UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE AUTOMAÇÃO E SISTEMAS

Leonardo Salsano de Assis

Operational Management of Crude Oil Supply: Models and Solution Strategies

Florianópolis
2020

Leonardo Salsano de Assis

Operational Management of Crude Oil Supply: Models and Solution Strategies

Tese submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas da Universidade Federal de Santa Catarina para a obtenção do título de Doutor em Engenharia de Automação e Sistemas.
Orientador: Prof. Eduardo Camponogara, Dr.

Florianópolis
2020

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

de Assis, Leonardo Salsano
Operational Management of Crude Oil Supply: Models and Solution Strategies / Leonardo Salsano de Assis ; orientador, Eduardo Camponogara, 2020.
148 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, Florianópolis, 2020.

Inclui referências.

1. Engenharia de Automação e Sistemas. 2. MILP, MINLP, Planning, Scheduling. 3. Crude Oil Scheduling, Maritime Inventory Routing, Management of Crude Oil Supply. 4. Blending, Bilinear Terms . 5. Clustering. I. Camponogara, Eduardo . II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Automação e Sistemas. III. Título.

Leonardo Salsano de Assis

Operational Management of Crude Oil Supply: Models and Solution Strategies

O presente trabalho em nível de Doutorado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Leandro Magatão, Dr.
Universidade Tecnológica Federal do Paraná-UTFPR

Prof. Erlon Cristian Finardi, Dr.
Universidade Federal de Santa Catarina-UFSC

Prof. Werner Kraus Junior, Dr.
Universidade Federal de Santa Catarina-UFSC

Laio O. Seman, Dr.
Universidade Federal de Santa Catarina-UFSC

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de Doutor em Engenharia de Automação e Sistemas.

Documento assinado digitalmente
Werner Kraus Junior
Data: 17/07/2020 16:12:48-0300
CPF: 531.085.239-53

Prof. Werner Kraus Junior, Dr.
Coordenador do Programa de
Pós-Graduação

Eduardo Camponogara:51398818020
Digitally signed by Eduardo Camponogara:51398818020
Date: 2020.07.17 15:48:57 -03'00'

Prof. Eduardo Camponogara, Dr.
Orientador

Florianópolis, 2020.

This work is dedicated to my dear parents Bruno and Rossana, my sister Larissa, my nephew Rafael and my wife Jessica.

ACKNOWLEDGEMENTS

This work is the result of five years of learning and effort. During this period, I had the chance to acquire skills, whether taking courses or working with research, that will be extremely important in my future as an independent researcher in academia or in industry. In addition, throughout the doctoral studies I had the chance to meet extraordinary people and the opportunity to work in amazing institutions (UFSC and CMU).

First, I would like to thank my Supervisor and friend Prof. Eduardo Camponogara for his enthusiasm, time, patience, willingness to teach and collaborate with the development of this thesis.

I am also deeply grateful to Prof. Ignacio Grossmann and his research group at Carnegie Mellon University, in which I visited twice (2015 and 2017).

A very important person during these five years was my wife Jessica who was always on my side, supporting me both in good and bad moments. I would like to thank my parents Bruno and Rossana, my sister Jessica and my nephew Rafael for all the support.

I thank my friends Alexandre, Fernando, Luis Fernando, Flávio and Sydney, that also started their doctoral studies with me, for helping reduce the stress throughout this period. I am also grateful to my research colleagues from GOS: Luiz, Eric, Thiago, Eduardo, Lauvir, Marco Aurélio, Caio, Ricardo, Pedro Henrique, Jean, Amanda and Angelo for having supported me when needed.

Finally, I thank the Federal University of Santa Catarina and the Department of Automation and Systems for having accepted and provided me extraordinary facilities and resources. In addition, I am very grateful to CNPq, CAPES and Petrobras for funding my doctoral studies, without which this thesis would not be possible.

“The power of rational thinking is the primary source of freedom in the world.”
(Neil deGrasse Tyson)

RESUMO

Esta tese de doutorado avança o estado da arte do gerenciamento de suprimento de óleo cru propondo modelos de programação matemática e algoritmos que tratam deste problema de forma integrada no nível de decisão operacional. Como resultado obtém-se o Gerenciamento Operacional do Suprimento de Petróleo (OMCOS). OMCOS considera ambos segmentos *upstream* (i.e., plataformas, navios e o terminal de petróleo cru) e *midstream* (i.e., CDUs nas refinarias). Em relação à literatura técnica, OMCOS combina *Maritime Inventory Routing* (MIR) com *Crude Oil Scheduling* (COS), considerando elementos do nível operacional (i.e., escalonamento e misturas de óleo cru) e do nível tático (i.e., controle de inventário e alocação de recursos). Esta integração gera modelos MINLP não convexos que são abordados nesta tese. O capítulo 2 propõe um algoritmo iterativo baseado em decomposição MILP-NLP, que aplica em cada iteração uma estratégia de redução de domínio para lidar com os termos bilineares encontrados no escalonamento de operações com óleo cru (COS). Um modelo MINLP não convexo para o OMCOS que agrega ao problema de suprimento de petróleo elementos do nível operacional encontrados em *Maritime Inventory Routing* e *Crude Oil Scheduling* é proposto no capítulo 3. Além disto, é apresentado um algoritmo baseado em decomposição MILP-NLP que utiliza envelopes de McCormick (para gerar uma relaxação MILP), redução de domínio (para reduzir a complexidade), e um *solver* NLP (para a obtenção de soluções factíveis). O capítulo 4 propõe um modelo de programação inteira mista (MILP) para clusterizar o OMCOS que possui os seguintes benefícios: (a) redução do número de rotas dos navios; (b) simplificação de operações de *offloading* e *unloading*; (c) imposição de regras para a mistura de diferentes tipos de petróleo em *storage tanks* de forma a minimizar a variação das propriedades; e (d) produção de limites em relação às propriedade dos petróleos nos *storage tanks* e *charging tanks* que são usados para linearizar os termos bilineares. Através da combinação de clusters e de uma decomposição MILP-NLP, boas soluções com custo computacional reduzido foram obtidas para um conjunto de instâncias do OMCOS.

Palavras-chave: MILP. MINLP. *Crude Oil Scheduling*. *Maritime Inventory Routing*. Suprimento de Petróleo. Terminal de Petróleo. Misturas. Termos Bilineares. *Clustering*.

RESUMO EXPANDIDO

Introdução

O suprimento de petróleo de plataformas *offshore* até refinarias é um dos principais problemas enfrentados por empresas verticalmente integradas que controlam a produção, transporte, armazenamento e refino. Em campos *offshore* de águas profundas, unidades Flutuantes de Produção, Armazenamento e Descarregamento (FPSOs) produzem e armazenam óleo cru. Este petróleo é transferido para terminais de petróleo *onshore* por frotas de navios. Quando estes navios chegam ao terminal, eles utilizam oleodutos para descarregarem sua carga de petróleo em *storage tanks*. O óleo é então bombeado para *charging tanks* e em seguida enviado para as colunas de destilação das refinarias. (CDUs).

Objetivos

O principal objetivo desta tese é desenvolver modelos de programação matemática para o gerenciamento operacional do suprimento de petróleo e estratégias de solução para lidarem com a complexidade dos modelos obtidos. Além disto, os objetivos específicos são os seguintes: (a) avaliar a contribuição acadêmica e industrial da solução do problema de gerenciamento operacional do suprimento de petróleo de forma integrada; entender e analisar o estado da arte no uso de programação matemática no contexto do gerenciamento operacional do suprimento de petróleo; desenvolver e avaliar modelos que integram o gerenciamento operacional do suprimento de petróleo; propor e avaliar a performance de estratégias de decomposição usadas para a solução do problema; desenvolver estratégias de clusterização e avaliar a sua eficácia na solução do problema de gerenciamento operacional do suprimento de petróleo; identificar limitações dos modelos e estratégias de solução, e apontar direções para pesquisas futuras.

Metodologia

Durante o processo de formulação dos modelos propostos nesta tese, o principal objetivo era representar as operações presentes no suprimento de petróleo o mais próximo possível da realidade. De forma geral, o gerenciamento operacional do suprimento de petróleo e seus subproblemas levam em consideração: (a) plataformas de petróleo e seus parâmetros relacionados a produção e armazenamento de petróleo; (b) navios aliviadores e seus parâmetros relacionados ao armazenamento de petróleo e tempos de viagem; (c) tanques de armazenamento em terminais de petróleo e seus parâmetros relacionados ao armazenamento e transferência de petróleo; (d) colunas de destilação e seus parâmetros relacionados a demanda e qualidade de petróleo. Como esperado, as formulações resultantes são não-lineares e apresentam um elevado número de variáveis e restrições que não podem ser resolvidas com *solvers* tradicionais. Desta forma, esta pesquisa pretende propor modelos e identificar estruturas que possam ser exploradas de forma a decompor o problema e reduzir o esforço computacional. Apesar de não existir um banco de dados de instâncias relativas a este problema, dados disponíveis em trabalhos da literatura foram utilizados para a geração das instâncias propostas nesta tese.

Resultados e Discussão

Esta tese de doutorado avança o estado da arte do gerenciamento de suprimento de óleo cru propondo modelos de programação matemática e algoritmos que tratam deste problema de forma integrada no nível de decisão operacional. Como resultado obtém-se o Gerenciamento Operacional do Suprimento de Petróleo (OMCOS). OMCOS considera ambos segmentos *upstream* (i.e., plataformas, navios e o terminal de petróleo cru) e *midstream* (i.e., CDUs nas refinarias). Em relação à literatura técnica, OMCOS combina *Maritime Inventory Routing* (MIR) com *Crude Oil Scheduling* (COS), considerando elementos do nível operacional (i.e., escalonamento e misturas de óleo cru) e do nível tático (i.e., controle de inventário e alocação de recursos). Esta integração gera modelos MINLP não convexos que são abordados nesta tese. O capítulo 2 propõe um algoritmo iterativo baseado em decomposição MILP-NLP, que aplica em cada iteração uma estratégia de redução de domínio para lidar com os termos bilineares encontrados no escalonamento de operações com óleo cru (COS). Um modelo MINLP não convexo para o OMCOS que agrega ao problema de suprimento de petróleo elementos do nível operacional encontrados em *Maritime Inventory Routing* e *Crude Oil Scheduling* é proposto no capítulo 3. Além disto, é apresentado um algoritmo baseado em decomposição MILP-NLP que utiliza envelopes de McCormick (para gerar uma relaxação MILP), redução de domínio (para reduzir a complexidade), e um *solver* NLP (para a obtenção de soluções factíveis). O capítulo 4 propõe um modelo de programação inteira mista (MILP) para clusterizar o OMCOS que possui os seguintes benefícios: (a) redução do número de rotas dos navios; (b) simplificação de operações de *offloading* e *unloading*; (c) imposição de regras para a mistura de diferentes tipos de petróleo em *storage tanks* de forma a minimizar a variação das propriedades; e (d) produção de limites em relação às propriedade dos petróleos nos *storage tanks* e *charging tanks* que são usados para linearizar os termos bilineares. Através da combinação de clusters e de uma decomposição MILP-NLP, boas soluções com custo computacional reduzido foram obtidas para um conjunto de instâncias do OMCOS.

Considerações Finais
O gerenciamento operacional do suprimento de petróleo consiste em coordenar o fluxo de petróleo de plataformas até refinarias. Com o objetivo de avançar o estado da arte, esta tese propõe modelos de programação matemática e algoritmos para a solução integrada do problema. Os capítulos apresentam uma série de contribuições na área de modelagem, estratégias de decomposição e de clusterização. Além disto, a tese propõe um conjunto de instâncias do problema que poderão ser utilizadas em pesquisas futuras. Os trabalhos futuros vão no sentido de: (a) melhorar as formulações do problema de forma a se obter relaxações *MILP* mais apertadas; (b) implementar estratégias de quebra de simetria; (c) propor novas estratégias de decomposição do problema; (d) aprimorar a estratégia de clusterização para diminuir ou eliminar possíveis soluções infactíveis.

Palavras-chave: MILP. MINLP. *Crude Oil Scheduling*. *Maritime Inventory Routing*. Suprimento de Petróleo. Terminal de Petróleo. Misturas. Termos Bilineares. *Clustering*.

ABSTRACT

This thesis advances the state of the art on the management of crude oil supply by proposing models and algorithms to consider elements of the operational decision level in an integrated fashion, which leads to the Operational Management of Crude Oil Supply (OMCOS). OMCOS comprises both the upstream (i.e., platforms, vessels and terminal) and the midstream (i.e., CDUs at the refinery) segments. In relation to the technical literature, OMCOS combines elements of Maritime Inventory Routing (MIR) with Crude Oil Scheduling (COS) by considering decisions at the operational level (i.e., scheduling and crude oil blending) and tactical level (i.e., inventory control and resource allocation). Such an integration leads to non-convex Mixed Integer Non-Linear Programming (MINLP) models that are addressed in this thesis. Chapter 2 proposes an iterative two-step MILP-NLP decomposition algorithm, which implements a domain-reduction strategy for handling bilinear terms in the scheduling of crude oil operations (COS). A non-convex MINLP model for OMCOS that brings elements of the operational level into the management of crude oil supply, thereby incorporating elements of maritime inventory routing and crude oil scheduling is proposed in Chapter 3. Further, an iterative MILP-NLP decomposition is presented to tackle the MINLP problem that relies on bivariate piecewise McCormick envelopes (to yield an MILP relaxation), domain reduction (to reduce complexity), and an NLP solver (to reach feasible solutions). Chapter 4 proposes an Mixed Integer Linear Programming (MILP) clustering formulation for OMCOS that offers the following benefits: (a) reduces the number of routes for the vessels; (b) simplifies offloading and unloading operations; (c) imposes rules for crude mixtures in clusters of storage tanks that minimize property variations; and (d) produces bounds on crude properties inside storage and charging tanks that are used to linearize the bilinear terms in blending constraints. Through the combination of clusters and an MILP-NLP decomposition, good solutions were obtained for a set of representative instances of OMCOS at a reduced computational cost.

Keywords: MILP. MINLP. Crude Oil Scheduling. Maritime Inventory Routing. Crude Oil Supply. Crude Oil Terminal. Blending. Bilinear Terms. Clustering.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

CCCP	Capacitated Centred Clustering Problem
CCP	Capacited Clustering Problem
CDU	Crude Distillation unit
CDUs	Crude Distillation units
COS	Crude Oil Scheduling
CPU	Central Processing Unit
CT	Charging Tank
CTs	Charging Tanks
DP	Dynamic Programming
FPSO	Floating, Production, Storage and Offloading unit
FPSOs	Floating, Production, Storage and Offloading units
GDP	Generalized Disjunctive Program
GO	Global Optimization
IRP	Inventory Routing Problem
LD	Lagrangean Decomposition
LM	Linearization Strategy
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Non-Linear Programming
MIP	Mixed Integer Programming
MIPGAP	Relative MIP Gap Tolerance
MIR	Maritime Inventory Routing
MOS	Multi-Operation Sequencing
MPC	Model Predictive Control
NLP	Non-Linear Programming
OMCOS	Operational Management of Crude Oil Supply
PCM	Piecewise McCormick based Strategy
RF	Refinery Planning
RHS	Rolling-Horizon Strategy
ST	Storage Tank
STs	Storage Tanks
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem

CONTENTS

1 INTRODUCTION

1.1 THE PETROLEUM INDUSTRY AND THE GLOBAL ENERGY MARKET

According to the *BP Statistical Review of World Energy 2019* (BP, 2019), the global growth of primary energy (i.e., oil, natural gas, coal, renewables, hydro and nuclear) consumption in 2018 was 2.9%, mostly driven by the energy demand increase in China, US and India. The larger shares in energy consumption come from coal (27%), natural gas (23%) and oil (33%). Although renewables only corresponded to a share of 4%, the larger increases in consumption were observed in renewables (14%) and natural gas (5%).

Regarding the oil industry, as illustrated in Fig. 1, in 2018 there was a 1.5% growth in oil consumption, totaling 99.8 million barrels per day (b/d). This corresponds to an increase of 1.4 million barrels per day (b/d) with China (680,000 b/d) and the US (500,000 b/d) being the largest contributors to the growth. On the production side, global oil production corresponded to 94.7 million barrels per day, an increase of 2, 2 million barrels per day in relation to 2017. Although declines on production were observed in Venezuela (-580,000 b/d) and Iran (-310,000 b/d), the US reported an increase of 2, 2 million barrels per day (b/d), which corresponded to the total net increase.

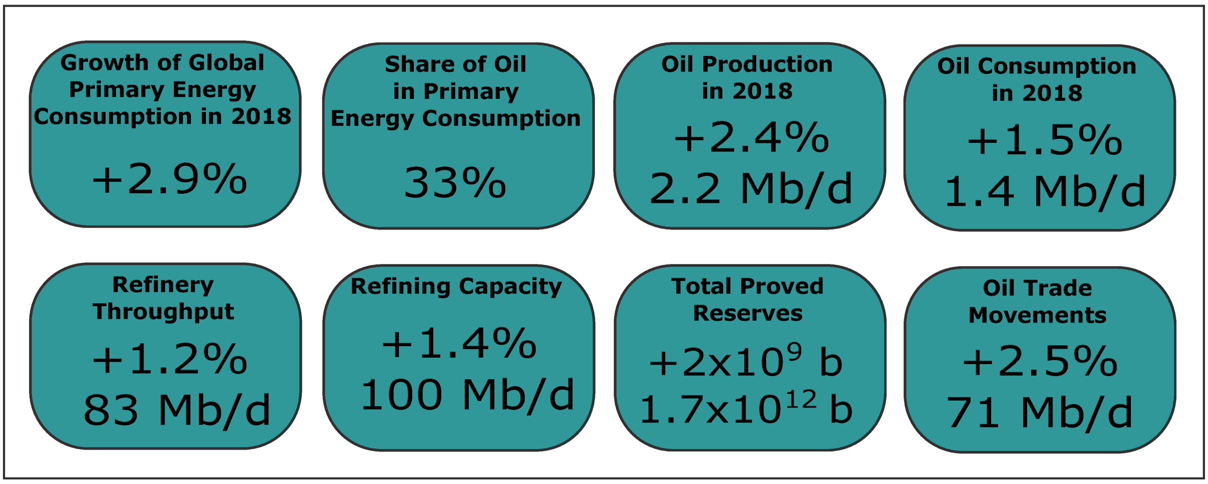


Figure 1 – Statistics for the global primary energy market in 2018.

In 2018, the refinery throughput rose by 960 000 b/d, an increase of 1.2%. Meanwhile, world's refining capacity jumped to 100 Mb/d, which represented an expansion of 1.4%.

The increases on production, consumption and refining capacity were also followed by an increase in global trade. In 2018, 71 million barrels were moved daily around the world as imports and exports. With proven reserves of 1.7 10^{12} barrels, an increase of almost 2 billion barrels in relation to the year of 2017, and a share of 33% of the total share of global primary energy consumption, oil is still a major source of energy in the world.

The numbers show that although renewables had the largest increase on their share in global energy consumption, oil still plays a relevant role in the global energy market. Therefore, there is still the need of developing decision supporting tools for the oil industry to assist decision making in problems related with: (a) oil exploration and production, (b) oil transportation, storage and supply to refineries, (c) refining, and (d) supply to final costumers. The next section introduces the basics elements and problems of the Petroleum Supply Chain, which is the main focus of this thesis.

1.2 PETROLEUM SUPPLY CHAIN

The petroleum industry is faced with a complex and economically relevant supply chain, which is composed by several sub-problems. The complexity is assured by the fact that the oil industry has a global marketplace, with oil production, refining and transportation spread all over the world.

As defined in Simchi-Levi et al. (2008), supply chain management consists of planning, managing, coordinating and integrating entities and activities of the chain in order to satisfy costumer demands, both in terms of quantity and time, while minimizing the overall cost. When it comes to oil companies, Sahebi et al. (2014) point out that in today's business world, these companies should take into account supply chain management concepts and tools in order to stay productive and competitive. In other words, it is highly important for enterprises to adopt supply chain management practices and decision supporting tools based on mathematical optimization to decrease costs and achieve operational efficiency, while maintaining quality.

Activities from the petroleum supply chain such as: (a) onshore and offshore oil exploration and production, (b) crude oil transportation and supply, (c) crude oil storage and refining, as well as (d) product storage and distribution to final costumers, involve a set of different operations with high revenues and costs (LIMA et al., 2016). The management of such activities is highly complex and it increases depending on the number of entities (see Table 1), decision levels (see Table 2), materials, and information involved, as well as the level of integration between them (BARBOSA-PÓVOA, 2014; BARBOSA-PÓVOA; PINTO, José Mauricio, 2020). To manage these activities in the best way, there is a need for developing decision making supporting tools based on mathematical optimization with the goal of providing an integrated and adaptive supply chain, as well as assessing risk and uncertainties (CAPGEMINI, 2008; MARQUES et al., 2020).

According to Sahebi et al. (2014), the petroleum supply chain can be divided into upstream, midstream and downstream segments. Decisions regarding oil field infrastructure, production and transportation of crude oil to the refineries belong to the upstream segment. Wellhead, well platform, production platforms, transportation vessels and crude oil terminals are the main entities of this segment. The midstream portion is concerned with the conversion of the crude oil into refined products at refineries and petrochemical plants. Finally, the downstream segment includes storage, primary and secondary distribution, and wholesale & retail market of refined products. Table 1 summarizes the entities of each segment of the petroleum supply chain.

Upstream				**Midstream**		**Downstream**	
Well Head (WH)	Well Platform (WP)	Production Platform (PP)	Crude Oil Terminal (CT)	Refinery (RF)	Petrochemical Plant (PC)	Distribution Center/Depot (DC)	Market/Customer (M/C)

Table 1 – List of entities of each segment of the petroleum supply chain.

As highlighted by Lazaros G. Papageorgiou (2009), the four main activities of the supply chain management are: supply chain design (strategic); supply chain planning (tactical) and scheduling (operational); and supply chain control (real-time management). All strategic, tactical and operational decision levels are present in the segments of the petroleum supply chain (BARBOSA-PÓVOA, 2014; GHIANI et al., 2004).

The strategic planning is at the highest level, where investment decisions are made considering the long-term (years). These decisions deal with network design problems such as: platform, refinery and warehouse location and capacity; fleet sizing; pipeline network design. Tactical planning decisions are taken for the medium-term (months) and are limited by the strategic planning. This level deals with the flow of materials across the chain: production and distribution planning, inventory management, and inventory allocation (SAHEBI et al., 2014; BARBOSA-PÓVOA, 2014). At the lowest level, operational decisions are taken in the short-term (days or weeks) in order to guarantee the tactical planning. These decisions define the day to day activities of the supply chain and include vessel scheduling (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016), scheduling of crude oil operations (ZIMBERG et al., 2015), refinery scheduling (PINTO, J. et al., 2000), pipeline scheduling (REJOWSKI; PINTO, J., 2004) and downstream vehicle routing (LIMA et al., 2016). Table 2 summarizes the main activities of each supply chain decision level.

Strategic Level	**Tactical Level**	**Operational Level**
Investment (project selection) Facility location (capacity determination) Facility relocation (capacity expansion) Facility allocation Technology selection, upgrading, downgrading Outsourcing	Project planning Oilfield and refinery production planning Inventory management Inventory allocation Distribution planning	Scheduling activities Routing activities

Table 2 – Main activities of each supply chain decision level.

1.3 MANAGEMENT OF CRUDE OIL SUPPLY

An important problem faced by oil companies is the supply of crude oil from offshore platforms to refineries, namely the crude oil supply planning problem (ROCHA et al., 2009). This problem is usually found in vertically integrated oil companies, which control production, transportation, storage and refining. Figure 2 illustrates the main aspects of the problem.

Since oil pipelines are not available in deep-water offshore oilfields, the oil company relies on Floating, Production, Storage and Offloading units (FPSOs), or simply platforms, to produce and store crude oil (CAMPONOGARA; PLUCENIO, 2014). From time to time shuttle tankers travel to FPSOs to collected the crude oil being stored by them. For a large number of platforms, a fleet of shuttle tankers is needed due to the high volume of oil that must be transferred from the platforms to crude oil terminals. It is important to highlight that the vessels' trips to offload FPSOs need to be such that there is always enough storage capacity to allow production at full capacity. For platforms closer to the coast, the production is transferred to onshore crude oil terminals by sub-sea pipelines (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016).

After arriving at the terminal, shuttle tankers unload crude oil through a pipeline to the tank farm, which can be composed by several Storage Tanks (STs) (ZIMBERG et al., 2015). Crude oil can be pumped between storage tanks and to the main pipeline that connects the crude oil terminal to the refinery. At the refinery, the crude oil arriving from the pipeline is stored in Charging Tanks (CTs) and subsequently sent to the Crude Distillation units (CDUs). The set of CDUs will produce oil products such as asphalt, diesel, gasoline, and fuel gas, among others, which will be delivered to chemical, pharmaceutical and energy industries, and end consumers.

Rocha et al. (2009) presented the decisions made in each decision level of the crude oil supply planning problem. The *strategic level* is responsible for defining the demands (i.e., total volume and type of crude oil) of the refinery for the long-term, as well as crude oil import/export decisions. The *tactical level* includes more detailed constraints and is concerned with the medium-term resource allocation. This level decides which platforms will feed each crude oil terminal; which terminal will supply each refinery; and the vessel fleet composition. Also, decisions involving material flow such as the shipments of crude oil between platforms and the crude oil terminal, as well as between the crude oil terminal and the refinery are carried out for each period of the planning horizon. The *operational level* is fed with upper level decisions and is concerned with routing and scheduling of operations. Maritime Inventory Routing (MIR) (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016) and Crude Oil Scheduling (COS) (MOURET et al., 2009) are the main sub-problems of this decision level.

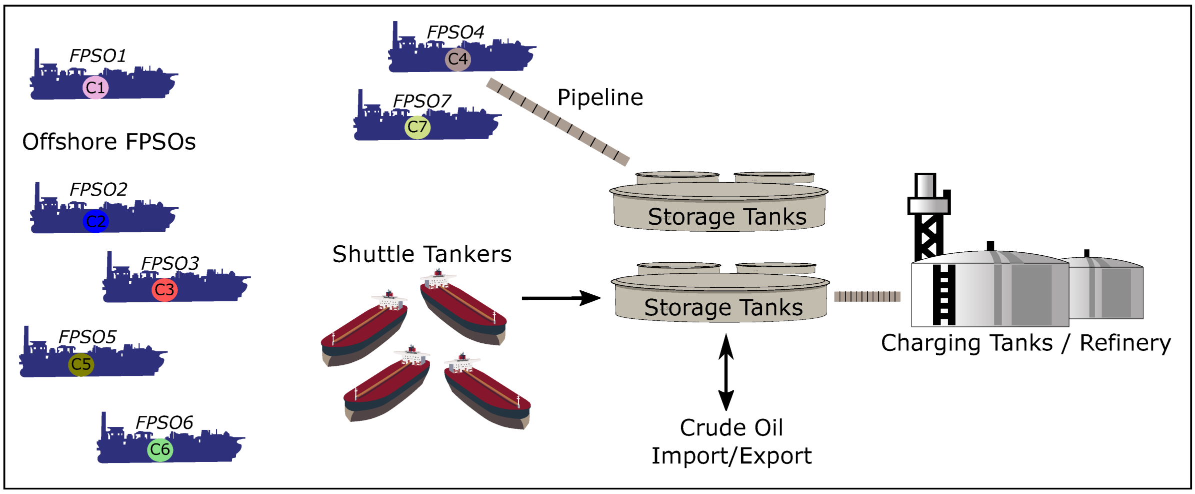


Figure 2 – The crude oil supply chain. Adapted from (ROCHA et al., 2009).

1.4 THESIS MOTIVATION, GOALS AND CHALLENGES

To the best of our knowledge, Aires et al. (2004), and then Rocha et al. (2009) were the first ones to address the problem of managing crude oil supply (see Section 1.3) in an integrated fashion (i.e., from FPSOs to CDUs). They proposed a Mixed Integer Linear Programming (MILP) formulation to allocate the crude oil produced by platforms to onshore terminals and subsequently to refineries in order to satisfy their demands (i.e., both in terms total volume and quality of crude oil). In addition, their formulation considered crude oil import, inventory control over the planning horizon and vessel fleet sizing decisions. Dealing with strategic/tactical level decisions, the limited number of vessels, scheduling of vessels, scheduling of operations in terminals and non-linearities due to blending were not addressed.

Extending the work on the management of crude oil supply, this research has the aim of proposing models and algorithms for tackling this problem considering elements of the operational decision level in an integrated fashion, namely the *Operational Management of Crude Oil Supply (OMCOS)*. According to the classification provided by Sahebi et al. (2014) (see Table 1), this problem comprises both upstream (i.e., production platforms, oil vessels and the crude oil terminal) and midstream (i.e., CDUs at the refinery) segments. Also, this research is concerned with decisions associated to the operational (i.e., scheduling and blending of crudes as seen in MIR and COS problems) and tactical (i.e., inventory control and resource allocation) levels, which can lead into non-convex Mixed Integer Non-Linear Programming (MINLP) models.

A vast number of works in the literature have approached strategic and/or tactical supply chain with mathematical programming models considering the integration of segments. However, the operational level is often found in the literature divided into subproblems: maritime inventory routing (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016), scheduling of crude oil operations (ZIMBERG et al., 2015), refinery scheduling

(PINTO, J. et al., 2000), pipeline scheduling (REJOWSKI; PINTO, J., 2004), and vehicle routing (LIMA et al., 2016). This can be explained by the fact that these problems may involve dozens of thousands of binary and continuous variables, linear and non-linear constraints, which required specialized solution techniques.

In that sense, one might ask if it is possible or even worth to develop mathematical programming models and solution strategies, which take into account elements of both tactical and operational decision levels, for integrating the crude oil supply at the operational level. This research is then justified by its potential to advance science and engineering, by developing new models and optimization strategies for the OMCOS. Further, Barbosa-Póvoa (2014), Barbosa-Póvoa and José Mauricio Pinto (2020) and Lazaros G. Papageorgiou (2009) pointed out the importance of integrating, if possible, tactical and operational level decisions, which can enhance supply chain performance.

In the process of formulating the models presented in this thesis, the main goal is to represent the operations as closely as possible in order to produce realistic models that can assist engineers plan crude oil supply. In general, the OMCOS and its sub-problems take into account:

- **FPSOs**. These resources are located in offshore oilfields producing different types of crude oil. The resources have a daily production rate, limited storage capacity and flow rate when offloading their volume into shuttle tankers. Also, each Floating, Production, Storage and Offloading unit (FPSO) produces only one type of crude which is characterized by certain properties (i.e., sulfur content, etc).

- **Shuttle Tankers or Vessels**. Similarly to FPSOs, vessels also have a storage capacity and flow rate limit when unloading crude oil into storage tanks at the terminal. Moreover, constraints are used for coordinating offloading operations in FPSOs and unloading of crudes into STs.

- **STs at the Crude Oil Terminal**. The storage tanks have the function of receiving crude oil from vessels and feeding charging tanks. Since mixtures of crudes are allowed inside a Storage Tank (ST), the individual inventory of each crude must be tracked. Also, when feeding a Charging Tank (CT), the proportion among crudes inside the storage tank must be equal to the proportion of crudes leaving the storage tank. Non-convex constraints are needed to enforce this blending rule. Depending on the nature of the problem, these constraints can lead to Non-Linear Programming (NLP) or MINLP formulations. Finally, other rules can also be considered for a storage tank: a vessels cannot feed more than one ST at a time; a ST cannot feed more than one CT at time; inlet and outlet operations in a ST cannot occur in the same time period; and maximum capacities and flow rate limits must be respected.

- **CTs at the Crude Oil Terminal**. Multiple CTs located near the refinery can receive crude oil from STs to feed the CDUs. Similar rules to the ones defined for STs can be applied to control the flow of crudes to and from CTs. Lower and upper bounds on the property specification of the crude blends are defined for each CTs. This means that the STs need to feed a CT such that the resulting crude blend inside the CT is within the specification. Consequently, the individual inventory of each crude must be tracked, which leads to non-convex constraints to guarantee the proportion of crudes in a feed operation to a Crude Distillation unit (CDU).

- **CDUs**. CDUs are the end point of OMCOS. Each CDU is connected to a subset of CTs and cannot be fed by more than one CT at a time. Further, a CDU has a total demand that needs to be satisfied both in terms of total amount and crude specification. Finally, each CDU must receive a minimum volume of crude each period of time.

As expected, the resulting Mixed Integer Programming (MIP) formulations will be composed by an expressive number of variables and constraints, which possibly will not be solved by off-the-shelf solvers such as SCIP (ACHTERBERG, 2009), BARON (SAHINIDIS, 2014), CPLEX (IBM, 2013), GUROBI (GUROBI OPTIMIZATION, 2016), and CONOPT (DRUD, 1985), among others. As highlighted by Floudas and Lin (2004) and Castro et al. (2018), scheduling problems with discrete decisions have a combinatorial nature, which when combined with non-linear constraints become challenging from the computational point of view. Therefore, this research intends to propose models and identify problem structure that can be exploited in order to decompose the problem and decrease the computation burden.

1.5 THESIS OBJECTIVES

The main objective of this thesis is to develop mathematical programming models for the operational management of crude oil supply and solution strategies to tackle the complexity of the resulting models. Other than that, the specific goals are the following:

- Assess the scientific and industrial value of tackling the operational management of crude oil supply in an integrated way.

- Understand and analyze the state of the art on the use of mathematical programming in the context of the operational management of crude oil supply.

- Develop and evaluate models that integrate the operational management of crude oil supply.

- Propose and assess the performance of decomposition schemes to tackle the problem of concern.
- Develop clustering strategies and evaluate their effectiveness in solving the operational management of crude oil supply.
- Identify limitations of the proposed models and solution strategies, and point out directions for future research.

