchemeng.2003.09.018

Oddsdottir, T.A., Grunow, M., Akkerman, R., 2013. Procurement planning in oil refining industries considering blending operations. Comput. Chem. Eng. 58, 1–13. https://doi.org/10.1016/j.compchemeng.2013.05.006

Oliveira, F., Gupta, V., Hamacher, S., Grossmann, I.E., 2013. A Lagrangean decomposition approach for oil supply chain investment planning under uncertainty with risk considerations. Comput. Chem. Eng. 50, 184–195. https://doi.org/10.1016/j.compchemeng.2012.10.012

Puranik, Y., Sahinidis, N. V., 2017. Domain reduction techniques for global NLP and MINLP optimization. Constraints 22, 338–376. https://doi.org/10.1007/s10601-016-9267-5

Sahinidis, N. V., 2004. BARON: A general purpose global optimization software package. J. Glob. Optim. 8, 201–205. https://doi.org/10.1007/bf00138693

Siamizade, M.R., 2019. Global Optimization of Refinery-wide Production Planning with Highly Nonlinear Unit Models. Ind. Eng. Chem. Res. 58, 10437–10454. https://doi.org/10.1021/acs.iecr.9b00887

Teles, J.P., Castro, P.M., Matos, H.A., 2013. Multi-parametric disaggregation technique for global optimization of polynomial programming problems. J. Glob. Optim. 55, 227–251. https://doi.org/10.1007/s10898-011-9809-8

Terrazas-Moreno, S., Trotter, P.A., Grossmann, I.E., 2011. Temporal and spatial Lagrangean

decompositions in multi-site, multi-period production planning problems with sequence-dependent changeovers. Comput. Chem. Eng. 35, 2913–2928. https://doi.org/10.1016/j.compchemeng.2011.01.004

Uribe-Rodriguez, A., Castro, P.M., Guillén-Gosálbez, G., Chachuat, B., 2020. Global optimization of large-scale MIQCQPs via cluster decomposition: Application to short-term planning of an integrated refinery-petrochemical complex. Comput. Chem. Eng. 140, 106883. https://doi.org/10.1016/j.compchemeng.2020.106883

Uribe-Rodríguez, A., Castro, P.M., Guillén-Gosálbez, G., Chachuat, B., 2023. Assessment of Lagrangean decomposition for short-term planning of integrated refinery-petrochemical operations. Comput. Chem. Eng. 174. https://doi.org/10.1016/j.compchemeng.2023.108229

Viswanathan, J., Grossmann, I.E., 1990. A combined penalty function and outer-approximation method for MINLP optimization. Comput. Chem. Eng. 14, 769–782. https://doi.org/10.1016/0098-1354(90)87085-4

Wenkai, L., Chi-Wai, H., Karimi, I.A., Srinivasan, R., 2007. A novel CDU model for refinery planning. ASIA - PACIFIC Chem. Eng. 2, 282–293. https://doi.org/10.1002/apj

WEO/IEA, 2016. World Energy Outlook 2016.

Wicaksono, D.S., Karimi, I.A., 2008. Piecewise MILP Under- and Overestimators for Global Optimization of Bilinear Programs. AIChE J. 54, 991–1008. https://doi.org/10.1002/aic

Wilhelm, M.E., Stuber, M.D., 2022. EAGO.jl: easy advanced global optimization in Julia. Optim. Methods Softw. 37, 425–450. https://doi.org/10.1080/10556788.2020.1786566

Wilson, Z.T., Sahinidis, N. V, 2017. The ALAMO approach to machine learning. Comput. Chem. Eng. 106, 785–795. https://doi.org/10.1016/j.compchemeng.2017.02.010

Yang, H., Bernal, D.E., Franzoi, R.E., Engineer, F.G., Kwon, K., Lee, S., Grossmann, I.E., 2020. Integration of crude-oil scheduling and refinery planning by Lagrangean Decomposition. Comput. Chem. Eng. 138, 106812. https://doi.org/10.1016/j.compchemeng.2020.106812

Zhang, B.J., Liu, K., Luo, X.L., Chen, Q.L., Li, W.K., 2015. A multi-period mathematical model for simultaneous optimization of materials and energy on the refining site scale. Appl. Energy 143, 238–250. https://doi.org/10.1016/j.apenergy.2015.01.044

Zhang, J., Wen, Y., Xu, Q., 2012. Simultaneous optimization of crude oil blending and purchase

planning with delivery uncertainty consideration. Ind. Eng. Chem. Res. 51, 8453–8464. https://doi.org/10.1021/ie102499p

Zhang, J., Zhu, X.X., Towler, G.P., 2001. A level-by-level debottlenecking approach in refinery operation. Ind. Eng. Chem. Res. 40, 1528–1540. https://doi.org/10.1021/ie990854w

Zhang, L., Yuan, Z., Chen, B., 2022. Adjustable Robust Optimization for the Multi-period Planning Operations of an Integrated Refinery-Petrochemical Site under Uncertainty. Comput. Chem. Eng. 160, 107703. https://doi.org/10.1016/j.compchemeng.2022.107703

Zhang, L., Yuan, Z., Chen, B., 2021. Refinery-wide planning operations under uncertainty via robust optimization approach coupled with global optimization. Comput. Chem. Eng. 146, 107205. https://doi.org/10.1016/j.compchemeng.2020.107205

Zhao, H., Ierapetritou, M.G., Shah, N.K., Rong, G., 2017. Integrated model of refining and petrochemical plant for enterprise-wide optimization. Comput. Chem. Eng. 97, 194–207. https://doi.org/10.1016/j.compchemeng.2016.11.020

Appendix A. SOLUTION OF THE BENCHMARKING POOLING PROBLEM USING GUROBI

We solve the benchmarking pooling problem described in the Section 4.4 using the branch and cut approach to solve MIQCQP problems recently developed by GUROBI.

The pooling structure of the problem is given by input streams (5), pools (3), output streams (5), properties (2) and bilinear terms (30). Moreover, this problem has 10 local optimal solutions: 1000, 1600, 1900, 2000, 2100, 2300, 2500, 2600, 2700, 2900, and 3500 as the global optimal.

GUROBI starts with a feasible solution of 0 and a relaxation at the root node of 6100. It then explores this node adding 12 RLT cutting planes, reaching the global optimal solution reported in the literature (optimality gap of 0%), with a CPU runtime below 1 second (see part of the log file in the Figure A. 1).

Table A. 1. Computational performance of the clustering approach, ANTIGONE, BARON and GUROBI applied to the benchmark pooling problem.

	Clustering approach	**ANTIGONE**	**BARON**	**GUROBI**
Global optimum	3,500	3,500	3,500	3,500
Optimality gap	5.20E-16*	9.99E-10	2.30E-01	0.00
Runtime [s]	16.85	9.67	3,600.00	0.13
*Relaxation gap is computed for the clustering approach				

The results for this benchmarking pooling problem shown that the four approaches were able to reach the global optimal solution. Moreover, ANTIGONE, GUROBI and the clustering approach were able to close the gap. In terms of computational expenses (CPU s), it is

remarkable that GUROBI can solve the problem using one order of magnitud less than ANTIGONE and the clustering approach (Table A. 1).

```
In [1]: runfile('/Users/wero/poolingV2.py', wdir='/Users/wero')
Set parameter NonConvex to value 2
Gurobi Optimizer version 10.0.2 build v10.0.2rc0 (mac64[x86])

CPU model: Intel(R) Core(TM) i5-7360U CPU @ 2.30GHz
Thread count: 2 physical cores, 4 logical processors, using up to 4 threads

Optimize a model with 26 rows, 81 columns and 109 nonzeros
Model fingerprint: 0xbb327dfc
Model has 16 quadratic constraints
Coefficient statistics:
  Matrix range     [1e+00, 2e+01]
  QMatrix range    [1e+00, 1e+00]
  Objective range  [1e+00, 1e+00]
  Bounds range     [1e+00, 1e+03]
  RHS range        [0e+00, 0e+00]
Presolve removed 3 rows and 12 columns

Continuous model is non-convex -- solving as a MIP

Found heuristic solution: objective -0.0000000
Presolve removed 12 rows and 21 columns
Presolve time: 0.01s
Presolved: 223 rows, 106 columns, 676 nonzeros
Presolved model has 49 bilinear constraint(s)
Variable types: 106 continuous, 0 integer (0 binary)

Root relaxation: objective 6.100000e+03, 91 iterations, 0.00 seconds (0.00 work units)

    Nodes    |    Current Node    |     Objective Bounds      |     Work
 Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time

     0     0 6100.00000    0   11   -0.00000 6100.00000      -     -    0s
     0     0 3500.00000    0   13   -0.00000 3500.00000      -     -    0s
     0     0 3500.00000    0   18   -0.00000 3500.00000      -     -    0s
     0     0 3500.00000    0   11   -0.00000 3500.00000      -     -    0s
     0     0 3500.00000    0   11   -0.00000 3500.00000      -     -    0s
H    0     0                     3500.0000000 3500.00000  0.00%     -    0s

Cutting planes:
  RLT: 12

Explored 1 nodes (216 simplex iterations) in 0.13 seconds (0.01 work units)
Thread count was 4 (of 4 available processors)

Solution count 2: 3500 -0
```

Figure A. 1. GUROBI log file for the benchmark pooling problem.

Appendix B. RELAXATION OF BILINEAR AND TRILINEAR TERMS

The terms $PF_{u,p}QF_u$, $PS_{u',s,p}Q_{u',s,u}$, $Q_{u',s,u}PS_{u',s,p}PS_{u's,SPG}$, $PS_{u,s,p}QS_{u,s}$, $PF_{u,vf}QF_u$, $QF_uPF_{u,p}PF_{u,SPG}$, $PS_{u,s,p}PF_{u',vf}$ and $PF_{u',vf}QS_{u,s}$ in problem **P** (section 3.2.1) involve the product of two or three continous variables. One way to obtain a relaxation for **P**, is to substitute each of these terms by a new continuous variable: $PFQF_{u,p}$, $PSQ_{u,s,u',p}$, $PSQS_{u,s,p}$, $PFQF_{u,vf}$, $PSPF_{u,s,u',p,vf}$ and $PFQS_{u,s,u',vf}$ leading to Eqs. (B-1) - (B-10). Note that Eq. (B-1) is applied for properties that mix on volume and weight bases, thus replacing Eqs. (3-3) - (3-7).

$$PFQF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} PSQ_{u,s,u',p} \quad \forall u \in U, p \in PI_u \tag{B-1}$$

$$QS_{u,s} = \sum_{vf \in P^{VF}} yield_{u,vf,} PFQF_{u,vf} \quad \forall u \in U^{CDU}, s \in SO_u \tag{B-2}$$

$$PSQS_{u,s,p} = \sum_{vf \in P^{VF}} prop_{vf,u,s,p} PFQF_{u,vf} \quad \forall u \in U^{CDU}, s \in SO_u, p \in PO_{u,s} \tag{B-3}$$

$$\sum_{u' \in U^{CDU}: u \in UVDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',RC} PFQS_{u,s,u',vf} = \sum_{vf \in P^{VF}} yield_{vf,u,s} PFQF_{u,vf} \quad \forall u \in U^{VDU}, s \in \{LVGO, HVGO, VR\}} \tag{B-4}$$

$$\sum_{u' \in U^{CDU}: u \in UBDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',LN} PFQS_{u,s,u',vf} = \sum_{vf \in P^{VF}} yield_{vf,u,s} PFQF_{u,vf} \quad \forall u \in U^{DBU}, s \in \{C5, DB\}} \tag{B-5}$$

$$\sum_{u' \in U^{CDU}: u \in UVDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',RC} PSPF_{u,s,u',p,vf} = \sum_{vf \in P^{VF}} prop_{vf,u,s} PF_{u,vf} \quad \forall u \in U^{VDU}, s \in \{LVGO, HVGO, VR\}, p \in PO_{u,s}} \tag{B-6}$$

$$\sum_{u' \in U^{CDU}: u \in UBDU_{u'}} \sum_{vf \in P^{VF}} yield_{vf,u',LN} PSPF_{u,s,u',p,vf} = \sum_{vf \in P^{VF}} prop_{vf,u,s} PF_{u,vf} \quad \forall u \in U^{DBU}, s \in \{C5, DB\}, p \in PO_{u,s}} \tag{B-7}$$

$$QS_{u,s} = a_{u,s}QF_u + \sum_{p \in PI_u} b_{u,s,p} PFQF_{u,p} \quad \forall u \in U^{CONV}, s \in SO_u \tag{B-8}$$

$$QSPS_{u,s,p} = c_{u,s,p}QF_u + \sum_{p' \in PI_u} d_{u,s,p',p} PFQF_{u,p'} \quad \forall u \in U^{CONV}, s \in SO_u, p \in PO_{u,s} \tag{B-9}$$

$$OpEx = \sum_{u \in U^{CR} \cup U^{CONV}} (\omega_u QF_u + \sum_{p \in PI_u} \psi_{u,p} PFQF_{u,p} + \sum_{u'} \sum_s \phi_{u',s,u} Q_{u',s,u}) \quad \text{(B-10)}$$

In order to improve the quality of the relaxation, additional constraints can be added to the problem that relate these new variables to the original ones and their lower and upper bounds. This will be done either through standard or piecewise McCormick envelopes. In case of the standard McCormick relaxation, a total of 4 linear inequality constraints are added for each bilinear term (McCormick, 1976). On the other hand, the piecewise McCormick relaxation, uses 9 mixed-integer linear constraints. It is tighter due to the use of binary variables for partitioning of the domain of one of the variables in every bilinear term (Bergamini et al., 2005; Castro, 2015).

Let w_{ij} represent the new variable replacing bilinear term $x_i x_j$, let x_j represent the partitioned variable in PMR, and let set BL include the pair of indexes (i, j) involved. For instance, in $PF_{u,p}QF_u$, QF_u is the partitioned variable. In general, to reduce the number of added binary variables, we select the flowrate ($QF_u, QS_{u,s}$ or $QS_{u',s,u}$) as the variable to partition. This is because x_i is defined by multisets derived from set U (such as PI_u and $PO_{u,s}$) making $|i| \geq |j|$. The exception occurs for term $PF_{u',vf}PS_{u,s,p}$, where the stream properties variable $PS_{u,s,p}$ is the one selected to be partitioned. Assuming the same number N of partitions for all variables, the overall number of binary variables introduced is $N \cdot (|U| + |SO_u| + |US_u| + |PO_{u,s}|)$. Even though the number of these binary variables is kept at a minimum for a given N, the MILP relaxation might still lead to an intractable problem. The alternative is to consider only a subset of the bilinear terms in the relaxation, for which a decomposition based on process structure may be foreseen.

Appendix C. CASE STUDY DATA

Raw material supply

Table C. 1. Domestic Crude.

Region	Crude	Max (kbbl/day)	Price (USD/bbl)
1	DC1	2.10	39.50
	DC2	31.10	49.70
	DC3	3.40	49.68
	DC4	2.20	35.43
2	DC5	5.60	45.27
	DC6	24.20	41.26
	DC7	12.60	40.28
	DC8	5.80	38.50
3	DC9	44.80	34.57
4	DC10	3.30	42.49
	DC11	3.70	40.13
5	DC12	32.00	40.86
6	DC13	24.10	52.00
	DC14	26.70	52.11
	DC15	13.50	42.78
7	DC16	10.30	30.77
8	DC17	11.80	35.08
	Total	**257.20**	

Table C. 2. Imported Crude.