1.6 CONTRIBUTIONS OF THE THESIS

The main contributions of this thesis are the following:

- **Chapter 2**. An iterative two-step MILP-NLP decomposition algorithm, which implements a domain-reduction strategy for handling bilinear terms in the scheduling of crude oil operations (COS).
- **Chapter 3**. A non-convex MINLP model for OMCOS that brings elements of the operational level into the management of crude oil supply, thereby incorporating elements of maritime inventory routing and crude oil scheduling. Further, an iterative MILP-NLP decomposition is presented to tackle the MINLP problem that relies on bivariate piecewise McCormick envelopes (to yield an MILP relaxation), domain reduction (to reduce complexity), and an NLP solver (to reach feasible solutions).
- **Chapter 4**. A Mixed Integer Linear Programming (MILP) clustering formulation for OMCOS that offers the following benefits: (a) reduces the number of routes for the vessels; (b) simplifies offloading and unloading operations; (c) imposes rules for crude mixtures in clusters of storage tanks that minimize property variations; and (d) produces bounds on crude properties inside storage and charging tanks that are used to linearize the bilinear terms in blending constraints. Through the combination of clusters and an MILP-NLP decomposition, good solutions were obtained for a set of representative instances of OMCOS at a reduced computational cost.

The main publications obtained during the development of this thesis are the following.

1. de Assis, L. S., Camponogara, E. A MILP model for planning the trips of dynamic positioned tankers with variable travel time. *Transportation Research Part E: Logistics and Transportation Review*, v. 93, p. 372-388, 2016.

2. Assis, L.S., Camponogara, E., Zimberg, B., Ferreira, F., Grossmann, I. E. A piecewise McCormick relaxation-based strategy for scheduling operations in a crude oil terminal. *Computers & Chemical Engineering*, v. 106, p. 309-321, 2017.

3. Camponogara, E., Guardini, L. A., Assis, L.S. Scheduling pumpoff operations in onshore oilfields with electric-power constraints and variable cycle time. *Computers & Operations Research*, v. 91, p. 247-257, 2018.

4. Assis, L.S., Camponogara, E., Menezes, B. C., Grossmann, I. E. A MINLP formulation for integrating the operational management of crude oil supply. *Computers & Chemical Engineering*, v. 123, p. 110-125, 2019.

5. Assis, L.S., Camponogara, E., Grossmann, I. E. A MILP-based clustering strategy for integrating the operational management of crude oil supply. *Submited to* Computers & Chemical Engineering, 2020.

1.7 THESIS ORGANIZATION

This thesis consists of a collection of technical papers published (or submitted for publication) during the development of the doctoral studies, namely: (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016), (ASSIS, Leonardo Salsano de et al., 2017), (ASSIS, Leonardo S. et al., 2019) and the paper submitted concerning Chapter 4.

Chapter 2 is concerned with the scheduling of crude oil operations in terminal. The discrete time MINLP formulation was first introduced by Zimberg et al. (2015) and applied to the crude oil terminal of the national refinery of Uruguay. The development of Chapter 2 is more focused on presenting the iterative two-step MILP-NLP decomposition algorithm to tackle the problem instances. The solution strategy is based on applying McCormick envelopes to relax the bilinear terms of the blending constraints, leading to a MILP relaxation of the original MINLP formulation. The relaxation is then solved and binary variables are fixed into the MINLP, yielding in an NLP problem. The iterative process is conducted by taking into account in each iteration a domain-reduction strategy, which makes the following iteration easier to be solved.

Although COS formulations also deal with the supply of crude oil to the CDUs, consider vessels arriving at a crude oil terminal and the unloading of crudes into STs, the arrival of vessels and the STs to unload their shipments are known usually in advance. Therefore, rises the need to complete the offshore portion of OMCOS by introducing the elements of MIR into the formulation. These elements were based on the main ideas proposed in the first paper published at the begging of the doctoral studies (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016).

Chapter 3 proposes an integrated non-convex MINLP model for OMCOS that takes into account elements of both MIR and COS. Also, a new iterative two-step

MILP-NLP solution strategy for tackling instances of the problem is proposed. This is necessary since off-the-shelf solvers are not able to tackle these instances. To the best of our knowledge, there does not exist an open data base of instances concerning the problem described in Chapter 3. Nevertheless, available data to generate adequate instances can be found in the works of Sérgio M.S. Neiro and José M. Pinto (2004), Rocha et al. (2009), Cerdá et al. (2015), Mouret et al. (2009), Zimberg et al. (2015), Jin-Hwa Song and Furman (2013), Nakano et al. (2009), Fraga et al. (2009), and Oliveira et al. (2016).

The final technical part of this thesis (Chapter 4) proposes a MILP clustering formulation to recommend how crudes should be mixed in STs located at the crude oil terminal. The model solution proposes which groups of FPSOs should feed which group of STs. This is done with the goal of minimizing the differences on chemical properties of the crudes assigned to the same cluster of STs. The cluster recommendation is then applied to OMCOS problem instances and solved using a MILP-NLP solution strategy. Nevertheless, the MILP is not a relaxation of the original MINLP as in Chapters 2 and 3. Now, the blending constraints are linearized based on bounds on crude properties obtained after solving the clustering problem. Results show that the use of clusters produce good quality solutions while, at the same time, decreasing the Central Processing Unit (CPU) time dramatically.

Finally, Chapter 5 provides the final remarks of the thesis and recommends topics for future research.

2 A PIECEWISE MCCORMICK RELAXATION-BASED STRATEGY FOR SCHEDULING OPERATIONS IN A CRUDE OIL TERMINAL

2.1 INTRODUCTION

As mentioned in the introduction, the petroleum supply chain can be divided into three segments: upstream, midstream and downstream. This work studies the scheduling of operations in a crude oil terminal within the midstream segment. The first challenge consists in deciding how the crude oil that arrives in vessels should be uploaded to the storage tanks. At the same time, the operations engineer must decide which storage tanks will feed the pipeline connected to the refinery in order to satisfy its demand. This work concerns the crude oil terminal of the national refinery of Uruguay. To schedule terminal operations, this work proposes an iterative two-step MILP-NLP algorithm based on piecewise McCormick relaxation and a domain-reduction strategy for handling bilinear terms. For small instances for which an optimal solution is known, the proposed strategy consistently finds optimal or near-optimal solutions. It also solves larger instances which are, in some cases, intractable by a global optimization solver.

The supply chain of the petroleum industry is arguably one of the most complex and economically relevant of today's society. According to Sahebi et al. (2014), the oil supply chain can be divided into upstream, midstream and downstream segments. Functions such as petroleum exploration, production (SILVA; CAMPONOGARA, 2014) and transportation (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016) of crude oil to the refineries belong to the upstream segment. Major components of the infrastructure are production platforms, transportation vessels and crude oil terminals. The midstream portion is concerned with the reception of commercial crude oil grades in terminals and conversion of the petroleum into refined products at refineries and petrochemical plants. Finally, the downstream segment includes storage, primary and secondary distribution, and wholesale and retail market of refined products.

One of the main challenges of the midstream segment concerns the schedule of operations at crude oil terminals. In general, the problem can be described as follows. After reaching the mooring buoy, a vessel unloads crude oil through a pipeline to the tank farm, which is composed by storage tanks. Crude oil can be pumped between tanks and to the main pipeline that connects the crude oil terminal to the refinery. Although mixtures of crudes with similar properties (e.g., specific gravity and sulfur concentration) are allowed in the storage tanks, they are not recommended in order to provide more flexibility to satisfy the demands of the refinery.

The main decisions of the scheduling problem are: a) determine the volume and quality of crude oil to be transferred from a vessel to each storage tank; and b) the volume and blend of crudes to be sent to the main pipeline in order to satisfy the refinery demands.

Blending equations are one of the most common constraints that appear in crude oil scheduling. These equations involve bilinear terms, which are non-convex functions, thereby, potentially giving rise to multiple local solutions. Algorithms can make use of the fact that, in a MINLP minimization problem, an MILP relaxation provides a lower bound on the original problem, while any feasible solution provides an upper bound (CASTRO, 2015). If these bounds are within a given tolerance, the global solution is achieved.

Standard McCormick envelopes (MCCORMICK, 1976) provide the tightest possible linear relaxation for bilinear terms. In this approach, the bilinear term $x_i x_j$ is replaced with a new continuous variable w_{ij} and four sets of linear constraints are added to the formulation. In order to strengthen the relaxation, one can partition the domain of one variable (x_j) involved in the bilinear terms into n disjoint regions. Then McCormick envelopes are constructed in each disjoint region and new binary variables are added to the formulation to select the best partition of x_j. This approach, known as piecewise McCormick (with univariate partitioning), was first proposed in the work of Bergamini et al. (2005), presenting uniform partitions and a linear increase of the binary variables with the number of partitions. Gounaris et al. (2009) present a comprehensive study on piecewise under- and over estimators for bilinear terms.

To our knowledge, Wicaksono and IA Karimi (2008) and Hasan and I.A. Karimi (2010) were the first to apply bivariate partitioning, which means partitioning the domain of both variables x_i and x_j. In the latter work, bivariate partitioning is applied to a benchmark process network synthesis problem, obtaining stronger relaxations than univariate partitioning. The former one achieved better relaxation using bivariate partitioning than univariate in moderate-size problems such as column sequencing for nonsharp distillation, integrated water use and treatment systems, generalized pooling problems on wastewater treatment networks, and synthesis of heat exchanger networks.

The technical paper (ASSIS, Leonardo Salsano de et al., 2017) is used as basis for the development of this chapter. The remainder of this chapter is organized as follows. A review of the literature on crude oil scheduling is presented in Section 2. The problem statement is given in Section 3. Section 4 introduces the proposed mathematical model. The solution algorithm based on piecewise McCormick envelopes is discussed in Section 5. The computational results and analysis are reported in Section 6. Finally, Section 7 presents conclusions and suggests directions for future research.

2.2 LITERATURE REVIEW

Scheduling problems are challenging optimization problems, both in terms of modeling and algorithmic solutions (MOURET et al., 2009). Of concern in this work, crude oil scheduling is a crucial part of the petroleum supply chain. To satisfy the demand for crudes in refineries, optimal or near-optimal decisions regarding the scheduling of crude oil operations can represent significant economic gains for companies. However, the problem of scheduling crude oil operations leads to a large non-convex mixed-integer non-linear programming (MINLP) model (CERDÁ et al., 2015), which is hard to solve with commercial solvers, usually requiring tailored algorithms. Thus, this problem has received significant attention in the literature.

2.2.1 Discrete and Continuous Time Models

Discrete- and continuous-time models are the two major approaches for modeling crude oil scheduling problems. Discrete-time models (SHAH, 1996; LEE, H. et al., 1996; REDDY, P. C. P. et al., 2004; HAMISU et al., 2013; CHEN, X. et al., 2014) are based on fixed duration of time intervals. The main advantage is the simplified modeling of material balance and flow constraints. The drawback is the large number of time intervals to correctly represent the problem, resulting sometimes in intractable problems. On the other hand, in continuous-time models (REDDY, P. P. et al., 2004; KARUPPIAH et al., 2008; MOURET et al., 2009; CASTRO; GROSSMANN, 2014; CERDÁ et al., 2015), the duration of time intervals is treated as continuous variables in the optimization model. Major advantages lie on the smaller size of the problem, the complete use of the time domain (KARUPPIAH et al., 2008), and usually tighter formulations when compared to discrete ones. However, for this time representation it is more difficult to keep track of material balances. In addition, it is not obvious how to define a-priori the number of time events that are needed. Floudas and Lin (2004), Xuan Chen et al. (2012) and Mouret et al. (2011a) present formulations and comparisons between continuous- and discrete-time models.

2.2.2 Physical Arrangements

Two types of topology for physically describing the system appear in the literature. The first one considers two sets of tanks: storage and charging tanks. In that case, storage tanks receive crude oil from vessels and charging tanks receive crudes from several storage tanks to produce the mixture demanded by the refinery. Finally, each distillation unit (CDU) is fed by only one charging tank at a time. This approach is used in the works of Heeman Lee et al. (1996), Mouret et al. (2009), Karuppiah et al. (2008), Hamisu et al. (2013), Castro and Grossmann (2014) and Xuan Chen et al. (2014). When no charging tanks are used (REDDY, P. C. P. et al., 2004, 2004; LI et al., 2012; CERDÁ et al., 2015), multiples storage tanks can feed a particular CDU at the same time. A different approach is presented by Zimberg et al. (2015), which considers the operations in a crude oil terminal, excluding charging tanks and CDUs.

2.2.3 Solution Approaches

Several solutions approaches have been studied for crude oil scheduling problems. The work of Shah (1996) is one of the first to use mathematical programming for solving the problem of scheduling the crude oil supply to a refinery. To tackle the high dimensionality of the model, the problem is decomposed into smaller ones (i.e., the downstream and upstream problems), which are solved sequentially. The latter (downstream) defines how the refinery operates and how it will be supplied by the pipeline, while the former (upstream) determines how the crude oil tanks feed the pipeline.

Heeman Lee et al. (1996) propose a discrete-time mixed-integer linear program for short-term crude oil scheduling. Bilinear terms in mixing equations are avoided by a linear approximation, which replaces the non-linear terms with individual component flows. P. Chandra Prakash Reddy et al. (2004) identify the periods of the planning horizon over which the composition in each tank does not change (before receiving crudes from a vessel or another tank), which results in an MILP without composition discrepancy. For the remaining periods, mixing constraints are dropped. This strategy proved to be attractive by producing near-optimal solutions in reasonable time. An extension of this strategy to deal with a continuous-time model is presented by P.Chandra Prakash Reddy et al. (2004). A discrete-time MILP is proposed by Zimberg et al. (2015), where discrete values of crude proportion are chosen from a discrete set by the optimization solver, transforming the non-linear mixing equations into linear terms.

The work of Karuppiah et al. (2008) present an outer-approximation algorithm for solving a continuous-time non-convex MINLP. First, the original MINLP is relaxed using McCormick envelopes, which results into an MILP capable of producing a lower bound on the original problem. The solution of this relaxation is used to obtain an upper bound for the MINLP. At each iteration, cutting planes derived from Lagrangean decomposition are added to the MILP. The process continues until the difference between the lower and upper bounds is within a given tolerance.

Mouret et al. (2009) propose a new continuous-time formulation for crude oil scheduling, denoted as single operation sequencing. For this approach, the solution schedule is represented as a single sequence of operations, which reduces the number of time slots required. Symmetry-breaking constraints are added to the model in order to avoid searching multiple equivalent solutions. In addition, a simple two-step MILP-NLP procedure is used for solving the original MINLP.

A different approach is proposed by Yüzgeç et al. (2010), which make use of Model Predictive Control (MPC) to tackle the scheduling of crude oil operations. One of the main advantages is that in case of disturbances or demand changes at any time over scheduling horizon, this framework can quickly updates the decisions.

Li et al. (2012) develop a robust continuous-time MINLP formulation under demand uncertainty. The authors propose a branch-and-bound global optimization algorithm to solve the deterministic robust counterpart optimization model.

A resource-task network is used in the work of Castro and Grossmann (2014) for modeling a continuous-time crude oil scheduling problem. The solution strategy is based on a two-step MILP-NLP algorithm, whereby the MILP relaxation is obtained via multiparametric disaggregation.

Cerdá et al. (2015) present a continuous-time MINLP model based on global-precedence sequencing variables for loading and unloading operations in tanks, and synchronized time slots for modeling the sequence of feedstock for each CDU. A two-step MILP-NLP, which reduces non-linear constraints with bounding constraints and valid cuts, is proposed as the solution algorithm. For handling large instances, a rolling-horizon scheme for continuous-time is presented.

Making use of clusters, Cerdá et al. (2018) proposed a sequential approach, which consists of defining (a) the clusters of charging tanks and their assignment to feed CDUs and (b) the solution of the resulting MINLP subproblems, one for each cluster-CDU pair.

Sergio Mauro da Silva Neiro et al. (2019) propose a continuous time MINLP model based on Multi-Operation Sequencing (MOS). The formulation, which focus on the operational features of a real-world existing refinery, presents constraints to tackle multiple tank outputs. Two algorithms to solve the problem were tested: a two-step MILP-NLP decomposition scheme and a Rolling-Horizon Strategy (RHS).

More recently, Yang et al. (2020) proposed an MINLP formulation for integrating short-term Crude Oil Scheduling with mid-term Refinery Planning (RF). The problem is solved by a Lagrangean Decomposition (LD) algorithm based on the fact that these problems are physically linked by CDUs and the use of their economic net values as their objectives.

2.2.4 Work Contribution

The key contribution of this work is an iterative two-step MILP-NLP decomposition algorithm, which implements a domain-reduction strategy for handling bilinear terms in the scheduling of crude oil operations. An MILP relaxation is obtained for the underlying MINLP problem by replacing the bilinear terms with piecewise McCormick envelopes. The relaxation is sufficiently general to be applied in domain partitions of one variable, which is commonly found in the literature, and of both variables of the bilinear term. The advantage of partitioning both variables is that it usually produces tighter relaxations, yielding stronger lower bounds. The solution of the MILP produces a lower bound and an initial solution for a NLP algorithm, which in turn, yields an upper bound when a feasible solution is found. Because the number of partitions tends to be large for an accurate modeling of the bilinear terms, a domain-reduction strategy is applied iteratively until convergence is achieved. The combination of the relaxation scheme and the domain-reduction strategy proved to be effective, generating better results than a global optimization solver and the strategy proposed in Zimberg et al. (2015).

Further, the model presented in this work advances the one of Zimberg et al. (2015) by including constraints that approximate more closely the mathematical model to the real operation:

- rules out inlet and outlet operations from taking place simultaneously, during the same period.
- ensures that outlet operations wait at least one period of time after uploading crude oil in a tank, in order to allow different types of crude to be fully mixed in a tank.

2.3 PROBLEM STATEMENT

This work is aimed at optimizing the operations at the crude oil terminal of ANCAP, the national refinery of Uruguay, which was initially addressed by Zimberg et al. (2015). This terminal, as illustrated in Fig. 3, consists of eight main tanks and one auxiliary tank that receive crude oil cargoes from marine vessels. The following operations are allowed: crude oil unloading from vessels to storage tanks and transfers from storage tanks to the pipeline that connects the terminal to the refinery. The strategic level is responsible for defining the refinery's demand and the type of crudes to arrive in vessels over the planning horizon. In addition, storage tanks may undergo maintenance services during certain periods.

Mixtures of crudes with similar properties (e.g., specific gravity and sulfur concentration) are allowed in the tanks. The objective seeks to minimize mixtures to provide more flexibility in order to satisfy the demands of the refinery.

The problem must account for critical constraints such as storage capacity, blending restrictions and maintenance requirements. Bilinear terms are needed to track the concentration of the different crudes in transfer operations between storage tanks and the main pipeline. Also, limits on the maximum number of storage tanks and volume of crude oil that can feed the main pipeline play a major role. The objective function aims to minimize the shortfall of crude supply at the refinery, minimize the mixtures of crudes in tanks, and meet the schedule for tank maintenance. The overall decisions of the scheduling problem consist in determining for the given planning horizon consisting of discrete time periods:

- The volume and quality of crude oil to be transferred from a vessel to each storage tank.

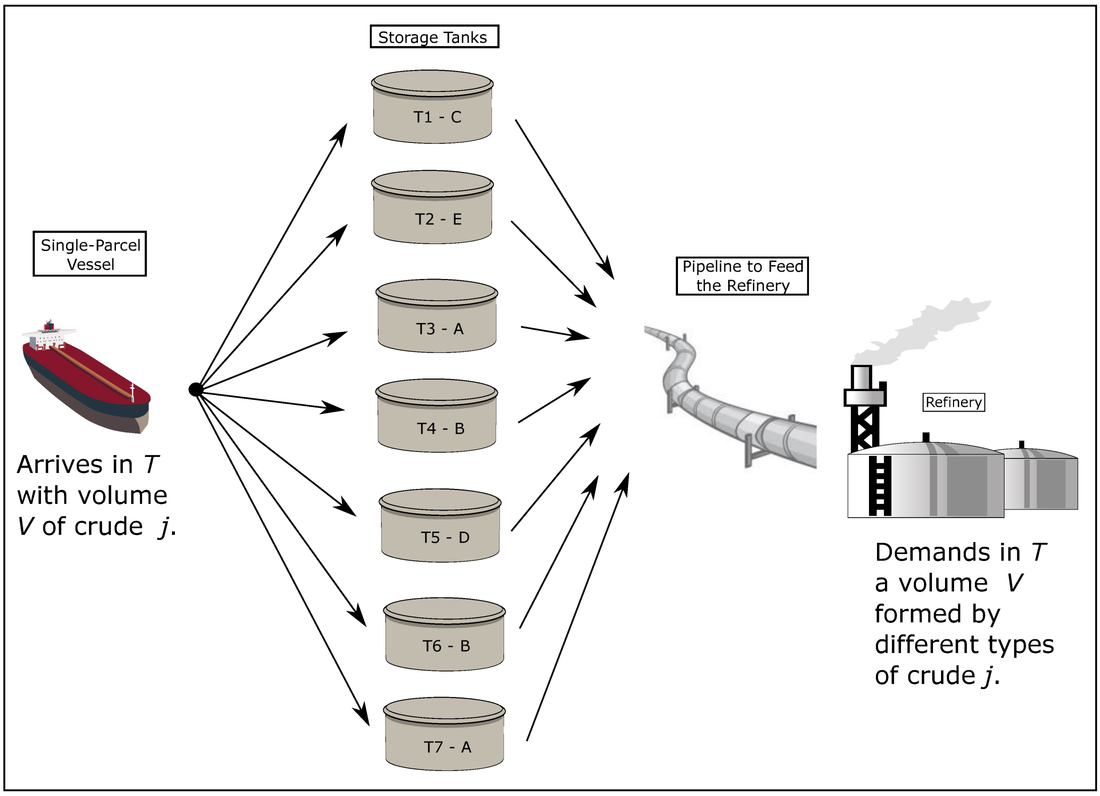


Figure 3 – ANCAP's crude oil terminal.

- The volume and blend of crudes to be sent to the main pipeline in order to satisfy the refinery demands.

2.3.1 Assumptions

The mathematical model is based on the following assumptions:

1. Inlet and outlet operations cannot be performed in a tank over the same period of time.
2. After a tank receives crude from a vessel, outlet operations must wait at least one period of time. This is necessary for brine separation. New and old crude qualities are mixed in a tank during vessel unloading.
3. Vessel-tank and tank-pipeline transfer operations are performed in one period of time.
4. A limited number of storage tanks can feed the pipeline simultaneously.
5. Vessel arrival times and their cargoes are known in advance.
6. The crude oil shipment from a vessel is not pre-allocated to a specific storage tank. This decision is made as part of the optimization.

7. To guarantee the quality of crude oil, rules on how crude oil can be blended are considered in the formulation.

2.4 MATHEMATICAL MODEL

2.4.1 Sets, Parameters and Variables

Before presenting the MINLP formulation, the sets, parameters, variables, constraints and the objective function are defined below.

2.4.1.1 Sets

The sets required for the problem definition are:

- $\mathcal{N} = \{1, \ldots, N\}$ is the set of tanks (index i).
- $\mathcal{J} = \{1, \ldots, J\}$ is the set of crude oil qualities (index j).
- $\mathcal{T} = \{1, \ldots, T\}$ is the set of time periods defining a discrete-time horizon of length T, over which operations are to be carried out. The periods of the horizon correspond to days (index t).

2.4.1.2 Parameters

The following parameters will be used in the model.

1. Demand Requirements
   - ΔRp^t is the total volume of crude oil demanded by the pipeline in period t.
   - ΓRpq_j^t is the percentage of crude j demanded by the pipeline in period t.
   - ΔRpq_j^t, where $\Delta Rpq_j^t = \frac{\Gamma Rpq_j^t * \Delta Rp^t}{100}$ is the volume of crude j demanded by the pipeline in period t.
2. Inventory
   - Vcq_j^t is the volume of crude j in a vessel that is scheduled to arrive at the crude oil terminal in period t.
   - Vc^t is the total volume of crude oil in a vessel that is scheduled to arrive at the crude oil terminal in period t. Notice that $Vc^t = \sum_{j \in J} Vcq_j^t$.
   - $\Delta Vcnq_{i,j}^0$ is the initial volume of crude j unloaded from a vessel to tank i.
   - $Vnq_{i,j}^0$ is the initial volume of crude j in tank i.
   - $\Delta Vpnq_{i,j}^0$ is the initial volume of crude j pumped from tank i to the pipeline.

- $\overline{Vn}_i$ is the maximum capacity of tank i.

3. Transfer Requirements

   - $Zcn_i^0 \in \{0, 1\}$ is 1 if crude oil is unloaded from a vessel to tank i in period 0; 0 otherwise.
   - $\underline{Dcn}$ is the minimum volume to be unloaded from a vessel into a tank.
   - $\overline{Dvp}$ is the maximum volume of crude oil that can be pumped into the pipeline at the same time.
   - $\overline{Pn}$ limits the number of tanks that can transfer crude oil to the pipeline at the same time.

4. Maintenance Requirements

   - $Zp_i^t \in \{0, 1\}$ is 1 if tank i can not transfer crude oil to the pipeline in period t; 0 otherwise. This parameter is used to express the need of having tank i out of service in period t, possibly due to maintenance in the pipeline or valves that connect the tank to the manifold.
   - $Xnmax_i^t \in \{0, 1\}$ is 1 if tank i is set to be at its maximum capacity in period t; 0 otherwise. This parameter is employed to induce tank i to be full in period t, mainly due to maintenance work on the tank roof.
   - $Xnmin_i^t \in \{0, 1\}$ is 1 if tank i is set to be empty in period t; 0 otherwise. This parameter is necessary to force tank i to be empty in period t in order to perform maintenance inside the tank (e.g., removal of solid deposits).

5. Blending Requirements

   - $\overline{Nq}$ is the maximum number of different crude qualities allowed in a tank.
   - $Xqq_{j,l} \in \{0, 1\}$ is 1 if crude j can be mixed with crude l; 0 otherwise.

2.4.1.3 Variables

Binary and continuous variables are needed. All variables are non-negative real variables, unless stated otherwise.

1. Oil Transfer

   - Δvcn_i^t is the volume of crude oil unloaded from a vessel to tank i in period t.
   - $\Delta vcnq_{i,j}^t$ is the volume of crude j unloaded from a vessel to tank i in period t.
   - Δvpn_i^t is the volume of crude oil delivered from tank i to the pipeline in period t.

- $\Delta vpnq_{i,j}^{t}$ is the volume of crude j delivered from tank i to the pipeline in period t.
- Δvp^{t} is the volume of crude oil delivered from the tanks to the pipeline in period t.
- Δvpq_{j}^{t} is the volume of crude j delivered from the tanks to the pipeline in period t.

2. Inventory Control

- vn_{i}^{t} is the total volume of crude oil in tank i in period t.
- $vnq_{i,j}^{t}$ is the volume of crude j in tank i in period t.

3. Demand Control

- $dqual_{j}^{t}$ is the difference between the volume of crude j sent to the pipeline in period t and the required demand.
- $dvol^{t}$ is the difference between the volume of crude oil sent to the pipeline in period t and the required demand.

4. Logistic

- $xnmax_{i}^{t} \in \{0, 1\}$ is 1 if tank i is full in period t; 0 otherwise.
- $zcn_{i}^{t} \in \{0, 1\}$ is 1 if a vessel uploads crude oil into tank i in period t; 0 otherwise.
- $z_{i}^{t} \in \{0, 1\}$ is 1 if tank i is full after a vessel uploads crude oil into the tank in period t; 0 otherwise.
- $xnq_{i,j}^{t} \in \{0, 1\}$ is 1 if crude j is present in tank i in period t; 0 otherwise.
- $zpn_{i}^{t} \in \{0, 1\}$ is 1 if crude oil is pumped from tank i to the pipeline in period t; 0 otherwise.

2.4.2 Constraints

2.4.2.1 Inventory Balance in Tanks

Eq. (1) defines the initial inventory of crude j in tank i, while Eq. (2) tracks the inventory of crude j in tank i for the remaining periods of the planning horizon. The total volume of crude oil in tank i is the sum of the volume of every crude j at the tank, which is tracked by Eq. (3). Lower and upper bounds on the storage capacity of each tank i

are determined by Eqs. (4) and (5).

$$vnq_{i,j}^{1} = Vnq_{i,j}^{0} + \Delta Vcnq_{i,j}^{0} - \Delta Vpnq_{i,j}^{0}, \quad \forall\, j \in \mathcal{J}, i \in \mathcal{N}, \tag{1}$$

$$vnq_{i,j}^{t} = vnq_{i,j}^{t-1} + \Delta vcnq_{i,j}^{t-1} - \Delta vpnq_{i,j}^{t-1}, \quad \forall\, j \in \mathcal{J}, i \in \mathcal{N}, t \in \mathcal{T}\backslash\{1\}, \tag{2}$$

$$vn_{i}^{t} = \sum_{j \in \mathcal{J}} vnq_{i,j}^{t}, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{3}$$

$$0 \leq vn_{i}^{t} \leq \overline{Vn}_{i}, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{4}$$

$$vn_{i}^{t} \geq \overline{Vn}_{i}\; xnmax_{i}^{t}, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}. \tag{5}$$

2.4.2.2 Crude Oil Mixture in Tanks

Although mixtures of different crude qualities are allowed, the operations in a crude oil terminal aim to minimize these mixtures in order to provide more flexibility to satisfy the demands of the refinery. Hence, constraints limiting the maximum number of crude qualities in the same tank and rules on how they can be mixed are needed. Eq. (6) enforces that the maximum number of different crude qualities in tank i is limited by parameter $\overline{Nq}$. Parameter $Xqq_{j,l}$ defines which crude qualities j and l can be mixed. Since not all crude qualities j and l can be mixed in tank i, Eq. (7) specifies which ones can be in the same tank. Further, if crude j is in tank i (i.e., $xnq_{i,j}^{t} = 1$), its volume can be at most the total of tank i, which is guaranteed by Eq. (8).

$$\sum_{j \in \mathcal{J}} xnq_{i,j}^{t} \leq \overline{Nq}, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{6}$$

$$Xqq_{j,l} \geq xnq_{i,j}^{t} + xnq_{i,l}^{t} - 1, \quad \forall\, i \in \mathcal{N}, j, l \in \mathcal{J}, t \in \mathcal{T}. \tag{7}$$

$$vnq_{i,j}^{t} \leq \overline{Vn}_{i}\; xnq_{i,j}^{t}, \quad \forall\, i \in \mathcal{N}, j \in \mathcal{J}, t \in \mathcal{T}, \tag{8}$$

2.4.2.3 Tank Completion

In practice, after emptying a tank by delivering oil to the refinery through the pipeline, small quantities of liquid or solid residues of hydrocarbons still remain at the bottom of the tank. It is desirable to dilute them by filling the tank when a vessel-storage tank uploading operation is performed. Variable z_{i}^{t} tracks if tank i is full after a vessel unloads crude oil. Eq. (9) defines the values of variable z_{i}^{t}, which takes a value of 1 if a vessel uploads crude oil into tank i in period $t-1$ (i.e., $zcn_{i}^{t-1} = 1$), and the tank is full in period t (i.e., $xnmax_{i}^{t} = 1$).

$$z_{i}^{t} = xnmax_{i}^{t}\; zcn_{i}^{t-1}, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}. \tag{9}$$

Notice that Eq. (9) is non-linear. Using standard linearization techniques, this equation can be recast as an equivalent system of integer linear equations given by Eqs. (10)-(14). Eqs. (10) and (11) treat unloading operations that have been performed before the considered planning horizon (parameter $Zcn_1^0 = 1$).

$$z_i^1 \geq xnmax_i^1 + Zcn_i^0 - 1, \;\; \forall\, i \in \mathcal{N}, \tag{10}$$

$$z_i^1 \leq Zcn_i^0, \;\; \forall\, i \in \mathcal{N}, \tag{11}$$

$$z_i^t \leq xnmax_i^t, \;\; \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{12}$$

$$z_i^t \geq xnmax_i^t + zcn_i^{t-1} - 1, \; \forall\, i \in \mathcal{N}, t \in (\mathcal{T} \backslash \{1\}), \tag{13}$$

$$z_i^t \leq zcn_i^{t-1}, \; \forall\, i \in \mathcal{N}, t \in (\mathcal{T} \backslash \{1\}). \tag{14}$$

2.4.2.4 Vessel-Tank Operation

The volume of crude oil in each vessel that arrives at the terminal is composed by a unique type of crude j. This load, which arrives in period t, is known in advance and determined by parameter Vcq_j^t. Further, the total volume of crude j is unloaded from the vessel into a tank (or tanks) in one period of time t, which is regulated by Eq. (15), while, the total volume of crude oil sent to tank i is determined by Eq. (16). Eq. (17) bounds the transfer of oil between a vessel and tank i.

$$Vcq_j^t = \sum_{i \in \mathcal{N}} \Delta vcnq_{i,j}^t, \;\; \forall\, j \in \mathcal{J}, t \in \mathcal{T}, \tag{15}$$

$$\Delta vcn_i^t = \sum_{j \in \mathcal{J}} \Delta vcnq_{i,j}^t, \;\; \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{16}$$

$$zcn_i^t \, \underline{Dcn} \leq \Delta vcn_i^t \leq Vc^t \; zcn_i^t, \;\; \forall\, j \in \mathcal{J}, t \in \mathcal{T}. \tag{17}$$

2.4.2.5 Tank-Pipeline Operation

Storage tanks must dispatch crude oil through the pipeline to satisfy the demands of the refinery in terms of total volume and crude quality. Eq. (18) tracks the total volume of oil sent from tank i to the pipeline in period t. The total amount of crude j sent from all tanks to the pipeline is represented in Eq. (19), while Eq. (20) determines the total volume of oil sent to the pipeline.