Crude	Max (kbbl/day)	Price (USD/bbl)
IC1	15.00	52.23
IC2	15.00	52.30
IC3	15.00	49.28
IC4	15.00	49.17
IC4	15.00	49.15
IC6	15.00	52.37
IC7	15.00	50.64
Total	**105.00**	

Table C. 3. Crude Oil Quality Specifications.

Crude	Sulphur (%wt)	TAN*	SPG
DC1	1.22	1.13	0.9103
DC2	0.51	0.10	0.8810
DC3	0.81	0.17	0.8635
DC4	1.93	0.53	0.9312
DC5	0.64	1.49	0.8868
DC6	0.93	2.14	0.9128
DC7	1.01	2.30	0.9225
DC8	0.96	3.13	0.9424
DC9	1.13	3.34	0.9327
DC10	1.22	1.68	0.8990
DC11	1.24	2.64	0.9176
DC12	1.85	0.12	0.9401
DC13	0.09	0.31	0.8075
DC14	0.05	0.07	0.8011
DC15	0.98	0.47	0.9121
DC16	1.14	0.14	0.9491
DC17	1.14	2.38	0.9370
IC1	0.16	0.63	0.8294
IC2	0.92	0.06	0.8283
IC3	0.25	0.59	0.8792
IC4	0.69	1.27	0.8810
IC4	0.61	0.47	0.8814
IC6	0.48	0.04	0.8272
IC7	0.16	0.61	0.8555

*TAN = Total Acid Number (mg KOH/g crude oil)

Table C. 4. Refined Products.

Market	Product	Max (Kbbl/day)	Price (USD/bbl)
Domestic	DRP1	0.70	22.61
Imports	IRP1	47.00	49.95
	IRP2	60.00	43.08

Fuel demand

Table C. 5. Fuel Demand.

Gasoline			Medium Distillate			Fuel Oil		
Grade	Max (kbbl/day)	Price (USD/bbl)	Grade	Max (kbbl/day)	Price (USD/bbl)	Grade	Max (kbbl/day)	Price (USD/bbl)
GL1	0.33	87.62	MD1	19.14	64.91	FO1	1.87	31.65
GL2	60.00	52.34	MD2	4.39	64.91	FO2	3.42	18.26
GL3	69.17	59.15	MD3	67.60	59.60	FO3	33.28	31.00
GL4	3.08	69.24	MD4	5.20	60.71	FO4	41.60	30.42
GL5	1.04	53.14	MD5	17.89	59.60			
GL6	24.52	54.42	MD6	16.90	31.00			
GL7	24.52	58.74						

Fuel specifications

Table C. 6. Gasoline.

Gasoline Grade	SPG	Sulphur (ppm)	RON		RVP
	Max	Max	Min	Max	Max
GL1	0.0400	5			
GL2	0.7351	120			
GL3	0.7800	100	85.50	90.00	8.60
GL4	0.7800	100	90.00	100.00	
GL5,GL6,GL7		50	48		50

Table C. 7. Medium Distillates.

Distillate Grade	SPG		Sulphur (ppm)	CBI	
	Min	Max	Max	Min	Max
MD1	0.7800	0.8400	300	30	100
MD2		0.8500	300		
MD3,MD4,MD6		0.9000	250		
MD5	0.8000	0.8750	40		

Table C. 8. Fuel Oil.

Fuel Oil Grade	SPG		Sulphur (ppm)		V50	
	Min	Max	Min	Max	Min	Max
FO1			1000	1700		
FO3	1.0000	1.3000	1000	1300		
FO4	0.9000	1.3000			34.05	34.35

Logistic

Table C. 9. Pipeline and river fleet routes for transportation of refined products.

Transport Mode	Commodity									
	LPG1	BUT1	GL6	MD4	MD6	FO3	FO4	IRP2	IRP3	DRP1
RF1	**1**	**1**	**1**							
RF2								**1**	**1**	**1**
RF3				**1**	**1**		**1**			
PL1*								**1**	**1**	
PL9**						**1**	**1**			

***** The pipeline PL1 delivers the imported refined products to the IRPC. The pipeline system PL1 - PL2 (not shown in Table C. 9) operates in bi-directional flow. Through PL2 (flowing in reverse direction compared to PL1), the IRPC delivers virgin naphtha and light cracked naphtha to other IRPC, which is part of the whole nationwide petroleum supply in Colombia. Since in this work we consider only one of those industrial complexes, PL2 is not modelled.

RF4 is a river fleet route delivering fuel-oil between the two IRPCs existing in Colombia. Since in this work we consider one IRPC, RF4 is set to zero.

Table C. 10. Pipeline routes for transportation of domestic and imported crude oil.

Transport Mode	Commodity																			
	DC1	DC2	DC3	DC4	DC5	DC6	DC7	DC8	DC9	DC10	DC11	DC12	DC13	DC14	DC15	DC16	DC17	IC1	...	IC7
PL3									1			1	1	1	1	1	1			
PL4								1		1	1									
PL5							1													
PL6					1	1														
PL7	1	1	1	1																
PL8**																		1	1	1

**Pipeline system PL8 - PL9 operates in bi-directional flow. Through PL8 the IRPC receives imported crude oil (IC1,…,IC7). On the other hand, through PL9 the IRPC delivers fuel-oil (FO3, FO4) from the IRPC to the domestic and exportation market.

Appendix D. CLUSTERING RESULTS FOR WRPS SCENARIO

Clustering results for the scenario without refinery - petrochemical integration (WRPS)

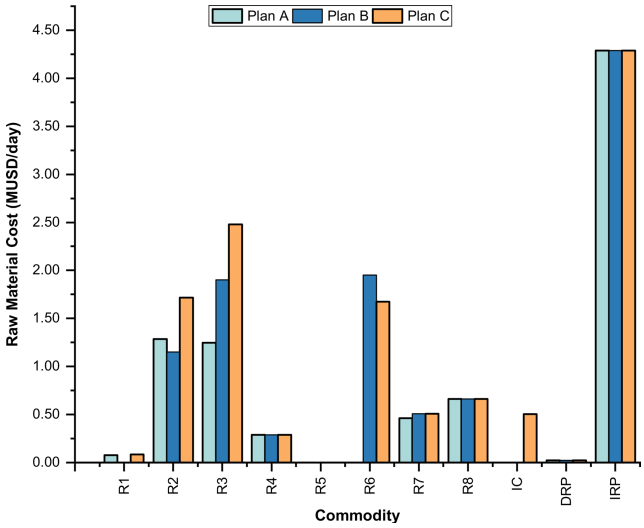


Figure D. 1. Raw material supply for the WRPS scenario.

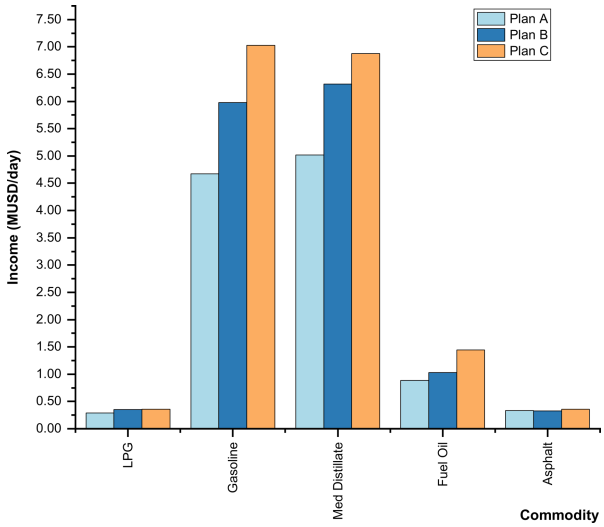


Figure D. 2. Commodities production for the WRPS scenario.

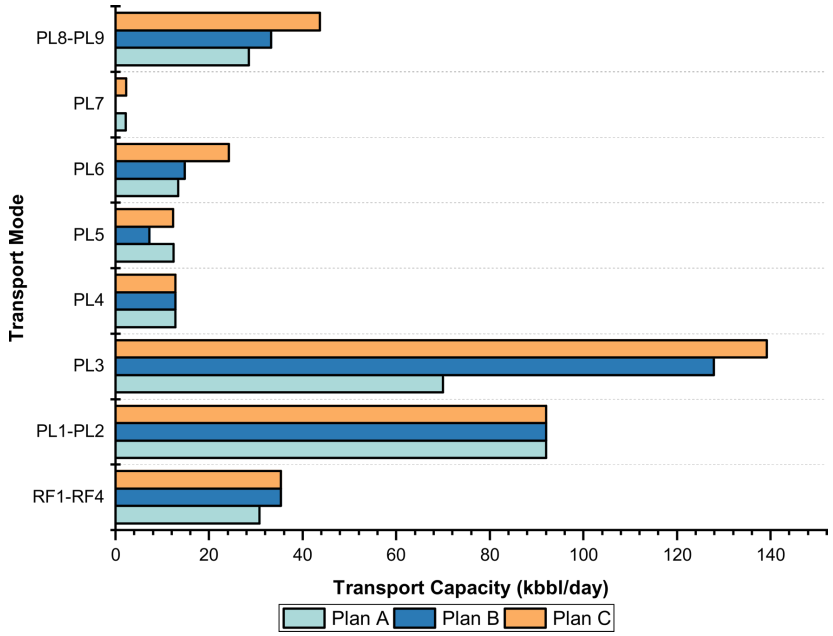


Figure D. 3. Logistic for the WRPS scenario.

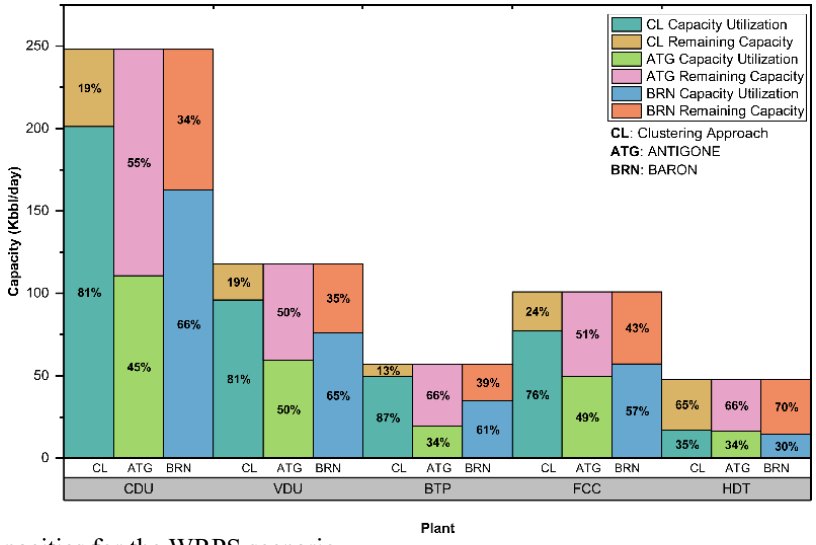


Figure D. 4. Plant capacities for the WRPS scenario.

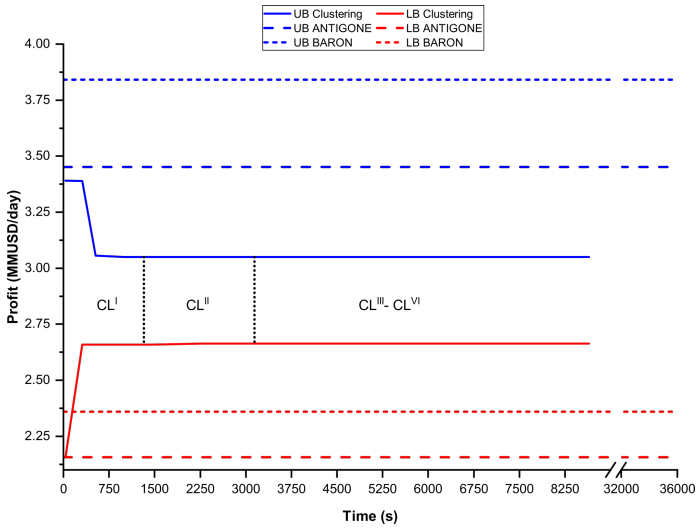


Figure D. 5. Solvers performance for the WRPS scenario.

Appendix E. CLUSTERING RESULTS FOR LDS SCENARIO

Clustering results for the logistic disruption scenario (LDS)

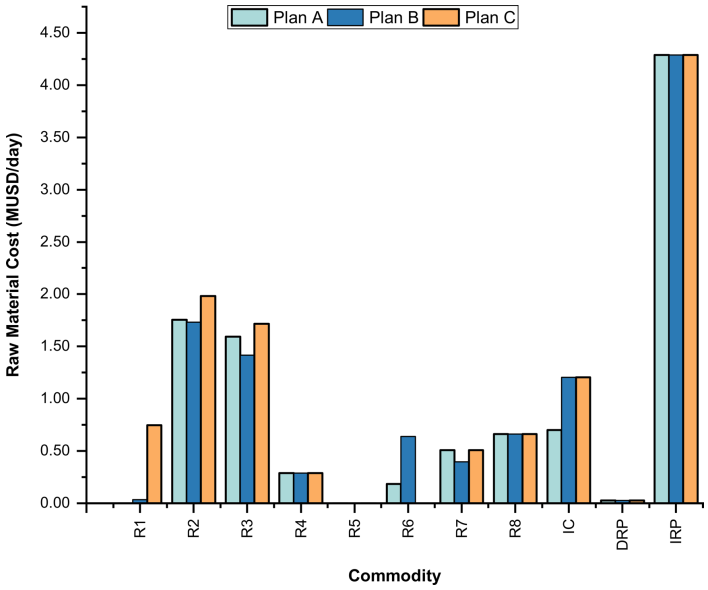


Figure E. 1. Raw material supply for the LDS scenario.

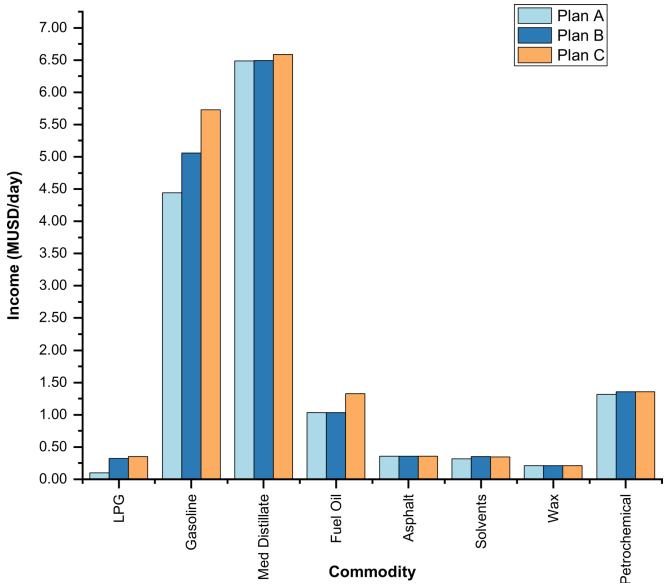


Figure E. 2. Commodities production for the LDS scenario.