Several bounds are imposed to the operation. First, a limit on the maximum number of tanks that can feed the pipeline at the same time t is set by Eq. (21). Likewise, Eq. (22) limits the maximum volume of crude oil that can be sent from tank i. Moreover, the total volume of crude oil that can be sent to the pipeline in period t is limited by Eq. (23). Finally, a crude oil can be sent from tank i to the pipeline only if the tank is not under maintenance, as imposed by Eq. (24).

$$\Delta vpn_i^t = \sum_{j \in J} \Delta vpnq_{i,j}^t, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{18}$$

$$\Delta vpq_j^t = \sum_{i \in \mathcal{N}} \Delta vpnq_{i,j}^t, \quad \forall\, j \in \mathcal{J}, t \in \mathcal{T}, \tag{19}$$

$$\Delta vp^t = \sum_{i \in \mathcal{N}} \Delta vpn_i^t, \quad \forall\, t \in \mathcal{T}, \tag{20}$$

$$\sum_{i \in \mathcal{N}} zpn_i^t \leq \overline{Pn}, \quad \forall\, t \in \mathcal{T}, \tag{21}$$

$$\Delta vpn_i^t \leq \overline{Vn}_i\ zpn_i^t, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{22}$$

$$\Delta vp^t \leq \overline{Dvp}, \quad \forall\, t \in \mathcal{T}. \tag{23}$$

$$zpn_i^t \leq (1 - Zp_i^t), \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}. \tag{24}$$

2.4.2.6 Tank Inlet and Outlet Operations

Equation (25) establishes that inlet and outlet operations in tank i cannot be performed at the same time. In addition, after unloading crude oil from a vessel to storage tank i, outlet operations in tank i must wait at least one period of time for the crudes to be fully mixed. This is guaranteed by Equation (26).

$$zcn_i^t + zpn_i^t \leq 1, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T}, \tag{25}$$

$$zcn_i^t + zpn_i^{t+1} \leq 1, \quad \forall\, i \in \mathcal{N}, t \in \mathcal{T} \setminus \{T\}. \tag{26}$$

2.4.2.7 Blending Constraints

A principle that must be respected during a transfer operation between tank i and the pipeline, is that the concentration of crudes sent to the pipeline is the same as the one inside the tank. Equation (27) assures that the concentration of crude j in a batch going from tank i to the pipeline is equal to the concentration inside the tank. Notice that this equation yields two bilinear terms, which are non-convex.

$$\frac{\Delta vpnq_{i,j}^t}{\Delta vpn_i^t} = \frac{vnq_{i,j}^t}{vn_i^t} \Longrightarrow \Delta vpnq_{i,j}^t\ vn_i^t = \Delta vpn_i^t\ vnq_{i,j}^t, \quad \forall\, i \in \mathcal{N}, j \in \mathcal{J}, t \in \mathcal{T}. \tag{27}$$

2.4.2.8 Discrete-Time Non-Convex MINLP Model

The objective penalizes for not satisfying refinery's demand, both in terms of total volume and crude quality; the non-filling of a tank after a vessel-tank uploading operation; not mixing qualities in a tank; and not satisfying maintenance requirements. Notice that the objective is linear, which means that the only non-linear terms of the model appear in Eq. (27).

$$
\begin{aligned}
P:\ \min\ f = & \sum_{t\in\mathcal{T}} C^{\mathrm{pv}}\ dvol^{t} + \sum_{j\in\mathcal{J}}\sum_{t\in\mathcal{T}} C_j^{\mathrm{pq}}\ dqual_j^{t} \\
& + \sum_{i\in\mathcal{N}}\sum_{\substack{t\in\mathcal{T}\\ t\neq 1}} C^{\mathrm{z}}\ (zcn_i^{t-1} - z_i^{t}) + \sum_{i\in\mathcal{N}} C^{\mathrm{z}}\ (Zcn_i^{0} - zcn_i^{1}) \\
& + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{J}}\sum_{t\in\mathcal{T}} C^{\mathrm{xnq}}\ xnq_{i,j}^{t} \\
& + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{J}}\sum_{t\in\mathcal{T}} C^{\mathrm{xn}}\ Xnmin_i^{t}\ xnq_{i,j}^{t} \\
& + \sum_{i\in\mathcal{N}}\sum_{t\in\mathcal{T}} C^{\mathrm{xn}}\ Xnmax_i^{t}(1 - xnmax_i^{t}) \qquad (28a)
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t. : } & (1)\text{-}(8); (10)\text{-}(27), && (28b)\\
& \Delta vp^{t} - \Delta Rp^{t} \leq dvol^{t},\ \ \forall\ t\in\mathcal{T}, && (28c)\\
& -\Delta vp^{t} + \Delta Rp^{t} \leq dvol^{t},\ \ \forall\ t\in\mathcal{T}, && (28d)\\
& \Delta vpq_j^{t} - \Delta Rpq_j^{t} \leq dqual_j^{t},\ \ \forall\ t\in\mathcal{T}, j\in\mathcal{J}, && (28e)\\
& -\Delta vpq_j^{t} + \Delta Rpq_j^{t} \leq dqual_j^{t},\ \ \forall\ t\in\mathcal{T}, j\in\mathcal{J}, && (28f)\\
& C^{\mathrm{pv}},\ C_j^{\mathrm{pq}},\ C^{\mathrm{z}},\ C^{\mathrm{xnq}},\ C^{\mathrm{xn}} \in \mathbb{R}_{+},\ \ \forall\ j\in\mathcal{J}, && (28g)\\
& vnq_{i,j}^{t},\ vcnq_{i,j}^{t},\ vpnq_{i,j}^{t} \in \mathbb{R}_{+},\ \ \forall\ i\in\mathcal{N}, t\in\mathcal{T}, j\in\mathcal{J}, && (28h)\\
& vn_i^{t},\ vcn_i^{t},\ vpn_i^{t} \in \mathbb{R}_{+},\ \ \forall\ i\in\mathcal{N}, t\in\mathcal{T}, && (28i)\\
& z_i^{t},\ zcn_i^{t},\ zpn_i^{t},\ xnmax_i^{t} \in \{0,1\},\ \ \forall\ i\in\mathcal{N}, t\in\mathcal{T}, && (28j)\\
& \Delta vpq_j^{t},\ dqual_j^{t} \in \mathbb{R}_{+},\ \ \forall\ t\in\mathcal{T}, j\in\mathcal{J}, && (28k)\\
& \Delta vp^{t},\ dvol^{t} \in \mathbb{R}_{+},\ \ \forall\ t\in\mathcal{T}. && (28l)
\end{aligned}
$$

The cost parameters are described as follows:

- C_j^{pq} is the cost of the difference between the crude oil j sent to the pipeline and the required demand.
- C^{pv} is the cost of the difference between the total volume sent to the pipeline and the required demand.
- C^{xn} is the cost for not respecting maintenance requirements in a tank.
- C^{xqn} is the cost of mixing qualities in a tank.
- C^{z} is the cost for not filling a tank after a vessel-tank uploading operation.

2.5 SOLUTION STRATEGY

The proposed solution approach consists of an iterative two-step MILP-NLP algorithm based on piecewise McCormick envelopes. In general terms (see Fig. 4), each iteration of the strategy has the following steps:

- First, an MILP relaxation is constructed by applying piecewise McCormick envelopes for relaxing the bilinear terms, providing a lower bound on the MINLP.
- Following, the solution of the MILP is used as an initial point and its logistics decisions (binary variables) are fixed into the MINLP, resulting in a non-linear programming (NLP) problem.
- Finally, after solving the NLP and obtaining an upper bound, the domain of each variable involved in the bilinear terms is tightened for the next iteration.
- The algorithm stops when the difference between the upper and lower bounds is within the tolerance or a maximum solution time is achieved.

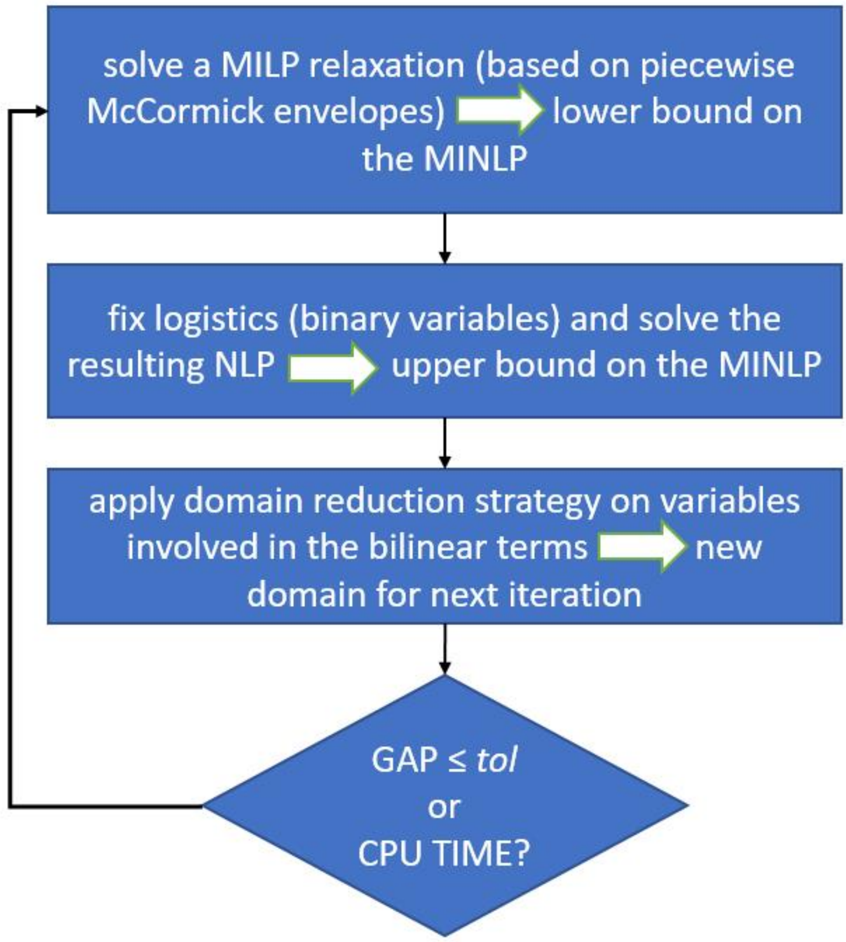


Figure 4 – Solution strategy diagram.

A more detailed explanation of the algorithm is presented below.

2.5.1 Relaxation of Bilinear Programs

McCormick envelopes (MCCORMICK, 1976) provide the tightest possible linear relaxation of the bilinear term $w = xy$. To obtain the relaxation, linear envelopes are built (see Fig. 5 (a)) over the domain of the variables x and y (see Fig. 6 (a)), which encloses the non-convex function $w = xy$. These inequalities relate variable w with x and x, and their lower and upper bounds (i.e., X^L, X^U, Y^L, Y^U).

In order to build a tighter relaxation, one can divide the domain of bilinear term $w = xy$ into $[o, p]$ partitions (see Fig. 6 (b)). Envelopes are constructed in each interval (see Fig. 5 (b)) and additional binary variables (i.e., $Z_{o,p}$) are included in the model in order to select the partition that provides the best relaxation. Notice that only one partition $[o, p]$ is selected, which is shown latter with the use of disjunctions to express the relaxation of the bilinear terms. The improved relaxation is referred to as piecewise McCormick with univariate partitioning (i.e., partition the domain of just one variable) or bivariate partitioning (i.e, partition the domain of both variables).

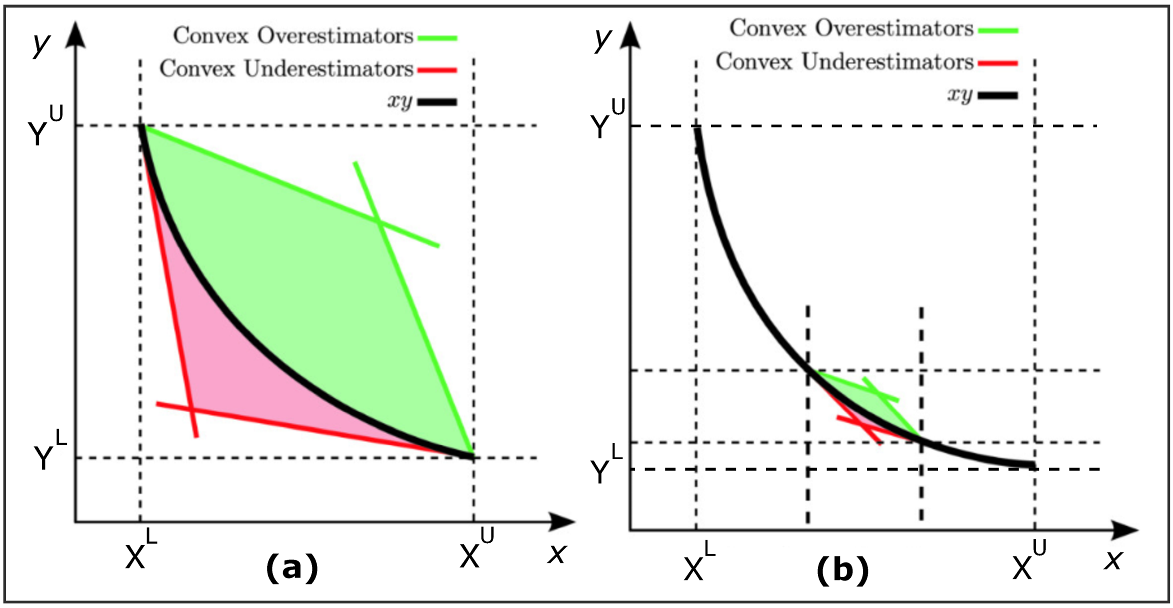


Figure 5 – Profile of Standard (a) and Piecewise (b) McCormick Envelopes (adapted from Miró et al. (2012)).

The present work proposes a strategy that makes use of piecewise McCormick envelopes, where domain partitions of a variable are uniform and binary variables increase linearly with the number of partitions. For the sake of limiting the size of this section, we will only describe the methodology for the right-hand side bilinear term $\eta^{RHS}_{i,j,t} = \Delta vpn^t_i vnq^t_{i,j}$. The same methodology is used for the left-hand side bilinear term $\eta^{LHS}_{i,j,t} = \Delta vpnq^t_{i,j} vn^t_i$.

Let the required sets, variables, and parameters be defined as follows:

- $\underline{DVPN}_{i,t} = 0$ and $\overline{DVPN}_{i,t} = \overline{Dvp}$ are the original lower and upper bounds of

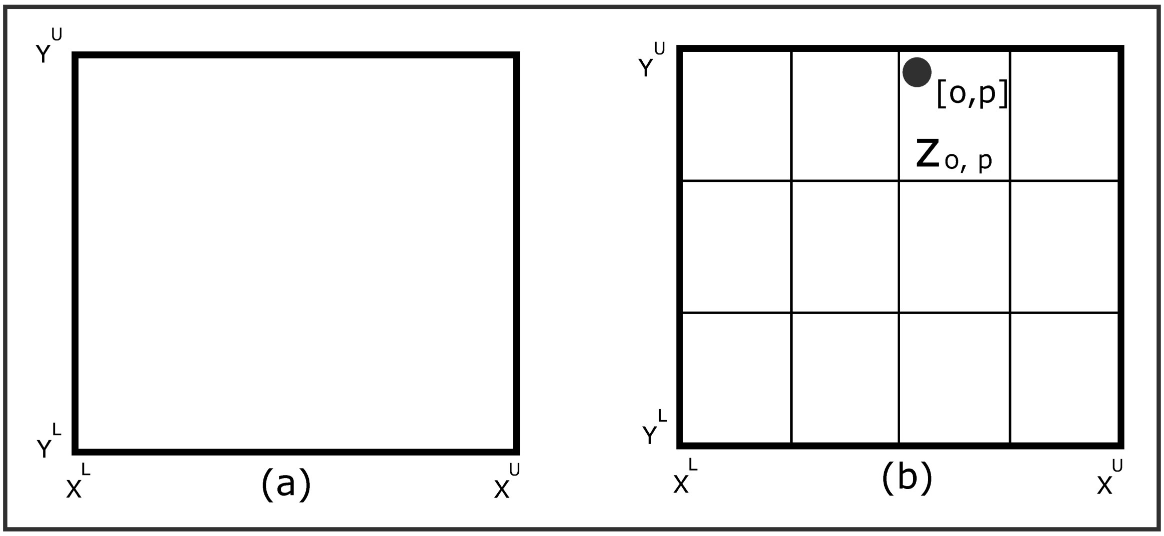


Figure 6 – Standard (a) and Piecewise (b) bilinear term domain.

variable Δvpn_i^t.

- $\underline{VNQ}_{i,j,t} = 0$ and $\overline{VNQ}_{i,j,t} = \overline{Vn}_i$ are the original lower and upper bounds of variable $vnq_{i,j}^t$.
- $\mathcal{O} = \{1, \ldots, O\}$ is the set of domain partitions of variable Δvpn_i^t (index o).
- $\mathcal{P} = \{1, \ldots, P\}$ is the set of domain partitions of variable $vnq_{i,j}^t$ (index p).
- $[\underline{DVPN}_{i,t,o}, \overline{DVPN}_{i,t,o}]$ are the bounds of each partition o of variable Δvpn_i^t.
- $[\underline{VNQ}_{i,j,t,p}, \overline{VNQ}_{i,j,t,p}]$ are the bounds of each partition p of variable $vnq_{i,j}^t$.
- $vnq_{t,i,j,o,p}$ is the value variable $vnq_{i,j}^t$ in partition $[o, p]$.
- $\Delta vpn_{t,i,j,o,p}$ is the value variable Δvpn_i^t in partition $[o, p]$.
- $y_{t,i,j,o,p}^{\text{RHS}} \in \{\text{True}, \text{False}\}$ indicates the selected partition $[o, p]$.

The relaxation of bilinear terms via piecewise McCormick envelopes can be expressed as a Generalized Disjunctive Program (GDP) (TRESPALACIOS; GROSSMANN, 2014). Eq. (29) constructs McCormick envelopes in each $[o, p]$ partition, while Eq. (30) indicates that only one $[o, p]$ partition is chosen. Eq. (31) defines lower and upper bounds of each partition. The following equations are defined for all $t \in \mathcal{T}$, $j \in \mathcal{J}$

and $i \in \mathcal{I}$.

$$\bigvee_{o=1}^{O}\bigvee_{p=1}^{P}\left[\begin{array}{c} y^{\text{RHS}}_{t,i,j,o,p} \\ \eta^{\text{RHS}}_{i,j,t} \geq \underline{VNQ}_{i,j,t,p}\Delta vpn^{t}_{i} + \underline{DVPN}_{i,t,o} vnq^{t}_{i,j} - \underline{VNQ}_{i,j,t,p}\underline{DVPN}_{i,t,o} \\ \eta^{\text{RHS}}_{i,j,t} \geq \overline{VNQ}_{i,j,t,p}\Delta vpn^{t}_{i} + \overline{DVPN}_{i,t,o} vnq^{t}_{i,j} - \overline{VNQ}_{i,j,t,p}\overline{DVPN}_{i,t,o} \\ \eta^{\text{RHS}}_{i,j,t} \leq \overline{VNQ}_{i,j,t,p}\Delta vpn^{t}_{i} + \underline{DVPN}_{i,t,o} vnq^{t}_{i,j} - \overline{VNQ}_{i,j,t,p}\underline{DVPN}_{i,t,o} \\ \eta^{\text{RHS}}_{i,j,t} \leq \underline{VNQ}_{i,j,t,p}\Delta vpn^{t}_{i} + \overline{DVPN}_{i,t,o} vnq^{t}_{i,j} - \underline{VNQ}_{i,j,t,p}\overline{DVPN}_{i,t,o} \\ \underline{DVPN}_{i,t,o} \leq \Delta vpn^{t}_{i} \leq \overline{DVPN}_{i,t,o} \\ \underline{VNQ}_{i,j,t,p} \leq vnq^{t}_{i,j} \leq \overline{VNQ}_{i,j,t,p} \end{array}\right], \tag{29}$$

$$y^{\text{RHS}}_{t,i,j,o,p} \in \{\text{True}, \text{False}\},\ \forall\ o \in \mathcal{O}, p \in \mathcal{P}, \tag{30}$$

$$\begin{cases} \underline{DVPN}_{i,t,o} = \underline{DVPN}_{i,t} + \frac{(\overline{DVPN}_{i,t} - \underline{DVPN}_{i,t})(o-1)}{O} \\ \overline{DVPN}_{i,t,o} = \underline{DVPN}_{i,t} + \frac{(\overline{DVPN}_{i,t} - \underline{DVPN}_{i,t})o}{O} \\ \underline{VNQ}_{i,j,t,p} = \underline{VNQ}_{i,j,t} + \frac{(\overline{VNQ}_{i,j,t} - \underline{VNQ}_{i,j,t})(p-1)}{P} \\ \overline{VNQ}_{i,j,t,p} = \underline{VNQ}_{i,j,t} + \frac{(\overline{VNQ}_{i,j,t} - \underline{VNQ}_{i,j,t})p}{P} \end{cases} \forall\ o \in \mathcal{O}, p \in \mathcal{P}. \tag{31}$$

Big-M and convex hull reformulations (BALAS, 1985; GROSSMANN; TRESPALACIOS, 2013) are common methodologies for transforming a GDP into an MILP. Along the lines of Hasan and I.A. Karimi (2010) and Castro (2015), the convex hull approach is chosen to reformulate the GDP into an MILP, since big-M usually presents poor relaxation quality (WICAKSONO; KARIMI, IA, 2008). The reformulation is presented as follows and are valid for all $t \in \mathcal{T}$, $j \in \mathcal{J}$ and $i \in \mathcal{I}$.

$$\eta^{\text{RHS}}_{i,j,t} \geq \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}}(\underline{VNQ}_{i,j,t,p}\Delta vpn_{t,i,j,o,p} + \underline{DVPN}_{i,t,o} vnq_{t,i,j,o,p} - \underline{VNQ}_{i,j,t,p}\underline{DVPN}_{i,t,o} y^{\text{RHS}}_{t,i,j,o,p}), \tag{32}$$

$$\eta^{\text{RHS}}_{i,j,t} \geq \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}}(\overline{VNQ}_{i,j,t,p}\Delta vpn_{t,i,j,o,p} + \overline{DVPN}_{i,t,o} vnq_{t,i,j,o,p} - \overline{VNQ}_{i,j,t,p}\overline{DVPN}_{i,t,o} y^{\text{RHS}}_{t,i,j,o,p}), \tag{33}$$

$$\eta^{\text{RHS}}_{i,j,t} \leq \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}}(\overline{VNQ}_{i,j,t,p}\Delta vpn_{t,i,j,o,p} + \underline{DVPN}_{i,t,o} vnq_{t,i,j,o,p} - \overline{VNQ}_{i,j,t,p}\underline{DVPN}_{i,t,o} y^{\text{RHS}}_{t,i,j,o,p}), \tag{34}$$

$$\eta^{\text{RHS}}_{i,j,t} \leq \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}}(\underline{VNQ}_{i,j,t,p}\Delta vpn_{t,i,j,o,p} + \overline{DVPN}_{i,t,o} vnq_{t,i,j,o,p} - \underline{VNQ}_{i,j,t,p}\overline{DVPN}_{i,t,o} y^{\text{RHS}}_{t,i,j,o,p}), \tag{35}$$

$$\begin{cases} \Delta vpn^{t}_{i} = \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}} \Delta vpn_{t,i,j,o,p} \\ vnq^{t}_{i,j} = \sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}} vnq_{t,i,j,o,p} \end{cases} \tag{36}$$

$$\begin{cases} y^{\text{RHS}}_{t,i,j,o,p}\underline{DVPN}_{i,t,o} \leq \Delta vpn_{t,i,j,o,p} \leq \overline{DVPN}_{i,t,o} y^{\text{RHS}}_{t,i,j,o,p} \\ y^{\text{RHS}}_{t,i,j,o,p}\underline{VNQ}_{i,j,t,p} \leq vnq_{t,i,j,o,p} \leq \overline{VNQ}_{i,j,t,p} y^{\text{RHS}}_{t,i,j,o,p} \end{cases} \forall\ o \in \mathcal{O}, p \in \mathcal{P}, \tag{37}$$

$$\sum_{o\in\mathcal{O}}\sum_{p\in\mathcal{P}} y^{\text{RHS}}_{t,i,j,o,p} = 1, \tag{38}$$

$$y^{\text{RHS}}_{t,i,j,o,p} \in \{0, 1\},\ \forall\ o \in \mathcal{O}, p \in \mathcal{P}, \tag{39}$$

$$\text{Eq.(31)}. \tag{40}$$

Finally, Eq. (41) is required to enforce Eq. (27).

$$\eta^{LHS}_{i,j,t} = \eta^{RHS}_{i,j,t}, \ \forall\ t \in \mathcal{T}, j \in \mathcal{J}, i \in \mathcal{I}. \tag{41}$$

Therefore, the MILP relaxation model consists of:

- Problem P, without the blending equation (Eq. (27));
- Set of constraints given by Eqs. (32)-(40);
- Set of constraints regarding the relaxation of the left-hand side bilinear term $\eta^{LHS}_{i,j,t} = \Delta vpnq^t_{i,j} vn^t_i$;
- Eq. (41).

As seen in this section, McCormick envelopes produce linear inequalities that enclose the bilinear functions, and as such, induce relaxation problems that can provide lower (upper) bounds for minimization (maximization) problems. Piecewise McCormick envelopes allow the refinement of the envelopes within regions of the domain partition that lead to tighter MILP relaxations, which can be combined with a primal algorithm to produce a feasible solution with a quality certificate. Nevertheless, this strategy may require a large number of partitions that make the computational solution of the resulting MILP too costly. This motivates the development of the domain-reduction procedure presented below.

2.5.2 NLP

Solving the MILP model provides a lower bound on the MINLP model. After finding the MILP solution, the logistics (binary) variables are fixed into the MINLP, resulting in an NLP problem. The NLP can be solved with a local solver, and its solution provides an upper bound.

2.5.3 Tightening the Domain of Bilinear Terms

The domain-reduction procedure for the right-hand side bilinear term $\eta^{RHS}_{i,j,t} = \Delta vpn^t_i vnq^t_{i,j}$ is shown in Fig. 7. Assuming $|\mathcal{O}|$ and $|\mathcal{P}|$ to be the cardinality of sets $\mathcal{O}$ and $\mathcal{P}$, Figure 7(a) illustrates the $|\mathcal{O}||\mathcal{P}|$ domain partitions of the bilinear term $\eta^{RHS}_{i,j,t}$. Consider that at iteration n, the MILP solution lies in partition $[o,p]$ (see Fig. 7(a)). After fixing the logistics (binary) variables into the MINLP, the resulting NLP is solved. The proposed strategy identifies the position where the NLP solution lies considering the partitions of the MILP problem. Having found the partitions of the MILP ($[o,p]$) and NLP ($[o',p']$) solutions, a new domain for the bilinear term $\eta^{RHS}_{i,j,t}$ is obtained by enclosing the extremes of both solutions (green shaded area indicated in Fig. 7(b)). The new domain is again divided into $|\mathcal{O}||\mathcal{P}|$ partitions and used for iteration $n+1$ (see Figs. 7(c) and 7(d)).

Mathematically, the new domain of the variables Δvpn_i^t and $vnq_{i,j}^t$ at iteration $n+1$ is given by the following equations:

$$\begin{cases} \underline{DVPN}_{i,t}^{n+1} = \min\{\underline{DVPN}_{i,t,o}^{n}\ ,\ \underline{DVPN}_{i,t,o'}^{n}\} \\ \overline{DVPN}_{i,t}^{n+1} = \max\{\overline{DVPN}_{i,t,o}^{n}\ ,\ \overline{DVPN}_{i,t,o'}^{n}\} \end{cases} \quad \forall\ o, o' \in \mathcal{O}, t \in \mathcal{T}, i \in \mathcal{I}. \tag{42}$$

$$\begin{cases} \underline{VNQ}_{i,j,t}^{n+1} = \min\{\underline{VNQ}_{i,j,t,p}^{n}\ ,\ \underline{VNQ}_{i,j,t,p'}^{n}\} \\ \overline{VNQ}_{i,j,t}^{n+1} = \max\{\overline{VNQ}_{i,j,t,p}^{n}\ ,\ \overline{VNQ}_{i,j,t,p'}^{n}\} \end{cases} \quad \forall\ p, p' \in \mathcal{P}, t \in \mathcal{T}, j \in \mathcal{J}, i \in \mathcal{I}. \tag{43}$$

The same procedure is applied for the left-hand side bilinear term $\eta_{i,j,t}^{\mathrm{LHS}} = \Delta vpnq_{i,j}^t vn_i^t$.

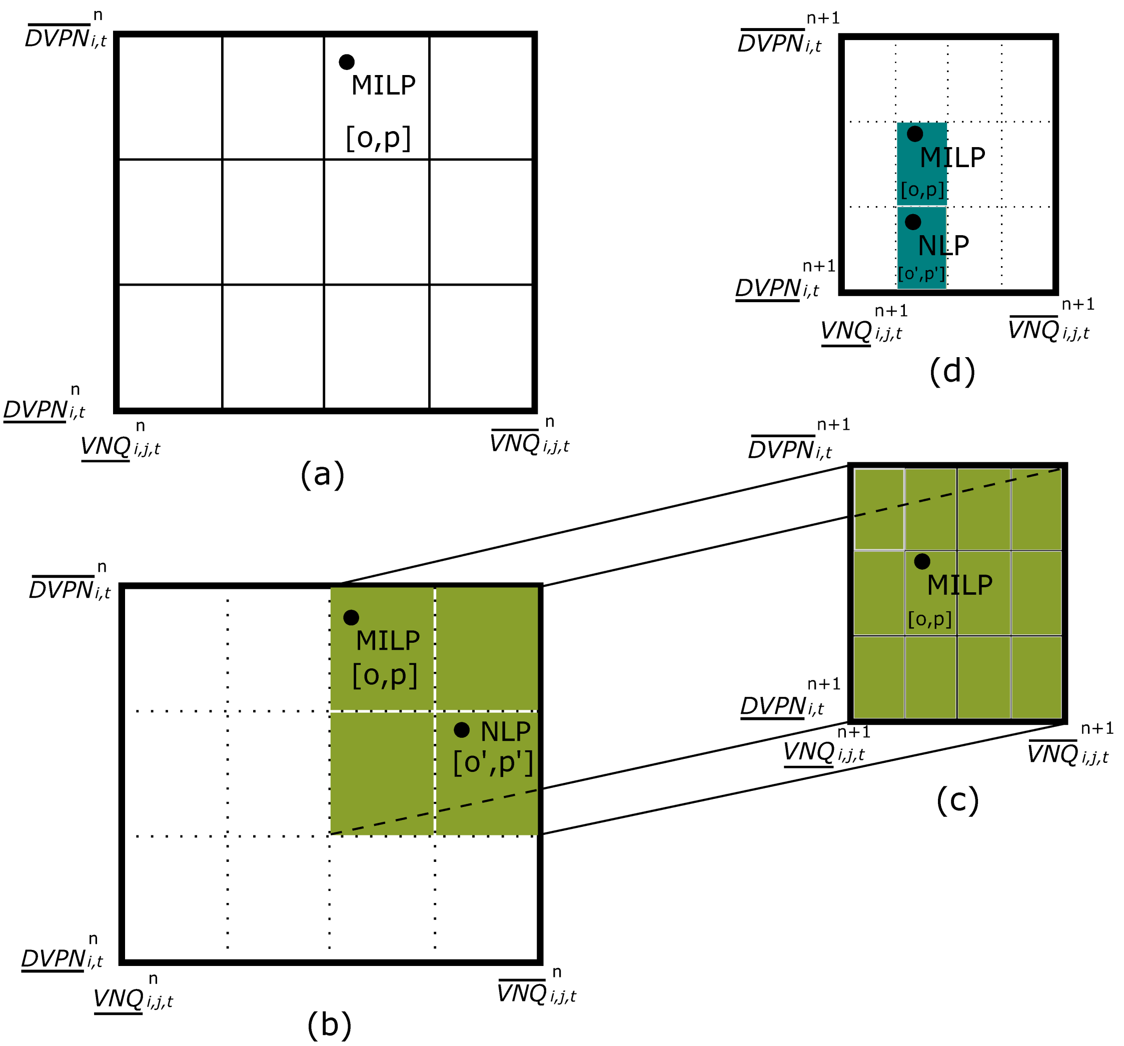


Figure 7 – Domain-reduction strategy for bilinear term $\eta_{i,j,t}^{\mathrm{RHS}} = \Delta vpn_i^t vnq_{i,j}^t$.

2.6 ANALYSIS

The proposed mathematical programming models are implemented in AMPL; MINLPs are solved using SCIP (ACHTERBERG, 2009); CPLEX is used for solving MILPs; and NLPs are solved by local solver CONOPT. The mathematical programming models and solution strategy were solved in a computer with two Intel Core Xeon E5-2630 v4 Processor (2.20 GHz), totaling 20 cores of 2 threads, 64 GB of RAM and a Ubuntu environment.

2.6.1 Problem Description

The crude oil terminal consists of 8 tanks, each one with a storage capacity of 64 000 m^3. As an operational rule, one of them is always on maintenance and 7 are operational (ZIMBERG et al., 2015). Initially, each storage tank has an initial volume and a single type of crude quality. In addition, 5 types of crude oil ({A, B, C, D, E}) are considered. Operation rules on how different types of crudes can be mixed are shown in Table 5. Cost parameters are detailed in Table 6.

Table 3 details an instance of the scheduling problem for a planning horizon of 30 days. Required demands are given in terms of total volume (in [$10^3 m^3$]). Also, the proportion of each crude demanded by the refinery is presented. A total of 6 vessels are scheduled to arrive, each one carrying a unique type of crude oil. Storage tank 3 is scheduled to be in maintenance between days 14 and 20.

2.6.2 Performance Analysis

The analysis consists in solving the scheduling problem for several instances, which are characterized by their cost and planning horizon. For example, instance LC-18Days considers the first 18 days of the planning horizon (Table 3) and low cost parameters (Table 6). A total of eight instances are considered for the analysis: LC-18Days, LC-22Days, LC-26Days and LC-30Days; HC-18Days, HC-22Days, HC-26Days and HC-30Days, in which LC stands for low cost and HC for high cost. Each instance is solved by the following strategies:

1. GO solver SCIP, refereed to as GO strategy.

2. LM proposed by Zimberg et al. (2015), refereed to as LM.

3. PCM (see Sec. 2.5), refereed to as PCM.

After an individual analysis of the performance of each strategy, an overall discussion comparing these strategies is presented.

A maximum CPU time of 1 hour (3 600 seconds) is set for the solution of each instance. All the statistics are obtained from the solver after it provides the solution.

For instances which the GO strategy finds a global solution within the 1 hour limit, the comparison with the LM and PCM strategies is concerned with their ability to obtain a global solution. However, if the GO strategy fails to find a global solution, but provides a feasible one, the comparison assesses whether the LM and PCM strategies can yield a better solution.

2.6.2.1 GO Strategy Performance

The GO strategy consists in solving the MINLP formulation *P* with SCIP. Table 7 presents the solutions (GO Solution), dual bounds and GAPs[1] provided by SCIP along with problem statistics, such as CPU time (in seconds), total number of variables and constraints, number of binary variables and non-linear constraints for the cases of low and high costs. Notice that SCIP is capable of finding, relatively fast, the global solution for instances LC-18Days and HC-18Days (highlighted in bold italic). For all other instances, SCIP reaches the CPU limit and halts with a GAP between lower and upper bounds that can be, in the worst case, up to 622% for instance LC-30Days and over 5 000% for instance HC-30Days.