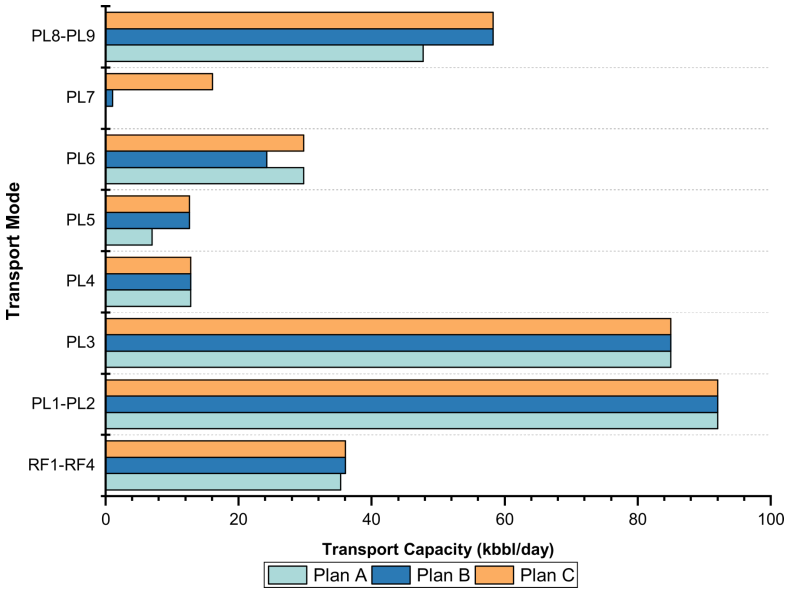


Figure E. 3. Logistic for the LDS scenario.

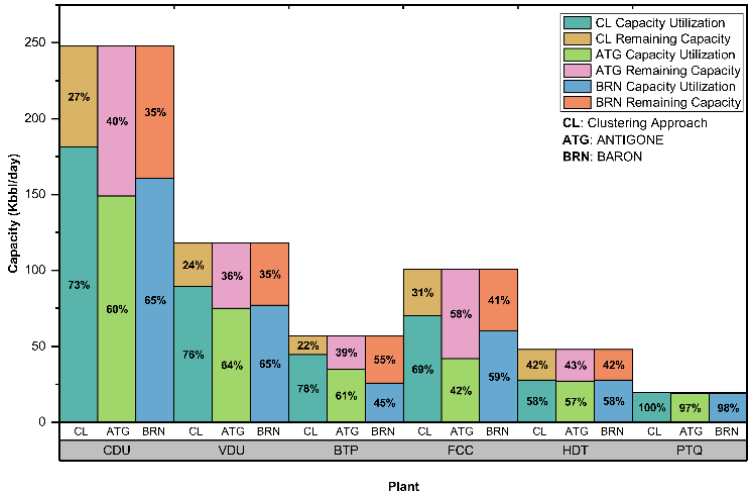


Figure E. 4. Plant capacities for the LDS scenario.

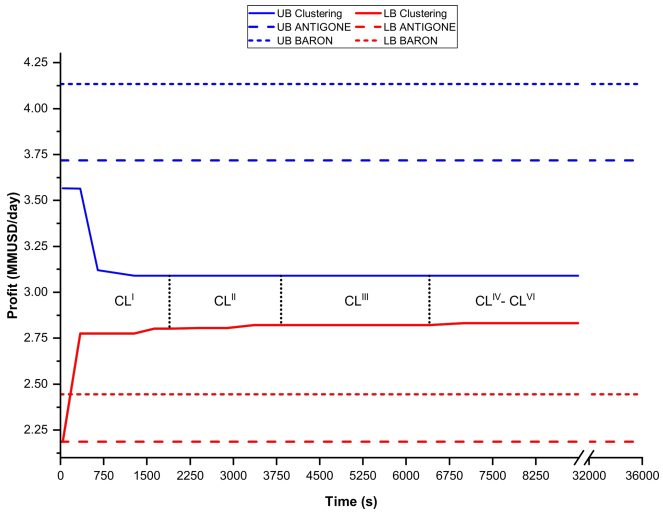


Figure E. 5. Solvers performance for the LDS scenario.

Appendix F. CLUSTERING RESULTS FOR DRS SCENARIO

Clustering results for the gasoline demand reduction scenario (DRS)

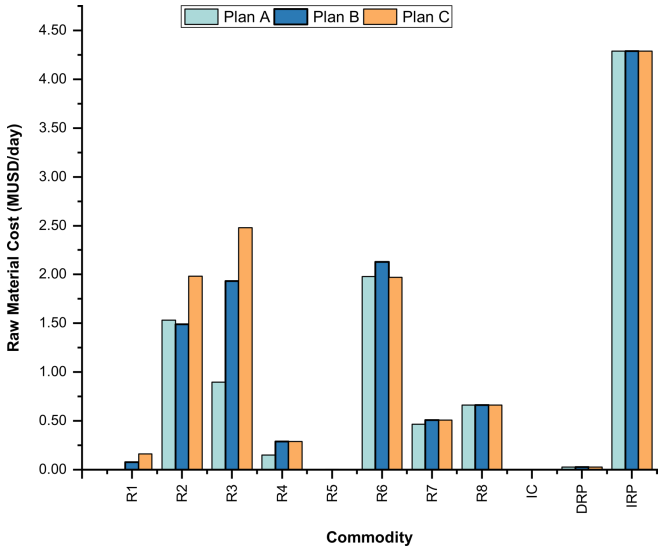


Figure F. 1. Raw material supply for the DRS scenario.

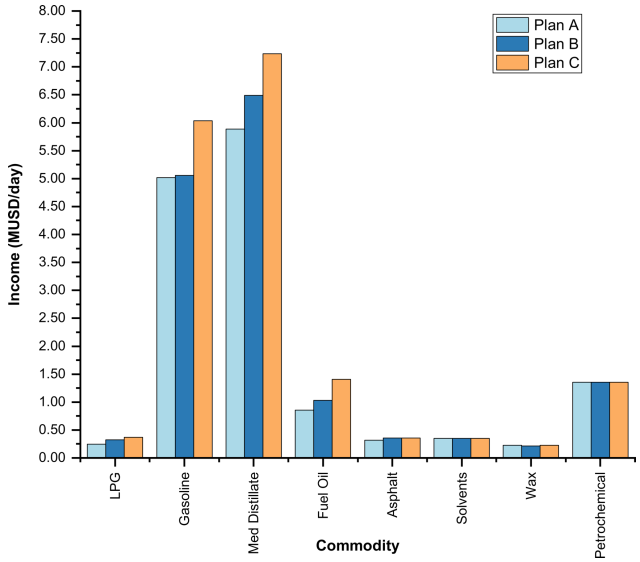


Figure F. 2. Commodities production for the DRS scenario.

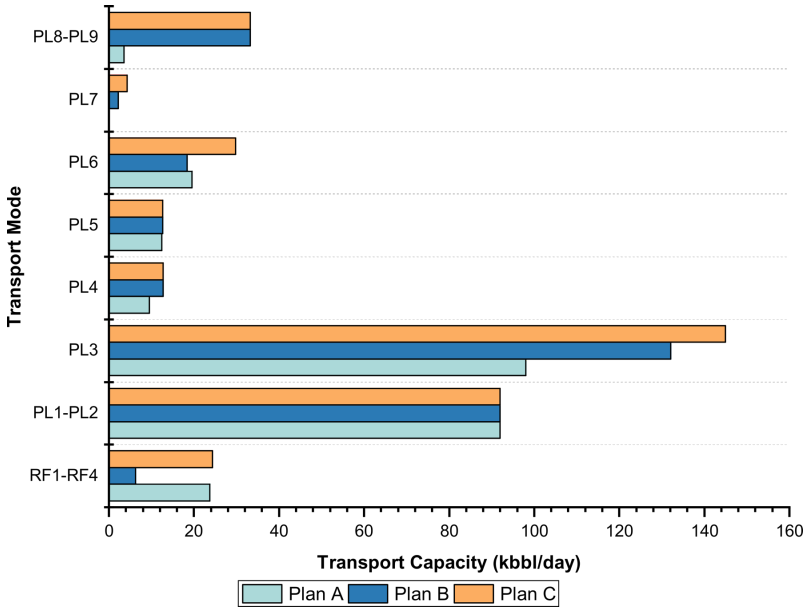


Figure F. 3. Logistic for the DRS scenario.

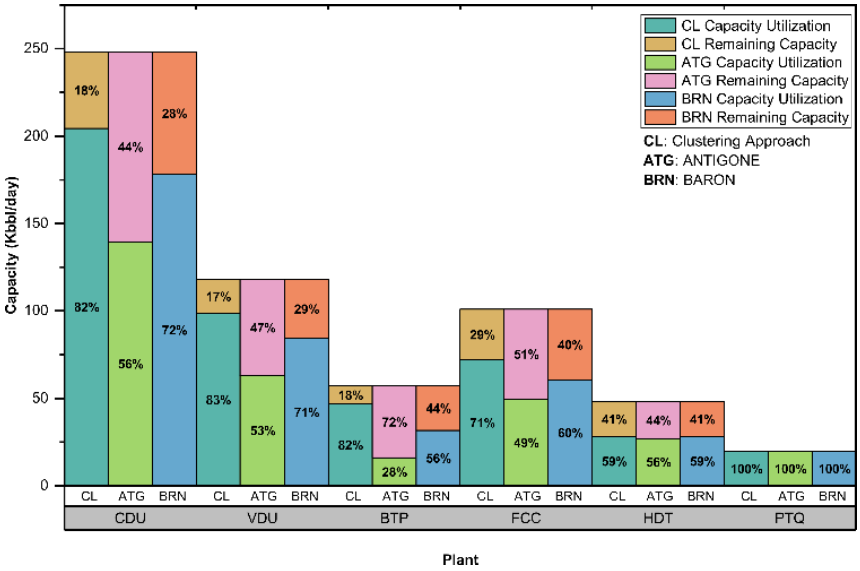


Figure F. 4. Plant capacities for the DRS scenario.

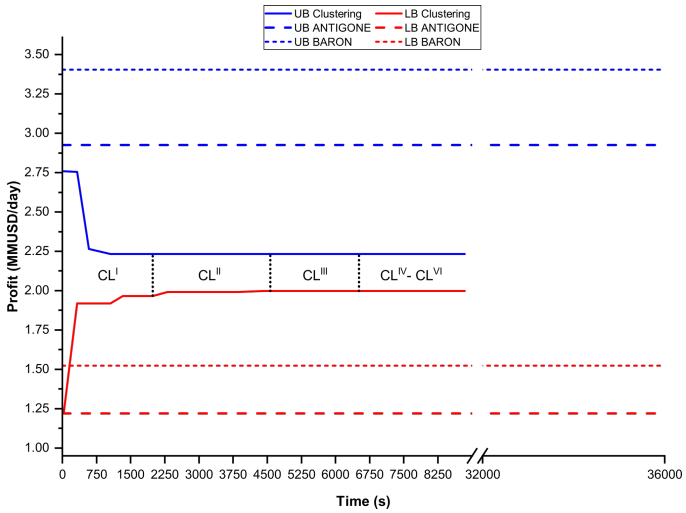


Figure F. 5. Solvers performance for the DRS scenario.

Appendix G. MODELS FOR EACH SUBPROBLEM IN THE LAGRANGEAN DECOMPOSITION

An integrated refinery petrochemical complex (**IRPC**) can be composed by four independent business units:

- Crude management section (**CM**)
- Refinery (**REF**)
- Petrochemicals (**PTQ**) and
- Fuel blending (**FB**)

Other possible schemes to disaggregate the **IRPC** into small business units are considered in this thesis. For instance, two sections including **CM** and **REF – PTQ - FB** or three sections considering **CM**, **REF – FB** and **PTQ**. Since the four subproblems are the most general decomposition of the IRPC considered in this thesis, we describe in detail its model formulation.

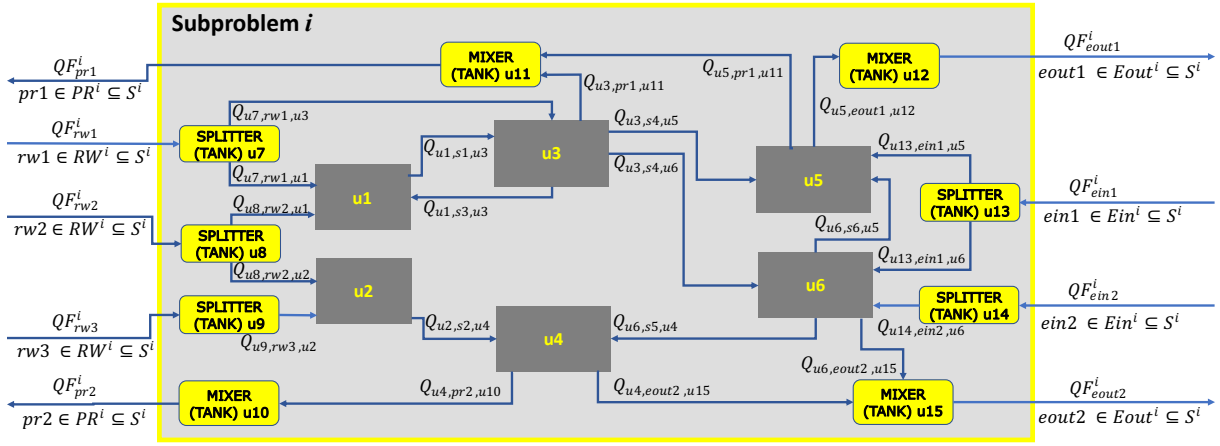
When the **IRPC** is decomposed in four subproblems, the interaction between each subproblem and their linking variables are summarized in Figure 5.3 and Figure 5.4.

It is highlighted that each group of linking variables among two subproblems is duplicated, for example $E^{CB} = \{CB1, \dots, CB8\}$ (Figure 5.4) , is identified in the subproblem i as: $Eout_j^i$ (in this example $Eout_{REF}^{CM} = E^{CB}$), and in the subproblem j as: Ein_i^j ($Ein_{CM}^{REF} = E^{CB}$). This group of linking variables comes out from one tank ($UEout_j^i$, in this example $UEout_{REF}^{CM}$) in the problem

where it is produced, and it is received in another tank ($UEin_i^j$, in this example $UEin_{CM}^{REF}$) in the destination problem. The flow and properties for this group of linking streams are calculated using the optimisation model of each subproblem. Since the results might be different, "checkpoints" (new restrictions) become necessary to address the integration between the subproblems, in order to guaranty the linking stream values are equalized.

Each problem i consists of a superstructure of units connected among them and a series of internal, linking, raw-material and final-product streams. Since the topology for the IRPC model addressed in this thesis comprises 155 units connected by intermediate streams, a generic superstructure is shown in Figure G. 1, to illustrate the problem modelling process.

The objective function of each problem is to maximize $z_1^{i,LD*}$, defined as the profit plus a term that is related to the integration of the problem i with the problem j which exchange materials between them by selling or buying commodities (See model identified as $\mathbf{LD_\lambda^i}$ in the Chapter 5).