2.6.2.2 LM Strategy Performance

Zimberg et al. (2015) proposed a linearization strategy to approximate the bilinear terms found in blending constraints. The strategy consists in defining a set of discrete values of predefined crude oil proportions which can be chosen by the optimization solver. It incorporates corrective terms for composition discrepancies, which are penalized in the objective function. A rolling-horizon approach was applied to the linearized model and optimized by CPLEX, producing more competitive solutions than SCIP when solving the non-linear model.

In order to make a fair comparison with the GO and PCM strategies, we applied the linearization approach of Zimberg et al. (2015) to model *P*, but without implementing the rolling horizon, which is then solved with CPLEX. Next, logistics decisions (binary variables) are fixed into the MINLP and the resulting NLP is solved.

Table 8 presents the solution (LM Solution) and CPU time (in seconds). While the GO strategy found global solutions for instances LC-18Days and HC-18Days, the results from Table 8 indicate that the LM strategy reaches global optimality only for scenario HC-18Days (highlighted in bold italic). On the other hand, the LM strategy provides better solutions for instances HC-22Days and HC-26Days than the GO strategy. Column LM Gain[2] of Table 8 gives the percent improvement. For all other instances, the LM strategy fails to provide improvements, not being able to even find a feasible solution for instances LC-30Days and HC-30Days.

[1] GAP between the GO Solution and its dual bound.

[2] LM Gain = $\frac{\text{GO Solution - LM Solution}}{\text{LM Solution}}100$.

2.6.2.3 PCM Strategy Performance

Tables 9 to 12 show the solution (PCM Solution), CPU time (in seconds) and number of iterations. For each instance, different numbers of domain partitions for the variables involved in the bilinear terms are tested. Each experiment runs until the algorithm reaches a CPU limit of 3 600 seconds or the gap between upper and lower bounds is less or equal to 10^{-2}%.

Besides the GO strategy, the PCM strategy also achieved the global optimum for instances LC-18Days and HC-18Days which is described in columns PCM Gain[3] of Table 9. For instance LC-18Days, global optimum is reached for almost all partitioning schemes, taking in some cases less than 10 seconds, while the GO strategy takes 50 seconds.

Significant improvements on the solutions found by the GO strategy are provided by the PCM strategy for low cost instances LC-22Days, LC-26Days and LC-30Days. As an example, partitioning [2, 2, 2, 2] (i.e., partitions related respectively to the domain of variables $\Delta vpnq^t_{i,j}$, vn^t_i, Δvpn^t_i and $vnq^t_{i,j}$) of instance LC-30Days (Table 12) can reach an improvement of 365% in relation to the solution provided by the GO strategy in not more than 42 minutes.

The PCM strategy also reports satisfactory performance for high cost instances. Instance HC-22Days is the only one that shows an advantage in favor of the LM strategy, reaching an improvement of 93% in almost 20 minutes, while the PCM strategy takes nearly 12 minutes to improve in 72% the solution provided by the GO strategy. Finally, impressive improvements are obtained for instances HC-26Days (1 995% with partitioning [2, 2, 3, 3]) and HC-30Days (4 274% with partitioning [3, 2, 3, 2]).

Figures 8 and 9 illustrate, respectively, the performance of the PCM strategy in relation to the GO strategy for instances LC-30Days and HC-30Days (hardest instances). Each figure presents the GO strategy solution (black horizontal line) and the four best results of the PCM strategy, with dashed lines representing univariate partitioning, while continuous ones represent bivariate. Iterations are marked by the symbol $\star$. The LM strategy was not able to provide a feasible solution for these instances. Hence, they are not presented in the figures. Notice that bivariate partitioning provides better results in fewer iterations (see Figure 9 zooming). However, these iterations are usually longer.

2.6.3 Discussion

An overall analysis suggests three major remarks regarding the results obtained by the PCM strategy:

1. **Solution Quality and CPU Time**. The results demonstrate that the proposed MILP relaxation is a good starting point for the NLP model to be sequentially

[3] PCM Gain = $\frac{\text{GO Solution - PCM Solution}}{\text{PCM Solution}} 100$.

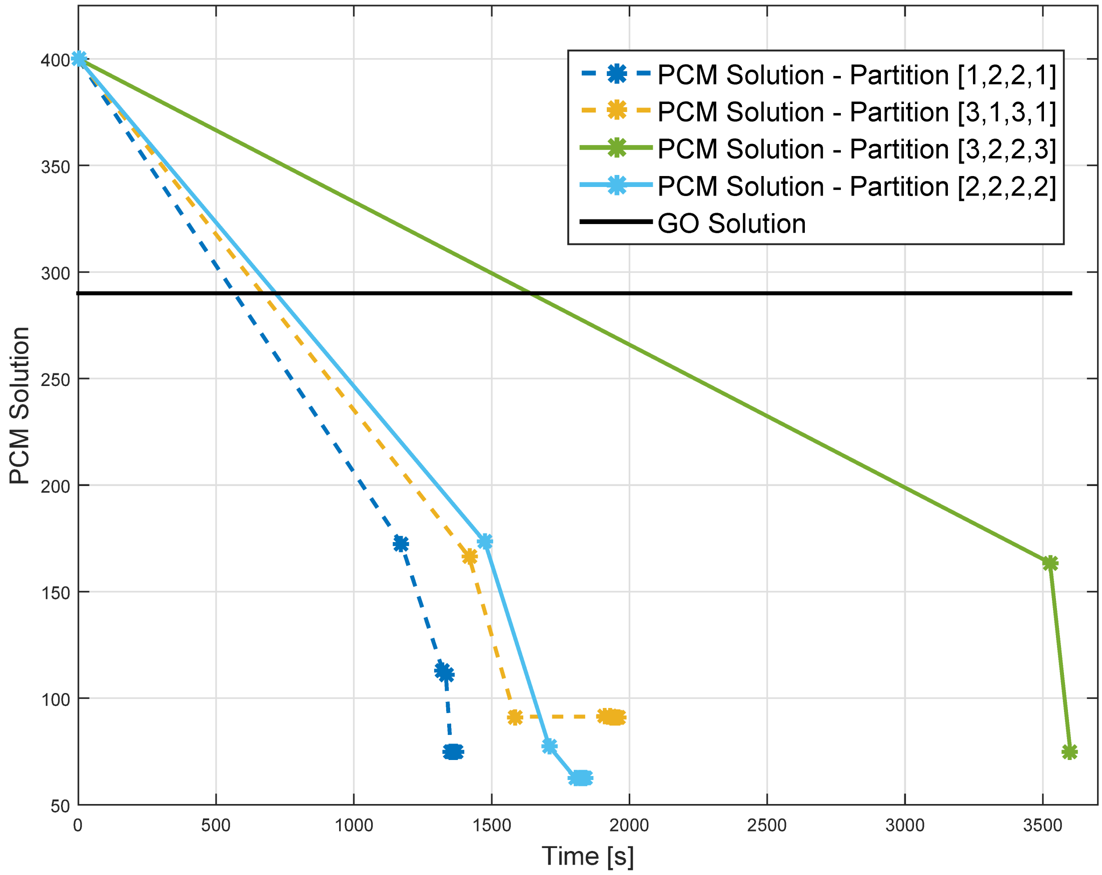


Figure 8 – PCM Strategy in relation to the GO Strategy for instance LC-30Days.

solved. In addition, the bivariate partitioning scheme usually provides a better relaxation than univariate, leading to better results in fewer iterations. However, the solution time was usually longer. This can be explained by the fact that univariate partitioning requires fewer binary variables than bivariate. In addition, the NLP solution is usually found in less than one second of CPU time, which means that the computation cost is mostly attributed to the solution of MILP relaxation.

2. **Variable Partitioning**. We shall recall that the objective (28a) of model P accounts for a number of goals by imposing penalties for: a) non-filling of a tank after a vessel-tank uploading operation; b) not respecting maintenance requirements; c) mixing qualities in a tank; and d) not satisfying refinery's demand, both in terms of total volume and crude quality. The last is directly related to the variables Δvpn_i^t (total volume sent from tank i to the pipeline in t) and $\Delta vpnq_{i,j}^t$ (total volume of crude j sent from tank i to the pipeline in t), which appear in the bilinear terms of Eq. (27). The results have shown that the number of domain partitions for variables $\Delta vpnq_{i,j}^t$ and Δvpn_i^t should be greater, or at least equal, to the number of domain partitions for the variables vn_i^t and $vnq_{i,j}^t$.

3. **Algorithm**. Although the solution strategy does not provide a guarantee of global optimality, the results reported in this work demonstrate that the PCM strategy

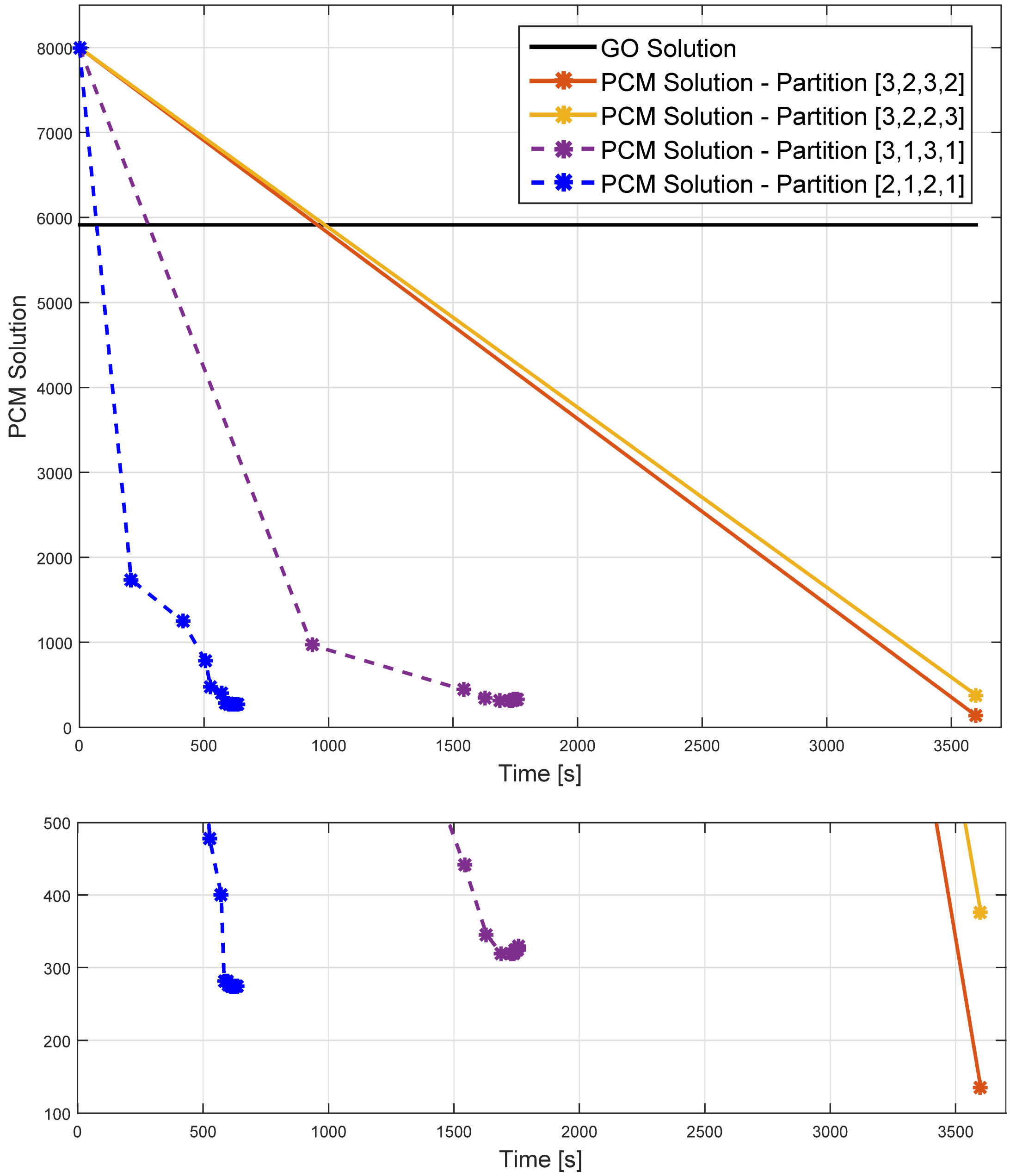


Figure 9 – PCM Strategy (and zooming) in relation to the GO Strategy for instance HC-30Days.

can be effective, both in terms of solution quality and computation time, when compared to the GO and LM strategies. The strategy was able to achieve an optimal solution in few iterations for instances LC-18Days and HC-18Days. In addition, significant gains in relation to the GO strategy solutions were observed for instances LC-22Days (18%) and HC-22Days (72%); LC-26Days (99%) and HC-26Days (1 995%); LC-30Days (365%) and HC-30Days (4 274%), which were the hardest instances (see Figures 8 and 9).

Table 3 – Requirements at the crude oil terminal.

Periods	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Demand [$10^3 m^3$]	4.0	4.5	4.0	3.0	8.0	6.0	7.0	5.0	6.0	6.0
Demand Proportion	1 C -	1 C -	1 C -	1 C -	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B
Vessel Cargo / Quality [$10^3 m^3$]	- -	- -	60 B	- -	- -	- -	- -	- -	- -	- -
Maintenance	-	-	-	-	-	-	-	-	-	-
Periods	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Demand [$10^3 m^3$]	7.0	7.0	6.5	5.0	6.0	6.0	7.0	6.0	6.0	6.0
Demand Proportion	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.2 D 0.8 B	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A
Vessel Cargo / Quality [$10^3 m^3$]	60 B	- -	- -	- -	- -	- -	60 A	- -	- -	- -
Maintenance	-	-	-	T3	T3	T3	T3	T3	T3	T3
Periods	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
Demand [$10^3 m^3$]	6.0	6.0	6.0	8.0	8.0	7.0	7.0	7.0	7.0	7.0
Demand Proportion	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A	0.3 E 0.7 A
Vessel Cargo / Quality [$10^3 m^3$]	- -	40 D	- -	- -	- -	70 A	- -	- -	30 C	- -
Maintenance	-	-	-	-	-	-	-	-	-	-

Table 4 – Initial conditions.

Tanks	1	2	3	4	5	6	7
Crude Quality	C	E	A	B	D	B	A
Volume [$10^3 m^3$]	47	54	4	10	25	14	8

Table 5 – Mixing rules (allow = 1, not allow = 0).

Crude Type	E	D	C	B	A
E	1	0	0	1	1
D	0	1	0	1	1
C	0	0	1	0	0
B	1	1	0	1	1
A	1	1	0	1	1

Table 6 – Cost parameters.

	C_j^{pq}	C^{pv}	C^{xn}	C^{xqn}	C^{z}
Low Cost	5	5	20	0.001	5
High Cost	50	50	20	0.001	5

Table 7 – Solutions obtained with the GO strategy.

Planning Horizon	Model Statistics				Low Cost				High Cost			
	Total Var.	Total Cons.	Binary Var.	Non-Linear Cons.	GO Solution	Dual Bound	GAP	CPU Time [s]	GO Solution	Dual Bound	GAP	CPU Time [s]
18 Days	3 618	10 397	828	220	[1] ***40.125***	40.125	-	50	[1] ***105.126***	105.126	-	11
22 Days	4 422	12 709	1 031	288	60.185	45.239	33%	3 600	251.802	105.179	139%	3 600
26 Days	5 226	15 021	1 241	380	107.184	41.453	158%	3 600	3 875.175	100.454	3 757%	3 600
30 Days	6 030	17 333	1 458	478	290.172	40.175	622%	3 600	5 915.184	105.196	5 523%	3 600

[1] Optimal solution.

Table 8 – Solutions obtained with the LM strategy.

Planning Horizon	Low Cost			High Cost		
	LM Solution	CPU Time [s]	LM Gain	LM Solution	CPU Time [s]	LM Gain
18 Days	[3]55.129	1 568	-	[1] ***105.126***	78	Optimal
22 Days	[4]110.135	3 600	-	[2]130.155	1 106	93%
26 Days	[4]115.159	3 600	-	[2]560.156	3 600	591%
30 Days	[5]-	3 600	-	[5]-	3 600	-

[1] Optimal solution.
[2] Improvement in relation to the GO strategy solution.
[3] Fails to reach optimality.
[4] Fails to provide improvement in relation to the GO strategy solution.
[5] No feasible solution within the time limit of 3 600 seconds.

Table 9 – Solutions obtained with the PCM strategy - LC-18Days and HC-18Days.

	Number of Domain Partitions				Low Cost				High Cost			
	$\Delta vpnq^t_{i,j}$	vn^t_i	Δvpn^t_i	$vnq^t_{i,j}$	PCM Solution	CPU Time [s]	Iterations	PCM Gain	PCM Solution	CPU Time [s]	Iterations	PCM Gain
Univariate	1	2	1	2	40.125	10	2	Optimal	[1]-	-	-	-
	1	2	2	1	40.125	14	2	Optimal	[1]-	-	-	-
	2	1	2	1	40.125	8	2	Optimal	105.126	9	3	Optimal
	2	1	1	2	40.125	9	1	Optimal	105.126	3	3	Optimal
	1	3	1	3	40.125	27	1	Optimal	[3]105.137	24	1	Near Optimal
	1	3	3	1	40.125	131	1	Optimal	105.126	44	2	Optimal
	3	1	3	1	40.125	21	3	Optimal	[1]-	-	-	-
	3	1	1	3	40.125	15	1	Optimal	[2]349.914	22	12	-
Bivariate	2	3	2	3	40.125	125	1	Optimal	[2]154.555	165	16	-
	2	3	3	2	[3]40.135	224	1	Near Optimal	[3]105.139	65	1	Near Optimal
	3	2	3	2	40.125	62	1	Optimal	[2]128.266	49	6	-
	3	2	2	3	[3]40.126	125	1	Near Optimal	[2]129.445	119	13	-
	2	2	2	2	40.125	17	1	Optimal	[2]125.144	26	5	-
	2	2	3	3	40.125	66	1	Optimal	105.126	27	1	Optimal
	3	3	2	2	40.125	51	1	Optimal	[1]-	-	-	-
	3	3	3	3	40.125	231	1	Optimal	[2]128.266	104	5	-

[1] Infeasible.
[2] Fails to reach optimality.
[3] Near Optimal: solution gap in relation to the GO strategy solution is less or equal to 1%.

Table 10 – Solutions obtained with the PCM strategy - LC-22Days and HC-22Days.

	Number of Domain Partitions				Low Cost				High Cost			
	$\Delta vpnq^t_{i,j}$	vn^t_i	Δvpn^t_i	$vnq^t_{i,j}$	PCM Solution	CPU Time [s]	Iterations	PCM Gain	PCM Solution	CPU Time [s]	Iterations	PCM Gain
Univariate	1	2	1	2	[2]174.821	137	17	-	[2]1 178.740	156	15	-
	1	2	2	1	[2]68.661	119	16	-	163.096	132	14	54%
	2	1	2	1	52.070	140	20	15%	[2]590.013	116	9	-
	2	1	1	2	[1]-	-	-	-	[2]691.084	31	20	-
	1	3	1	3	[2]165.994	322	8	-	[2]809.247	445	14	-
	1	3	3	1	[2]63.117	423	11	-	221.356	531	13	13%
	3	1	3	1	50.726	186	11	18%	244.980	201	14	2%
	3	1	1	3	[2]143.153	151	10	-	[2]895.226	257	12	-
Bivariate	2	3	2	3	50.667	813	13	18%	234.336	578	16	7%
	2	3	3	2	50.848	822	10	18%	223.236	1 197	15	12%
	3	2	3	2	[1]-	-	-	-	[2]269.368	469	12	-
	3	2	2	3	[2]67.379	733	12	-	179.835	327	14	40%
	2	2	2	2	50.896	249	19	18%	173.085	197	21	45%
	2	2	3	3	[2]63.271	1 268	12	-	146.266	722	11	72%
	3	3	2	2	50.668	1 476	11	18%	[1]-	-	-	-
	3	3	3	3	50.668	1 645	12	18%	[1]-	-	-	-

[1] Infeasible.
[2] Fails to provide improvement in relation to the GO strategy solution.

Table 11 – Solutions obtained with the PCM strategy - LC-26Days and HC-26Days.

	Number of Domain Partitions				Low Cost				High Cost			
	$\Delta vpnq^t_{i,j}$	vn^t_i	Δvpn^t_i	$vnq^t_{i,j}$	PCM Solution	CPU Time [s]	Iterations	PCM Gain	PCM Solution	CPU Time [s]	Iterations	PCM Gain
Univariate	1	2	1	2	[2]147.203	258	13	-	992.751	350	17	290%
	1	2	2	1	69.585	468	16	54%	279.314	376	15	1 287%
	2	1	2	1	73.600	351	25	45%	238.579	177	15	1 524%
	2	1	1	2	66.765	74	18	60%	547.717	80	19	607%
	1	3	1	3	79.593	1 052	14	34%	794.595	1 805	12	387%
	1	3	3	1	58.077	2 113	13	84%	540.492	1 727	9	616%
	3	1	3	1	72.359	492	11	48%	485.544	483	15	698%
	3	1	1	3	[2]144.815	604	13	-	573.512	122	10	575%
Bivariate	2	3	2	3	83.720	3 600	1	28%	272.112	2 957	2	1 324%
	2	3	3	2	96.353	3 600	1	11%	212.706	3 600	1	1 721%
	3	2	3	2	67.513	2 518	9	58%	-[1]	-	-	-
	3	2	2	3	63.102	2 629	11	69%	-[1]	-	-	-
	2	2	2	2	60.496	993	19	77%	323.088	559	10	1 099%
	2	2	3	3	53.701	2 777	11	99%	184.915	3 005	8	1 995%
	3	3	2	2	95.561	3 600	1	12%	273.211	2 797	1	1 318%
	3	3	3	3	[2]141.799	3 600	1	-	1 944.315	3 600	1	99%

[1] Infeasible.
[2] Fails to provide improvement in relation to the GO strategy solution.

Table 12 – Solutions obtained with the PCM strategy - LC-30Days and HC-30Days.

	Number of Domain Partitions				Low Cost				High Cost			
	$\Delta vpnq_{i,j}^t$	vn_i^t	Δvpn_i^t	$vnq_{i,j}^t$	PCM Solution	CPU Time [s]	Iterations	PCM Gain	PCM Solution	CPU Time [s]	Iterations	PCM Gain
Univariate	1	2	1	2	137.695	813	17	111%	1 551.652	3 600	5	281%
	1	2	2	1	74.716	1 371	16	288%	467.315	742	14	1 165%
	2	1	2	1	137.347	1 072	22	111%	274.568	637	20	2 054%
	2	1	1	2	164.022	372	18	77%	616.226	179	22	859%
	1	3	1	3	163.837	3 600	3	77%	1 140.261	3 600	2	418%
	1	3	3	1	130.193	3 600	1	122%	536.440	3 600	2	1 002%
	3	1	3	1	91.306	1 960	10	217%	324.075	1 758	11	1 725%
	3	1	1	3	162.052	1 421.24	12	79%	657.649	321	11	799%
Bivariate	2	3	2	3	227.742	3 600	1	27%	1 312.125	3 600	1	350%
	2	3	3	2	263.304	3 600	1	10%	651.706	3 600	1	807%
	3	2	3	2	182.343	3 600	1	59%	135.223	3 600	1	4 274%
	3	2	2	3	74.814	3 600	2	287%	376.705	3 600	1	1 470%
	2	2	2	2	62.326	1 839	8	365%	685.257	3 600	17	763%
	2	2	3	3	133.779	3 600	1	116%	660.551	3 600	1	795%
	3	3	2	2	197.312	3 600	1	47%	675.560	3 600	1	775%
	3	3	3	3	177.563	3 600	1	63%	504.030	3 600	1	1 073%

2.7 CONCLUSION

This work has proposed an extended model for the scheduling of operations at the crude oil terminal of ANCAP, the national refinery of Uruguay. First presented by Zimberg et al. (2015), the new model advances the previous one by considering more operational constraints.

The key contribution of this work is an iterative two-step MILP-NLP decomposition algorithm based on a piecewise McCormick relaxation, which implements a domain-reduction strategy for handling bilinear terms in the scheduling of crude oil operations. On small instances for which an optimal solution is known, the proposed strategy consistently finds optimal or near-optimal solutions. It also solves larger instances which are, in some cases, intractable by a global optimization solver and the MILP linearization strategy proposed by Zimberg et al. (2015). By solving several instances of the scheduling problem, it has been shown that the bivariate partitioning scheme usually provides a stronger relaxation than univariate, leading to better results in fewer iterations. On the other hand, the CPU time is usually higher. Another conclusion is that domain partitioning decisions should prioritize to have more, or at least equal, domain partitions for variables $\Delta vpnq_{i,j}^{t}$ (total volume of crude j sent from tank i to the pipeline in t) and Δvpn_{i}^{t} (total volume sent from tank i to the pipeline in t) when compared to the number of domain partitions for the variables vn_{i}^{t} and $vnq_{i,j}^{t}$.

As seen in this chapter, COS formulations usually consider the arrival period of vessels at the crude oil terminal and their cargo as a known parameter, and the details of the offshore operations are not taken into account. Therefore, if one wants to consider the OMCOS, there is the need to complete the offshore portion of the OMCOS by introducing the elements of MIR (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016) into the formulation. Chapter 3 proposes an integrated non-convex MINLP model and a solution strategy for the OMCOS that takes into account elements of both MIR and COS.

3 AN MINLP FORMULATION FOR INTEGRATING THE OPERATIONAL MANAGEMENT OF CRUDE OIL SUPPLY

3.1 INTRODUCTION

As addressed in Chapter 2, COS formulations deal with the supply of crude oil from the arrival of vessels at crude oil terminals to the feed of crudes to the CDUs. Nevertheless, in order to consider the extended chain of crude oil supply, there is the need to take into account the offshore elements that are associated to vessel's trips and FPSOs. Therefore, this chapter proposes an MINLP model for OMCOS (i.e., from FPSOs to CDUs), considering the scheduling of vessels and operations in a terminal, which means that it incorporates elements of two known problems in the literature: MIR and COS. To tackle this problem, an iterative MILP-NLP decomposition scheme with domain reduction is applied.

Chapter 1 states that the supply of crude oil from offshore oil fields to refineries is one of the major problems faced by vertically integrated oil companies (i.e., companies that control production, transportation, storage and refining). For offshore oil production, such as in the Brazilian Pre-Salt layer (FRAGA et al., 2009), the oil company relies on floating production, storage and offloading units (FPSOs), or simply platforms, to produce and store crude oil. After production, the crude oil is then transferred to onshore terminals by sub-sea pipelines or vessels. Since oil pipelines are not available in deep-water offshore oilfields, a fleet of vessels is deployed to transfer crude oil to the terminals (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016).

After arriving in a terminal, oil vessels unload crude oil through a pipeline to the tank farm, which is composed by storage tanks (STs) (ASSIS, Leonardo Salsano de et al., 2017). At this point mixtures cannot be avoided due to the large number of different types of crude oil in the market and the limited storage capacity of a terminal. At the refinery, the crude oil arriving from the storage tanks through a pipeline network is stored in charging tanks (CTs), which subsequently feed the crude distillation units (CDUs).

In the work of Rocha et al. (2009), the authors describe the main decisions at each level of the crude oil supply management problem. The decisions associated to the *strategic level* are concerned with defining for the long term the demands (i.e., both in terms of quantity and quality) of the refinery, as well as either to import or not crude oil. Meanwhile, medium term resource allocation and material flow are related to the *tactical level*. For example, one must decide which platforms will feed each crude oil terminal; which terminal will supply each refinery; the volumes of crude to be transferred between resources; and the vessel fleet composition. Finally, short term decisions such as routing and scheduling of vessels, and scheduling of operations in terminals are associated to the *operational level*.

Table 13 – List of items of each segment of the petroleum supply chain (SAHEBI et al., 2014).

Upstream				Midstream		Downstream	
Well Head (WH)	Well Platform (WP)	Production Platform (PP)	Crude Oil Terminal (CT)	Refinery (RF)	Petrochemical Plant (PC)	Distribution Center/Depot (DC)	Market/Customer (M/C)

The main items of each *upstream*, *midstream* and *downstream* segment of the petroleum supply chain are presented in Table 13. According to this classification, the management of crude oil supply integrates items of both upstream and midstream segments, involving production platforms in offshore oil fields, transportation, storage in crude oil terminals, and finally the feed of CDUs in refineries. The works of Aires et al. (2004), Rocha et al. (2009), Rocha et al. (2013) and Rocha et al. (2017) tackle the management of crude oil supply in its whole extension, dealing with strategic/tactical level decisions. This means that operational issues such as scheduling of vessels, blending and scheduling of operations in a terminal are not considered.

At the operational level, the management of crude oil supply has elements of maritime inventory routing (MIR) and crude oil scheduling (COS) problems, as stated by Aires et al. (2004). However, the independent solution of the MIR and COS problems for an integrated supply chain, such as the one that arises in a vertically integrated oil company, can lead to loss of information along the chain. Motivated by this lack of information sharing, this work develops a model for the operational management of crude oil supply. The proposed model integrates elements of MIR and COS problems, such as the scheduling of vessels, the scheduling of operations in the terminal and blending of crudes. Such an integration is advocated by Barbosa-Póvoa (2014) and Lazaros G. Papageorgiou (2009), who pointed out the importance of integrating, if possible, tactical and operational level decisions, to enhance the supply chain performance.

The independent solution of maritime inventory routing and crude oil scheduling may fail to coordinate the access to shared resources, such as storage tanks, which define the boundary between MIR and COS problems. For instance, the MIR problem usually considers that storage tanks are available for receiving crudes from vessels that arrive from time to time at the terminal. However, the dynamics of storage tanks are also affected by outlet operations that are managed by the COS problem, leading to a potential mismatch between the level and composition of crudes in those tanks. Further, operational constraints, such as the rule that prevents simultaneous inlet and outlet operations in storage tanks, may not be fulfilled in a solution obtained by solving the MIR and COS problems independently. Such limitations motivate the integration of the operational management of crude oil supply, taking into account elements of the MIR and COS problems.

The technical paper (ASSIS, Leonardo S. et al., 2019) is used as basis for the development of this chapter. The remainder of this work is organized as follows. A review of the literature is presented in Section 3.2. The problem definition is given in Section 3.3. The mathematical formulation is described in Section 3.4, while the proposed solution strategy is presented in Section 3.5. Problem instances and computational results are shown in Section 3.6. Finally, the conclusion and directions for future work are described in Section 3.7.

3.2 LITERATURE REVIEW

This section is divided into four parts. First, we present an overview of maritime inventory routing (MIR) and crude oil scheduling (COS), since the problem of concern has elements of both domains. The second part discusses works that attempt to integrate segments of the petroleum supply chain. Since we apply McCormick envelopes to obtain a linear relaxation of the model, the third part presents works from the literature that apply these envelopes in their solution strategies. Finally, the section ends with a discussion that positions the current work with respect to the technical literature and further states the contributions.

3.2.1 Overview on Maritime Inventory Routing and Crude Oil Scheduling

Maritime inventory routing consists of scheduling the trips of a set of vessels between ports in order to satisfy certain demands for products, while respecting lower and upper limits of inventory at production and consumption ports. The work of David Ronen (1983) is the first review on ship scheduling problems. Further models started to incorporate inventory control in ports such as in D. Ronen (2002) and Camponogara and Plucenio (2014). The latter derived valid inequalities and proposed a Lagrangean based strategy to solve the MILP model. An extensive review on MIR problems is provided by Christiansen et al. (2013).

The works of Dimitri J. Papageorgiou et al. (2018) and Agra et al. (2017) have addressed strategies to deal with deep sea (i.e., where ports are spread in different continents) and short sea (i.e., where ports are near each other) MIR problems, respectively. The former analyzes the use of several matheuristics like rolling horizon heuristics, K-opt heuristics, and local branching, while the latter proposes discrete and continuous time formulations, extended formulations and valid inequalities that strengthen the formulations.

Recently, several techniques to handle uncertainties in vessels' travel time are addressed by Rodrigues et al. (2019), namely robust optimization, stochastic programming, deterministic model with inventory buffers and models that incorporate conditional value-at-risk measures.

In crude oil scheduling problems, the main objective is to satisfy the demands of

CDUs (i.e., both in terms of total volume and quality of crude oil). To achieve that, one must schedule a set of operations including the unloading of crude oil into storage tanks, the transfers between storage and charging tanks, and the feed of CDUs performed by charging tanks.

Heeman Lee et al. (1996) were the first to address the crude oil scheduling problem. The authors proposed and solved a discrete time MILP model where blending constraints were not considered and replaced by a linear approximation. On the other hand, a continuous time MINLP formulation coined as single operation sequencing was proposed by Mouret et al. (2009), whose main advantage is that the number of time slots is significantly smaller than traditional formulations. Further, a combination of symmetry-breaking constraints and two-step MILP-NLP decomposition is applied to solve the problem.

The work of Leonardo Salsano de Assis et al. (2017) considers the operations in ANCAP's crude oil terminal (i.e., national refinery of Uruguay), excluding charging tanks and CDUs. To tackle the problem, the authors proposed a piecewise McCormick based MILP-NLP decomposition with domain reduction.

Similar to short-term crude oil scheduling, the scheduling and feed of concentrates is a common problem in the metal refining industry (SONG, Y. et al., 2018). This problem, which consists of a MINLP formulation, has the goal of defining the blend of concentrates and the sequence in which each blend is fed to the smelter in order to satisfy its demand both in terms of quantity and specification.

3.2.2 Management of Integrated Crude Oil Supply

Despite not considering the integrated problem of supplying crude oil from FPSOs to CDUs, other works have also tackled problems involving multiple segments of the crude oil supply chain. For instance, Escudero et al. (1999) proposed a linear programming (LP) model for defining optimal material flows of an oil company. Known as multi-period the Supply, Transformation and Distribution (STD) problem, it consists of a network with storage tanks, transformation sites, transshipment nodes, and destination depots. Uncertainties such as spot selling price, spot supply cost, and product demand are also taken into account. Dempster et al. (2000) also considered the multi-period STD with uncertainties in product demands and spot supply costs. The authors suggest a stochastic LP model to produce, supply, and distribute crude oil products from a consortium of oil companies.

On the refinery side, Sérgio M.S. Neiro and José M. Pinto (2004) and Sérgio M. S. Neiro and José M. Pinto (2005) tackled a network made up by a set of crude oil terminals, pipelines, refineries, and distribution centers. The authors proposed a large-scale MINLP model, which considered connections between refineries and the non-linearities associated to product blending. Main decisions consist of defining stream flow rates, operational variables, inventory management, and facility assignment. Al-Othman et al. (2008) extended the work of Sérgio M.S. Neiro and José M. Pinto (2004) and Sérgio M. S. Neiro and José M. Pinto (2005) by integrating a petrochemical plant into the network. Moreover, the model considered penalties on missed demands and backlogs in the objective function, while dealing with uncertainties in market demands and prices. Guyonnet et al. (2009) proposed an extension to the work of Sérgio M.S. Neiro and José M. Pinto (2004) by adding the scheduling of operations in the crude oil terminal and the distribution of final products.

To the best of our knowledge, Aires et al. (2004) were the first to tackle the integrated problem of supplying crude oil (i.e., from FPSOs to CDUs). Dealing with strategic/tactical level decisions, they propose an MILP formulation for allocating crude oil produced by a set of platforms to a set of terminals in order to satisfy the demands of a set of refineries (i.e., both in terms total volume and quality of crude oil). In addition, crude oil import, inventory management and vessel fleet sizing decisions are also considered. Rocha et al. (2009) combined a heuristic method to find a feasible solution with a local search procedure to improve it. Tighter reformulations of material balance constraints based on knapsack inequalities are presented by Rocha et al. (2013), providing substantial gains on solution time for the tested instances. Recently, Rocha et al. (2017) proposed an efficient decomposition algorithm and made use of material balance constraints reformulation, valid inequalities and an extended formulation related to the offload of platforms for solving the problem. The limited number of vessels, scheduling of vessels, scheduling of operations in terminals and non-linearities due to blending were not addressed.

3.2.3 Overview on McCormick Envelopes Based Strategies

McCormick envelopes provide the tightest possible linear relaxation for bilinear terms (MCCORMICK, 1976). In this approach, the bilinear term $x_i x_j$ is replaced with a new continuous variable w_{ij} and four sets of linear constraints are considered for the formulation.

The standard McCormick envelopes can be strengthened by partitioning the domain of one of the variables, say x_j, into n disjoint regions and then approximating the bilinear term within each partition, a process that requires binary variables to select the best partition for x_j. First proposed by Bergamini et al. (2005), this partitioning approach is known as univariate piecewise McCormick as a means to tighten the approximation.

To the best our knowledge, Wicaksono and IA Karimi (2008) were the first to propose the domain partitioning of both variables, x_i and x_j, a strategy that came to be known as bivariate piecewise McCormick. In this work, bivariate partitioning yielded a stronger relaxation than univariate partitioning in moderate-size problems, such as column sequencing for nonsharp distillation, integrated water use and treatment systems, generalized pooling problems on wastewater treatment networks, and synthesis of heat exchanger networks. Later, Hasan and I.A. Karimi (2010) applied bivariate partitioning to a benchmark process network synthesis problem, obtaining stronger relaxations than univariate partitioning. A comprehensive study on piecewise under- and over-estimators for bilinear terms was presented by Gounaris et al. (2009).

A number of works on Piecewise McCormick envelopes have appeared in the recent literature. Castro (2015) proposed an iterative algorithm based on univariate piecewise McCormick with bound contraction, which was applied for the design of water system networks. In the work Leonardo Salsano de Assis et al. (2017), the authors developed an MINLP model for the optimization of operations in a crude oil terminal, for which univariate and bivariate piecewise McCormick envelopes were applied as part of a MILP-NLP decomposition. Castillo Castillo et al. (2017) presented an iterative strategy based on approximation of bilinear terms to solve a class of MINLP that arise from oil refinery planning. Their strategy produces tight MILP relaxation by dynamically discretizing the bilinear terms using either piecewise McCormick or normalized multiparametric disaggregation.

3.2.4 Work Contribution

In summary, a few works address the integrated management of crude oil supply (i.e., from FPSOs to CDUs) at the strategic/tactical decision levels, but without taking into account operational issues such as scheduling of vessels, blending and scheduling of operations in a terminal. To this end, the present work contributes to the literature by developing a model for the management of crude oil supply at the operational level, incorporating elements of maritime inventory routing and crude oil scheduling (AIRES et al., 2004).

To the best of our knowledge, this is the first work to integrate the management of crude oil supply at the operational level by taking into account the scheduling of vessels, the scheduling of operations in the terminal and the non-convex non-linearities associated to the blending of crudes. To tackle this problem, we propose a discrete time MINLP formulation to be solved by an iterative MILP-NLP decomposition, which relies on domain reduction, bivariate piecewise McCormick envelopes to yield the MILP relaxation, and a NLP solver to reach feasible solutions.

3.3 PROBLEM STATEMENT

Figure 10 depicts an instance of the operational management of crude oil supply, which is composed by the following set of resources: FPSOs (*FPSO1* and *FPSO2*), vessels (*Vessel1* and *Vessel2*), storage tanks (*ST1* and *ST2*), charging tanks (*CT1* and *CT2*) and crude oil distillation units (*CDU1*). Moreover, the arrows illustrate all possible operations along the chain: FPSO offloading, vessel traveling, vessel waiting at the terminal, vessel unloading into storage tanks, transfers from storage to charging tanks, and from charging tanks to CDUs.

In Figure 10, *Vessel1* and *Vessel2* offload crude oil from *FPSO1* through operations *v1* and *v2*, respectively. Likewise, *FPSO2* is offloaded by *Vessel1* and *Vessel2* through operations *v3* and *v4*, respectively.

Vessel1 travels back and forth between the crude oil terminal and *FPSO1* and *FPSO2* according with operations *v5* and *v7*, respectively, which explains why the arrows are bidirectional. Similarly, *Vessel2* performs the same tasks through operations *v6* and *v8*, respectively. Note that vessels must be empty when traveling from the crude oil terminal to an FPSO. After reaching an FPSO, offloading operations are performed until no storage capacity remains in the vessel, which means that only one type of crude oil is transported at time. This policy of offloading from an FPSO until full capacity is consistent with real world operations, since FPSOs have a larger storage capacity than the capacity of vessels. Besides maximizing the free capacity of FPSOs, such a policy minimizes the inventory holding at the FPSOs and reduces the number of vessel trips to and from FPSOs. This policy further avoids the mixing of crude oil during vessel transportation, which is allowed only at the terminal.

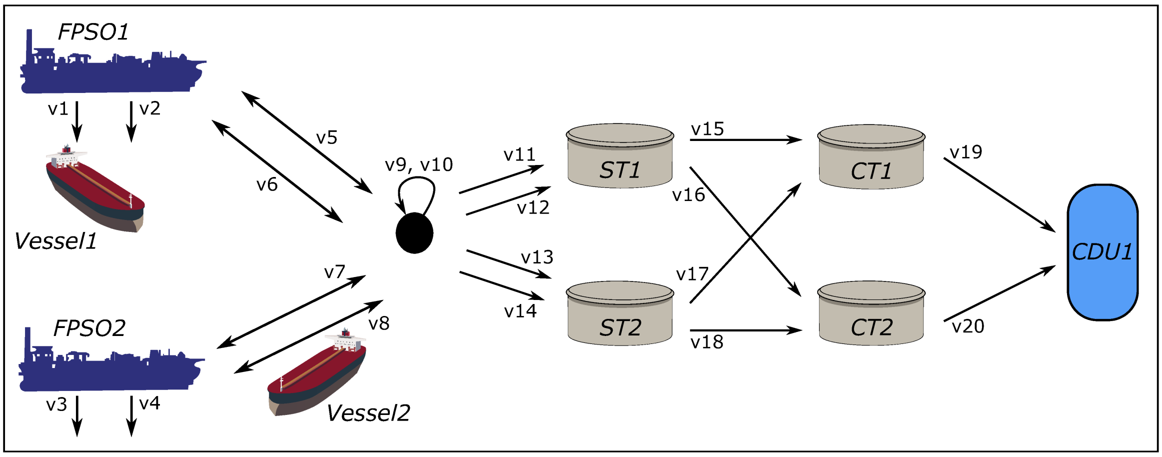


Figure 10 – The management of crude oil supply.

After arriving at the terminal, vessels can directly unload crude oil into storage tanks or stand by if another vessel is performing unloading operations. *Vessel1* and *Vessel2* can wait at the terminal through operations *v9* and *v10*, respectively. (Although demurrage costs can have an impact on the problem solution, they are not considered in this work). Unloads into storage tanks *ST1* and *ST2* are performed by *Vessel1* through operations *v11* and *v13*, respectively; and by *Vessel2* through operations *v12* and *v14*.

At the storage tanks, different types of crude oil can be mixed. Therefore, transfer operations *v15, v16, v17* and *v18* from storage to charging tanks are ruled by blending constraints. Note that in the storage tanks there are no restrictions on how crude oils are mixed.

Charging tanks *CT1* and *CT2* will feed *CDU1* through operations *v19* and *v20*, respectively. Because different types of crude oil can also be mixed in the charging tanks, transfer operations from charging tanks (*v19* and *v20*) are also submitted to blending constraints.

The feed of crude oil to CDUs must satisfy two types of demands, namely the total volume delivered over the planning horizon and its composition must be within given ranges. For this reason, the blends of crudes in the charging tanks should be within the composition ranges imposed by the CDUs which, in turn, means that the storage tanks must feed the charging tanks in order to yield the required compositions.

The main operational rules are:

(a) a vessel must be empty before traveling to offload an FPSO;

(b) a vessel must offload an FPSO to fully fill its storage tanks, until no residual capacity is left;

(c) an FPSO can load at most one vessel at a time;

(d) a vessel must unload all its volume into the storage tanks at the terminal, but not necessarily in the same tank;

(e) at most one vessel at a time can unload into the storage tanks (i.e., only one single buoy mooring (SBM) is available);

(f) at most one (inlet or outlet) operation can be performed during the same time period at a storage or charging tank;

(g) at least one distillation operation must be carried out in a time period.

The optimization problem consists in determining, for the planning horizon, the optimal schedule of operations associated to all resources in order to satisfy the demands of CDUs (i.e., both in terms of quality and quantity), while maximizing the gross margin. To this end, we propose a discrete time MINLP model, whose main decisions consist in selecting what operations take place at each time, the level of crudes in each resource, and the volume of crude oil transferred between resources.

3.4 MATHEMATICAL MODEL

Before proposing the discrete time MINLP model for the problem of concern, sets, parameters, variables, constraints, and the objective function are presented below.

3.4.1 Sets, Parameters and Variables

1. **Sets**

   The following sets are required for the problem formulation:

   - $\mathcal{T} = \{1, \ldots, PH\}$: set of discrete time periods which define the planning horizon PH.
   - $\mathcal{RF}$, $\mathcal{RV}$, $\mathcal{RS}$, $\mathcal{RC}$ and $\mathcal{RD}$: respectively the set of FPSOs, vessels, storage tanks, charging tanks, and CDUs.
   - $\mathcal{R} = \mathcal{RF} \cup \mathcal{RV} \cup \mathcal{RS} \cup \mathcal{RC} \cup \mathcal{RD}$: set of all resources.
   - $\mathcal{WL}$, $\mathcal{WU}$, $\mathcal{WW}$, $\mathcal{WT}$, $\mathcal{WF}$ and $\mathcal{WD}$: respectively, the set of offloading operations, unloading operations, wait operations, travel operations, tank-to-tank feed operations and distillation operations.
   - $\mathcal{W} = \mathcal{WL} \cup \mathcal{WU} \cup \mathcal{WW} \cup \mathcal{WT} \cup \mathcal{WF} \cup \mathcal{WD}$: set of all operations.
   - $\mathcal{I}_r \subset \mathcal{W}$: set of inlet operations on each resource $r \in \mathcal{R}$.
   - $\mathcal{O}_r \subset \mathcal{W}$: set of outlet operations on each resource $r \in \mathcal{R}$.
   - $\mathcal{D}_r \subset \mathcal{W}$: set of wait operations of each vessel $r \in \mathcal{RV}$.
   - $\mathcal{TR}_r \subset \mathcal{W}$: set of travel operations of each vessel $r \in \mathcal{RV}$.
   - $\mathcal{G}_r = (\mathcal{N}_r, \mathcal{E}_r)$ is a graph representing the flow between operations associated to each vessel $r \in \mathcal{RV}$, where $\mathcal{N}_r = \mathcal{I}_r \cup \mathcal{O}_r \cup \mathcal{D}_r \cup \mathcal{TR}_r$ is the set of nodes and $\mathcal{E}_r = \mathcal{N}_r \times \mathcal{N}_r$ is the set of arcs. The nodes $\mathcal{N}_r$ are the operations regarding vessel r, while the edges $(v, u) \in \mathcal{E}_r$ indicate the possibility to flow from operation v to u ($v, u \in \mathcal{N}_r$). Since not every flow between operations is allowed, set $\mathcal{VD}_r \subset \mathcal{E}_r$ contains the possible flows between operations related to vessel r. For example, if vessel r is performing an unloading or a waiting operation, it must execute a travel operation before offloading an FPSO.
   - $\mathcal{IOP}_r \subset (\mathcal{I}_r \cup \mathcal{O}_r \cup \mathcal{D}_r \cup \mathcal{TR}_r)$: set with the initial operation to be performed by vessel $r \in \mathcal{RV}$.
   - $\mathcal{C}$: set of crude oils.
   - $\mathcal{K}$: set of crude oil properties.
   - $\mathcal{CL}$: set of cliques of conflicting operations. Let $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$ be a graph whose vertice set $\mathcal{V}_c = \mathcal{W}$ consists of all operations, and whose edge set $\mathcal{E}_c \subseteq \mathcal{V}_c \times \mathcal{V}_c$ corresponds to the conflicting operations. This means that two operations u and v cannot take place simultaneously if and only if $(u, v) \in \mathcal{E}_c$. Rather than expressing a constraint for each pair $(u, v) \in \mathcal{E}_c$, $\mathcal{CL}$ can be defined as the set of all maximal cliques which ensures a coverage of all conflicting constraints.

- $\mathcal{WCL}_{cl}$: set of operations in a clique *cl*. $\mathcal{WCL}_{cl} \in \mathcal{W}$ is the set of conflicting operations in a clique $cl \in \mathcal{CL}$.

2. **Parameters**

   The following parameters should be considered:

   - G_c: gross margin of crude oil $c \in \mathcal{C}$, in dollars per thousand barrels [$/10^3 bbl$].
   - $PROD_{r,c}$: production rate of crude oil $c \in \mathcal{C}$ in FPSO $r \in \mathcal{RF}$, in 10^3 barrels per day [10^3 bbl/day]. An FPSO *r* is capable of producing crude oil *c* only if $PROD_{r,c} > 0$.
   - $VTT_{r,v}$: number of periods taken for executing travel operation $v \in \mathcal{TR}_r$ associated to vessel $r \in \mathcal{RV}$.
   - [$\underline{FR}_v$, $\overline{FR}_v$]: flowrate lower and upper bounds for operation $v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT})$, in 10^3 barrels per day [10^3 bbl/day]. Bounds on the flowrate of crude oil are imposed when offloading an FPSO, unloading a vessel, in transfers between storage and charging tanks, and between charging tanks and CDUs.
   - [$\underline{CAP}_r$, $\overline{CAP}_r$]: capacity lower and upper bounds of resource $r \in \mathcal{R} \setminus \mathcal{RD}$, in 10^3 barrels [10^3 bbl].
   - TIL_r: initial level of crude oil in resource $r \in \mathcal{R} \setminus \mathcal{RD}$, in 10^3 barrels [10^3 bbl].
   - $CIL_{r,c}$: initial level of crude oil *c* in resource $r \in \mathcal{R} \setminus \mathcal{RD}$, in 10^3 barrels [10^3 bbl].
   - $PR_{k,c}$ is the weight fraction of property $k \in \mathcal{K}$ in crude oil $c \in \mathcal{C}$.
   - [$\underline{DEMC}_{v,k}$, $\overline{DEMC}_{v,k}$]: lower and upper bounds on the weight fraction of property *k* of the blend of crudes transferred during operation $v \in \mathcal{WD}$ from charging tanks to the CDUs. In other words, the weight fraction of property *k* in the blend of crudes flowing in operation *v*, from a charging tank to a CDU, must be within the bounds $\underline{DEMC}_{v,k}$ and $\overline{DEMC}_{v,k}$.
   - [$\underline{DEM}_r$, $\overline{DEM}_r$]: lower and upper bounds on the total volume of crude oil demanded by CDU $r \in \mathcal{RD}$ over the planning horizon, in 10^3 barrels [10^3 bbl].

3. **Decision Variables**

   Binary assignment and continuous operation-state variables are needed.

   a) Logistic Variables.

      - $z_{i,v} \in \{0,1\}$, $i \in \mathcal{T}$ and $v \in \mathcal{W}$. Operation variable $z_{i,v} = 1$ if operation *v* is assigned to be executed in period *i*. Otherwise, $z_{i,v} = 0$.

- $s_{i,r,v,u} \in \{0,1\}$, $i \in (\mathcal{T} \setminus \{PH\})$, $r \in \mathcal{RV}$ and $(v,u) \in \mathcal{VD}_r$. Flow variable $s_{i,r,v,u} = 1$ if vessel r flows from executing operation v in period i to executing operation u in period $i+1$. Otherwise, $s_{i,r,v,u} = 0$.

b) Level and Flow Variables.

- $vt_{i,v} \geq 0$, $i \in \mathcal{T}$ and $v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT})$. Variable $vt_{i,v}$ is the total volume of crude oil transferred in period i by operation v.
- $vct_{i,v,c} \geq 0$, $i \in \mathcal{T}$, $v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT})$ and $c \in \mathcal{C}$. Variable $vct_{i,v,c}$ is the volume of crude oil c transferred in period i by operation v.
- $lr_{i,r} \geq 0$, $i \in \mathcal{T}$ and $r \in \mathcal{R} \setminus \mathcal{RD}$. Variable $lr_{i,r}$ is the total level of crude oil in resource r at the end of period i.
- $lcr_{i,r,c} \geq 0$, $i \in \mathcal{T}$, $r \in \mathcal{R} \setminus \mathcal{RD}$ and $c \in \mathcal{C}$. Variable $lcr_{i,r,c}$ is the level of crude oil c in resource r at the end of period i.

3.4.2 Constraints

3.4.2.1 Material Balance and Resource Capacity

The set of equations from (44) to (51) track the total level of crude oil ($lr_{i,r}$) and the specific level of each crude oil $c \in \mathcal{C}$ ($lcr_{i,r,c}$) in all resources $r \in \mathcal{R} \setminus \mathcal{RD}$ (except from CDUs). These levels directly depend on the flow of crudes (i.e., defined by variables $vct_{i,v,c}$ and $vt_{i,v}$) associated to inlet and outlet operations on each resource (i.e., operations defined in sets $\mathcal{I}_r$ and $\mathcal{O}_r$). There is no need to track the inventory of CDUs, since it is assumed that the daily flow of crude oil to the distillation units are according to its processing capacity.

Eqs. (44) to (47) are related to the inventory control of FPSOs, where parameter $PROD_{r,c}$ is the fixed daily production rate of crude oil c at FPSO r. Note that $PROD_{r,c} = 0$ if FPSO r cannot produce crude oil $c \in \mathcal{C}$. Parameters $CIL_{r,c}$ and TIL_r in Eqs. (44) and (46) correspond, respectively, to the initial volume of crude oil c and the total initial volume of crude oil in resource r.

$$lcr_{i,r,c} = CIL_{r,c} + PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}, \ \forall\ r \in \mathcal{RF}, i \in \mathcal{T}, c \in \mathcal{C}, i = 1, \tag{44}$$

$$lcr_{i,r,c} = lcr_{i-1,r,c} + PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}, \ \forall\ r \in \mathcal{RF}, i \in \mathcal{T}, c \in \mathcal{C}, i \neq 1, \tag{45}$$

$$lr_{i,r} = TIL_r + \sum_{c \in \mathcal{C}} PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vt_{i,v}, \ \forall\ r \in \mathcal{RF}, i \in \mathcal{T}, i = 1, \tag{46}$$

$$lr_{i,r} = lr_{i-1,r} + \sum_{c \in \mathcal{C}} PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vt_{i,v}, \ \forall\ r \in \mathcal{RF}, i \in \mathcal{T}, c \in \mathcal{C}, i \neq 1. \tag{47}$$

Likewise, Eqs. (48) to (51) track the volume of crude oil at storage and charging tanks. The main difference from Eqs. (44) to (47) is that the volume of crude c associated to inlet operations $v \in \mathcal{I}_r$ in these resources are variables (i.e., $vct_{i,v,c}$ and $vt_{i,v}$), while at the FPSOs they correspond to the daily production parameter $PROD_{r,c}$.

$$lcr_{i,r,c} = CIL_{r,c} + \sum_{v \in \mathcal{I}_r} vct_{i,v,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}, \ \forall\, r \in \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, c \in \mathcal{C}, i = 1, \quad (48)$$

$$lcr_{i,r,c} = lcr_{i-1,r,c} + \sum_{v \in \mathcal{I}_r} vct_{i,v,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}, \ \forall\, r \in \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, c \in \mathcal{C}, i \neq 1, \quad (49)$$

$$lr_{i,r} = TIL_r + \sum_{v \in \mathcal{I}_r} vt_{i,v} - \sum_{v \in \mathcal{O}_r} vt_{i,v}, \ \forall\, r \in \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, i = 1, \quad (50)$$

$$lr_{i,r} = lr_{i-1,r} + \sum_{v \in \mathcal{I}_r} vt_{i,v} - \sum_{v \in \mathcal{O}_r} vt_{i,v}, \ \forall\, r \in \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, i \neq 1. \quad (51)$$

Eq. (52) states that the total level of crude oil in a resource r is the sum of the levels of each crude oil c in that resource. Moreover, each resource r is bounded by limits on its capacity (i.e., $\underline{CAP}_r$ and $\overline{CAP}_r$), which are imposed by Eq. (53).

$$lr_{i,r} = \sum_{c \in \mathcal{C}} lcr_{i,r,c}, \ \forall\, i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}, \quad (52)$$

$$\underline{CAP}_r \leq lr_{i,r} \leq \overline{CAP}_r, \ \forall\, i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}. \quad (53)$$

3.4.2.2 Vessel Operations Scheduling

Consider a graph $\mathcal{G}_r = (\mathcal{N}_r, \mathcal{E}_r)$ of flow between operations associated to vessel $r \in \mathcal{RV}$. The set of nodes $\mathcal{N}_r = \mathcal{I}_r \cup \mathcal{O}_r \cup \mathcal{D}_r \cup \mathcal{TR}_r$ represents all operations allowed for a vessel: waiting ($\mathcal{D}_r$), traveling ($\mathcal{TR}_r$), unloading ($\mathcal{O}_r$), and offloading ($\mathcal{I}_r$). Meanwhile, edges $(v, u) \in \mathcal{VD}_r \subset \mathcal{E}_r = \mathcal{N}_r \times \mathcal{N}_r$ indicate the precedence or allowed flow between operations.

Figure 11 illustrates the graphs for *Vessel1* and *Vessel2* for the instance in Fig. 10. Take *Vessel1* for example. If at time i the vessel is unloading crude oil into storage tank *ST1* (i.e., operation v11), then in i+1 the vessel can: travel to *FPSO1* (i.e., operation v5); travel to *FPSO2* (i.e., operation v7); continue to unload into *ST1*; unload into *ST2* (i.e., operation v13); or wait at the terminal (i.e., operation v9). Note that *Vessel1* can only offload *FPSO1* (i.e., operation v1) or *FPSO2* (i.e., operation v3) after traveling to them.

Eq. (54) defines the initial flow between operations v and u, where the initial operation v to be executed by vessel r at period i = 1 is defined in the set $\mathcal{IOP}_r$. For the first period of time, all other possible flows are set to zero, which are defined by Eq. (55).

$$\sum_{(v,u) \in \mathcal{VD}_r} s_{1,r,v,u} = 1, \ \forall\, r \in \mathcal{RV}, v \in \mathcal{IOP}_r, \quad (54)$$

$$s_{1,r,v,u} = 0, \ \forall\, r \in \mathcal{RV}, v \in \mathcal{N}_r \setminus \mathcal{IOP}_r, (v, u) \in \mathcal{VD}_r. \quad (55)$$

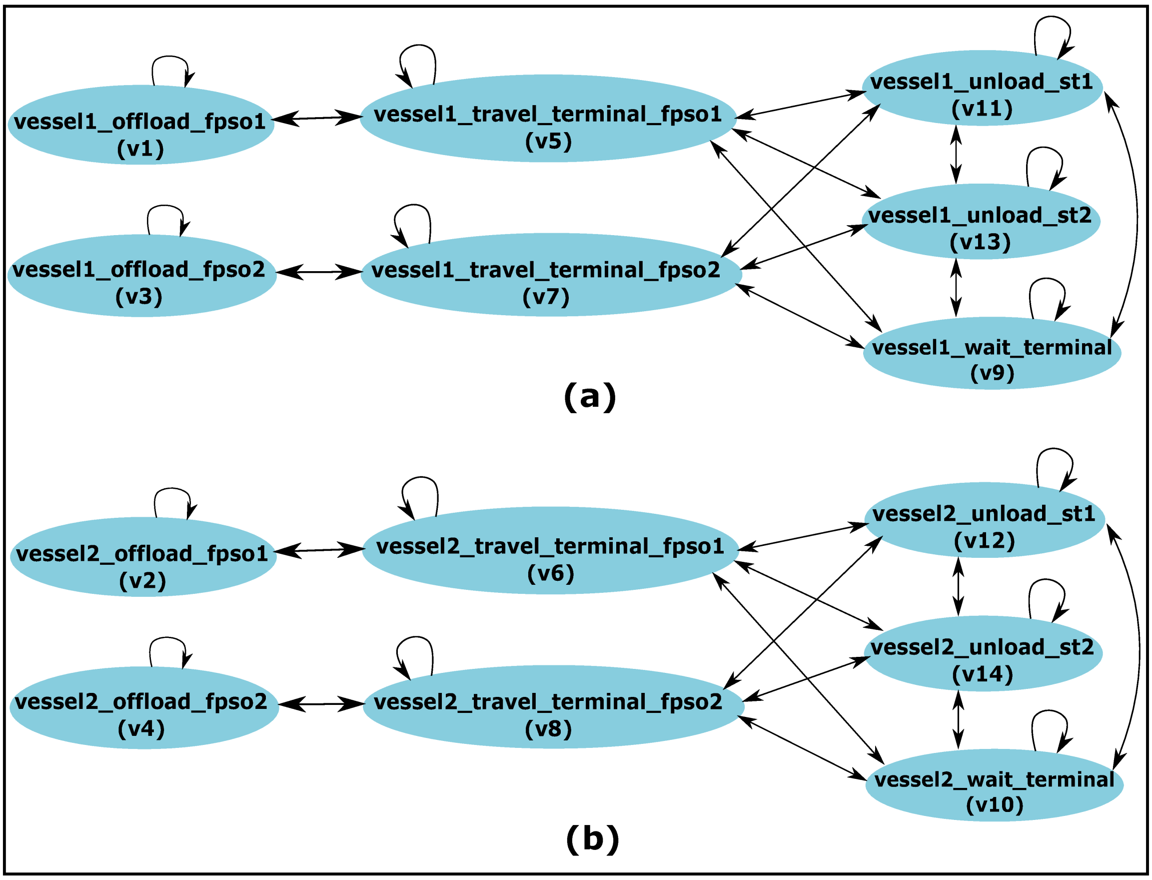


Figure 11 – Figures (a) and (b) illustrate the graphs for the flow of operations associated to *Vessel1* and *Vessel2* respectively.

The conservation of flow with respect to the graph is defined by Eq. (56).

$$\sum_{(v,u)\in\mathcal{VD}_r} s_{i,r,v,u} = \sum_{(u,v)\in\mathcal{VD}_r} s_{i+1,r,u,v}, \ \forall\ i \in \mathcal{T},\ r \in \mathcal{RV}, u \in \mathcal{N}_r, i \neq PH. \tag{56}$$

Eq. (57) states that if there is a flow in period i from operation v to operation u in $i+1$ (i.e., $s_{i,r,v,u} = 1$) then operation v must be executed in period i (i.e., $z_{i,v} = 1$).

$$\sum_{(v,u)\in\mathcal{VD}_r} s_{i,r,v,u} = z_{i,v}, \ \forall\ i \in \mathcal{T},\ r \in \mathcal{RV},\ v \in \mathcal{N}_r. \tag{57}$$

3.4.2.3 Vessel Travel Times

Travel operations $u \in \mathcal{TR}_r$ are required for each vessel r to move between FPSOs and the crude oil terminal. For each travel operation u, parameter $VTT_{r,u}$ defines the number of periods required for vessel r to perform the travel operation.

The dynamics of offshore trips of a vessel r (i.e., from the terminal to the FPSOs) is given by Eq. (58). Suppose that vessel r is performing an unloading or a waiting operation $v \in \mathcal{O}_r \cup \mathcal{D}_r$ at the terminal during period i. If the vessel r starts a traveling operation $u \in \mathcal{TR}_r$ at time $(i+1)$, then the vessel must arrive $VTT_{r,u}$ periods later at an

FPSO, in order to start the offloading of crude oil (i.e., performing operation $z \in \mathcal{I}_r$).

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r:\\ z\in\mathcal{I}_r}} s_{i+VTT_{r,u},r,u,z}, \ \forall\, i \in \mathcal{T},\ r \in \mathcal{RV}, \quad u \in \mathcal{TR}_r, i \leq PH - VTT_{r,u}. \quad (58)$$

After performing an offloading operation $v \in \mathcal{I}_r$ from an FPSO at period i, suppose that vessel r starts a travel operation $u \in \mathcal{TR}_r$ at time $(i + 1)$, as flagged by $s_{i,r,v,u} = 1$. Then, the vessel must arrive $VTT_{r,u}$ periods later at the terminal to begin an unloading or waiting operation $z \in \mathcal{O}_r \cup \mathcal{D}_r$ at period $(VTT_{r,u} + 1)$, which is defined by Eq. (59).