$Internal\ streams\ of\ subproblem\ i = Sint^i = \{s1, s2, s3, s4, s5, s6\}$
$Raw\ material\ streams\ of\ subproblem\ i = RW^i = \{rw1, rw2, rw3\}$
$Product\ streams\ of\ subproblem\ i, sold\ to\ external\ market = PR^i = \{pr1, pr2\}$
$Linking\ streams\ of\ subproblem\ i\ to\ other\ subproblems = Eout^i = \{eout1, eout2\}$
$Linking\ streams\ of\ other\ subproblems\ to\ subproblem\ i = Ein^i = \{ein1, ein2\}$
$Streams\ of\ subproblem\ i = S^i = Sint^i \cup RW^i \cup PR^i \cup Eout^i \cup Ein^i$

Figure G. 1. General subproblem model developed in Chapter 5.

For problem i, the objective function (profit) to be maximized can be cast as follows:

$$f_0^i(x^i, y^i) = profit^i = totalrevenue^i - totalcost^i \quad \text{(G-1)}$$

Where x^i represents the continuous - and y^i the binary-variables.

Total revenue of problem i is calculated using Eq. (G-2):

$$totalrevenue^i = revenue\_pr^i + revenue^i \quad \text{(G-2)}$$

Where $revenue\_pr^i$ is the sum of revenues obtained by selling all the final products from the problem i to external market:

$$revenue\_pr^i = \sum_{pr \in PR^i} \alpha_{pr}. QF_{pr}^i \quad \text{(G-3)}$$

Where α_{pr} is the selling price for product pr ($/bbl or $/ton).

The second term in the Eq. (G-2), $revenue^i$, is the sum of revenues obtained from the exchange of commodities sold to the business unit j with $j \neq i$. These streams are traded at $\lambda_e^{i,j}$ and $\lambda_{e,p}^{i,j}$ price, which correspond to the Lagrange multipliers.

$$revenue^i = \sum_{j \neq i} \sum_{e \in Eout_j^i} \lambda_e^{i,j}. QF_e^i + \sum_{j \neq i} \sum_{e \in Eout_j^i} \sum_{p \in PI_e^i} \lambda_{e,p}^{i,j}. PF_{e,p}^i \quad \text{(G-4)}$$

It is assumed that each linking stream from problem i is sent to only one problem j.

In the case of costs of problem i, given in the Eq. (G-5), three types of costs are considered: the cost of raw materials at the plant gate ($cost\_rw^i$), the logistic cost ($cost\_lgstc^i$) and the cost of linking streams from another subproblems ($cost^i$).

$$totalcost^i = cost\_rw^i + cost\_lgstc^i + Cost\_OpEx^i + cost^i \quad \text{(G-5)}$$

The cost of raw materials ($cost\_rw^i$) are defined by Eq. (G-6) as the sum of the cost of the raw materials at their source: oil fields for domestic crudes, or national ports for imported crudes and imported refined products.

$$cost\_rw^i = \sum_{rw \in RW^i} \beta_{rw} \cdot QF^i_{rw} \quad \text{(G-6)}$$

Where β_{rw} is the purchasing cost for raw material rw ($/bbl or $/ton).

The logistic cost ($cost\_lgstc^i$) is defined as the cost of transportation of raw materials from their source to the plant gate, plus the cost of transportation of final products to the buyer's storage facilities or exportation ports, generally by pipeline system (mt):

$$cost\_lgstc^i = \sum_{mt \in MT^i} \gamma_{mt} \cdot QT^i_{mt} \quad \text{(G-7)}$$

Where γ_{mt} is the transportation cost of the transport mode mt ($/bbl), MT^i is the set of all incoming and outgoing piping systems associated with problem i and QT^i_{mt} is the total flowrate, (total volume of raw materials or final products of subproblem i), through that transportation mode $mt \in MT^i$.

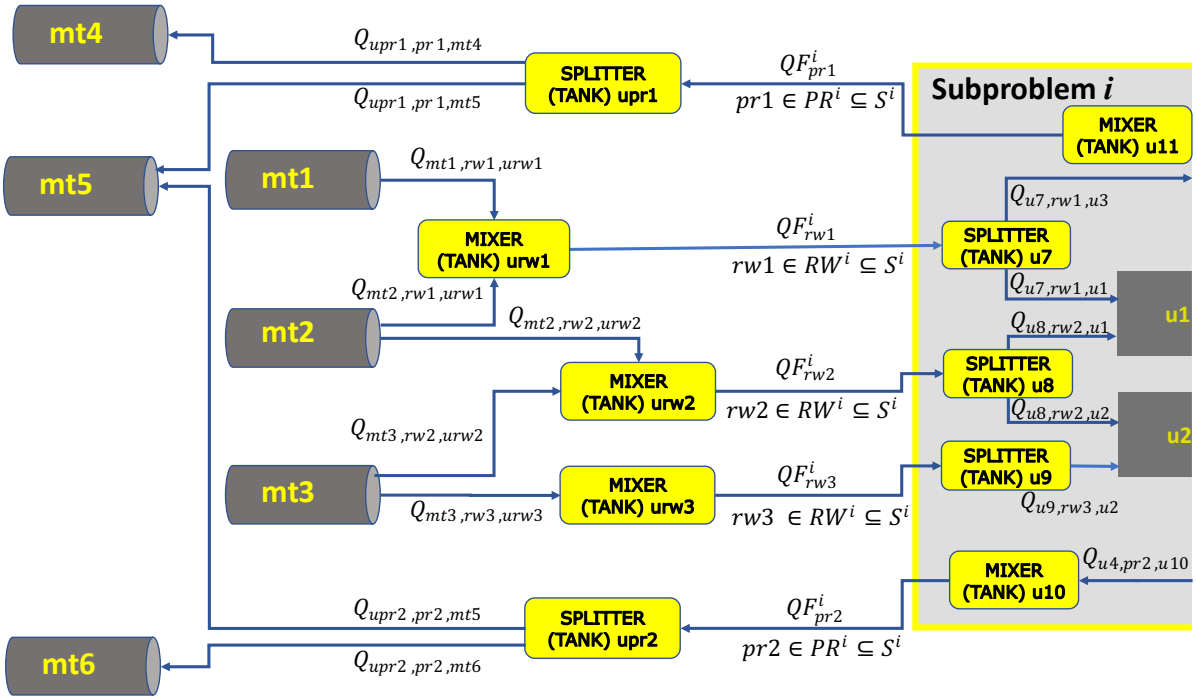


Figure G. 2. Raw material and product stream transportation scheme.

Unidirectional flow is assumed for each pipeline system. Each raw material or product stream is allowed to be transported through one, two or more transportation modes (pipeline systems), as shown in Figure G. 2. Then, QT_{mt}^{i} can be calculated by Eq. (G-8):

$$QT_{mt}^{i} = \sum_{(upr,pr)\in UPR_{mt}^{i}} Q_{upr,pr,mt} + \sum_{(rw,urw)\in URW_{mt}^{i}} Q_{mt,rw,urw} \; \forall \; mt \in MT^{i} \qquad \text{(G-8)}$$

Where UPR_{mt}^{i} is a subset indicating all product streams being delivered through transportation mode mt, and their corresponding product unit from which they are outgoing. In Figure G. 2, the subsets UPR_{mt}^{i} for each $mt \in MT^{i}$ are presented as follows:

$$UPR_{mt1}^{i} = \{\quad\} \qquad UPR_{mt4}^{i} = \{upr1 \quad pr1\}$$

$$UPR_{mt2}^{i} = \{\quad\} \qquad UPR_{mt5}^{i} = \begin{Bmatrix} upr1 & pr1 \\ upr2 & pr2 \end{Bmatrix}$$

$$UPR_{mt3}^{i} = \{\quad\} \qquad UPR_{mt6}^{i} = \{upr2 \quad pr2\}$$

Analogously,

$$URW_{mt1}^{i} = \{urw1 \quad rw1\} \qquad URW_{mt4}^{i} = \{\quad\}$$

$$URW_{mt2}^{i} = \begin{Bmatrix} urw1 & rw1 \\ urw2 & \mathrm{rw2} \end{Bmatrix} \qquad URW_{mt5}^{i} = \{\quad\}$$

$$URW_{mt3}^{i} = \begin{Bmatrix} urw2 & rw2 \\ urw3 & rw3 \end{Bmatrix} \qquad URW_{mt6} = \{\quad\}$$

The following restriction on the total flow by the transportation mode mt must be applied to avoid surpassed the limit of the piping system capacity:

$$QT_{mt}^{L} \leq QT_{mt}^{i} \leq QT_{mt}^{U} \forall mt \in MT^{i} \qquad \text{(G-9)}$$

The Operating costs ($Cost\_OpEx^{i}$) for problem i is defined as a summatory of the operating costs of all units inside the problem, as follows:

$$Cost\_OpEx^i = \sum_{u \in UInt^i} \left(\omega_u . QF_u^i + \sum_{p \in PI_u} \psi_{u,p} . QF_u^i . PF_{u,p}^i + \sum_{u' \in US_u} \sum_{s \in SO_{u'}} \phi_{u',s,u} . Q_{u',s,u}^i \right) \quad \text{(G-10)}$$

Where ω_u, $\psi_{u,p}$ and $\phi_{u',s,u}$ are positive costing indexes that can be null for certain units inside the problem i.

Finally, the cost of linking streams coming in from another problem j to problem i ($cost^i$) is calculated in the Eq. (G-11) as a function of the Lagrange multipliers:

$$cost^i = \sum_{j \neq i} \sum_{e \in Ein_j^i} \lambda_e^{j,i} . QF_e^i + \sum_{j \neq i} \sum_{e \in Ein_j^i} \sum_{p \in P^i} \lambda_{e,p}^{j,i} . PF_{e,p}^i \quad \text{(G-11)}$$

It is assumed that each linking stream entering problem i is coming from only one problem j.

Each subproblem i (Figure G. 1) has three types of process units:

- Internal units: that represent the unit operations $(u_1, u_2, \dots, u_6)$ transforming the raw materials and exchanged streams into sellable products or commodities to trade with other business units.
- Farm tanks for the products that are sold at the gate of the business unit i to the market through units u_{10} and u_{11} or to other business units j managing the tanks u_{12} and u_{15}, with $j \neq i$
- Farm tanks to manage the reception of raw materials (u_7, u_8, u_9) at the business unit i and the exchange of commodities (u_{13}, u_{14}) coming in from the business units j , with $j \neq i$.

The modelling approach for each business unit and the monolithic IRPC problem is based on the framework proposed by Neiro and Pinto (2004). The process network topology can be represented by a sequence of mixers, process units (e.g. CDUs, FCCs, HDTs, tanks, etc.) and splitters as it is shown in the Figure G. 3. The variable $Q_{u',s,u}$ represents the flowrate of the stream s, which is an

outlet stream from the process u' and it is routed to process u. The variable QF_u manages the feedstock for the unit u and the outlet stream s from process u is accounted by variable $QS_{u,s}$. The variables $PF_{u,p}$ and $PS_{u,s,p}$ represent the characterization or property p for the feedstock to unit u and the stream s leaving unit u. The binary variable $y_{w,u}$ represents the operational conditions w which can be non-exclusive or mutually exclusive campaigns for the unit u.

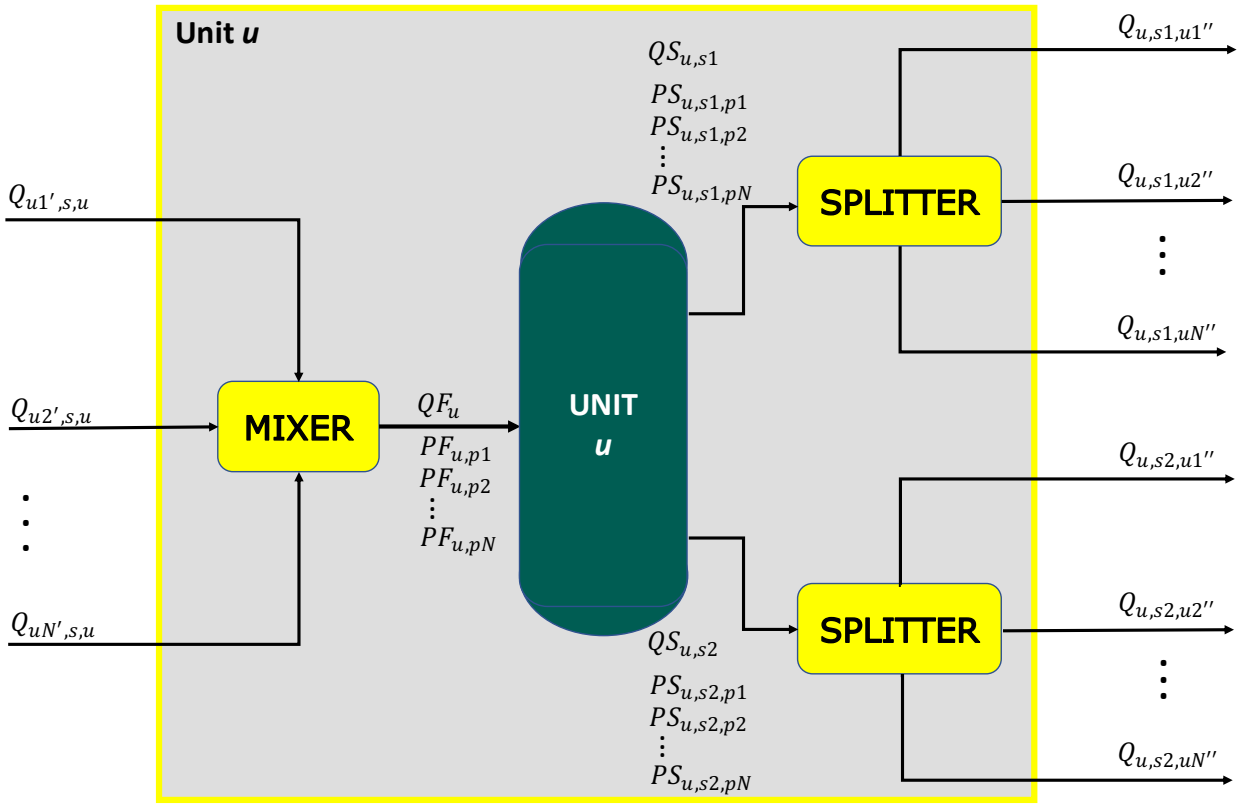


Figure G. 3. General unit model adapted from Neiro and Pinto (2004) to consider raw materials streams, linking streams between subproblems and product streams to external market.

There are three main indexes, process units (u), process streams (s) and stream properties (p). These main sets are used to build the process network connectivity as follows: the set SO_u denotes the duple outlet stream s from unit u, US_u is a tuple to represent the routing for stream s from unit

u' to unit u. Moreover, the property p for the feedstock at process unit u is represented by the duple PI_u while the property for the oulet stream s is denoted by the tuple $PO_{u,s}$.

Inputs to the model of the unit u itself include the feedstock volume (QF_u), properties ($PF_{u,p}$) and the operational conditions ($y_{w,u}$) which can be non-exclusive or mutually exclusive campaigns. These inputs are used to predict the flowrate ($QS_{\mathrm{u,s}}$) of each outlet stream s from unit u and its corresponding properties ($PS_{\mathrm{u,s,p}}$) through a process model that may be based on conservation principles, yield or constitutive relations or empirical correlations. Afterwards, each outlet stream can then take different paths to other units.