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in\mathcal{I}_r}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r:\\ z\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i+VTT_{r,u},r,u,z}, \ \forall\, i \in \mathcal{T},\ r \in \mathcal{RV}, \quad u \in \mathcal{TR}_r, i \leq PH - VTT_{r,u}. \quad (59)$$

3.4.2.4 Vessel Offloading and Unloading Rules

Eq. (60) states that once a vessel starts offloading from an FPSO, it will continue offloading until utilizing its total storage capacity. Suppose that vessel r executes a travel operation $v \in \mathcal{TR}_r$ at time i, and starts offloading an FPSO through operation $u \in \mathcal{I}_r$ at time $(i + 1)$, which is indicated by $s_{i,r,v,u} = 1$. If one such action takes place, then the right-hand side of Eq. (60) will be $\overline{CAP}_r$, forcing vessel r to fill its storage capacity by offloading from the FPSO at a rate $\overline{FR}_u$, from time $(i + 1)$ until time $\lceil \overline{CAP}_r/\overline{FR}_u \rceil$. Otherwise, if $s_{i,r,v,u} = 0$ for all pairs of operations (v, u) then Eq. (60) becomes innocuous.

$$\sum_{c\in\mathcal{C}} \sum_{t=(i+1)}^{i+\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \rceil} vct_{t,u,c} \geq \overline{CAP}_r - \overline{CAP}_r\Big(1 - \sum_{\substack{(v,u)\in\mathcal{VD}_r\\ v\in\mathcal{TR}_r}} s_{i,r,v,u}\Big), \quad \forall\, i \in \mathcal{T},\ r \in \mathcal{RV},\ \ u \in \mathcal{I}_r, i \leq PH - \left\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \right\rceil. \quad (60)$$

Moreover, after $\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \rceil$ periods of offloading, vessel r must start the return trip to the terminal, which is established by Eq. (61). If vessel r arrives at an FPSO at time i and starts offloading at time $(i + 1)$, which is indicated by $s_{i,r,v,u} = 1$, then the offloading must be completed at time $(i + \lceil \overline{CAP}_r/\overline{FR}_u \rceil)$, and the vessel must start a travel operation z at the following period of time.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in\mathcal{TR}_r}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r:\\ z\in\mathcal{TR}_r}} s_{i+\lceil\frac{\overline{CAP}_r}{\overline{FR}_u}\rceil,r,u,z}, \; \forall\, i \in \mathcal{T},\; r \in \mathcal{RV},$$
$$u \in \mathcal{I}_r, i \leq PH - \left\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \right\rceil. \tag{61}$$

Eq. (62) states that if a vessel r starts to unload ($v \in \mathcal{O}_r$) crude oil into a storage tank in period i, it can only start a waiting operation $u \in \mathcal{D}_r$ in period $i+1$ if the vessel is empty.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in\mathcal{O}_r}} s_{i,r,v,u} \leq \frac{\overline{CAP}_r - lr_{i,r}}{\overline{CAP}_r}, \; \forall\, i \in \mathcal{T},\; r \in \mathcal{RV}, u \in \mathcal{D}_r. \tag{62}$$

In the case that a vessel starts unloading crude oil in the terminal, waiting and travel operations from the terminal are allowed only after the vessel becomes empty, as enforced by Eq. (63). Assume that at period i a vessel r is unloading crude oil in a storage tank or waiting at the terminal, respectively, operations $v \in \mathcal{O}_r \cup \mathcal{D}_r$. Then, at period $(i+1)$, the vessel can initiate a travel operation $u \in \mathcal{TR}_r$ to offload an FPSO only with an empty tank. In this case, the total level of crude oil in the vessel is $lr_{i,r} = 0$ at time i, and therefore the right-hand side of Eq. (63) becomes $\overline{CAP}_r/\overline{CAP}_r = 1$, which allows $s_{i,r,v,u}$ to assume value 1 for a travel operation u at time $(i+1)$. Otherwise, if there is crude oil remaining in the vessel (i.e., $lr_{i,r} > 0$), then the right-hand side will be smaller than 1, forcing $s_{i,r,v,u} = 0$ for all variables on the left-hand side.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i,r,v,u} \leq \frac{\overline{CAP}_r - lr_{i,r}}{\overline{CAP}_r}, \; \forall\, i \in \mathcal{T},\; r \in \mathcal{RV}, u \in \mathcal{TR}_r. \tag{63}$$

3.4.2.5 Transfer Constraints

Eq. (64) determines that if a transfer operation $v \in \mathcal{W}\setminus(\mathcal{WW}\cup\mathcal{WT})$ is performed in period i, meaning $z_{i,v} = 1$, then the flow of crude oil between resources is bounded by the lower ($\underline{FR}_v$) and upper ($\overline{FR}_v$) bounds on the flowrate of each operation v. This constraint is valid for all operations, except waiting and traveling operations performed by vessels (i.e., $\mathcal{WW}$ and $\mathcal{WT}$), which do not involve transfer of crudes between resources. Eq. (65) states that the total volume of crude oil $vt_{i,v}$ transferred in operation v is the sum of the volumes $vct_{i,v,c}$ of all crudes c transferred in the same operation.

$$z_{i,v}\underline{FR}_v \leq vt_{i,v} \leq \overline{FR}_v z_{i,v}, \; \forall\, i \in \mathcal{T}, v \in \mathcal{W}\setminus(\mathcal{WW}\cup\mathcal{WT}), \tag{64}$$
$$vt_{i,v} = \sum_{c\in\mathcal{C}} vct_{i,v,c}, \; \forall\, i \in \mathcal{T}, v \in \mathcal{W}\setminus(\mathcal{WW}\cup\mathcal{WT}). \tag{65}$$

As discussed in Section 3.3, blending of different types of crudes takes place in storage and charging tanks. This means that a total level $lr_{i,r}$ of crude oil, and a specific level $lcr_{i,r,c}$ of each crude type c, is associated with these resources $r \in \mathcal{RS} \cup \mathcal{RC}$. Likewise, a total volume $vt_{i,v}$ of crude oil and a specific volume $vct_{i,v,c}$ of each crude type c must be accounted, for every transfer operation $v \in \mathcal{O}_r$ outleting a storage or charging tank r. Note that the proportion $lcr_{i,r,c}/lr_{i,r}$ inside each resource r and the proportion $vct_{i,v,c}/vt_{i,v}$ in each transfer operation v must be the same for composition consistency. This condition enforces that crude oil compositions inside storage, and charging tanks, remain the same when batches of crude oil are transferred between storage and charging tanks, and between charging tanks and CDUs. The blending condition is imposed by Eq. (66), which is the only constraint in the model that involves non-linear non-convex terms.

$$\frac{vct_{i,v,c}}{vt_{i,v}} = \frac{lcr_{i,r,c}}{lr_{i,r}} \Rightarrow vct_{i,v,c} lr_{i,r} = vt_{i,v} lcr_{i,r,c}, \ \forall\, i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r, c \in \mathcal{C}. \tag{66}$$

3.4.2.6 CDUs

CDUs are set up to work over determined operating ranges of crude oil composition, which translates into a feasible range for each property k of the crude oil transferred to the CDU. Put another way, the flow of an operation $v \in \mathcal{WD}$ from a charging tank r to the CDU must have its property k within the range $[\underline{DEMC}_{v,k},\ \overline{DEMC}_{v,k}]$, a condition enforced by Eq. (67). Given that an operation v defines the transfer of crude from a charging tank to a CDU, the bounds $\underline{DEMC}_{v,k}$ and $\overline{DEMC}_{v,k}$ are defined according with the CDU. Parameter $PR_{k,c}$ defines the weight fraction of property k associated to crude c.

$$\underline{DEMC}_{v,k} vt_{i,v} \leq \sum_{c \in \mathcal{C}} vct_{i,v,c} PR_{k,c} \leq \overline{DEMC}_{v,k} vt_{i,v}, \ \forall\, i \in \mathcal{T},\ v \in \mathcal{WD},\ k \in \mathcal{K}. \tag{67}$$

Besides restrictions on composition of crudes, Eq. (68) states that over the planning horizon the total volume of crude oil demanded from each charging tank r by the CDUs is bounded by $[\underline{DEM}_r,\ \overline{DEM}_r]$.

$$\underline{DEM}_r \leq \sum_{i \in \mathcal{T}} \sum_{v \in \mathcal{O}_r} vt_{i,v} \leq \overline{DEM}_r, \ \forall\, r \in \mathcal{RC}. \tag{68}$$

Finally, the CDUs must receive crude oil from charging tanks in all periods over the planning horizon.

$$\sum_{v \in \mathcal{I}_r} z_{i,v} = 1, \ \forall\, i \in \mathcal{T}, r \in \mathcal{RD}. \tag{69}$$

3.4.2.7 Operational Rules

Certain sets of operations cannot be performed during the same period of time due to logistic rules inherent to the problem. In Fig. 10 for instance:

- *Vessel1* and *Vessel2* can offload *FPSO1* performing operations $v1$ and $v2$, respectively. Let set FPSO-OFFLOAD = $\{v1, v2\}$.
- *CDU1* can receive loads of crude oil from charging tanks *CT1* and *CT2* through operations $v19$ and $v20$, respectively. Let set CDU1-INPUT = $\{v19, v20\}$.

In both sets FPSO-OFFLOAD and CDU1-INPUT the operations cannot be performed in the same time period. In other words, *FPSO1* has only one pump to transfer crude oil into the vessels and *CDU1* can only receive streams of oil from one charging tank at a time. For this example, let the set of cliques be $\mathcal{CL}$ = {FPSO-OFFLOAD, CDU1-INPUT} and let set $\mathcal{WCL}_{cl}$ contain the operations of each clique $cl \in \mathcal{CL}$. Eq. (70) guarantees that at most one operation $v \in \mathcal{WCL}_{cl}$ will be performed in a time period.

$$\sum_{v \in \mathcal{WCL}_{cl}} z_{i,v} \leq 1, \; \forall \, i \in \mathcal{T}, \; cl \in \mathcal{CL}. \tag{70}$$

3.4.3 Nonconvex Discrete Time MINLP Formulation

Having introduced the notation and constraints, the operational management of crude oil supply is cast as:

$$P: \; \max \; f = \sum_{i \in \mathcal{T}} \sum_{r \in \mathcal{RD}} \sum_{v \in \mathcal{I}_r} \sum_{c \in \mathcal{C}} G_c vct_{i,v,c} \tag{71a}$$

$$\begin{aligned}
\text{s.t.}: \; & (44)\text{-}(70), && (71b)\\
& z_{i,v} \in \{0,1\}, \; \forall \, i \in \mathcal{T}, v \in \mathcal{W}, && (71c)\\
& s_{i,r,v,u} \in \{0,1\}, \; \forall \, i \in (\mathcal{T} \setminus \{PH\}), r \in \mathcal{RV}, (v,u) \in \mathcal{VD}_r, && (71d)\\
& vt_{i,v} \geq 0, \; \forall \, i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}) && (71e)\\
& vct_{i,v,c} \geq 0, \; \forall \, i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}), c \in \mathcal{C}, && (71f)\\
& lr_{i,r} \geq 0, \; \forall \, i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}, && (71g)\\
& lcr_{i,r,c} \geq 0, \; \forall \, i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}, c \in \mathcal{C}. && (71h)
\end{aligned}$$

The problem of concern is a discrete-time mixed-integer nonlinear program (MINLP). All the constraints in model (71) are linear except the compositional Eq. (66), which involves bilinear terms.

3.5 SOLUTION STRATEGY

This section details the proposed solution strategy (see Fig. 12), which consists of an iterative MILP-NLP decomposition with domain contraction:

- First, the approach solves the MILP relaxation of formulation (71), in which Eq. (66) is dropped and replaced by McCormick envelopes, providing an upper bound to the MINLP problem. Since the problem addressed in this work involves elements of two known complex scheduling problems (i.e., maritime inventory routing and crude oil scheduling), depending on the instance, the solution time may increase exponentially. We make use of two options available in solvers to overcome this issue while solving the relaxation: TIMELIMIT[1] and Relative MIP Gap Tolerance (MIPGAP)[2].
- We then fix the logistics decisions (i.e., binary variables) from the MILP into the MINLP, yielding into a continuous non-linear program (NLP). Its solution yields a lower bound to the MINLP problem.
- After finding MILP and NLP solutions, the domain of variables associated to bilinear terms is contracted for the next iteration.
- Finally, the iterative process ends in the following cases: (a) if there is no improvement on the NLP bound between successive iterations; (b) if at any part of the process the MILP or NLP solutions are infeasible (e.g., NLP is infeasible after fixing binaries from the MILP); (c) if the maximum number of iterations is achieved.

Next, the MILP relaxation and the domain contraction are explained.

3.5.1 MILP Relaxation

It is known in the literature that the tightest possible linear relaxation of the bilinear term $x_i x_j$ is given by the McCormick envelopes (MCCORMICK, 1976). The relaxation consists of linear envelopes, built over the domain of the variables x_i and x_j, which encloses the non-convex function $x_i x_j$. This can be done by replacing $x_i x_j$ by continuous variable $w_{i,j}$ and considering four sets of linear inequalities to the formulation. These inequalities relate variable $w_{i,j}$ with x_i and x_j, and their lower and upper bounds.

A tighter relaxation can be obtained by partitioning the domain of one variable (CASTRO, 2015) or both (WICAKSONO; KARIMI, IA, 2008) into intervals. Envelopes are constructed in each interval and additional binary variables are included in the

[1] Maximum CPU time allowed for solving the MILP relaxation.

[2] Relative tolerance between the best integer solution and the best bound for optimizing the MILP relaxation.

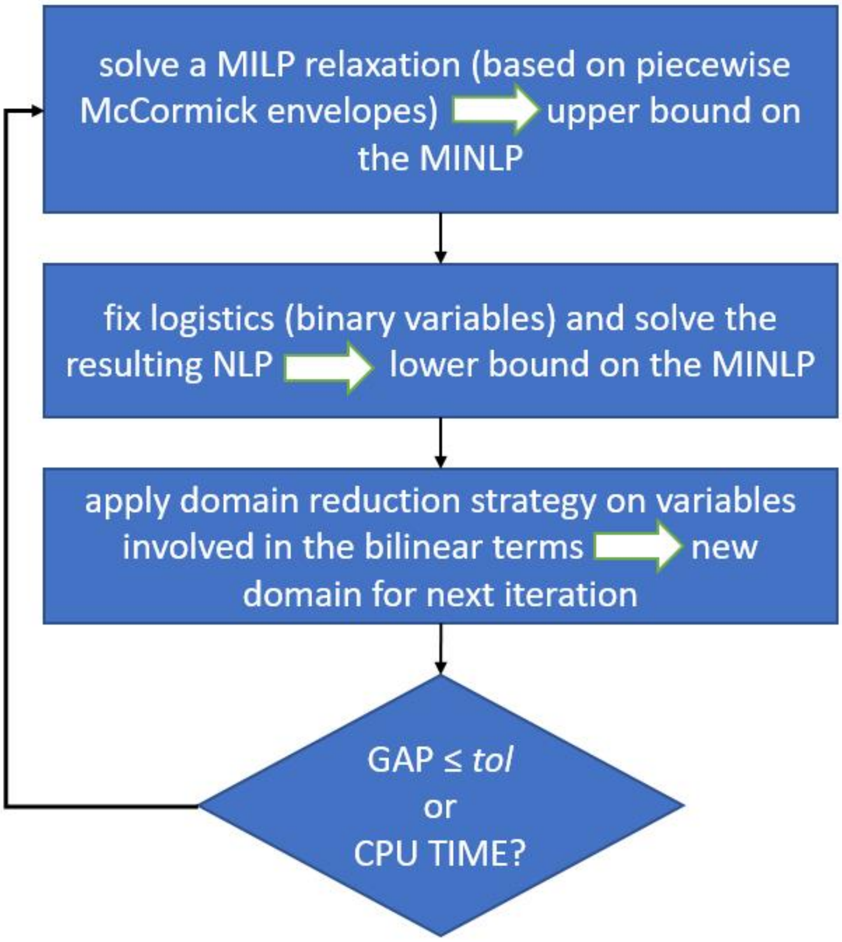


Figure 12 – Solution strategy diagram.

model in order to select the partition that provides the best relaxation. The improved relaxation is referred to as piecewise McCormick with univariate partitioning (i.e., partition the domain of just one variable) or bivariate partitioning (i.e, partition the domain of both variables). More details on estimators for bilinear terms can be found in the work of Gounaris et al. (2009).

Reefer to Sec. 2.5 and Figs. 5 and 6, which illustrate in more details how the relaxation of the bilinear terms is constructed.

Table 14 – Sets, variables, and parameters for the disjunctive formulation.

Sets, Variables, and Parameters	
$[\underline{LCR}_{i,r,c}, \overline{LCR}_{i,r,c}]$	$\underline{LCR}_{i,r,c} = 0$ and $\overline{LCR}_{i,r,c} = \overline{CAP}_r$ are the bounds for variable $lcr_{i,r,c}$.
$[\underline{VT}_{i,v}, \overline{VT}_{i,v}]$	$\underline{VT}_{i,v} = 0$ and $\overline{VT}_{i,v} = \overline{FR}_v$ are the bounds for variable $vt_{i,v}$.
$\mathcal{P} = \{1, \ldots, n\}$	Set of domain partitions for variable $lcr_{i,r,c}$ (index p).
$\mathcal{Q} = \{1, \ldots, m\}$	Set of domain partitions for variable $vt_{i,v}$ (index q).
$[\underline{LCR}_{i,r,c,p}, \overline{LCR}_{i,r,c,p}]$	Bounds of each partition p of variable $lcr_{i,r,c}$.
$[\underline{VT}_{i,v,q}, \overline{VT}_{i,v,q}]$	Bounds of each partition q of variable $vt_{i,v}$.
$y^{\text{RHS}}_{i,r,v,c,q,p}$	$y^{\text{RHS}}_{i,r,v,c,q,p} \in \{\text{True}, \text{False}\}$ indicates the selected partition $[q, p]$.

One can make use of Generalized Disjunctive Programming (GDP) (TRESPALACIOS; GROSSMANN, 2014) for modeling the relaxation of bilinear terms through piecewise McCormick envelopes. In order to limit the size of this section, the description of the formulation is only presented for the right-hand bilinear term of Eq. (66), namely $\eta^{\text{RHS}}_{i,r,v,c} = vt_{i,v}lcr_{i,r,c}$. Table 14 describes all required sets, variables and parameters to build the relaxation in terms of disjunctions. As seen in Eq. (72) envelopes are constructed in each partition $[q,p]$. Eq. (73) imposes that only partition $[q,p]$ will be chosen. Lower and upper bounds of each partition are defined by Eqs. (74) and (75). The following equations are defined for all $i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}$.

$$\bigvee_{q=1}^{Q}\bigvee_{p=1}^{P}\left[\begin{array}{c} y^{\text{RHS}}_{i,r,v,c,q,p} \\ \eta^{\text{RHS}}_{i,r,v,c} \geq \underline{VT}_{i,v,q}lcr_{i,r,c} + \underline{LCR}_{i,r,c,p}vt_{i,v} - \underline{VT}_{i,v,q}\underline{LCR}_{i,r,c,p} \\ \eta^{\text{RHS}}_{i,r,v,c} \geq \overline{VT}_{i,v,q}lcr_{i,r,c} + \overline{LCR}_{i,r,c,p}vt_{i,v} - \overline{VT}_{i,v,q}\overline{LCR}_{i,r,c,p} \\ \eta^{\text{RHS}}_{i,r,v,c} \leq \overline{VT}_{i,v,q}lcr_{i,r,c} + \underline{LCR}_{i,r,c,p}vt_{i,v} - \overline{VT}_{i,v,q}\underline{LCR}_{i,r,c,p} \\ \eta^{\text{RHS}}_{i,r,v,c} \leq \underline{VT}_{i,v,q}lcr_{i,r,c} + \overline{LCR}_{i,r,c,p}vt_{i,v} - \underline{VT}_{i,v,q}\overline{LCR}_{i,r,c,p} \\ \underline{LCR}_{i,r,c,p} \leq lcr_{i,r,c} \leq \overline{LCR}_{i,r,c,p} \\ \underline{VT}_{i,v,q} \leq vt_{i,v} \leq \overline{VT}_{i,v,q} \end{array}\right], \tag{72}$$

$$\underset{q=1}{\overset{Q}{\veebar}}\ \underset{p=1}{\overset{P}{\veebar}}\ y^{\text{RHS}}_{i,r,v,c,q,p}, \tag{73}$$

$$\left\{\begin{array}{l} \underline{LCR}_{i,r,c,p} = \underline{LCR}_{i,r,c} + \frac{(\overline{LCR}_{i,r,c}-\underline{LCR}_{i,r,c})(p-1)}{|\mathcal{P}|} \\ \overline{LCR}_{i,r,c,p} = \underline{LCR}_{i,r,c} + \frac{(\overline{LCR}_{i,r,c}-\underline{LCR}_{i,r,c})p}{|\mathcal{P}|} \end{array}\right. \quad \forall\, p \in \mathcal{P}, \tag{74}$$

$$\left\{\begin{array}{l} \underline{VT}_{i,v,q} = \underline{VT}_{i,v} + \frac{(\overline{VT}_{i,v}-\underline{VT}_{i,v})(q-1)}{|\mathcal{Q}|} \\ \overline{VT}_{i,v,q} = \underline{VT}_{i,v} + \frac{(\overline{VT}_{i,v}-\underline{VT}_{i,v})q}{|\mathcal{Q}|} \end{array}\right. \quad \forall\, q \in \mathcal{Q}, \tag{75}$$

$$y^{\text{RHS}}_{i,r,v,c,q,p} \in \{\text{True}, \text{False}\},\ \forall\, q \in \mathcal{Q},\ p \in \mathcal{P}. \tag{76}$$

There are two common methods for transforming a GDP into an MILP, namely big-M and convex hull reformulations (BALAS, 1985; GROSSMANN; TRESPALACIOS, 2013). Wicaksono and IA Karimi (2008) stressed out the poor quality of solutions provided by big-M. Therefore, the convex hull reformulation is chosen as in Castro (2015) and Leonardo Salsano de Assis et al. (2017). Please refer to Appendix A, which describes in details the resulting set of equations for the bilinear term $\eta^{\text{RHS}}_{i,r,v,c} = vt_{i,v}lcr_{i,r,c}$. Similar equations are needed for the bilinear term $\eta^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c}lr_{i,r}$.

The MILP relaxation is constructed as follows: problem P, without the blending equation (66); with the additional MILP constraints originated from the disjunctive reformulation for both bilinear terms $vt_{i,v}lcr_{i,r,c}$ (see Appendix A) and $vct_{i,v,c}lr_{i,r}$; and Eq. (77), which enforces Eq. (66).

$$\eta^{\text{LHS}}_{i,r,v,c} = \eta^{\text{RHS}}_{i,r,v,c},\ \forall\, i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}. \tag{77}$$

The choice on the type of relaxation (see Table 15) depends on how the sets $\mathcal{P}$ and $\mathcal{Q}$ (and sets for the left-hand side bilinear term) are handled. For instance, the relaxation scheme SME means having sets with cardinality $|\mathcal{P}| = 1$ and $|\mathcal{Q}| = 1$. On the other hand, PWL_n would have sets with cardinality $|\mathcal{P}| = n$ and $|\mathcal{Q}| = 1$ (i.e., univariate partitioning). Bivariate partitioning (i.e., $\text{PWB}_{n,m}$) would have sets with cardinality $|\mathcal{P}| = n$ and $|\mathcal{Q}| = m$. An iterative approach, such as IPWL_n, would increase the size of set $\mathcal{P}$ as iterations proceed, starting with $|\mathcal{P}| = n$.

Table 15 – Relaxation schemes for bilinear terms $vct_{i,v,c}lr_{i,r}$ and $vt_{i,v}lcr_{i,r,c}$.

Relax. Sche.	Reference
SME	Standard McCormick envelopes.
PWL_n	Piecewise McCormick envelopes on level variables (i.e., $lr_{i,r}$ and $lcr_{i,r,c}$).
PWT_n	Piecewise McCormick envelopes on transfer variables (i.e., $vt_{i,v}$ and $vct_{i,v,c}$).
$\text{PWB}_{n,m}$	Piecewise McCormick envelopes on all variables.
IPWL_n	Increases the number of envelopes on level variables at each iteration.
IPWT_n	Increases the number of envelopes on transfer variables at each iteration.
$\text{IPWB}_{n,m}$	Increases the number of envelopes on all variables at each iteration.

3.5.2 Domain Contraction

The contraction of domains at iteration *it* consists of the following steps:

(a) divide the domain of each bilinear term into $n^{\text{grid}} \times m^{\text{grid}}$ grids (see Fig. 13(a)). The number of grids should be larger than the number of partitions for bilinear terms in order to have a finer discretization of the variable domains (i.e., $n^{\text{grid}} > n$ and $m^{\text{grid}} > m$).

(b) Identify the position of the variables $[vt_{i,v},\ lcr_{i,r,c}]^{(it)}_{\text{MILP}}$ defined by the MILP solution, and the position of $[vt_{i,v},\ lcr_{i,r,c}]^{(it)}_{\text{NLP}}$ defined by the NLP solution, inside the grid (see Fig. 13(a));

(c) obtain the contracted domain for the partition of the bilinear terms for iteration $(it+1)$ as the smallest region that encloses both solutions (the region indicated with the color gray in Fig. 13(b)). For instance, update the bounds $\underline{VT}^{(it+1)}_{i,v}$, $\overline{VT}^{(it+1)}_{i,v}$, $\underline{LCR}^{(it+1)}_{i,r,c}$, and $\overline{LCR}^{(it+1)}_{i,r,c}$ for the variables associated with the bilinear terms $\eta^{\text{RHS}}_{i,r,v,c}$, yielding a contracted domain at the next iteration (see Fig. 13(c));

The same procedure must be applied to the bilinear term $\eta^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c}lr_{i,r}$. Section 3.6 discusses the influence of the number of grids on problem solution.

3.5.3 Formalization of the Solution Strategy

Herein, all the elements of the solution strategy are brought together in a more structured form. Algorithm 1 formalizes the steps of the solution strategy that were

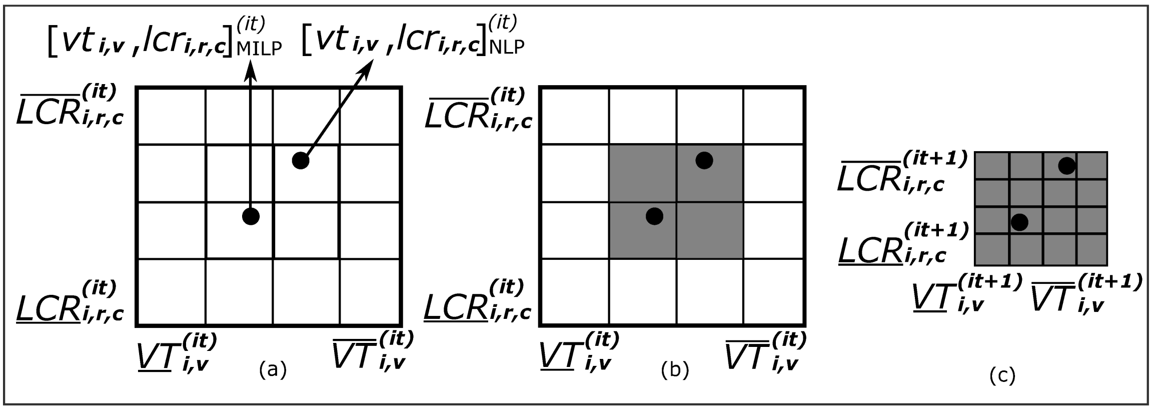


Figure 13 – Domain contraction schemes for the domain of bilinear term $\eta^{RHS}_{i,r,v,c} = vt_{i,v} lcr_{i,r,c}$.

discussed above[3].

3.6 ANALYSIS

3.6.1 Problem Instances

Based on the works of Rocha et al. (2009), Fraga et al. (2009) and Mouret et al. (2009), a set of instances for the operational management of crude oil supply was put together for the purpose of computational analysis. The model was implemented in AMPL (FOURER et al., 2003), optimized with CPLEX (IBM, 2013) and CONOPT (DRUD, 1985), and solved in a computer with two Intel Core Xeon E5-2630 v4 Processor (2.20 GHz), totaling 20 cores of 2 threads, 64 GB of RAM and a Ubuntu environment. The total number of variables and constraints, binary variables and non-linear constraints are given in Table 16. Consider 2F-2V-2ST-2CT-1CDU-2C-1P-15D for referring to an instance with: (a) 2 FPSOs; (b) 2 vessels; (c) 2 storage tanks; (d) 2 charging tanks; (e) 1 CDU; (f) 2 types of crude oil; (g) 1 property; and (h) 15 days of planning horizon.

Figure 22 (a), illustrates the network and the parameters, previously defined in Section 3.4, for instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D. In this instance, *FPSO*1 and *FPSO*2 produce crude *cA* and *cB* with production rates of 130 kbbl/d and 110 kbbl/day respectively. Also, both initial volumes are 500 kbbl of crude oil. Vessels are allowed to travel to all platforms and unload crude oil into all storage tanks. *Vessel*1 has an initial volume of 300 kbbl of crude *cA* and its initial operation is to unload crude into *ST*1. On the other hand, in the beginning of the planning horizon, *Vessel*2 is waiting at the terminal to unload its initial cargo of 500 kbbl of crude *cB*.

The initial volumes and crude types in storage and charging are indicated in Fig. 22 (a). All STs are allowed to feed all CTs, and similarly, all CTs can supply crude oil

[3] In case that a solution to the MILP relaxation or the NLP problem is not reached, the algorithm will halt with the best primal solution found thus far. This control policy was omitted to keep the code as simple as possible.

Algorithm 1: Iterative MILP-NLP Decomposition

Input: Initial conditions of resources, data defining problem instance (i.e., Tables 17 and 18), sets $\mathcal{IOP}_r$ of initial vessel operations, and *MaxIter*.

Output: The best decision variables $\theta_{NLP} = (z_{i,v}, s_{i,r,v,u}, vt_{i,v}, vct_{i,v,c}, lr_{i,r}, lcr_{i,r,c})$ yielded by the NLP problem and its objective value f_{NLP}.

- $it := 0$;
- $\theta^{(0)} := \emptyset$, $f^{(0)} := 0$; `/* Initial values for primal solution. */`
- Define initial bounds $\underline{VT}^{(0)}_{i,v} := 0$, $\overline{VT}^{(0)}_{i,v} := \overline{FR}_v$, $\underline{LCR}^{(0)}_{i,r,c} := 0$, $\overline{LCR}^{(0)}_{i,r,c} := \overline{CAP}_r$ `/* The bounds for the left-hand side of the bilinear terms` $\eta^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} lr_{i,r}$ `are defined likewise. */`
- Choose the relaxation scheme as in Table 15 and the number of partitions n and m;
- Choose the number of partitions n^{grid} and m^{grid} for domain reduction;

repeat

  - Define the bounds $\underline{VT}^{(it)}_{i,v,q}$, $\overline{VT}^{(it)}_{i,v,q}$, $\underline{LCR}^{(it)}_{i,r,c,p}$, $\overline{LCR}^{(it)}_{i,r,c,p}$ of each partition $[q, p]$ of bilinear terms $\eta^{\text{RHS}}_{i,r,v,c}$ as in Eqs. (74) and (75); `/* Likewise, define the bounds for the bilinear terms` $\eta^{\text{LHS}}_{i,r,v,c}$. `*/`
  - Solve the MILP relaxation;
  - Fix the logistics decisions $z_{i,v}$ and $s_{i,r,v,u}$ to obtain an NLP problem;
  - Solve the NLP problem to obtain a primal solution $\theta^{(it)}$ for problem P given in (71);
  - **If** $it \geq 1$ and $f^{(it)} < f^{(it-1)}$ **then break**;
  - Divide the domain of bilinear terms $\eta^{\text{RHS}}_{i,r,v,c}$ into grids as in Figure 13(a);
  - Identify the position of the variables $[vt_{i,v},\ lcr_{i,r,c}]^{(it)}_{\text{MILP}}$ in the MILP solution, and the position of $[vt_{i,v},\ lcr_{i,r,c}]^{(it)}_{\text{NLP}}$ in the NLP solution, inside the grid (see Fig. 13(a));
  - Contract the variable domain to the smallest region that encloses both solutions (Figure 13 (b)), for all bilinear terms $\eta^{\text{RHS}}_{i,r,v,c}$ and $\eta^{\text{LHS}}_{i,r,v,c}$;
  - Update the bounds $\underline{VT}^{(it+1)}_{i,v}$, $\overline{VT}^{(it+1)}_{i,v}$, $\underline{LCR}^{(it+1)}_{i,r,c}$, $\overline{LCR}^{(it+1)}_{i,r,c}$ at the next iteration $(it + 1)$ for the variables $vt_{i,v}$ and $lcr_{i,r,c}$ of $\eta^{\text{RHS}}_{i,r,v,c}$ (see Fig. 13(c));
  - Perform the same domain contraction for the variables $vct_{i,v,c}$ and $lr_{i,r}$ of $\eta^{\text{LHS}}_{i,r,v,c}$;
  - $it := it + 1$;

until *iter > MaxIter*;

- **return** $(\theta_{\text{NLP}} := \theta^{(it-1)}, f_{\text{NLP}} := f^{(it-1)})$;

Table 16 – Instances statistics.

Instances	Total Vars.	Total Cons.	Binary Vars.	Non-Linear Cons.
2F-2V-2ST-2CT-1CDU-2C-1P-15D	2 160	2 483	1 170	180
2F-2V-2ST-2CT-1CDU-2C-1P-15D	17 490	11 593	8 310	1 920
4F-4V-10ST-6CT-5CDU-8C-1P-15D	29 175	15 954	15 000	2 880

to the CDU. Notice that above each CT there are the bounds on property S that the blend of crudes inside the tank must be in order to satisfy the CDU's request. Finally, the planning horizon is 15 days.

The data associated to instances 4F-4V-6ST-4CT-3CDU-8C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D can be found in Tables 17 and 18, respectively. Also, these instances are illustrated in Figs. 23 and 24.

Table 17 – Data for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D.

Crudes	Sulfur Conc. [Weight Frac.]	Gross Margin [\$/10^3 *bbl*]
Crude A	0.010	10 000
Crude B	0.030	8 000
Crude C	0.045	6 500
Crude D	0.060	5 000
Crude E	0.012	9 500
Crude F	0.026	8 500
Crude G	0.041	7 000
Crude H	0.056	5 500

Flow Rates	Bounds [10^3 *bbl/day*]
FPSO-Vessel	[0, 500]
Vessel-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

STs	Init. Level [10^3 *bbl*]	Init. Crude
ST1	350	Crude A
ST2	0	-
ST3	350	Crude C
ST4	0	-
ST5	350	Crude D
ST6	0	-

Vessels	Init. Level [10^3 *bbl*]	Init. Crude
Vessel1	300	Crude A
Vessel2	500	Crude B
Vessel3	0	-
Vessel4	0	-

Plan. Horizon	**Discret.**
15 days	1 day

CTs	Init. Level [10^3 *bbl*]	Init. Crude	Bounds Sulfur Conc.	Demand [10^3 *bbl*]
CT1	500	Crude E	[0.005, 0.015]	[800, 1200]
CT2	500	Crude F	[0.020, 0.030]	[800, 1200]
CT3	500	Crude G	[0.035, 0.045]	[800, 1200]
CT4	500	Crude H	[0.050, 0.060]	[800, 1200]

FPSOs	Init. Level [10^3 *bbl*]	Produced Crude	Prod. Rate [10^3 *bbl/day*]	Travel Time [*day*]
FPSO1	500	Crude A	130	1
FPSO2	500	Crude B	110	2
FPSO3	1000	Crude C	100	1
FPSO4	1000	Crude D	110	1

Resource Capacity Bounds	FPSOs [10^3 *bbl*]	Vessels [10^3 *bbl*]	STs [10^3 *bbl*]	CTs [10^3 *bbl*]
	[200, 1500]	[0, 1000]	[0, 1000]	[0, 1000]

Vessels Initial Operation	Vessel1	Vessel2	Vessel3	Vessel4
	Unload-ST1	Wait-Terminal	Wait-Terminal	Wait-Terminal

3.6.2 Computational Results

The first strategy to tackle the problem instances is the use of global solvers. We made use of solvers BARON, SCIP and COUENNE, and defined a maximum solving time of 10 hours. For instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D, BARON presented the best performance, finding the optimal solution with an objective of \$22 800 10^3 in 2.15 hours. SCIP also found the optimal solution, however in 2.48 hours. COUENNE, on the other hand, found a feasible solution with an objective of \$22 679 10^3 in 10 hours. For instances 4F-4V-6ST-4CT-3CDU-8C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D, and a maximum time of 10 hours, the solvers did not find a feasible solution.

Table 19 presents the computational results of the instances in Table 16. The first column indicates the instance of concern, while the next three columns define the MIPGAP, the number of grids for each variable [$vct_{i,v,c}$, $lr_{i,r}$, $vt_{i,v}$, $lcr_{i,r,c}$] and the relaxation scheme (see Table 15 for details). Next, the best value for the MILP relaxation and the problem solution after convergence (i.e., NLP solution) are presented. Finally, we display the gap in % between MILP and NLP solutions, the number of iterations and total CPU time in seconds of the iterative process. For each iteration, a maximum time of 10 hours is defined.

Table 18 – Data for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D.