The outlet stream s from unit t u is calculated by Eq. (G-12), where function $f_{u,s}$ can be an empirical correlation, a data-driven model, a fixed yield approach or a model based on conservation principles.

$$f_{u,s}\left(QF_u, QS_{u,s}, PF_{u,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u, u \in UInt^i \subseteq U \tag{G-12}$$

Eq. G-13 calculates the property p for the outlet stream s from unit u, where $g_{u,p}$ could have a similar mathematical structure as $f_{u,s}$.

$$g_{u,p}\left(QF_u, PF_{u,p}, QS_{u,s}, PS_{u,s,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u,\ p \in PO_{u,s}, u \in UInt^i \tag{G-13}$$

The mathematical representation of functions $f_{u,s}$ and $g_{u,p}$ in the equations (G-12) and (G-13) respectively unit u to be modeled. For example, models for blender tanks and pipelines are presented by Neiro and Pinto (2004). Whilst, crude fractionation units (atmospheric distillation, vacuum distillation and debutanizer columns) and conversion units, the models are presented in the electronic supplementary material of our previous paper (Uribe-Rodriguez et al., 2020).

The binary variables denoted by $y_{w,u}$ are implicit in the process model (Eqs G-12 - G-13) and are used to represent mutually exclusive, or inclusive, modes of operation of the unit u. Given w

representing a real processing unit that has been divided into a set of virtual units UW_w, one for each operating mode, and W^{ME} including the units with mutually exclusive modes, for such units, only one operating condition is allowed to be selected, then Eq. (G-14) is formulated as an equality constraint, where the binary variable $y_{w,u}$ takes a value of 1 if mode u (virtual unit) is selected. Eq. (G-15) then determines the bounds on the flowrate QF_u. The maximum capacity available for units with inclusive operating modes is given by Eq. (G-16), where cap_w^L/cap_w^U and sf_w represent the minimum/maximum capacity and service factor for the real unit w. sf_w is a value in the range $[0,1]$ indicating the fraction of the process capacity available in the real unit w.

$$\sum_{u\in WU_w} y_{w,u} \leq 1 \; \forall w \in W^{ME} \tag{G-14}$$

$$cap_w^L.sf_w.y_{w,u} \leq QF_u \leq cap_w^U.sf_w.y_{w,u} \; \forall w \in W^{ME}, u \in UW_w \tag{G-15}$$

$$cap_w^L.sf_w \leq \sum_{u\in UW_w} QF_u \leq cap_w^U.sf_w \; \forall w \in W \backslash W^{ME} \tag{G-16}$$

If a specific unit does not have several operating modes, $w = u,\ y_{u,u} = 1$

Bounds for the variables for $PF_{u,p}$, $QS_{u,s}$ and $PS_{u,s,p}$ are imposed by Eqs. (G-17) – (G-19).

$$PF_{u,p}^L \leq PF_{u,p} \leq PF_{u,p}^U \;\; \forall u \in UInt^i, p \in PI_u \tag{G-17}$$

$$QS_{u,s}^L \leq QS_{u,s} \leq QS_{u,s}^U \;\; \forall u \in UInt^i, s \in SO_u \tag{G-18}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U \;\; \forall u \in UInt^i, s \in SO_u, p \in PO_{u,s} \tag{G-19}$$

When the streams are mixtures of several components, for example crude blends, the volume fraction of each component is included as an element of the subset of properties denoted by P^{VF}. The summatory of all the volume fractions is equal to one:

$$\sum_{p\in PI_u \cap P^{VF}} PF_{u,p} = 1 \;\; \forall u \in UInt^i \subseteq U \tag{G-20}$$

$$\sum_{p\in PO_{s,u} \cap P^{VF}} PS_{u,s,p} = 1 \;\; \forall s \in SO_u, u \in UInt^i \subseteq U \tag{G-21}$$

The flowrate QF_u is calculated by Eq. (G-22), which represents a mixer, where all the streams $Q_{u',s,u}$ are routed to the unit u. Moreover, the feedstock properties $PF_{u,p}$ are estimated by equations (G-23) and (G-24) using volumetric and weight blend respectively.

$$QF_u = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall u \in UInt^i \subseteq U \tag{G-22}$$

Where US_u the subset of upstream processes connected to u. The second part of the model is intended to calculate the properties of the of the inlet stream to the unit itself:

$$QF_u PF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall \, u \in UInt^i \subseteq U, p \in PI_u \cap P^V \tag{G-23}$$

$$QF_u PF_{u,p} PF_{u,SPG} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall u \in UInt^i \subseteq U, p \in PI_u \backslash P^V \tag{G-24}$$

Note that trilinear terms in the previous equation can be expressed as bilinear terms, defining auxiliary variables with the form of $QFaux_u = QF_u P_{u,s,SPG}$ and $Qaux_{u',s,u} = Q_{u',s,u} PS_{u,s,SPG}$.

On the other hand, the model for each splitter can be stated as follows:

$$QS_{u,s} = \sum_{u' \in UO_u} Q_{u,s,u'} \quad \forall \, u \in UInt^i \subseteq U, s \in SO_u \tag{G-25}$$

Where UO_u is the subset of downstream units connected to unit u. For each outlet stream $s \in SO_u$ of the unit u, property values ($PS_{u,s,p}$) are preserved before and after the splitter.

$Q_{u,s,u'}$ is bounded:

$$Q^L_{u,s,u'} \leq Q_{u,s,u'} \leq QS^U_{u,s,u'} \quad \forall u \in UInt^i \subseteq U, s \in SO_u, u'' \in UO_u \tag{G-26}$$

The second type of units present in the problem i are the product units, which are mixer tanks. These units manage products sold to the market or the linking streams between problems i and j, with $j \neq i$. See equations (G-27) – (G-33).

$$QF^i_e = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall \, e \in \left(Eout^i \cup PR^i\right), u \in u^i \cap (UEout^i \cup U^{PR}) \tag{G-27}$$

$$QF_e^i . PF_{e,p}^i = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^e \cap P^V , e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-28)}$$

$$QF_e^i . PF_{e,p}^i . PF_{e,SPG}^i = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^e \backslash P^V , e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-29)}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^i = 1 \quad \forall e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-30)}$$

$$QF_e^L \leq QF_e^i \leq QF_e^U \quad \forall e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-31)}$$

It is highlighted that in this type of unit, $s = e$, $QF_u = QS_{u,s} = QF_e^i$.

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^i = 1 \quad \forall e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-32)}$$

$$QF_e^L \leq QF_e^i \leq QF_e^U \quad \forall e \in \left(Eout^i \cup PR^i\right) \quad \text{(G-33)}$$

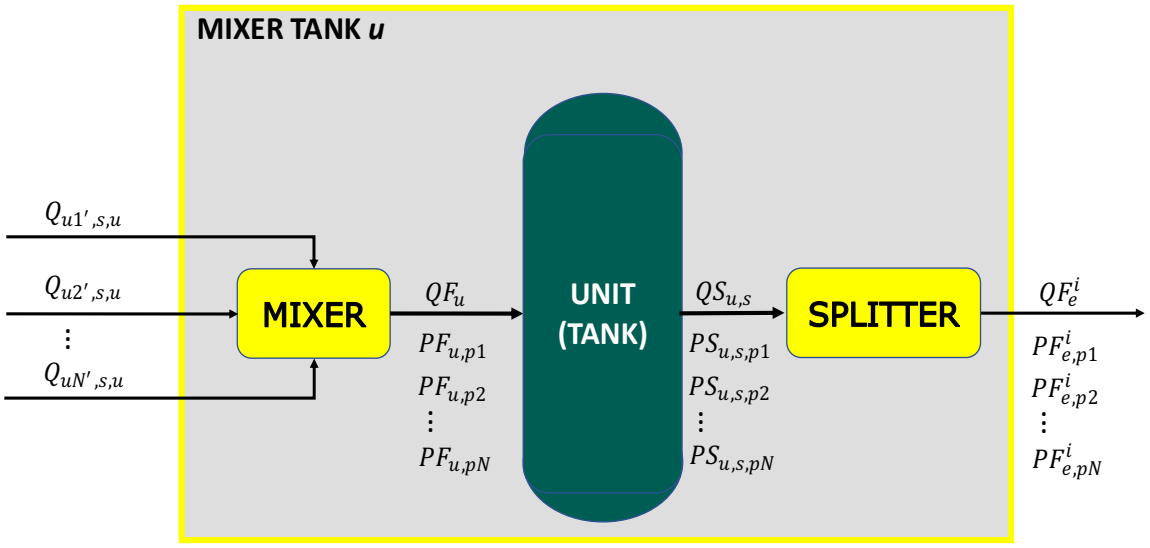


Figure G. 4. Product unit scheme.

The third type of unit inside the subproblem is the raw material unit (splitter tank), the scheme and model (assumed as a mere splitter) are presented as follows:

$$QF_e^i = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall \, e \in \left(Ein^i \cup RW^i\right), u \in u^i \cap \left(UEin^i \cup U^{RW}\right) \quad \text{(G-34)}$$

$$QF_e^L \leq QF_e^i \leq QF_e^U \; \forall \, e \in \left(Ein^i \cup RW^i\right) \quad \text{(G-35)}$$

$$Q^{L}_{u,s,u''} \leq Q_{u,s,u''} \leq QF^{U}_{u,s,u''} \; \forall \, u \in u^{i} \cap (UEin^{i} \cup U^{RW}), s \in \left(Ein^{i} \cup RW^{i}\right), u'' \in UO_{u} \tag{G-36}$$

In this type of unit, $s = e$.

Stream property values are preserved before and after each raw material stream unit:

$$PS_{u,s,p} = PF^{i}_{e,p} \;\; \forall u \in u^{i} \in u^{i} \cap (UEin^{i} \cup U^{RW}), e \in \left(Ein^{i} \cup RW^{i}\right), p \in PE^{e} \tag{G-37}$$

A restriction on the values of the properties should be applied to monitor that subproblem is receiving the right raw material or linking stream:

$$PS^{L}_{u,s,p} \leq PS_{u,s,p} \leq PS^{U}_{u,s,p} \;\; \forall u \in u^{i} \cap (UEin^{i} \cup U^{RW}), s \in \left(Ein^{i} \cup RW^{i}\right), p \in PE^{e} \tag{G-38}$$

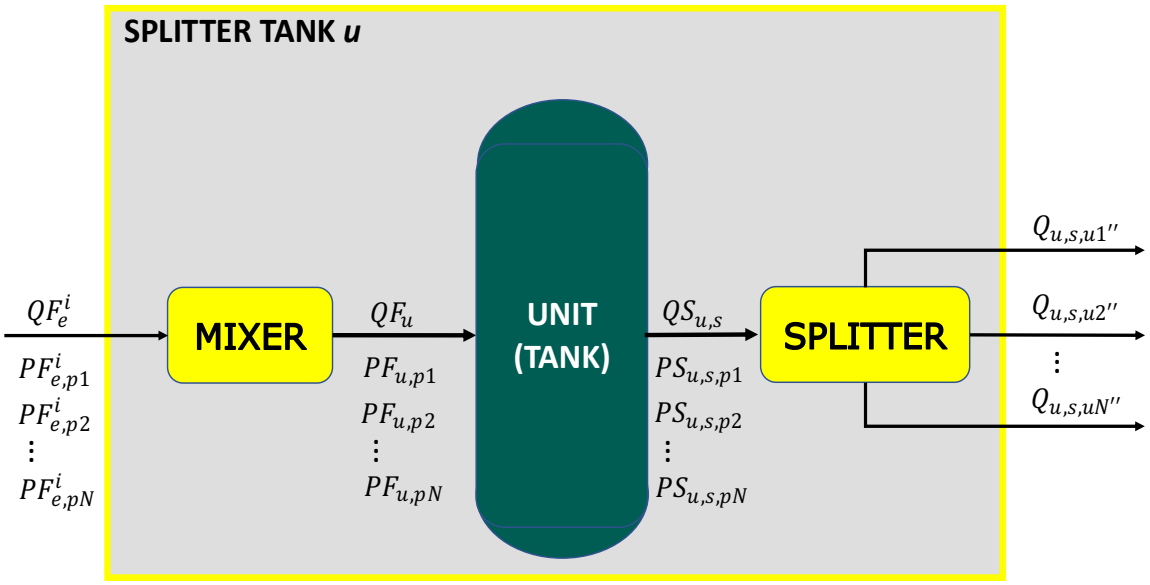


Figure G. 5. Raw material stream unit scheme.

Crude Management (CM) model

Crude management (CM) buy domestic or imported crude oils from external market (domestic fields or national ports), mix them and send the crude blends to the refinery section (REF). Note that if the two optimisation problems are solved independently, there will be an imbalance between

the flowrate, bulk properties and composition of the crude blends leaving CM and arriving REF at the CDU charge tanks. Specifically, CM will maximize its profit by buying cheap crude oil from the market, minimizing the transportation cost to the refinery and selling crude blends at the highest price, without being concerned with the operational performance of REF. In contrast, REF will maximize its profit by buying enough quantity of good-quality crude blends from CM at a cheap price, without being concerned with the costs incurred by CM in purchasing and delivering the crude.

The crude management (CM) optimisation problem is a quadratically constrained quadratic program (QCQP), comprising a set of equations representing transport capacity (Eqs. G-41 - G-42), mass balances, capacity constraints and bounds for variables in splitter tanks (Eqs. G-43 - G-47) and mixer tanks (Eqs. G-48 - G-54). Whereas equation G-51 ensures the volumetric composition of crude blends. The nonconvexities arise from mixing on volume G-49 and weight G-50 basis to estimate stream properties. Thus, the objective function $\mathbf{OF^{CM}}$ is to maximize profit given by the revenue obtained from selling crude blend flows QF_e^{CM} at price $\lambda_e^{CM,REF}$, considering a quality bonus $\lambda_{e,p}^{CM,REF}$ for property $PF_{e,p}^{CM}$, where $e \in Eout_{REF}^{CM} = E^{CB} = \{CB1, \dots, CB9\}$ refers to the crude blends going from CM to REF, and $p \in P^{CB}$ can be a bulk property (specific gravity, sulphur content or TAN) or the volumetric composition of the crude blend; minus the outcome represented by the crude oil QF_{rw}^{CM} purchases at cost β_{rw}, where $rw \in RW^{CM} = DC \cup IC$ (domestic crudes plus imported crudes), and the total volume delivered QT_{mt}^{CM} at transport fee γ_{mt}. Since all the energy to pump the crudes and crude blends is assumed to be included in the logistic costs for each transportation mode (mt), the OpEx for every unit inside CM is taken as null.

$$f_0^{CM}(x^{CM}) = profit^{CM} = totalrevenue^{CM} - totalcost^{CM} \qquad (\mathbf{OF^{CM}})$$

Where x^{CM} is denoting the continuous variables. No binary variables are present in CM optimisation problem.

$$totalrevenue^{CM} = \sum_{e \in Eout_{REF}^{CM}} \lambda_e^{CM,RPB} QF_e^{CM} + \sum_{e \in Eout_{REF}^{CM}} \sum_{p \in P^{CB}} \lambda_{e,p}^{CM,RPB} PF_{e,p}^{CM} \quad \text{(G-39)}$$

$$totalcost^{CM} = \sum_{rw \in RW^{CM}} \beta_{rw} \,.\, QF_{rw}^{CM} + \sum_{mt \in MT^{CM}} \gamma_{mt} \,.\, QT_{mt}^{CM} \quad \text{(G-40)}$$

s. t.