Crudes	Sulfur Conc. [Weight Frac.]	Gross Margin [\$/10^3 *bbl*]
Crude A	0.010	10 000
Crude B	0.030	8 000
Crude C	0.045	6 500
Crude D	0.060	5 000
Crude E	0.012	9 500
Crude F	0.026	8 500
Crude G	0.041	7 000
Crude H	0.056	5 500

Flow Rates	Bounds [10^3 *bbl*/*day*]
FPSO-Vessel	[0, 500]
Vessel-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

Plan. Horizon	**Discret.**
15 days	1 day

STs	Init. Level [10^3 *bbl*]	Init. Crude
ST1	100	Crude A
ST2	100	Crude A
ST3	100	Crude A
ST4	100	Crude B
ST5	100	Crude B
ST6	100	Crude C
ST7	100	Crude C
ST8	100	Crude D
ST9	100	Crude D
ST10	100	Crude D

Vessels	Init. Level [10^3 *bbl*]	Init. Crude
Vessel1	300	Crude A
Vessel2	500	Crude B
Vessel3	0	-
Vessel4	0	-

CTs	Init. Level [10^3 *bbl*]	Init. Crude	Bounds Sulfur Conc.	Demand [10^3 *bbl*]
CT1	500	Crude E	[0.005, 0.015]	[800, 1200]
CT2	500	Crude E	[0.005, 0.015]	[800, 1200]
CT3	500	Crude F	[0.020, 0.030]	[800, 1200]
CT4	500	Crude G	[0.035, 0.045]	[800, 1200]
CT5	500	Crude H	[0.050, 0.060]	[800, 1200]
CT6	500	Crude H	[0.050, 0.060]	[800, 1200]
FPSOs	Init. Level [10^3 *bbl*]	Produced Crude	Prod. Rate [10^3 *bbl*/*day*]	Travel Time [*day*]
FPSO1	500	Crude A	130	1
FPSO2	500	Crude B	110	2
FPSO3	1000	Crude C	100	1
FPSO4	1000	Crude D	110	1
Resource Capacity Bounds	FPSOs [10^3 *bbl*]	Vessels [10^3 *bbl*]	STs [10^3 *bbl*]	CTs [10^3 *bbl*]
	[200, 1500]	[0, 1000]	[0, 1000]	[0, 1000]
Vessels Initial Operation	Vessel1 Unload-ST1	Vessel2 Wait-Terminal	Vessel3 Wait-Terminal	Vessel4 Wait-Terminal

The results show that small instances (i.e., 2F-2V-2ST-2CT-1CDU-4C-1P-15D) do not need the use of MIPGAP other than 0% since the MILP-NLP decomposition strategy is able to solve the problem almost to optimality in less than 3 minutes, while BARON and SCIP take more than 2 hours. The solutions of all relaxation schemes are consistent and provide good initial points for the NLP problems, which find feasible solutions with small gaps. Nevertheless, bivariate schemes (i.e., partitioning the domain of all variable $vct_{i,v,c}$, $lr_{i,r}$, $vt_{i,v}$, and $lcr_{i,r,c}$) usually take much longer since a number of new variables and constraints are added to the MILP problem.

Looking at instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D, it is possible to observe how much difference there is on the CPU time when using MIPGAP=0% and MIPGAP = 1.5%. Figs. 14 and 15 illustrate that difference by reporting CPU time and solutions of several relaxation schemes compared to the best bound found for the instance (i.e., black horizontal line with value of \$38 857.9 thousand). One can see that the solution quality is not affected that much, with an average gap for MIPGAP = 0% about 1.5% and about 2% for MIPGAP = 1.5%. Notice that an outlier result of 0.75% gap is found,

Table 19 – Solutions for set of instances.

Instances	MIPGAP	Grids	Relax. Scheme	MILP Sol. [10^3\$]	NLP Sol. [10^3\$]	GAP [%]	Iter.	CPU Time [s]
2F-2V-2ST-2CT 1CDU-4C-1P-15D	0%	[4,4,4,4]	SME	22 800	22 756.6	0.19	2	1
	0%	[4,4,4,4]	$IPWL_1$	22 800	22 756.6	0.19	2	2
	0%	[4,4,4,4]	PWL_2	22 800	22 800	0	2	8
	0%	[4,4,4,4]	$IPWT_1$	22 800	22 756.6	0.19	2	2
	0%	[4,4,4,4]	PWT_2	22 800	22 768.7	0.13	3	6
	0%	[4,4,4,4]	$PWB_{2,2}$	22 800	22 789.5	0.04	3	21
	0%	[4,4,4,4]	$PWB_{2,3}$	22 800	22 789.4	0.04	3	155
	0%	[4,4,4,4]	$PWB_{3,2}$	22 800	22 798.9	0.004	3	30
	0%	[4,4,4,4]	$PWB_{3,3}$	22 800	22 712.5	0.38	2	96
4F-4V-6ST-4CT 3CDU-8C-1P-15D	0%	[2,2,2,2]	SME	38 885	38 065.7	2.10	2	863
	0%	[4,4,4,4]	SME	38 885	38 277.6	1.56	4	476
	0%	[7,7,7,7]	SME	38 885	38 096.7	2.02	5	496
	0%	[2,2,2,2]	$IPWL_1$	38 885	38 065.7	2.10	2	3 403
	0%	[4,4,4,4]	$IPWL_1$	38 885	38 162.7	1.85	4	8 114
	0%	[7,7,7,7]	$IPWL_1$	38 885	38 116.6	1.97	4	2 641
	0%	[2,2,2,2]	PWL_2	38 857.9	38 253.7	1.55	3	5 488
	0%	[4,4,4,4]	PWL_2	38 857.9	37 773.9	2.78	4	4 339
	0%	[7,7,7,7]	PWL_2	38 857.9	37 655.1	3.09	4	4 066
	0%	[2,2,2,2]	$IPWT_1$	38 885	38 065.7	2.10	2	1 338
	0%	[4,4,4,4]	$IPWT_1$	38 885	38 280.1	1.55	4	2 762
	0%	[7,7,7,7]	$IPWT_1$	38 885	38 124.9	1.95	5	4 932
	0%	[2,2,2,2]	PWT_2	38 885	38 453.8	1.10	2	2 054
	0%	[4,4,4,4]	PWT_2	38 885	38 460.5	1.09	3	2 018
	0%	[7,7,7,7]	PWT_2	38 885	38 453.8	1.10	2	1 974
	1.5%	[2,2,2,2]	SME	38 523.2	37 910.2	1.59	2	195
	1.5%	[4,4,4,4]	SME	38 523.2	37 910.2	1.59	2	195
	1.5%	[7,7,7,7]	SME	38 523.2	37 910.2	1.59	2	191
	1.5%	[2,2,2,2]	$IPWL_1$	38 523.2	37 910.2	1.59	2	1 144
	1.5%	[4,4,4,4]	$IPWL_1$	38 523.2	38 060.9	1.20	3	551
	1.5%	[7,7,7,7]	$IPWL_1$	38 523.2	37 910.2	1.59	2	393
	1.5%	[2,2,2,2]	PWL_2	38 603.6	37 698	2.34	3	471
	1.5%	[4,4,4,4]	PWL_2	38 603.6	38 042.2	1.45	3	361
	1.5%	[7,7,7,7]	PWL_2	38 603.6	38 018.2	1.51	4	431
	1.5%	[2,2,2,2]	$IPWT_1$	38 523.2	38 047.2	1.23	3	342
	1.5%	[4,4,4,4]	$IPWT_1$	38 523.2	38 014.4	1.32	3	269
	1.5%	[7,7,7,7]	$IPWT_1$	38 523.2	37 910.2	1.59	2	207
	1.5%	[2,2,2,2]	PWT_2	38 536.8	38 563.6	0.06	2	509
	1.5%	[4,4,4,4]	PWT_2	38 536.8	38 563.6	0.06	2	446
	1.5%	[7,7,7,7]	PWT_2	38 536.8	38 563.6	0.06	2	458
4F-4V-10ST-6CT 5CDU-8C-1P-15D	3%	[4,4,4,4]	SME	56 412.2	56 149.8	0.46	2	1 347
	3%	[4,4,4,4]	$IPWL_1$	56 412.2	56 149.8	0.46	2	1 349
	3%	[4,4,4,4]	PWL_2	55 650	54 440.1	2.17	2	36 054
	3%	[4,4,4,4]	$IPWT_1$	56 412.2	56 149.8	0.46	2	1 373
	3%	[4,4,4,4]	PWT_2	56 380	56 167.9	0.37	2	8 843

when using PWT_2, but it can not be taken as a rule since the MILP is not solved to optimality (i.e., MIPGAP = 1.5%).

The use of a small MIPGAP in large instances can make the solution time prohibitive. For the largest instances (i.e., 4F-4V-10ST-6CT-5CDU-8C-1P-15D), results in Table 19 show that, for MIPGAP = 3%, the strategy finds feasible solutions in reasonable CPU time, except when using PWL_2.

It is not straightforward to define the number of grids for each variable $vct_{i,v,c}$, $lr_{i,r}$, $vt_{i,v}$ and $lcr_{i,r,c}$. Nevertheless, experiments on instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D indicate that one should choose a number of grids such that the resulting contracted domain would not remain the same of previous iteration or be contracted in excess. In the latter case, the search space would decrease too much, potentially

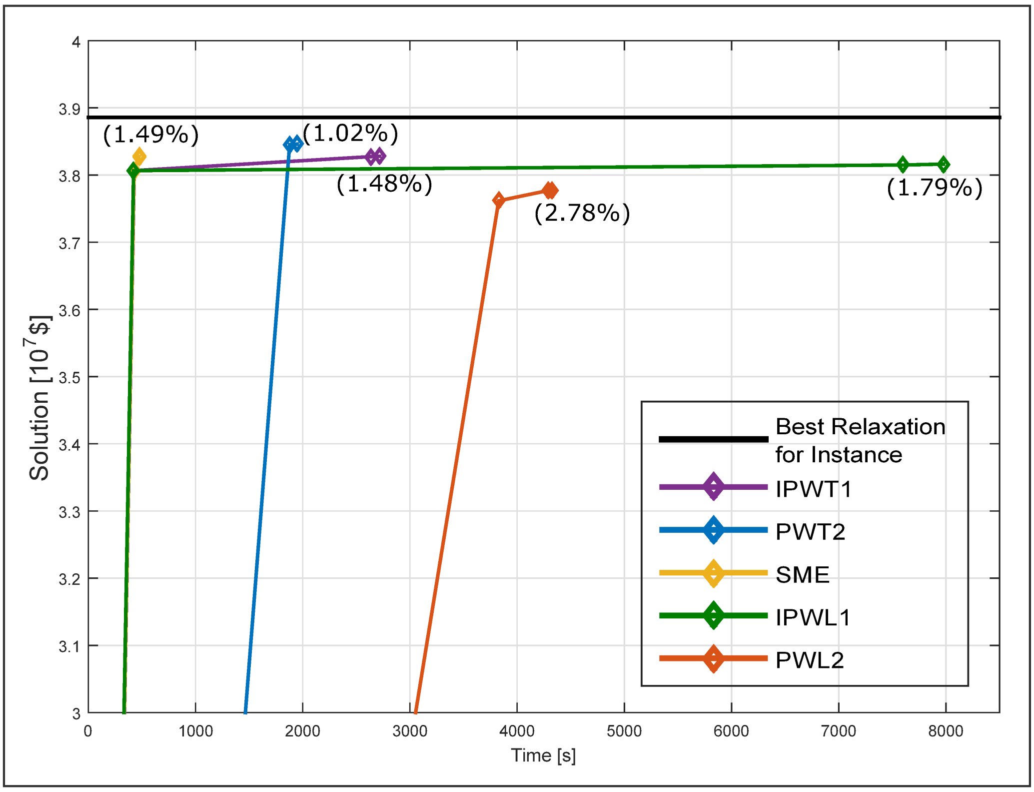


Figure 14 – Solution for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D for MIPGAP = 0% and [4, 4, 4, 4] grid.

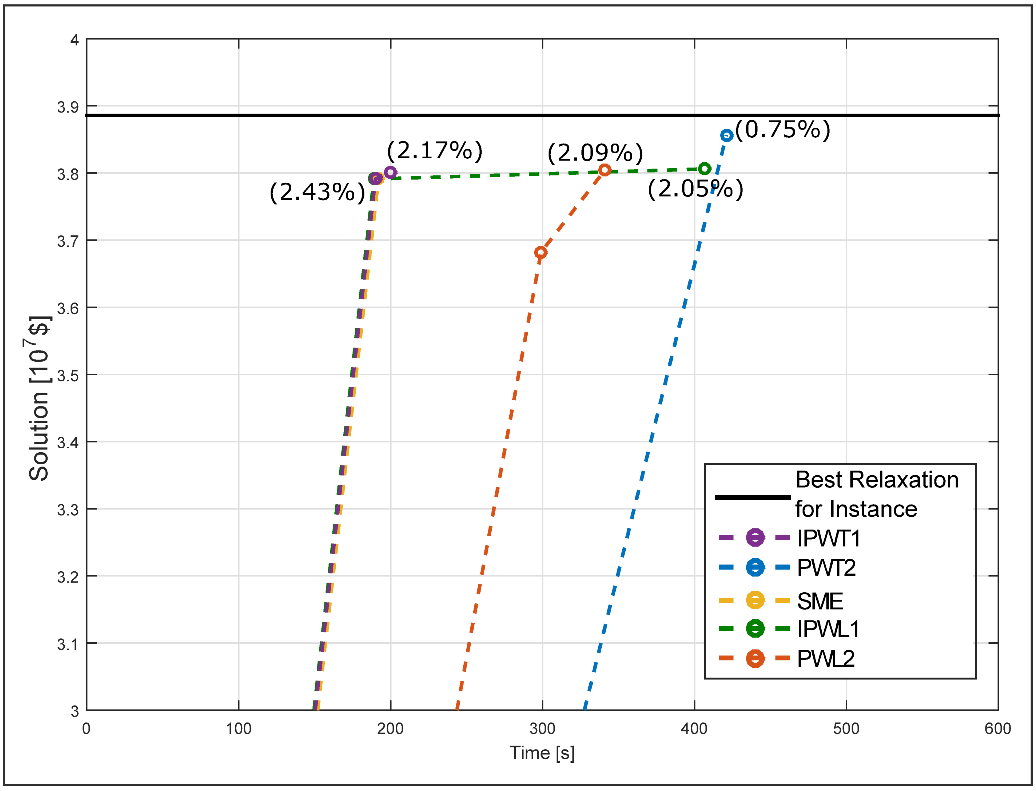


Figure 15 – Solution for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D for MIPGAP = 1.5% and [4, 4, 4, 4] grid.

compromising the solution quality.

Although a wide set of relaxation schemes is tested, on average, relaxations that favor domain partitioning on flow variables (i.e., $vct_{i,v,c}$ and $vt_{i,v}$) produce improved NLP bounds, which is illustrated in Figs. 14 and 15 (i.e., respectively PWT_2 in blue and in dashed blue). This point was observed by Leonardo Salsano de Assis et al. (2017) and can be explained by the fact that the flow variables have tighter bounds.

In summary, the solution strategy based on a MILP-NLP decomposition, with the use of MIPGAP and domain contraction, is able to find small gap solutions on small and medium size instances (i.e., 2F-2V-2ST-2CT-1CDU-2C-1P-15D and 4F-4V-6ST-4CT-3CDU-8C-1P-15D), and find good feasible solutions for the larger instance (i.e., 4F-4V-10ST-6CT-5CDU-8C-1P-15D) within a reasonable CPU time.

3.6.3 Illustration of Operations and Inventories

For problem instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D, whose operations are depicted in Figure 10, this subsection illustrates the schedule of operations and the level of the resources according with the solution found by the proposed solution strategy.

Figure 16 displays the Gantt diagram of the operations. Notice that *CDU1* operates continuously receiving crude oil from both charging tanks through operations *v19* and *v20*, which do not overlap in time, conditions that meet the requirements from the problem statement. The operations that transfer crude from the storage tanks to a charging tank (i.e., the outlet operations *v15* and *v17* from *ST1*), and the one that feeds the CDU from this charging tank (respectively, *v19*), cannot be performed concurrently as seen in the Gantt diagram. With respect to *Vessel2*, the schedule shows that the vessel waits at the terminal (operation *v10*) while *Vessel1* unloads crude into *ST1* and *ST2* (operations *v11* and *v13*). Upon completing the unloading, *Vessel1* travels to offload *FPSO1* (operations *v5* and *v1*), while *Vessel2* unloads into *ST1* (operation *v12*). After finalizing the unloading, *Vessel2* travels to offload *FPSO2* according to operations *v8* and *v4*, respectively. The diagram shows that the proposed model yields a schedule that effectively integrates the operations related to maritime inventory routing and crude oil scheduling.

Figure 17 shows the level of the resources over the planning horizon, namely the level for the FPSOs, vessels, storage and charging tanks. Note that the level of the FPSOs increases over time at the given production rates until a vessel offloads crude, for instance *Vessel1* offloads 10^6 bbl from *FPSO1* during periods 5 and 6 (operation *v1*). At the terminal, storage tank *ST1* initially receives crude oil from *Vessel1* (until time period 2) as dictated by operation *v11*, and then starts transfer crude oil to charging tanks as dictated by operations *v16* and *v15*. It can be noticed that the capacities of the resources are not violated, and vessels travel to FPSOs with empty tanks and return to the terminal with full tanks, as required by the problem statement.

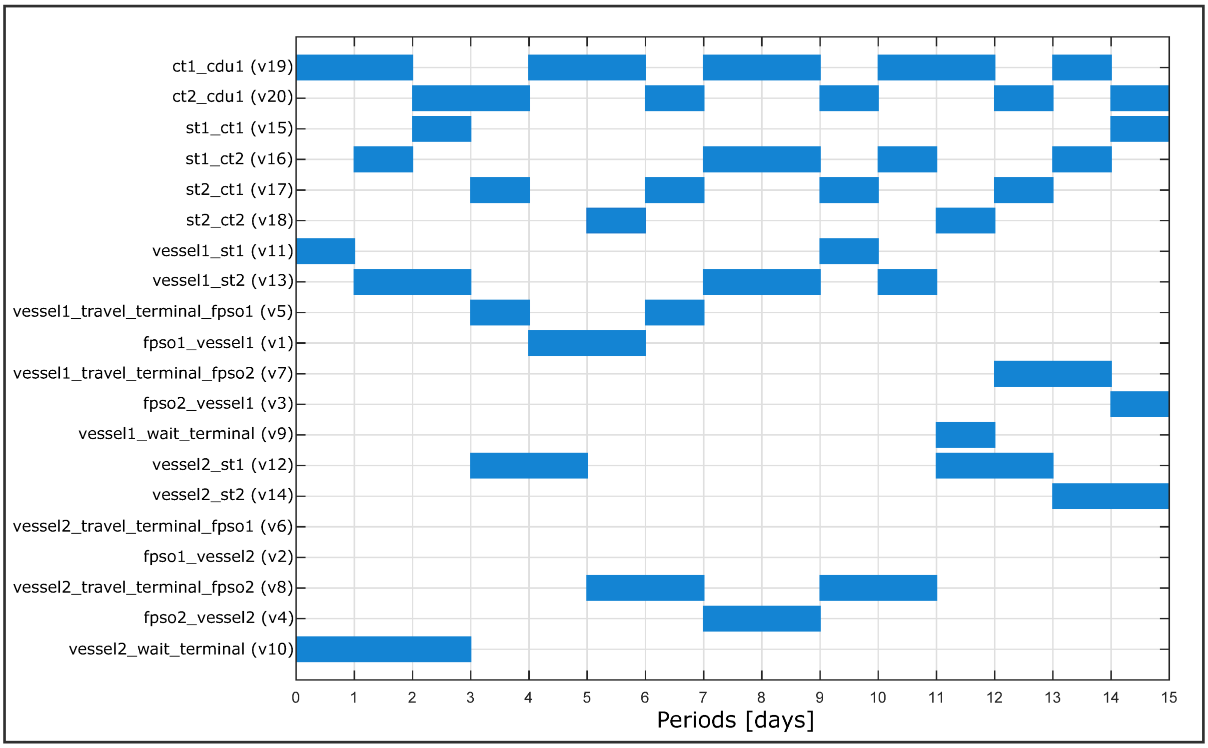


Figure 16 – Schedule of operations for instance 2F-2V-2ST-2CT-2CDU-2C-1P-15D, relaxation scheme PWL_2 and MIPGAP = 0%.

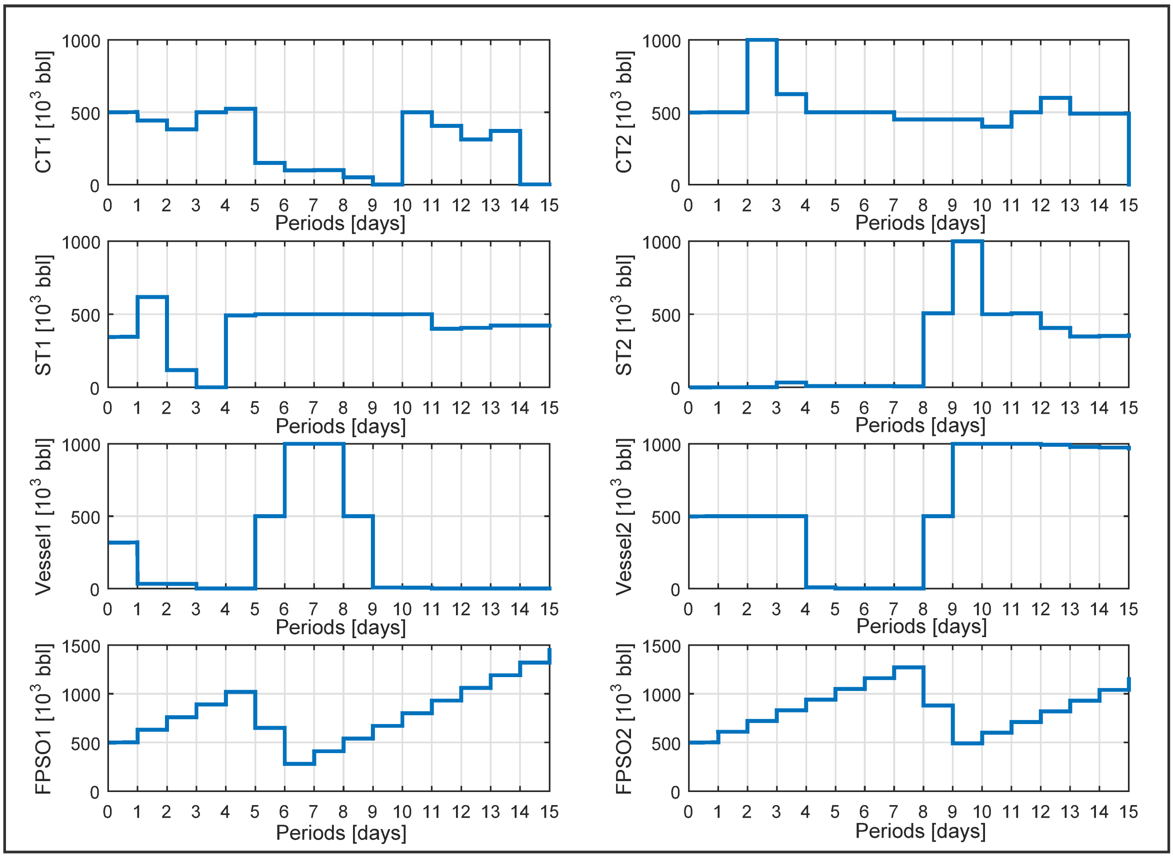


Figure 17 – Level on resources in instance 2F-2V-2ST-2CT-2CDU-2C-1P-15D, PWL_2 and MIPGAP = 0%.

3.7 CONCLUSION

The management of crude oil supply is a problem faced by vertically integrated oil companies, which control production, transportation, storage and refining. It consists of supplying crude oil from FPSOs to CDUs at refineries in order to satisfy their demands, both in terms of crude quantity and quality. The resulting problem incorporates elements of known problems in the literature, namely maritime inventory routing and crude oil scheduling.

To the best of our knowledge, this is the first work to integrate the management of crude oil supply at the operational level by taking into account scheduling of vessels, scheduling of operations in the terminal, and the non-convex non-linearities associated to the blend of crudes.

To tackle this problem, we proposed an MINLP formulation, which is solved by an iterative MILP-NLP decomposition scheme with domain reduction. The solution strategy was able to find small gap solutions on small and medium size instances (i.e., 2F-2V-2ST-2CT-1CDU-2C-1P-15D and 4F-4V-6ST-4CT-3CDU-8C-1P-15D), and found good feasible solutions for the larger instance (i.e., 4F-4V-10ST-6CT-5CDU-8C-1P-15D) within a reasonable CPU time.

The next chapter tackles the difficulty of solving large instances of the non-convex MINLP model developed for the OMCOS. The proposed strategy consists in organizing groups of platforms and storage tanks in clusters, so that crudes are transferred from platforms to storage tanks that belong to the same cluster. Besides reducing the number of routes for vessels, clustering can be done in such a way to minimize the mixing of crudes in storage tanks. In other words, crudes with similar properties would stay in the same cluster, avoiding, for instance, crudes with high content of sulfur being together with crudes with low content of sulfur.

This strategy can also play a part in the optimization. For each cluster, lower and upper bounds can be inferred for the compositions in the storage and charging tanks, allowing the bilinear terms to be approximated with these bounds, rather than using McCormick envelopes. Subsequently, an MILP model would be solved with the implementation of the clusters and approximation of the bilinear terms. The final step would be solving an NLP problem to obtain a primal solution.

4 AN MILP FORMULATION FOR CLUSTERING CRUDES AND ITS EFFECTS ON THE SOLUTION OF THE OPERATIONAL MANAGEMENT OF CRUDE OIL SUPPLY

4.1 INTRODUCTION

The main goal of OMCOS defined in Chapter 3 (Problem (71)) is to define the schedule of vessels' trips and crude oil operations in the terminal in order to deliver to the CDUs the required feed of crudes within the specification. Apart from the rules and constraints defined in Sections 3.3 and 3.4, there are no specific rules on how to perform vessel trips between FPSOs and storage tanks, or how the unloading of crude oil into storage tanks should be performed. This means that vessels are free to travel to and from all FPSOs and unload their cargo in all storage tanks, leading to a broad range of possible blends in these tanks. As shown in the work of Leonardo S. Assis et al. (2019), this creates a highly combinatorial problem, which can be hard to tackle depending on the size of the instance. Additionally, by allowing random mixtures in the storage tanks, crudes with distinct property values may be mixed (e.g., a mixture between crude c_1 with a sulfur content of 0.060 and crude c_2 with 0.010), decreasing the flexibility of operations at the terminal to produce the required blends in the charging tanks.

Ideally, each crude oil arriving at the terminal would have a dedicated storage tank to be unloaded. If this is the case, crudes would only be mixed at the charging tanks in order to produce blends within the specifications required by the CDUs. Nevertheless, when the number of crudes is higher than the number of storage tanks, the pigeonhole principle (KELLY et al., 2017b) suggests that eventually two or more crudes will be mixed in a storage tank. Mixtures may also happen when a vessel needs to unload its cargo in two or more tanks due to the lack of storage capacity in a single one. Finally, a storage tank may be unavailable to receive crudes (i.e., the tank is under maintenance or in operation feeding a charging tank), which forces the vessel to unload in another tank, thereby leading to mixtures.

Since mixtures in the storage tanks seem inevitable, a mathematical formulation is proposed to define clusters (or groups) of crudes and storage tanks, such that the difference among the property values of the crudes assigned to a cluster of storage tanks is as low as possible. From the solution of the clustering problem, bounds on the crude property can be defined for the storage tanks. These bounds are used to build the relaxation of blending constraints in Problem (71), resulting in a relaxed MILP formulation, which is then used in a MILP-NLP decomposition scheme. Further, by knowing the crudes that can be grouped or clustered, their origin platforms and the storage tanks they are assigned to, the number of arcs concerning the traveling operations of vessels (see Fig. 11) can be limited, which decreases the number of logistic decisions (binary variables) and consequently the complexity of the MILP problem.

4.2 LITERATURE REVIEW

This section presents a review on the use of clustering strategies in combination with mathematical programming models for the Inventory Routing Problem (IRP), Vehicle Routing Problem (VRP), COS and others. Regarding IRP and VRP, clusters are typically designed to reduce the complexity of solution strategies for the routing problems. On the other hand, clusters are employed in COS to impose rules on crude mixtures and define groups of tanks to feed CDUs. Nevertheless, few works from the literature consider clusterization for problems that involve COS.

In the VRP literature, Gillett and Miller (1974) were among the first to propose the use of the *cluster-first* and *route-second* strategy. In their work, the solution strategy consists in two sequential steps: (a) define groups of customers according to their polar coordinates and assign vehicles according to capacities; and (b) solve a Traveling Salesman Problem (TSP) for each group.

Mathematical programming techniques are also used by Mulvey and Beck (1984) to model the Capacited Clustering Problem (CCP), which has applications in salesforce allocation and VRP. The problem consists in constructing clusters that are as homogeneous as possible (i.e., minimize the sum of distances between each element in a cluster) without violating the capacity of each cluster.

Liu (1999) make use of clusters to tackle the stock location and order-picking problems in a distribution center. In this problem, the goal is to cluster items in the slots of racks and to sequence the picking lists by customers in order to minimize the total travel distance of a picker in the distribution center.

An extension of the CCP is the Capacitated Centred Clustering Problem (CCCP) (NEGREIROS; PALHANO, 2006), which does not consider necessarily as center value of a cluster one of the elements' attributes in that cluster. Instead, the center value is defined with respect to all elements of the cluster, which introduces non-linearities into the formulation.

Dondo and Cerdá (2007) tackle a multi-depot VRP with a heterogeneous fleet and time windows. The cluster-based solution strategy is a combination of three sequential steps: (a) identify the set of feasible clusters of customers that are cost-effective; (b) assign clusters to vehicles and sequence them on each tour; and (c) define within a cluster the order of nodes and the schedule of vehicles arrival times at customer locations for each tour.

Ganesh and Narendran (2007) address the VRP with delivery and pick-up nodes. The authors proposed a solution strategy where nodes are first clustered based on proximity, then routes are defined for each cluster of nodes, and finally vehicles are allocated to the routes. Qi et al. (2012) tackle a large-scale VRP with time windows. In this work, clusters of customers are defined based on both their spatial location and temporal information. The authors manage to represent time and space in the same coordinate system, and therefore measure the space-temporal distance between two customers.

Nambirajan et al. (2016) extend the classic IRP formulation by considering replenishment tasks at a central depot and different warehouses in a three echelon supply chain. First, the replenishment schedules from suppliers to a single depot is defined using Dynamic Programming (DP). Then, the routing of vehicles from the central depot to multiple warehouses is planned using an extension of a three-stage heuristic based on clustering, allocation, and routing (RAMKUMAR, 2011).

Kelly et al. (2017b) propose an MILP model for defining the assignment of crudes, considering different properties, from external sources to storage tanks in a crude oil terminal. The assignment is done such that the deviation of properties of crudes assigned to a cluster is minimized. Despite considering a large number of crudes and properties, the clustering model does not take into account availability of crudes, flow rate limits, timing, number of storage tanks and storage capacity limits for defining the clusters. Further, Kelly et al. (2017a) discuss how to use their clustering formulation (KELLY et al., 2017b) as part of a pre-scheduling step to reduce the original search space and tackle large scale instances of COS problems.

Cerdá et al. (2018) also make use of a clustering strategy for tackling a COS problem which considers charging tanks, pipelines and CDUs. They proposed a decomposition strategy based on two decision levels. First, an MINLP model is solved to (a) grouping charging tanks into as many clusters as the number of CDUs and (b) assign each cluster of charging tanks to feed a unique CDU. Then, an MINLP model for each cluster-CDU pair is solved to defined the scheduling of crude oil operations. The results show for the tested instances a reduced degradation in solution quality and a strong reduction of the computational burden.

When it comes to food grains procurement and their transportation, the use of clusters can play a major role in order to decrease the complexity of resulting optimizations models. Mogale et al. (2019) propose an MILP formulation to determine the number and location of procurement centers while minimizing the total supply chain network costs. The first step of the solution strategy consists of using genetic algorithms to group grain suppliers to clusters and then allocate each cluster to a candidate location of procurement center. Then, the MILP is solved.

More recently, an extension of the storage assignment and warehouse order-picking problems is proposed by In Gyu Lee et al. (2020). The solution strategy consists of two steps: clustering and assignment. In the clustering stage, an optimization model to group items balances both travel time and picking delays caused by traffic congestion and it is solved by evolutionary algorithms. The latter step (assignment) consists of distributing items in each cluster to empty storage locations.

4.2.1 Work Contribution

Usually, clustering formulations are tailor-made for problems like IRP, VRP or COS, and cannot be fully applied to an integrated approach as the operational management of crude oil supply (OMCOS).

Therefore, this chapter proposes an MILP clustering formulation for OMCOS that has the following benefits: (a) decreasing the number of routes available for the vessels; (b) decreasing offloading and unloading operations; and (c) defining rules for crude mixtures in clusters of storage tanks such that the property deviation is minimized. Further, in order to define the clusters, the proposed MILP formulation takes into account the availability of crudes, flow rate limits, timing, availability of resources (i.e., FPSOs, STs, CTs and CDUs), storage capacity limits and demand satisfaction.

The use of clusters also plays a part in the optimization. After defining clusters, bounds on crude properties inside storage and charging tanks can be defined. These bounds are used to linearize the bilinear terms in blending constraints, which yields to an MILP linearization of the OMCOS MINLP formulation. Through the combination of clusters and an MILP-NLP decomposition strategy, OMCOS is solved for a set of instances, presenting good solution quality and reduced computational complexity.

4.3 PROBLEM STATEMENT

Figure 18 (a) illustrates a problem instance with 7 platforms ($FPSO1$ to $FPSO7$), 7 crudes ($C1$ to $C7$), 5 storage tanks ($ST1$ to $ST5$), 3 charging tanks ($CT1$ to $CT3$) and 2 CDUs ($CDU1$ to $CDU2$). For the sake of simplicity, the operations between resources are not shown (i.e., arrows in Fig. 10).

Fig. 18 (b) illustrates the network to be considered for the clustering formulation. Notice that platform-cluster 1 is linked to st-cluster 1, platform-cluster 2 is linked to st-cluster 2, and so forth. Further, all st-clusters are connected to the CT Group (i.e., aggregate of all charging tanks), which is linked to the CDU Group.

For the sake of exemplification, assume that one wants to cluster the set of platforms $FPSO1$ to $FPSO7$, and consequently the set of crudes $C1$ to $C7$, in two groups (i.e., platform-cluster 1 and platform-cluster 2). Similarly, the set of storage tanks $ST1$ to $ST5$ will be clustered in two groups (i.e., st-cluster 1 and st-Cluster 2), and therefore, it is possible to define which group of platforms are allowed to feed each group of storage tanks. Each platform-cluster will have a combined production rate, storage and flow rate capacity that corresponds to the sum of the rates and capacities of the platforms assigned to the cluster. The same holds for the clusters of storage tanks. Each st-cluster will have a combined capacity and flow rate. As indicated in Fig.