Transport capacity for CM:

$$QT_{mt}^{CM} = \sum_{(upr,pr) \in UPR_{mt}^{CM}} Q_{upr,pr,mt} + \sum_{(rw,urw) \in URW_{mt}^{CM}} Q_{mt,rw,urw} \; \forall \; mt \in MT^{CM} \quad \text{(G-41)}$$

$$QT_{mt}^{L} \leq QT_{mt}^{CM} \leq QT_{mt}^{U} \forall mt \in MT^{CM} \quad \text{(G-42)}$$

Equations for units in CM receiving raw materials (domestic or imported crudes). In this type of units $e = s = rw$:

$$QF_{rw}^{CM} = \sum_{u'' \in UO_u} Q_{u,s,u''} \;\; \forall \; rw \in RW^{CM}, u \in U^{CM} \cap U^{RW} \quad \text{(G-43)}$$

$$QF_{rw}^{L} \leq QF_{rw}^{CM} \leq QF_{rw}^{U} \; \forall \; rw \in RW^{CM} \quad \text{(G-44)}$$

$$Q_{u,s,u''}^{L} \leq Q_{u,s,u''}^{CM} \leq Q_{u,s,u''}^{U} \; \forall \; u \in U^{CM} \cap U^{RW}, s \in RW^{CM}, u'' \in UO_u \quad \text{(G-45)}$$

$$PS_{u,s,p} = PF_{rw,p}^{CM} \; \forall \; u \in U^{CM} \cap U^{RW}, rw \in RW^{CM}, p \in PE^{rw} \quad \text{(G-46)}$$

$$PS_{u,s,p}^{L} \leq PS_{u,s,p} \leq PS_{u,s,p}^{U} \; \forall \; u \in U^{CM} \cap U^{RW}, s \in RW^{CM}, p \in PE^{rw} \quad \text{(G-47)}$$

Equations for units in CM producing the crude blends ($s = e, e \in Eout_{REF}^{CM} = E^{CB}, PE^{e} = PE^{CB}$):

$$QF_e^{CM} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \;\; \forall \; e \in Eout_{REF}^{CM}, u \in UEout_{REF}^{CM} \quad \text{(G-48)}$$

$$QF_e^{CM} . PF_{e,p}^{CM} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^{e} \cap P^{V} \,, e \in Eout_{REF}^{CM} \quad \text{(G-49)}$$

$$QF_e^{CM} . PF_{e,p}^{CM} . PF_{e,SPG}^{CM} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^{e} \backslash \mathrm{P}^{V} \,, e \in Eout_{REF}^{CM} \quad \text{(G-50)}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^{CM} = 1 \ \forall\, e \in Eout_{REF}^{CM} \quad \text{(G-51)}$$

$$QF_e^L \leq QF_e^{CM} \leq QF_e^U \ \ \forall e \in Eout_{REF}^{CM} \quad \text{(G-52)}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{e,p}^{CM} \ \forall\, u \in UEout_{REF}^{CM}, e \in Eout_{REF}^{CM}, p \in PE^e \quad \text{(G-53)}$$

$$PF_{e,p}^L \leq PF_{e,p}^{CM} \leq PF_{e,p}^U \ \forall e \in Eout_{REF}^{CM}, p \in PE^e \quad \text{(G-54)}$$

Refinery (REF) model

The refinery (REF) problem is cast as a mixed-integer quadratically constrained quadratic program (MIQCQP). The objective function $\mathbf{OF^{REF}}$ represents the profit to be maximized. The income has three components, $\sum_{pr \in PR^{REF}} \alpha_{pr}.\, QF_{pr}^{REF}$ indicates the revenue for selling refined products to external market. The second element $\sum_{e \in Eout_{PTQ}^{REF}} \lambda_e^{REF,PTQ}.\, QF_e^{REF} + \sum_{e \in Eout_{PTQ}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,PTQ}.\, PF_{e,p}^{REF}$ corresponds to the refined streams sold to the petrochemical plants (PTQ). And the third term $\sum_{e \in Eout_{FB}^{REF}} \lambda_e^{REF,FB}.\, QF_e^{REF} + \sum_{e \in Eout_{FB}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,FB}.\, PF_{e,p}^{REF}$ represents the sales to the fuel blending section. The outcomes are given by crude blends bought to the crude management (CM) section and the purchase of streams from the petrochemical sections to improve the gasoline blending, provide raw material to specialty solvent production, and hydrogen for hydrotreating. These are represented by the terms $\sum_{e \in Ein_{PTQ}^{REF}} \lambda_e^{PTQ,REF}.\, QF_e^{REF} + \sum_{e \in Ein_{PTQ}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{PTQ,REF}.\, PF_{e,p}^{REF}$ and $\sum_{e \in Ein_{CM}^{REF}} \lambda_e^{CM,REF}.\, QF_e^{REF} + \sum_{e \in Ein_{CM}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{CM,REF}.\, PF_{e,p}^{REF}$ respectively. Other component of the outcome is the requirement of refined products to fulfill the feedstock for some refining units and improve fuels quality $\sum_{rw \in RW^{REF}} \beta_{rw}.\, QF_{rw}^{REF}$, the refining units operating costs $\sum_{u \in UInt^{REF}} \left(\omega_u.\, QF_u^{REF} + \sum_{p \in PI_u} \psi_{u,p}.\, QF_u^{REF}.\, PF_{u,p}^{REF} + \sum_{u' \in US_u} \sum_{s \in SO_{u'}} \phi_{u',s,u}.\, Q_{u',s,u}^{REF} \right)$, and the transportation cost $\sum_{mt \in MT^{REF}} \gamma_{mt}.\, QT_{mt}^{REF}$:

$$f_0^{REF}(x^{REF}, y^{REF}) = \sum_{pr \in PR^{REF}} \alpha_{pr}. QF_{pr}^{REF} + \sum_{e \in Eout_{PTQ}^{REF}} \lambda_e^{REF,PTQ}. QF_e^{REF} +$$
$$\sum_{e \in Eout_{PTQ}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,PTQ}. PF_{e,p}^{REF} + \sum_{e \in Eout_{FB}^{REF}} \lambda_e^{REF,FB}. QF_e^{REF} +$$
$$\sum_{e \in Eout_{FB}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,FB}. PF_{e,p}^{REF} - \sum_{e \in Ein_{PTQ}^{REF}} \lambda_e^{PTQ,REF}. QF_e^{REF} -$$
$$\sum_{e \in Ein_{PTQ}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{PTQ,REF}. PF_{e,p}^{REF} - \sum_{e \in Ein_{CM}^{REF}} \lambda_e^{CM,REF}. QF_e^{REF} - \quad (\mathbf{OF^{REF}})$$
$$\sum_{e \in Ein_{CM}^{REF}} \sum_{p \in PE^e} \lambda_{e,p}^{CM,REF}. PF_{e,p}^{REF} - \sum_{rw \in RW^{REF}} \beta_{rw} . QF_{rw}^{REF} - \sum_{u \in UInt^{REF}} \left(\omega_u. QF_u^{REF} + \right.$$
$$\left. \sum_{p \in PI_u} \psi_{u,p}. QF_u^{REF}. PF_{u,p}^{REF} + \sum_{u' \in US_u} \sum_{s \in SO_{u'}} \phi_{u',s,u}. Q_{u',s,u}^{REF} \right) - \sum_{mt \in MT^{REF}} \gamma_{mt} . QT_{mt}^{REF}$$

Where x^{REF} is denoting the continuous variables and y^{REF} corresponds to the binary variables present in REF optimisation problem.

s. t.

Transport capacity for REF:

$$QT_{mt}^{REF} = \sum_{(upr,pr) \in UPR_{mt}^i} Q_{upr,pr,mt} + \sum_{(rw,urw) \in URW_{mt}^i} Q_{mt,rw,urw} \; \forall \; mt \in MT^{REF} \quad \text{(G-55)}$$

$$QT_{mt}^L \leq QT_{mt}^{REF} \leq QT_{mt}^U \forall mt \in MT^{REF} \quad \text{(G-56)}$$

Equations for units in REF receiving raw materials (alkylate and gasoil). In this type of units $e = s = rw$:

$$QF_{rw}^{REF} = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall rw \in RW^{REF}, u \in U^{REF} \cap U^{RW} \quad \text{(G-57)}$$

$$QF_{rw}^L \leq QF_{rw}^{REF} \leq QF_{rw}^U \;\; \forall rw \in RW^{REF} \quad \text{(G-58)}$$

$$Q_{u,s,u''}^L \leq Q_{u,s,u''} \leq QF_{u,s,u''}^U \;\; \forall \; u \in U^{REF} \cap U^{RW}, s \in RW^{REF}, u'' \in UO_u \quad \text{(G-59)}$$

$$PS_{u,s,p} = PF_{rw,p}^{REF} \;\; \forall u \in U^{REF} \cap U^{RW}, rw \in RW^{REF}, p \in PE^{rw} \quad \text{(G-60)}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U \;\; \forall u \in U^{REF} \cap U^{RW}, s \in RW^{REF}, p \in PE^{rw} \quad \text{(G-61)}$$

Equations for units in REF receiving linking streams from CM (crude blends). In this type of units, $s = e, e \in Ein_{CM}^{REF} = E^{CB}$:

$$QF_e^{REF} = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall e \in Ein_{CM}^{REF}, u \in UEin_{CM}^{REF} \quad \text{(G-62)}$$

$$QF_e^L \leq QF_e^{REF} \leq QF_e^U \;\; \forall e \in Ein_{CM}^{REF} \quad \text{(G-63)}$$

$$Q_{u,s,u''}^L \leq Q_{u,s,u''} \leq QF_{u,s,u''}^U \;\; \forall u \in UEin_{CM}^{REF}, s \in Ein_{CM}^{REF}, u'' \in UO_u \quad \text{(G-64)}$$

$$PS_{u,s,p} = PF_{e,p}^{REF} \;\; \forall u \in UEin_{CM}^{REF}, e \in Ein_{CM}^{REF}, p \in PE^e \quad \text{(G-65)}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U \;\; \forall u \in UEin_{CM}^{REF}, s \in Ein_{CM}^{REF}, p \in PE^e \quad \text{(G-66)}$$

Equations for units in REF receiving linking streams from PTQ (H2, raffinate).

$$QF_e^{REF} = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall e \in Ein_{PTQ}^{REF}, u \in UEin_{PTQ}^{REF} \quad \text{(G-67)}$$

$$QF_e^L \leq QF_e^{REF} \leq QF_e^U \forall e \in Ein_{PTQ}^{REF} \quad \text{(G-68)}$$

$$Q_{u,s,u''}^L \leq Q_{u,s,u''} \leq QF_{u,s,u''}^U \forall u \in UEin_{PTQ}^{REF}, s \in Ein_{PTQ}^{REF}, u'' \in UO_u \quad \text{(G-69)}$$

$$PS_{u,s,p} = PF_{e,p}^{REF}, \forall e \in Ein_{PTQ}^{REF}, p \in PE^e \quad \text{(G-70)}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U \forall s \in Ein_{PTQ}^{REF}, p \in PE^e \quad \text{(G-71)}$$

Equations for units in REF delivering final products to the external market. (In this type of units, e = $s = pr$, $QF_u = QS_{u,s} = QF_{pr}^{REF}$):

$$QF_{pr}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \;\; \forall \, pr \in PR^{REF}, u \in U^{REF} \cap (U^{PR}) \quad \text{(G-72)}$$

$$QF_{pr}^{REF} . PF_{pr,p}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^{pr} \cap P^V, pr \in PR^{REF} \quad \text{(G-73)}$$

$$QF_{pr}^{REF} . PF_{pr,p}^{REF} . PF_{pr,SPG}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^{pr} \backslash P^V, pr \in PR^{REF} \quad \text{(G-74)}$$

$$\sum_{p \in PI_u \cap P^{VF}} PF_{pr,p}^{REF} = 1 \;\; \forall \, pr \in PR^{REF} \quad \text{(G-75)}$$

$$QF_{pr}^L \leq QF_{pr}^{REF} \leq QF_{pr}^U \;\; \forall pr \in PR^{REF} \quad \text{(G-76)}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{pr,p}^{REF} \;\; \forall u \in U^{REF} \cap U^{PR}, pr \in PR^{REF}, p \in PE^{pr} \quad \text{(G-77)}$$

$$PF_{pr,p}^L \leq PF_{pr,p}^{REF} \leq PF_{pr,p}^U \;\; \forall pr \in PR^{REF}, p \in PE^{pr} \quad \text{(G-78)}$$

Equations for units in REF producing linking streams to PTQ:

$$QF_e^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall\, e \in Eout_{PTQ}^{REF}, u \in UEout_{PTQ}^{REF} \tag{G-79}$$

$$QF_e^{REF}.PF_{e,p}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^e \cap P^V, e \in Eout_{PTQ}^{REF} \tag{G-80}$$

$$QF_e^{REF}.PF_{e,p}^{REF}.PF_{e,SPG}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^e \backslash \mathrm{P}^V, e \in Eout_{PTQ}^{REF} \tag{G-81}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^{REF} = 1 \quad \forall\, e \in Eout_{PTQ}^{REF} \tag{G-82}$$

$$QF_e^L \le QF_e^{REF} \le QF_e^U \quad \forall e \in Eout_{PTQ}^{REF} \tag{G-83}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{e,p}^{REF} \; \forall\, u \in UEout_{PTQ}^{REF}, e \in Eout_{PTQ}^{REF}, p \in PE^e \tag{G-84}$$

$$PF_{e,p}^L \le PF_{e,p}^{REF} \le PF_{e,p}^U \; \forall e \in Eout_{PTQ}^{REF}, p \in PE^e \tag{G-85}$$

Equations for units in REF producing linking streams to FB:

$$QF_e^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall\, e \in Eout_{FB}^{REF}, u \in UEout_{FB}^{REF} \tag{G-86}$$

$$QF_e^{REF}.PF_{e,p}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^e \cap P^V, e \in Eout_{FB}^{REF} \tag{G-87}$$

$$QF_e^{REF}.PF_{e,p}^{REF}.PF_{e,SPG}^{REF} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^e \backslash \mathrm{P}^V, e \in Eout_{FB}^{REF} \tag{G-88}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^{REF} = 1 \quad \forall\, e \in Eout_{FB}^{REF} \tag{G-89}$$

$$QF_e^L \le QF_e^{REF} \le QF_e^U \quad \forall e \in Eout_{FB}^{REF} \tag{G-90}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{e,p}^{REF} \; \forall\, u \in UEout_{FB}^{REF}, e \in Eout_{FB}^{REF}, p \in PE^e \tag{G-91}$$

$$PF_{e,p}^L \le PF_{e,p}^{REF} \le PF_{e,p}^U \; \forall e \in Eout_{FB}^{REF}, p \in PE^e \tag{G-92}$$

Equations for internal units in REF:

$$f_{u,s}\left(QF_u, QS_{u,s}, PF_{u,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u, u \in UInt^{REF} \tag{G-93}$$

$$g_{u,p}\left(QF_u, PF_{u,p}, QS_{u,s}, PS_{u,s,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u,\; p \in PO_{u,s}, u \in UInt^{REF} \tag{G-94}$$

As mentioned before, the specific form that can take the previous equations (G-93) - (G-94) in the model of each unit u depends on the type of unit being considered. The general forms of these equations for crude fractionation units (atmospheric distillation, vacuum distillation and debutanizer columns) and conversion units are presented in the electronic supplementary material of our previous paper (Uribe-Rodriguez et al., 2020) and can be classified in three main types: based on conservation principles, yield or constitutive relations and empirical correlations. The specific yields, factors or coefficients for specific units and feedstocks cannot be disclosed due to confidentiality issues.

$$\sum_{u \in WU_w} y_{w,u} \leq 1 \; \forall w \in W^{ME} \tag{G-95}$$

$$cap_w^L . sf_w . y_{w,u} \leq QF_u \leq cap_w^U . sf_w . y_{w,u} \; \forall w \in W^{ME}, u \in UW_w \tag{G-96}$$

$$cap_w^L . sf_w \leq \sum_{u \in UW_w} QF_u \leq cap_w^U . sf_w \; \forall w \in W \backslash W^{ME} \tag{G-97}$$

If a specific unit does not have several operating modes, $w = u$, $y_{u,u} = 1$

$$PF_{u,p}^L \leq PF_{u,p} \leq PF_{u,p}^U \;\; \forall u \in UInt^{REF}, p \in PI_u \tag{G-98}$$

$$QS_{u,s}^L \leq QS_{u,s} \leq QS_{u,s}^U \;\; \forall u \in UInt^{REF}, s \in SO_u \tag{G-99}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U \;\; \forall u \in UInt^{REF}, s \in SO_u, p \in PO_{u,s} \tag{G-100}$$

$$\sum_{p \in PI_u \cap P^{VF}} PF_{u,p} = 1 \;\; \forall u \in UInt^{REF} \tag{G-101}$$

$$\sum_{p \in PO_{s,u} \cap P^{VF}} PS_{u,s,p} = 1 \;\; \forall s \in SO_u, u \in UInt^{REF} \tag{G-102}$$

$$QF_u = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \;\; \forall u \in UInt^{REF} \tag{G-103}$$

$$QF_u PF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall \, u \in UInt^{REF}, p \in PI_u \cap P^V \tag{G-104}$$

$$QF_u PF_{u,p} PF_{u,SPG} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall u \in UInt^{REF}, p \in PI_u \backslash P^V \tag{G-105}$$

$$QS_{u,s} = \sum_{u'' \in UO_u} Q_{u,s,u''} \;\; \forall\, u \in UInt^{REF}, s \in SO_u \tag{G-106}$$

$$Q^L_{u,s,u''} \le Q_{u,s,u''} \le QS^U_{u,s,u''} \;\; \forall u \in UInt^{REF}, s \in SO_u, u'' \in UO_u \tag{G-107}$$

The nonconvexities in problem REF arise from Eqs. G-73, G-74, G-80, G-81, G-87, G-88, G-93, G-94, G-104, G-105.

Petrochemicals (PTQ) model

The petrochemical (PTQ) problem is formulated as a MIQCQP. The objective function $\mathbf{OF^{PTQ}}$ represents the profit to be maximized:

$$\begin{aligned}
&f_0^{PTQ}(x^{PTQ}, y^{PTQ}) = \sum_{pr \in PR^{PTQ}} \alpha_{pr}.\, QF_{pr}^{PTQ} + \sum_{e \in Eout_{REF}^{PTQ}} \lambda_e^{PTQ,REF}.\, QF_e^{PTQ} + \\
&\sum_{e \in Eout_{REF}^{PTQ}} \sum_{p \in PE^e} \lambda_{e,p}^{PTQ,REF}.\, PF_{e,p}^{PTQ} + \sum_{e \in Eout_{FB}^{PTQ}} \lambda_e^{PTQ,FB}.\, QF_e^{PTQ} + \\
&\sum_{e \in Eout_{FB}^{PTQ}} \sum_{p \in PE^e} \lambda_{e,p}^{PTQ,FB}.\, PF_{e,p}^{PTQ} - \sum_{e \in Ein_{REF}^{PTQ}} \lambda_e^{REF,PTQ}.\, QF_e^{PTQ} - \\
&\sum_{e \in Ein_{REF}^{PTQ}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,PTQ}.\, PF_{e,p}^{PTQ} - \sum_{rw \in RW^{PTQ}} \beta_{rw}\, .\, QF_{rw}^{PTQ} - \sum_{u \in UInt^{PTQ}} \Big(\omega_u.\, QF_u^{PTQ} + \\
&\sum_{p \in PI_u} \psi_{u,p}.\, QF_u^{PTQ}.\, PF_{u,p}^{PTQ} + \sum_{u' \in US_u} \sum_{s \in SO_{u'}} \phi_{u',s,u}.\, Q_{u',s,u}^{PTQ} \Big) - \sum_{mt \in MT^{PTQ}} \gamma_{mt}\, .\, QT_{mt}^{PTQ}
\end{aligned} \tag{$\mathbf{OF^{PTQ}}$}$$

Where x^{PTQ} is denoting the continuous variables and y^{PTQ} corresponds to the binary variables present in REF optimisation problem.

s. t.

Transport capacity for PTQ:

$$QT_{mt}^{PTQ} = \sum_{(upr,pr) \in UPR_{mt}^i} Q_{upr,pr,mt} + \sum_{(rw,urw) \in URW_{mt}^i} Q_{mt,rw,urw} \;\forall\, mt \in MT^{PTQ} \tag{G-108}$$

$$QT_{mt}^L \le QT_{mt}^{PTQ} \le QT_{mt}^U \forall mt \in MT^{PTQ} \tag{G-109}$$

Equations for units in PTQ receiving raw materials. In this type of units $e = s = rw$:

$$QF_{rw}^{PTQ} = \sum_{u'' \in UO_u} Q_{u,s,u''} \;\forall rw \in RW^{PTQ}, u \in U^{PTQ} \cap U^{RW} \tag{G-110}$$

$$QF_{rw}^{L} \leq QF_{rw}^{PTQ} \leq QF_{rw}^{U} \ \forall rw \in RW^{PTQ} \quad \text{(G-111)}$$

$$Q_{u,s,u''}^{L} \leq Q_{u,s,u''} \leq QF_{u,s,u''}^{U} \ \forall u \in U^{PTQ} \cap U^{RW}, s \in RW^{PTQ}, u'' \in UO_u \quad \text{(G-112)}$$

$$PS_{u,s,p} = PF_{rw,p}^{PTQ} \ \forall u \in U^{PTQ} \cap U^{RW}, rw \in RW^{PTQ}, p \in PE^{rw} \quad \text{(G-113)}$$

$$PS_{u,s,p}^{L} \leq PS_{u,s,p} \leq PS_{u,s,p}^{U} \ \forall u \in U^{PTQ} \cap U^{RW}, s \in RW^{PTQ}, p \in PE^{rw} \quad \text{(G-114)}$$

Equations for units in PTQ receiving linking streams from REF.

$$QF_{e}^{PTQ} = \sum_{u'' \in UO_u} Q_{u,s,u''} \ \forall e \in Ein_{REF}^{PTQ}, u \in UEin_{REF}^{PTQ} \quad \text{(G-115)}$$

$$QF_{e}^{L} \leq QF_{e}^{PTQ} \leq QF_{e}^{U} \forall e \in Ein_{REF}^{PTQ} \quad \text{(G-116)}$$

$$Q_{u,s,u''}^{L} \leq Q_{u,s,u''} \leq QF_{u,s,u''}^{U} \forall u \in UEin_{REF}^{PTQ}, s \in Ein_{REF}^{PTQ}, u'' \in UO_u \quad \text{(G-117)}$$

$$PS_{u,s,p} = PF_{e,p}^{PTQ}, \forall e \in Ein_{REF}^{PTQ}, p \in PE^{e} \quad \text{(G-118)}$$

$$PS_{u,s,p}^{L} \leq PS_{u,s,p} \leq PS_{u,s,p}^{U} \forall s \in Ein_{REF}^{PTQ}, p \in PE^{e} \quad \text{(G-119)}$$

Equations for units in PTQ delivering final products to the external market. (In this type of units, e = $s = pr$, $QF_u = QS_{u,s} = QF_{pr}^{PTQ}$):

$$QF_{pr}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \ \forall pr \in PR^{PTQ}, u \in U^{PTQ} \cap (U^{PR}) \quad \text{(G-120)}$$

$$QF_{pr}^{PTQ}.PF_{pr,p}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \ \forall p \in PE^{pr} \cap P^{V}, pr \in PR^{PTQ} \quad \text{(G-121)}$$

$$QF_{pr}^{PTQ}.PF_{pr,p}^{PTQ}.PF_{pr,SPG}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \ \forall p \in PE^{pr} \backslash P^{V}, pr \in PR^{PTQ} \quad \text{(G-122)}$$

$$\sum_{p \in PI_u \cap P^{VF}} PF_{pr,p}^{PTQ} = 1 \ \forall pr \in PR^{PTQ} \quad \text{(G-123)}$$

$$QF_{pr}^{L} \leq QF_{pr}^{PTQ} \leq QF_{pr}^{U} \ \forall pr \in PR^{PTQ} \quad \text{(G-124)}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{pr,p}^{PTQ} \ \forall u \in U^{PTQ} \cap U^{PR}, pr \in PR^{PTQ}, p \in PE^{pr} \quad \text{(G-125)}$$

$$PF_{pr,p}^{L} \leq PF_{pr,p}^{PTQ} \leq PF_{pr,p}^{U} \ \forall pr \in PR^{PTQ}, p \in PE^{pr} \quad \text{(G-126)}$$

Equations for units in PTQ producing linking streams to REF:

$$QF_e^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall\, e \in Eout_{REF}^{PTQ}, u \in UEout_{REF}^{PTQ} \tag{G-127}$$

$$QF_e^{PTQ}.PF_{e,p}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^e \cap P^V, e \in Eout_{REF}^{PTQ} \tag{G-128}$$

$$QF_e^{PTQ}.PF_{e,p}^{PTQ}.PF_{e,SPG}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^e \backslash \mathrm{P}^V, e \in Eout_{REF}^{PTQ} \tag{G-129}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^{PTQ} = 1 \;\; \forall\, e \in Eout_{REF}^{PTQ} \tag{G-130}$$

$$QF_e^L \leq QF_e^{PTQ} \leq QF_e^U \;\; \forall e \in Eout_{REF}^{PTQ} \tag{G-131}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{e,p}^{PTQ} \; \forall\, u \in UEout_{REF}^{PTQ}, e \in Eout_{REF}^{PTQ}, p \in PE^e \tag{G-132}$$

$$PF_{e,p}^L \leq PF_{e,p}^{PTQ} \leq PF_{e,p}^U \; \forall e \in Eout_{REF}^{PTQ}, p \in PE^e \tag{G-133}$$

Equations for units in PTQ producing linking streams (gasoline components) to FB:

$$QF_e^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \quad \forall\, e \in Eout_{FB}^{PTQ}, u \in UEout_{FB}^{PTQ} \tag{G-134}$$

$$QF_e^{PTQ}.PF_{e,p}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^e \cap P^V, e \in Eout_{FB}^{PTQ} \tag{G-135}$$

$$QF_e^{PTQ}.PF_{e,p}^{PTQ}.PF_{e,SPG}^{PTQ} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^e \backslash \mathrm{P}^V, e \in Eout_{FB}^{PTQ} \tag{G-136}$$

$$\sum_{p \in PE^e \cap P^{VF}} PF_{e,p}^{PTQ} = 1 \;\; \forall\, e \in Eout_{FB}^{PTQ} \tag{G-137}$$

$$QF_e^L \leq QF_e^{PTQ} \leq QF_e^U \;\; \forall e \in Eout_{FB}^{PTQ} \tag{G-138}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{e,p}^{PTQ} \; \forall\, u \in UEout_{FB}^{PTQ}, e \in Eout_{FB}^{PTQ}, p \in PE^e \tag{G-139}$$

$$PF_{e,p}^L \leq PF_{e,p}^{PTQ} \leq PF_{e,p}^U \; \forall e \in Eout_{FB}^{PTQ}, p \in PE^e \tag{G-140}$$

Equations for internal units in PTQ:

$$f_{u,s}\left(QF_u, QS_{u,s}, PF_{u,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u, u \in UInt^{PTQ} \tag{G-141}$$

$$g_{u,p}\left(QF_u, PF_{u,p}, QS_{u,s}, PS_{u,s,p}, y_{w,u}\right) = 0 \quad \forall\, s \in SO_u,\ p \in PO_{u,s}, u \in UInt^{PTQ} \tag{G-142}$$

$$\sum_{u \in WU_w} y_{w,u} \leq 1\ \forall w \in W^{ME} \tag{G-143}$$

$$cap_w^L . sf_w . y_{w,u} \leq QF_u \leq cap_w^U . sf_w . y_{w,u}\ \forall w \in W^{ME}, u \in UW_w \tag{G-144}$$

$$cap_w^L . sf_w \leq \sum_{u \in UW_w} QF_u \leq cap_w^U . sf_w\ \forall w \in W \backslash W^{ME} \tag{G-145}$$

If a specific unit does not have several G-149 operating modes, $w = u,\ y_{u,u} = 1$

$$PF_{u,p}^L \leq PF_{u,p} \leq PF_{u,p}^U\ \forall u \in UInt^{PTQ}, p \in PI_u \tag{G-146}$$

$$QS_{u,s}^L \leq QS_{u,s} \leq QS_{u,s}^U\ \forall u \in UInt^{PTQ}, s \in SO_u \tag{G-147}$$

$$PS_{u,s,p}^L \leq PS_{u,s,p} \leq PS_{u,s,p}^U\ \forall u \in UInt^{PTQ}, s \in SO_u, p \in PO_{u,s} \tag{G-148}$$

$$\sum_{p \in PI_u \cap P^{VF}} PF_{u,p} = 1\ \forall u \in UInt^{PTQ} \tag{G-149}$$

$$\sum_{p \in PO_{s,u} \cap P^{VF}} PS_{u,s,p} = 1\ \ \forall s \in SO_u, u \in UInt^{PTQ} \tag{G-150}$$

$$QF_u = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u}\ \forall u \in UInt^{PTQ} \tag{G-151}$$

$$QF_u PF_{u,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p}\ \forall\, u \in UInt^{PTQ}, p \in PI_u \cap P^V \tag{G-152}$$

$$QF_u PF_{u,p} PF_{u,SPG} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG}\ \forall u \in UInt^{PTQ}, p \in PI_u \backslash P^V \tag{G-153}$$

$$QS_{u,s} = \sum_{u'' \in UO_u} Q_{u,s,u''}\ \ \forall\, u \in UInt^{PTQ}, s \in SO_u \tag{G-154}$$

$$Q_{u,s,u''}^L \leq Q_{u,s,u''} \leq QS_{u,s,u''}^U\ \ \forall u \in UInt^{PTQ}, s \in SO_u, u'' \in UO_u \tag{G-155}$$

The nonconvexities in problem PTQ arise from Eqs. G-121, G-122, G-128, G-129, G-135, G-136, G-141, G-142, G-152, G-153.

Fuel blending (FB) model

The fuel blending (FB) optimisation problem is QCQP, where objective function $\mathbf{OF^{FB}}$ represents the profit to be maximized. The nonconvexities arise from mixing on volume (Eq. G-176) and weight (Eq. G-177) basis to estimate stream properties.

$$f_0^{FB}(x^{FB}) = profit^{FB} = totalrevenue^{FB} - totalcost^{FB} \quad (\mathbf{OF^{FB}})$$

Where x^{FB} is denoting the continuous variables. No binary variables are present in FB optimisation problem.

$$totalrevenue^{FB} = \sum_{pr \in PR^{FB}} \alpha_{pr}. QF_{pr}^{FB} \quad \text{(G-156)}$$

$$totalcost^{FB} = \sum_{e \in Ein_{REF}^{FB}} \lambda_e^{REF,FB}. QF_e^{FB} + \sum_{e \in Ein_{REF}^{FB}} \sum_{p \in PE^e} \lambda_{e,p}^{REF,FB}. PF_{e,p}^{FB} +$$

$$+ \sum_{e \in Ein_{PTQ}^{FB}} \lambda_e^{PTQ,FB}. QF_e^{FB} + \sum_{e \in Ein_{PTQ}^{FB}} \sum_{p \in PE^e} \lambda_{e,p}^{PTQ,FB}. PF_{e,p}^{FB} + \quad \text{(G-157)}$$

$$\sum_{rw \in RW^{FB}} \beta_{rw}. QF_{rw}^{FB} + \sum_{mt \in MT^{FB}} \gamma_{mt}. QT_{mt}^{FB}$$

s. t.

Transport capacity for FB:

$$QT_{mt}^{FB} = \sum_{(upr,pr) \in UPR_{mt}^{FB}} Q_{upr,pr,mt} + \sum_{(rw,urw) \in URW_{mt}^{FB}} Q_{mt,rw,urw} \; \forall \; mt \in MT^{FB} \quad \text{(G-158)}$$

$$QT_{mt}^{L} \leq QT_{mt}^{FB} \leq QT_{mt}^{U} \forall mt \in MT^{FB} \quad \text{(G-159)}$$

Equations for units in FB receiving raw materials (diesel, gasoline). In this type of units $e = s = rw$:

$$QF_{rw}^{FB} = \sum_{u'' \in UO_u} Q_{u,s,u''} \;\; \forall \; rw \in RW^{FB}, u \in U^{FB} \cap U^{RW} \quad \text{(G-160)}$$

$$QF_{rw}^{L} \leq QF_{rw}^{FB} \leq QF_{rw}^{U} \; \forall \; rw \in RW^{FB} \quad \text{(G-161)}$$

$$Q_{u,s,u''}^{L} \leq Q_{u,s,u''}^{FB} \leq Q_{u,s,u''}^{U} \; \forall \; u \in U^{FB} \cap U^{RW}, s \in RW^{FB}, u'' \in UO_u \quad \text{(G-162)}$$

$$PS_{u,s,p} = PF_{rw,p}^{FB} \; \forall \; u \in U^{FB} \cap U^{RW}, rw \in RW^{FB}, p \in PE^{rw} \quad \text{(G-163)}$$

$$PS^{L}_{u,s,p} \leq PS_{u,s,p} \leq PS^{U}_{u,s,p} \; \forall \; u \in U^{FB} \cap U^{RW}, s \in RW^{FB}, p \in PE^{rw} \quad \text{(G-164)}$$

Equations for units in FB receiving linking streams from REF.

$$QF^{FB}_{e} = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall e \in Ein^{FB}_{REF}, u \in UEin^{FB}_{REF} \quad \text{(G-165)}$$

$$QF^{L}_{e} \leq QF^{FB}_{e} \leq QF^{U}_{e} \forall e \in Ein^{FB}_{REF} \quad \text{(G-166)}$$

$$Q^{L}_{u,s,u''} \leq Q_{u,s,u''} \leq QF^{U}_{u,s,u''} \forall u \in UEin^{FB}_{REF}, s \in Ein^{FB}_{REF}, u'' \in UO_u \quad \text{(G-167)}$$

$$PS_{u,s,p} = PF^{FB}_{e,p}, \forall e \in Ein^{FB}_{REF}, p \in PE^{e} \quad \text{(G-168)}$$

$$PS^{L}_{u,s,p} \leq PS_{u,s,p} \leq PS^{U}_{u,s,p} \forall s \in Ein^{FB}_{REF}, p \in PE^{e} \quad \text{(G-169)}$$

Equations for units in FB receiving linking streams from PTQ:

$$QF^{FB}_{e} = \sum_{u'' \in UO_u} Q_{u,s,u''} \; \forall e \in Ein^{FB}_{PTQ}, u \in UEin^{FB}_{PTQ} \quad \text{(G-170)}$$

$$QF^{L}_{e} \leq QF^{FB}_{e} \leq QF^{U}_{e} \forall e \in Ein^{FB}_{PTQ} \quad \text{(G-171)}$$

$$Q^{L}_{u,s,u''} \leq Q_{u,s,u''} \leq QF^{U}_{u,s,u''} \forall u \in UEin^{FB}_{PTQ}, s \in Ein^{FB}_{PTQ}, u'' \in UO_u \quad \text{(G-172)}$$

$$PS_{u,s,p} = PF^{FB}_{e,p}, \forall e \in Ein^{FB}_{PTQ}, p \in PE^{e} \quad \text{(G-173)}$$

$$PS^{L}_{u,s,p} \leq PS_{u,s,p} \leq PS^{U}_{u,s,p} \forall s \in Ein^{FB}_{PTQ}, p \in PE^{e} \quad \text{(G-174)}$$

Equations for units in FB delivering final products to the external market. (In this type of units, e = $s = pr$, $QF_u = QS_{u,s} = QF^{FB}_{pr}$):

$$QF^{FB}_{pr} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} \;\; \forall \; pr \in PR^{FB}, u \in U^{FB} \cap (U^{PR}) \quad \text{(G-175)}$$

$$QF^{FB}_{pr} . PF^{FB}_{pr,p} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} \; \forall p \in PE^{pr} \cap P^{V}, pr \in PR^{FB} \quad \text{(G-176)}$$

$$QF^{FB}_{pr} . PF^{FB}_{pr,p} . PF^{FB}_{pr,SPG} = \sum_{u' \in US_u} \sum_{s \in SO_{u'}} Q_{u',s,u} PS_{u',s,p} PS_{u',s,SPG} \; \forall p \in PE^{pr} \backslash P^{V}, pr \in PR^{FB} \quad \text{(G-177)}$$

$$\sum_{p \in PI_u \cap P^{VF}} PF^{FB}_{pr,p} = 1 \;\; \forall \; pr \in PR^{FB} \quad \text{(G-178)}$$

$$QF^{L}_{pr} \leq QF^{FB}_{pr} \leq QF^{U}_{pr} \;\; \forall pr \in PR^{FB} \quad \text{(G-179)}$$

$$PF_{u,p} = PS_{u,s,p} = PF_{pr,p}^{FB} \quad \forall u \in U^{FB} \cap U^{PR}, pr \in PR^{FB}, p \in PE^{pr} \tag{G-180}$$

$$PF_{pr,p}^{L} \leq PF_{pr,p}^{FB} \leq PF_{pr,p}^{U} \quad \forall pr \in PR^{FB}, p \in PE^{pr} \tag{G-181}$$

The performance of local solvers such as SBB and DICOPT for solving problem the short-term IRPC planning problem was also investigated and their results are shown in the Table G. 1. For the BCS scenario, the local solution (1.367 MMUSD/day) reached for these solvers is 54% worse than the best solution found (2.964 MMUSD/day), which corresponds to the clustering approach with six clusters. Similar figures are reported by SBB and DICOPT for the WRPS and DRS scenarios, with a local optimal about 48% and 64% worse than the best solutions found, which again correspond to six clusters. Interestingly, for the LDS scenario, the local solvers reached an optimal value, representing a loss in profit of 0.152 MMUSD/day. Recall that the best solution reported for this scenario is 2.664 MMUSD/day.

These findings highlight the relevance of an initialization procedure to provide good initial points to the local solvers and on the other hand, the potential benefits in profit that can be achieved applying global optimisation techniques, such as the Lagrangean-based decomposition and the clustering approach.

Overall, local solvers such as SBB and DICOPT are not reliable to solve the large-scale MIQCQP addressed in this work, since they might provide a poor local optimal solution or as in the logistic disruption scenario (LDS) they can reach a local optimal representing a loss in profit. Recall that for the LDS the best solution found is 2.664 MMUSD/day.

Table G. 1. Results from local MINLP solvers SBB and DICOPT compared to the Lagrangean decomposition algorithm and the Clustering approach.

	Base case scenario (BCS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**|Relaxation Gap [%]**	**Runtime [h]**
Lagrangean Decomposition*	2,953	3,181	7.2%	10.03
Clustering Approach**	2,964	3,205	7.5%	5.70
SBB/DICOPT***	1,367	N/A	N/A	0.002
	Without refinery-petrochemical integration scenario (WRPS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap [%]**	**Runtime [h]**
Lagrangean Decomposition*	2,006	2,022	0.8%	0.72
Clustering Approach**	2,009	2,233	10.0%	5.84
SBB/DICOPT***	1,050	N/A	N/A	0.002
	Logistic disruption scenario (LDS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap[%]**	**Runtime [h]**
Lagrangean Decomposition*	2,661	2,814	5.4%	10.08
Clustering Approach**	2,664	3,050	12.7%	3.68
SBB/DICOPT***	-152	N/A	N/A	0.002
	Demand reduction scenario (DRS)			
	LB [kUSD/day]	**UB [kUSD/day]**	**Relaxation Gap[%]**	**Runtime [h]**
Lagrangean Decomposition*	2,804	2,908	3.6%	5.84
Clustering Approach**	2,833	3,090	8.3%	5.80
SBB/DICOPT***	1,008	N/A	N/A	0.002

*Using three subproblems: CM, RB and PTQ

**Using six clusters

*** Optimality gap is applied for SBB and DICOPT

Appendix H. PRELIMINARY RESULTS

NMDT IN THE CLUSTERING APPROACH

The performance for the NMDT relaxation integrated into the clustering decomposition approach is shown in Figure H. 1, Figure H. 2 and Figure H. 3 for WRPS, LDS and DRS scenarios, respectively. By applying NMDT (problem **PR'**), we were able to reach a better lower bound for the BCS and WRPS scenarios, whilst for LDS and DRS scenarios the LB obtained for PCM (problem **PR**) and NMDT are the same. This finding highlights the capability of the algorithm to integrate both approaches. However, in terms of duality gap, the NMDT led to the worst figures for all the scenarios. Thus, there is not strong evidence that applying NMDT we can obtain at least similar results as the PCM. This is a further research topic.

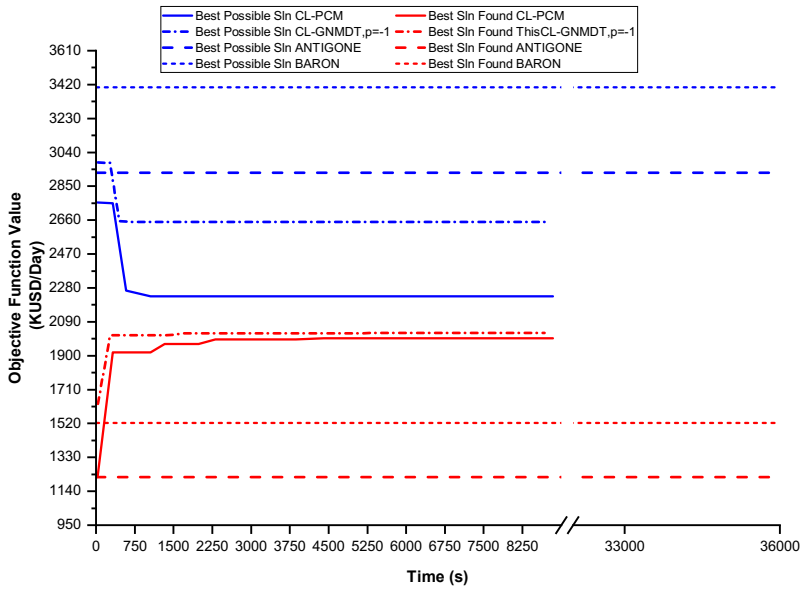


Figure H. 1. Clustering approach performance for NMDT applied to WRPS.

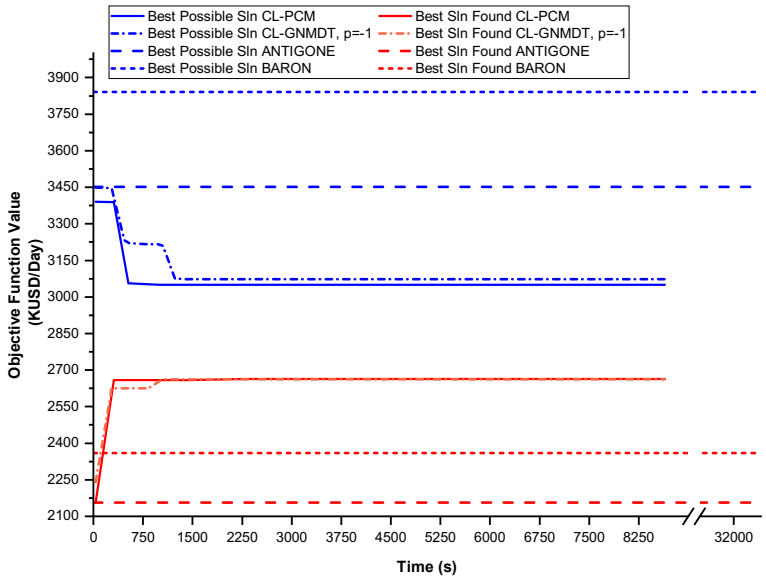


Figure H. 2. Clustering approach performance for NMDT applied to LDS.

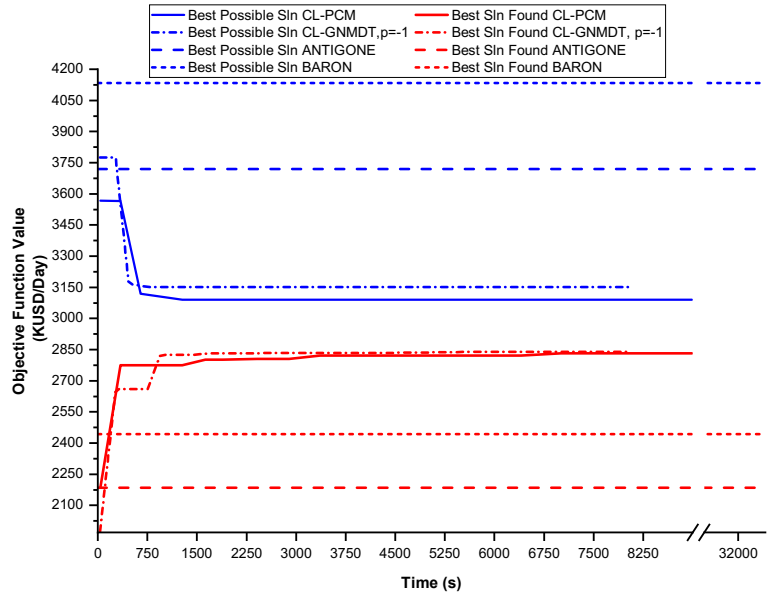


Figure H. 3. Clustering approach performance for NMDT applied to DRS.