18 (b), all charging tanks are grouped into a single CT Group and, in the same manner, the CDUs are integrated in a single CDU Group. The crude oil demand of the CDU Group is the sum of the individual demands of all CDUs. Likewise, the storage and flow rate capacities of the CT Group depend on characteristics of the charging tanks.

The solution of the clustering problem can be seen in Fig. 18 (c). The objectives of satisfying the demand of the CDU Group while minimizing the deviation among the crude property values in a cluster drives a solution where platforms $FPSO1$, $FPSO4$, $FPSO6$ and $FPSO7$ are clustered and assigned to the group of storage tanks $ST1$, $ST2$ and $ST3$. Meanwhile, the group of platforms $FPSO2$, $FPSO3$ and $FPSO5$ is assigned to storage tanks $ST4$ and $ST5$.

Note that for the instance presented in Fig. 18 (a), if the number of clusters is defined as 1 (one), all storage tanks would be assigned to this cluster, and all platforms would be allowed to feed all storage tanks, leading to the original problem defined in Sections 3.3 and 3.4 of the previous chapter. On the other extreme, if the number of clusters is defined as 5 (five), each cluster would have only one storage tank. Thus the baseline operational management problem of crude oil supply can be reduced to its cluster-based version, meaning that the latter is more general than the former.

Main operational rules can be defined:

(a) a platform must be assigned to a unique platform-cluster. Likewise, a storage tank must be assigned to a unique st-cluster;

(b) a platform-cluster must contain at least one platform. The same holds for the st-cluster. At least one storage tank must belong to a st-cluster;

(c) a st-cluster can perform at most one (i.e., receiving crudes from a platform-cluster or sending crudes to the CT Group) operation during the same time period;

(d) at most one st-cluster can feed the CT Group during the same time period;

(e) a distillation operation (i.e., sending crudes from the CT Group the CDU Group) must be carried out in all time periods.

The optimization problem consists in determining, for the planning horizon, the optimal cluster of platforms, storage tanks and consequently crude oils, while maximizing the flow of crudes to the CDU Group and minimizing the deviation of crude property values in a st-cluster. To this end, we propose a discrete time MILP model, whose main decisions consist in selecting the assignments of storage tanks and platforms to clusters, what operations take place at each time, the level of crudes in each cluster and group of resources, and the volume transferred between clusters and group of resources.

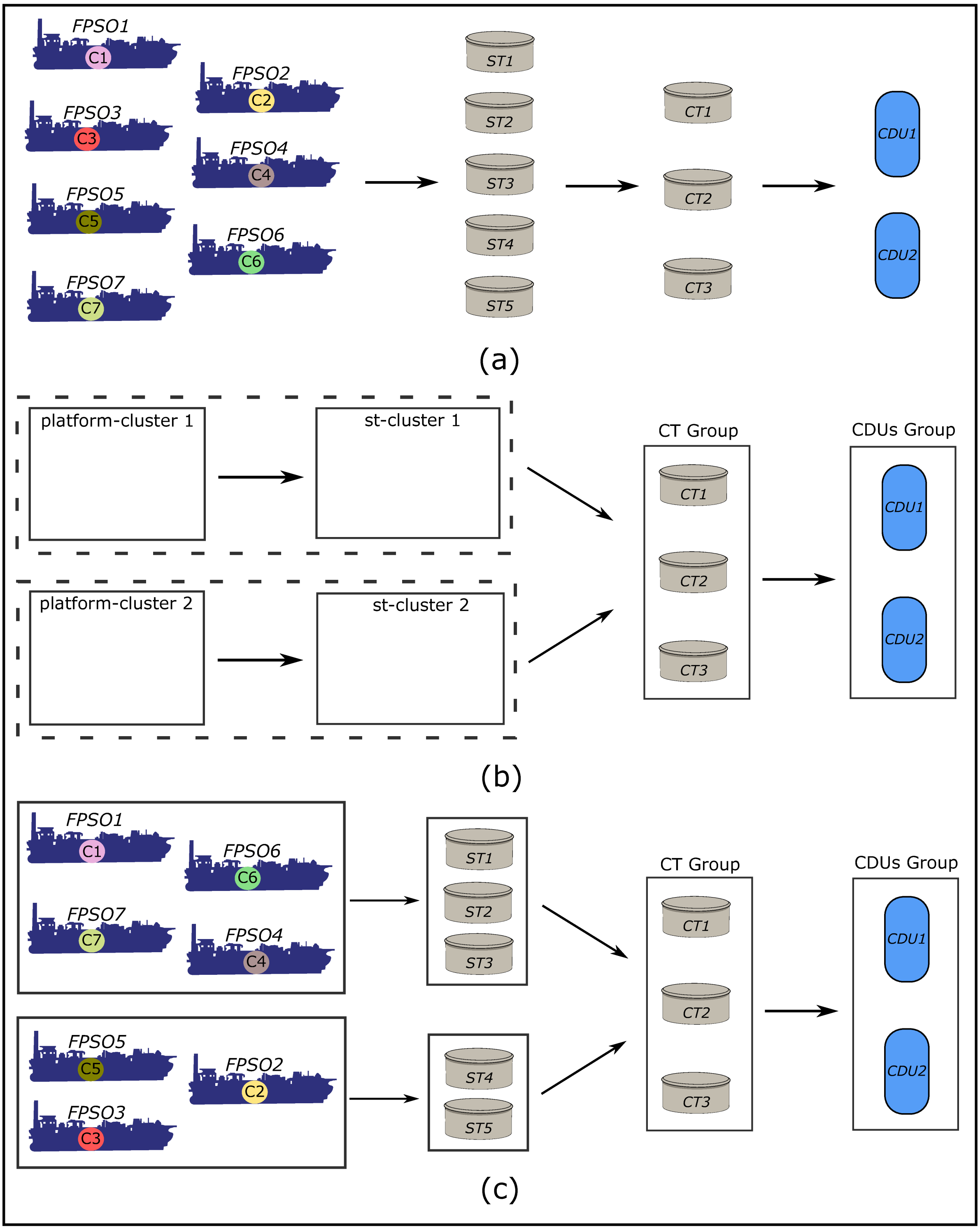


Figure 18 – Clustering procedure considering 2 clusters.

4.4 MATHEMATICAL MODEL FOR CLUSTER DESIGN

Before presenting the constraints and objective function of the mixed integer clustering formulation, the required sets, parameters and variables are defined (some previously defined terms appear here to ease readability).

1. **Sets**.

- $\mathcal{T}$. Set of periods. Index i.
- $\mathcal{RF}$, $\mathcal{RS}$, $\mathcal{RC}$, $\mathcal{RD}$. Set of platforms, storage tanks, charging tanks and CDUs. Index r.
- $\mathcal{R} = \mathcal{RF} \cup \mathcal{RS} \cup \mathcal{RC} \cup \mathcal{RD}$. Set of all resources. Index r.
- $\mathcal{RFCS} = \{1, ..., NCS\}$. Set of platform-clusters. Cardinality $|RFCS| = NCS$, where NCS is the number of clusters. Index rr and rrr.
- $\mathcal{STCS} = \{1, ..., NCS\}$. Set of st-clusters. Cardinality $|STCS| = NCS$, where NCS is the number of clusters. Index rr and rrr.
- $\mathcal{NRC}$. Single-element set representing all charging tanks (namely, CT Group). Index rr and rrr.
- $\mathcal{NRD}$. Single-element set representing all CDUs (namely, CDU Group). Index rr and rrr.
- $\mathcal{NR} = \mathcal{RFCS} \cup \mathcal{STCS} \cup \mathcal{NRC} \cup \mathcal{NRD}$. Set of all new resources. Index rr and rrr.
- $\mathcal{N} \subset \mathcal{NR} \times \mathcal{NR}$. Links between aggregated resources to represent the network illustrated in Fig. 18 (b).
- $\mathcal{C}$. Set of crudes. Index c.
- $\mathcal{K}$. Set of properties. Index k.

2. **Parameters**.

- $RATE_r$. Production rate of platform $r \in \mathcal{RF}$ in 10^3 bbl/day.
- $PR_{k,c}$. Value of property k associated to crude c.
- $CFPSO_{c,r} \in \{0, 1\}$. Indicates if crude c is produced in platform $r \in \mathcal{RF}$.
- $[\underline{CAP}_r, \ \overline{CAP}_r]$. Lower and upper bounds on the capacity of each resource $r \in \mathcal{R}$.
- $[\underline{DEM}_r, \ \overline{DEM}_r]$. Lower and upper bounds on the total volume of crude oil demanded from each charging tank $r \in \mathcal{RC}$ by the CDUs.
- TIL_r. Initial level of crude oil in resource $r \in \mathcal{R} \setminus \mathcal{RD}$.
- $[\underline{\widetilde{CAP}}_{rr}, \ \overline{\widetilde{CAP}}_{rr}]$. Lower and upper bounds on the capacity of new resource $rr \in \mathcal{NR}$.
- $[\underline{\widetilde{DEM}}_{rr}, \ \overline{\widetilde{DEM}}_{rr}]$. Lower and upper bounds on the total demand of the CDU Group $rr \in \mathcal{NRD}$ over the planning horizon. Notice that $\underline{\widetilde{DEM}}_{rr} = \sum_{r \in \mathcal{RC}} \underline{DEM}_r$ and $\overline{\widetilde{DEM}}_{rr} = \sum_{r \in \mathcal{RC}} \overline{DEM}_r$.
- $[\underline{\widetilde{FR}}_r, \ \overline{\widetilde{FR}}_r]$. Lower and upper bounds on the outlet flow rate of resource $r \in \mathcal{R} \setminus \mathcal{RD}$.

- $[\underline{PR}_k,\ \overline{PR}_k]$. Lower and upper bounds on property k among all crudes c.
- $[\underline{PRST}_{k,r},\ \overline{PRST}_{r,k}]$. Lower and upper bounds of property k in each storage tank $r \in \mathcal{RS}$. These bounds are defined after finalizing the cluster optimization, as calculated by Eqs. (124) and (125) which will be introduced later, and then used to linearize the blending constraint define by Eq. (66).

3. **Variables**.

- $assignSTCS_{r,rr} \in \{0,1\}$. Is 1 if storage tank $r \in \mathcal{RS}$ is assigned to st-cluster $rr \in \mathcal{STCS}$.
- $assignRFCS_{r,rr} \in \{0,1\}$. Is 1 if platform $r \in \mathcal{RF}$ is assigned to platform-cluster $rr \in \mathcal{RFCS}$.
- $assign_{i,rr,rrr} \in \{0,1\}$. Is 1 if there is flow of crude oil between resources $rr, rrr \in \mathcal{NR}$ in period i.
- $crudeSTCS_{c,rr} \in \{0,1\}$. Is 1 if crude c is assigned to st-cluster rr.
- $\widetilde{lr}_{i,rr} \geq 0$. Total level of crude oil in the cluster resource $rr \in \mathcal{NR}$ in period i.
- $\widetilde{vt}_{i,rr,rrr} \geq 0$. Flow of crude between cluster resources $(rr, rrr) \in \mathcal{N}$ in period i.
- $\widetilde{il}_{i,rr} \geq 0$. Initial level of crude oil in cluster resource $rr \in \mathcal{NR} \setminus \mathcal{NRD}$.
- $tg_{k,rr} \geq 0$. Value associated to property k in st-cluster $rr \in \mathcal{STCS}$ such that the difference between $tg_{k,rr}$ and the property of crudes assigned to rr is minimized.
- $epr_{i,r,k} \geq 0$. Is the value of property k in storage or charging tank r in period i.

Having introduced the notation, we are in a position to present the constraints that define the formulation for clustering platforms and storage tanks.

4.4.1 Clustering Rules

Eq. (78) states that at least one platform $r \in \mathcal{RF}$ must be assigned to a platform-cluster $rr \in \mathcal{RFCS}$.

$$\sum_{r \in \mathcal{RF}} assignRFCS_{r,rr} \geq 1,\ \forall\ rr \in \mathcal{RFCS}. \tag{78}$$

Also, Eq. (79) defines that a platform r is assigned to one platform-cluster rr.

$$\sum_{rr \in \mathcal{RFCS}} assignRFCS_{r,rr} = 1,\ \forall\ r \in \mathcal{RF}. \tag{79}$$

Similar rules can be defined for the storage tank clusters. At least one storage tank $r \in \mathcal{RS}$ must be assigned to a storage-tank cluster $rr \in \mathcal{STCS}$.

$$\sum_{r \in \mathcal{RS}} assignSTCS_{r,rr} \geq 1, \; \forall \; rr \in \mathcal{STCS}. \tag{80}$$

Further, a storage tank r can only be assigned to one storage-tank cluster rr.

$$\sum_{rr \in \mathcal{STCS}} assignSTCS_{r,rr} = 1, \; \forall \; r \in \mathcal{RS}. \tag{81}$$

Eqs. (82)-(83) track which crude c is assigned to each storage-tank cluster rrr. If platform r, which produces crude c, is assigned to platform cluster rr and the connection from the platform cluster rr to storage-tank cluster rrr is defined in the network $\mathcal{N}$, then crude c will be delivered to storage-tank cluster rrr. Put another way, this constraint states that $crudeSTCS_{c,rrr} = 1$ when: platform r produces crude c, $CFPSOc, r = 1$; the platform is assigned to platform cluster rr, $assignRFCS_{r,rr} = 1$; and the platform cluster rr feeds storage-tank cluster rrr, a condition established by the link $(rr, rrr) \in \mathcal{N}$.

$$crudeSTCS_{c,rrr} \geq assignRFCS_{r,rr},$$
$$\forall \; c \in \mathcal{C}, \; r \in \mathcal{RF}, \; rrr \in \mathcal{STCS}, \; (rr, rrr) \in \mathcal{N} : CFPSO_{c,r} = 1, \tag{82}$$

$$crudeSTCS_{c,rrr} \leq \sum_{\substack{r \in \mathcal{RF}: \\ CFPSO_{c,r}=1}} assignRFCS_{r,rr},$$
$$\forall \; c \in \mathcal{C}, \; rrr \in \mathcal{STCS}, (rr, rrr) \in \mathcal{N}. \tag{83}$$

4.4.2 Inventory Control

Eq. (84) defines the initial volume il_{rr} of crude, in each platform cluster $rr \in \mathcal{RFCS}$, as the sum of initial volume TIL_r in each platform $r \in \mathcal{RF}$ assigned to rr (i.e., $assignRFCS_{r,rr} = 1$). Similarly, Eq. (85) defines the initial volume in each storage-tank cluster $rr \in \mathcal{STCS}$. The proposed cluster framework considers a unique group $rr \in \mathcal{NRC}$ of charging tanks (see Fig. 18 (b)). Therefore, Eq. (86) defines the initial volume il_{rr} of the CT Group as the sum of the initial volume TIL_r in each charging tank $r \in \mathcal{RC}$.

$$il_{rr} = \sum_{r \in \mathcal{RF}} TIL_r \; assignRFCS_{r,rr}, \; \forall \; rr \in \mathcal{RFCS}, \tag{84}$$

$$il_{rr} = \sum_{r \in \mathcal{RS}} TIL_r \; assignSTCS_{r,rr}, \; \forall \; rr \in \mathcal{STCS}, \tag{85}$$

$$il_{rr} = \sum_{r \in \mathcal{RC}} TIL_r, \; \forall \; rr \in \mathcal{NRC}. \tag{86}$$

The inventory $\widetilde{lr}_{i,rr}$ of crude of platform cluster rr in period i, as described by Eqs. (87) and (88), is defined as the previous inventory $\widetilde{lr}_{i-1,rr}$ (or il_{rr} for $i = 1$) added to the production rate $RATE_r$ of all platforms r assigned to platform cluster rr (i.e., $assignRFCS_{r,rr} = 1$) and subtracted the flow $\widetilde{vt}_{i,rr,rrr}$ of crude oil entering the storage-tank cluster rrr.

$$\widetilde{lr}_{i,rr} = il_{rr} + \sum_{r \in \mathcal{RF}} RATE_r \; assignRFCS_{r,rr} - \widetilde{vt}_{i,rr,rrr},$$
$$\forall\; rr \in \mathcal{RFCS},\; (rr, rrr) \in \mathcal{N},\; i = 1, \quad (87)$$

$$\widetilde{lr}_{i,rr} = \widetilde{lr}_{i-1,rr} + \sum_{r \in \mathcal{RF}} RATE_r \; assignRFCS_{r,rr} - \widetilde{vt}_{i,rr,rrr}$$
$$\forall\; rr \in \mathcal{RFCS},\; (rr, rrr) \in \mathcal{N},\; i \in \mathcal{T} \setminus \{1\}. \quad (88)$$

Likewise, Eqs. (89) and (90) enforce the inventory control in each storage-tank cluster and in the CT Group for the initial period $i = 1$ and the remaining planning horizon, respectively. For each storage-tank cluster $rr \in \mathcal{STCS}$, the level of crude $\widetilde{lr}_{i,rr}$ in period i is defined as its previous level $\widetilde{lr}_{i-1,rr}$ (or il_{rr} for $i = 1$) plus the inlet flow from platform cluster $rrr \in \mathcal{RFCS} : (rrr, rr) \in \mathcal{N}$, subtracted the outlet flow to the CT Group $rrr \in \mathcal{NRC} : (rr, rrr) \in \mathcal{N}$.

$$\widetilde{lr}_{i,rr} = il_{rr} + \sum_{(rrr,rr) \in \mathcal{N}} \widetilde{vt}_{i,rrr,rr} - \sum_{(rr,rrr) \in \mathcal{N}} \widetilde{vt}_{i,rr,rrr},$$
$$\forall\; rr \in (\mathcal{STCS} \cup \mathcal{NRC}),\; i = 1, \quad (89)$$

$$\widetilde{lr}_{i,rr} = \widetilde{lr}_{i-1,rr} + \sum_{(rrr,rr) \in \mathcal{N}} \widetilde{vt}_{i,rrr,rr} - \sum_{(rr,rrr) \in \mathcal{N}} \widetilde{vt}_{i,rr,rrr},$$
$$\forall\; rr \in (\mathcal{STCS} \cup \mathcal{NRC}),\; i \in \mathcal{T} \setminus \{1\}. \quad (90)$$

In the case of rr being CT Group, the level $\widetilde{lr}_{i,rr}$ of crude in period i is defined as its previous level $\widetilde{lr}_{i-1,rr}$ (or il_{rr} for $i = 1$) plus the inlet flow from storage-tank cluster $rrr \in \mathcal{STCS} : (rrr, rr) \in \mathcal{N}$, minus the outlet flow to the CDU Group $rrr \in \mathcal{NRD} : (rr, rrr) \in \mathcal{N}$.

4.4.3 Resource Capacity

The capacity bounds on each platform cluster, storage-tank cluster and the charging-tank group are imposed by Eqs. (91)-(93).

$$\widetilde{lr}_{i,rr} \leq \sum_{r \in \mathcal{RF}} \overline{CAP}_r \; assignRFCS_{r,rr},\; \forall\; i \in \mathcal{T},\; rr \in \mathcal{RFCS}, \quad (91)$$

$$\widetilde{lr}_{i,rr} \leq \sum_{r \in \mathcal{RS}} \overline{CAP}_r \; assignSTCS_{r,rr},\; \forall\; i \in \mathcal{T},\; rr \in \mathcal{STCS}, \quad (92)$$

$$\widetilde{lr}_{i,rr} \leq \sum_{r \in \mathcal{RC}} \overline{CAP}_r,\; i \in \mathcal{T},\; \forall\; rr \in \mathcal{NRC}. \quad (93)$$

4.4.4 Flow Rate Limits

Eq. (94) defines that when there is flow of crude oil out of platform-cluster *rr* to st-cluster *rrr* in period *i* (i.e., $assign_{i,rr,rrr} = 1$), it is limited by the sum of maximum flow rate $\overline{\widetilde{FR}}_r$ allowed out of each platform *r* that is assigned to platform-cluster *rr* (i.e., $assignRFCS_{r,rr} = 1$).

$$\widetilde{vt}_{i,rr,rrr} \leq \left(\sum_{r \in \mathcal{RF}} \overline{\widetilde{FR}}_r \; assignRFCS_{r,rr} \right) assign_{i,rr,rrr}, \quad \forall\, i \in \mathcal{T}, rr \in \mathcal{RFCS},\; (rr,rrr) \in \mathcal{N}, \quad (94)$$

Notice that this equation is non-linear and can be linearized by set of Eqs. (95).

$$\begin{cases} \widetilde{vt}_{i,rr,rrr} & \geq 0, \\ \widetilde{vt}_{i,rr,rrr} & \leq \sum_{r \in \mathcal{RF}} \overline{\widetilde{FR}}_r \; assignRFCS_{r,rr}, \\ \widetilde{vt}_{i,rr,rrr} & \leq \left(\sum_{r \in \mathcal{RF}} \overline{\widetilde{FR}}_r \right) assign_{i,rr,rrr}. \end{cases} \quad \forall\, i \in \mathcal{T}, rr \in \mathcal{RFCS}, (rr,rrr) \in \mathcal{N}. \quad (95)$$

A similar constraint can be defined to limit the flow of crudes out of a st-cluster *rr* as stated in Eq. (96). Like Eq. (94), the flow out of a st-cluster is limited by the sum of maximum flow rate $\overline{\widetilde{FR}}_r$ allowed out of each storage tank *r* assigned to the cluster *rr* (i.e., $assignSTCS_{r,rr} = 1$).

$$\widetilde{vt}_{i,rr,rrr} \leq \left(\sum_{r \in \mathcal{RS}} \overline{\widetilde{FR}}_r \; assignSTCS_{r,rr} \right) assign_{i,rr,rrr}, \quad \forall\, i \in \mathcal{T},\; rr \in \mathcal{STCS},\; (rr,rrr) \in \mathcal{N}, \quad (96)$$

Likewise, this equation can be linearized by the set of Eqs. (97).

$$\begin{cases} \widetilde{vt}_{i,rr,rrr} & \geq 0, \\ \widetilde{vt}_{i,rr,rrr} & \leq \sum_{r \in \mathcal{RS}} \overline{\widetilde{FR}}_r \; assignSTCS_{r,rr}, \\ \widetilde{vt}_{i,rr,rrr} & \leq \left(\sum_{r \in \mathcal{RS}} \overline{\widetilde{FR}}_r \right) assign_{i,rr,rrr}. \end{cases} \quad \forall\, i \in \mathcal{T}, rr \in \mathcal{STCS}, (rr,rrr) \in \mathcal{N}. \quad (97)$$

Finally, the total flow of crude oil from the CT Group into the CDU Group is limited by Eq. (98). Notice that this constraint is linear since there is only one CT Group which contains all charging tanks.

$$\widetilde{vt}_{i,rrr,rr} \leq \left(\sum_{r \in \mathcal{RC}} \overline{\widetilde{FR}}_r \right) assign_{i,rr,rrr}, \; \forall\, i \in \mathcal{T},\; rr \in \mathcal{NRC},\; (rr,rrr) \in \mathcal{N}. \quad (98)$$

4.4.5 Demand Satisfaction

Eqs. (99) and (100) define respectively the lower and upper bounds on crude oil demand for the group of CDUs. For instance, the lower demand $\underline{\widetilde{DEM}}_{rrr}$ of the group of CDUs $rrr \in \mathcal{NRD}$ (Eq. (99)) is defined as the sum of minimum supply of crude oil $\underline{DEM}_r$ that each charging tank r needs to provide to the CDUs. A similar definition is also valid for the upper bound in Eq. (100).

Eq. (101) ensures that the total flow $\widetilde{vt}_{i,rr,rrr}$ from the CT Group $rr \in \mathcal{NRC}$ to the CDU Group $rrr \in \mathcal{NRD}$, over the planning horizon, must be within the lower and upper bounds $[\underline{\widetilde{DEM}}_{rrr}, \overline{\widetilde{DEM}}_{rrr}]$ on the overall crude demand requested by the CDUs.

$$\underline{\widetilde{DEM}}_{rrr} = \sum_{r \in \mathcal{RC}} \underline{DEM}_r, \ \forall \ rrr \in \mathcal{NRD}, \tag{99}$$

$$\overline{\widetilde{DEM}}_{rrr} = \sum_{r \in \mathcal{RC}} \overline{DEM}_r, \ \forall \ rrr \in \mathcal{NRD}. \tag{100}$$

$$\underline{\widetilde{DEM}}_{rrr} \leq \sum_{i \in \mathcal{T}} \sum_{(rr,rrr) \in \mathcal{N}} \widetilde{vt}_{i,rr,rrr} \leq \overline{\widetilde{DEM}}_{rrr}, \ \forall \ rrr \in \mathcal{NRD}. \tag{101}$$

4.4.6 Operation Rules

Rules on inlet and outlet operations, similar to the ones defined in Section 3.3, can be applied for the cluster network in Fig 18 (b). Eq. (102) ensures that at most one st-cluster rr can feed the CT Group rrr in period i.

$$\sum_{\substack{rr \in STCS: \\ (rr,rrr) \in \mathcal{N}}} assign_{i,rr,rrr} \leq 1, \ \forall \ i \in \mathcal{T}, \ rrr \in \mathcal{NRC}. \tag{102}$$

Eq. (103) states that, in period i, an inlet operation from platform-cluster rr into a st-cluster rrr can not be performed at the same time as an outlet operation from the same st-cluster rrr.

$$assign_{i,rr,rrr} + assign_{i,rrr,rrrr} \leq 1, \\ \forall \ i \in \mathcal{T}, rrr \in \mathcal{STCS}, (rr, rrr) \in \mathcal{N}, (rrr, rrrr) \in \mathcal{N}. \tag{103}$$

Further, as stated in Sec. 3.3, Eq. (104) defines the continuous distillation condition, which means that in all periods of time $i \in \mathcal{T}$ the CT Group $rr \in \mathcal{NRC}$ must be assigned to supply crude to the CDU Group $rrr \in \mathcal{NRD}$.

$$assign_{i,rr,rrr} = 1, \ \forall \ i \in \mathcal{T}, \ rrr \in \mathcal{NRD}, \ (rr, rrr) \in \mathcal{N}. \tag{104}$$

4.4.7 Objective

The objective (105) of the clustering problem is threefold:

$$C: \max\ f = \left(\sum_{i \in \mathcal{T}} \sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{NRC}}} \widetilde{vt}_{i,rr,rrr} \right) / \overline{\widetilde{DEM}}_{CDUGroup}$$
$$- blendWeight \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \sum_{rr \in \mathcal{STCS}} \frac{|PR_{k,c} - tg_{k,rr}|}{\overline{PR}_k - \underline{PR}_k} crudeSTCS_{c,rr}$$
$$- \sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{RFCS}}} \sum_{\substack{(rrrr,rrrrr) \in \mathcal{N}: \\ rrrr \in \mathcal{RFCS}}} \left| \frac{\sum_{r \in \mathcal{RS}} assignSTCS_{r,rrr}}{\sum_{r \in \mathcal{RF}} assignRFCS_{r,rr}} - \frac{\sum_{r \in \mathcal{RS}} assignSTCS_{r,rrrrr}}{\sum_{r \in \mathcal{RF}} assignRFCS_{r,rrrr}} \right| \quad (105)$$

- In the first term, similar to the baseline operational management problem defined in Eq. (71), the clustering problem aims to maximize the flow of crude supplied to the CDU Group. Since this term has as denominator the maximum demand of the CDU Group, it will assume a maximum value of 1. This is done to bring this term to an order that can be comparable with the other terms in the objective.

- In the second term, it aims to define a single target value for each property k for all crudes to be stored in a given storage-tank cluster rr, namely the value $tg_{k,rr}$. Then, the objective seeks to minimize the deviation of the property k of each crude c, $PR_{k,c}$, that can be delivered to the ST cluster rr from the target value $tg_{k,rr}$. As the property k may vary depending on the type of crude c, this objective seeks to group FPSOs with similar crudes to feed the same st-cluster.

  This term, which is the L_1 distance metric, is typically found in K-medoids MILP formulations for building clusters (PAPAGEORGIOU, Dimitri J; TRESPALACIOS, 2018; NEMHAUSER; WOLSEY, 1988; KAUFMAN; ROUSSEEUW, 1987). The choice of the L_1 distance metric is also endorsed by Kelly et al. (2017b), which propose a clustering MILP model for assigning crudes from external sources to storage tanks in a crude oil terminal.

- Finally, the last term of the objective balances the proportion between the number of storage tanks and platforms in a (platform-cluster, st-cluster) pair with the proportion of these resources in the remaining pairs.

In Eq. (105), the deviations from target values for each property k are expressed in terms of absolutes values, and normalized by the range of maximum and minimum values of the corresponding crude property. By definition, the value $PR_{k,c}$ will assume values within the interval [$\underline{PR}_k$, $\overline{PR}_k$]. Thus, the minimization of the second term of Eq. (105) ensures that, at optimality, the target value $tg_{k,rr}$ will also be within the same bounds. For instance, if $tg_{k,rr} < \underline{PR}_k$ then the values $PR_{k,c} - tg_{k,rr} > PR_{k,c} - \underline{PR}_k$ for all c and therefore the objective would be reduced by setting $tg_{k,rr} = \underline{PR}_k$. Similarly reasoning lead us to conclude that $tg_{k,rr} \leq \overline{PR}_k$ and that the term $\frac{|PR_{k,c} - tg_{k,rr}|}{\overline{PR}_k - \underline{PR}_k} \in [0, 1]$.

In order to linearize Eq. (105), the auxiliary variable $deviation_{k,c,rr} \in \mathbb{R} \geq 0$, $k \in \mathcal{K}$, $c \in \mathcal{C}$, and $rr \in \mathcal{STCS}$, is introduced as an upper bound on the value of term $\frac{|PR_{k,c} - tg_{k,rr}|}{\overline{PR}_k - \underline{PR}_k}$. Eqs. (106) and (107) ensure the consistency of the upper bound induced by $deviation_{k,c,rr}$.

$$\frac{PR_{k,c} - tg_{k,rr}}{\overline{PR}_k - \underline{PR}_k} \leq deviation_{k,c,rr}, \forall\, k \in \mathcal{K}, c \in \mathcal{C}, rr \in \mathcal{STCS}, \tag{106}$$

$$-\frac{(PR_{k,c} - tg_{k,rr})}{\overline{PR}_k - \underline{PR}_k} \leq deviation_{k,c,rr}, \forall\, k \in \mathcal{K}, c \in \mathcal{C}, rr \in \mathcal{STCS}. \tag{107}$$

This leads the second term to be replaced by Eq. (108), which remains non-linear though.

$$\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \sum_{rr \in \mathcal{STCS}} deviation_{k,c,rr}\; crudeSTCS_{c,rr}. \tag{108}$$

Because the target value $tg_{k,rr} \in [\underline{PR}_k, \overline{PR}_k]$ and $deviation_{k,c,rr}$ is an upper bound for $|PR_{k,c} - tg_{k,rr}|/(\overline{PR}_k - \underline{PR}_k)$, the minimization of the deviations in Eq. (108) ensure that $deviation_{k,c,rr} \in [0, 1]$.

Notice that the variables $deviation_{k,c,rr}$ are defined only, and only if, crude c is delivered to the st-cluster rr, a condition flagged by $crudeSTCS_{c,rr} = 1$. To take advantage of this conditions, another auxiliary variable $\widehat{deviation}_{k,c,rr} \in \mathbb{R} \geq 0$ is introduced to assume the value $deviation_{k,c,rr}$ when crude c is received by st-cluster rr. This is implemented for all $k \in \mathcal{K}$, $c \in \mathcal{C}$, and $rr \in \mathcal{STCS}$ as follows:

$$\begin{cases} \widehat{deviation}_{k,c,rr} & \geq 0, \\ \widehat{deviation}_{k,c,rr} & \leq deviation_{k,c,rr}, \\ \widehat{deviation}_{k,c,rr} & \leq crudeSTCS_{c,rr}, \\ \widehat{deviation}_{k,c,rr} & \geq deviation_{k,c,rr} - (1 - crudeSTCS_{c,rr}). \end{cases} \tag{109}$$

The set of Eqs. (109) define that if crude c is not assigned to storage tank-cluster rr ($crudeSTCS_{c,rr} = 0$), variable $\widehat{deviation}_{k,c,rr}$ is set to zero. Likewise, if crude c is assigned to cluster rr, variable $\widehat{deviation}_{k,c,rr}$ is set to $deviation_{k,c,rr}$. Finally, variable $\widehat{deviation}_{k,c,rr}$ replaces the bilinear term $deviation_{k,c,rr}\; crudeSTCS_{c,rr}$ in Eq. (108)

The third term of the objective has the goal to maintain the relation between the number of storage tanks and platforms in a (platform-cluster, st-cluster) pair. For instance, consider 2 (platform-cluster, st-cluster) pairs, 4 platforms and 8 storage tanks. If the first platform-cluster has 1 platform and the second 3 platforms, then the first st-cluster would contain 2 storage tanks and the second 6 storage tanks. Notice that the third term needs to be linearized since it has non-linear fractions and the modulus operator.

Consider a (platform-cluster, st-cluster) pair, with platform-cluster $rr \in \mathcal{RFCS}$ and st-cluster $rrr \in \mathcal{STCS}$. This means that pair $(rr, rrr) \in \mathcal{N}$. Also, variable $rfCluster_{rr}$ is the number of platforms assigned to platform-cluster rr and $stCluster_{rrr}$ is the number of storage tanks assigned to st-cluster rrr. Notice that at least one platform must be assigned to a platform-cluster and at least one storage tank must be assigned to a st-cluster, which means $rfCluster_{rr} \geq 1$ and $stCluster_{rrr} \geq 1$. Moreover, these variables can only assume integer values, which implies that both $rfCluster_{rr},\ stCluster_{rrr} \in \mathbb{N}_+^*$. These variables are defined by Eqs. (110) and (111).

$$rfCluster_{rr} = \sum_{r \in \mathcal{RF}} assignRFCS_{r,rr},\ \forall\ rr \in \mathcal{RFCS}. \tag{110}$$

$$stCluster_{rrr} = \sum_{r \in \mathcal{RS}} assignSTCS_{r,rrr},\ \forall\ rrr \in \mathcal{STCS}. \tag{111}$$

Then, as defined by Eq. (112), variable $proportion_{rr,rrr}$ assumes the value of the ratio between the number of storage tanks and platforms in (platform-cluster, st-cluster) pair $(rr, rrr) \in \mathcal{N}$. This variable is defined as $proportion_{rr,rrr} \in \mathbb{R} \geq 0$.

$$proportion_{rr,rrr} = \frac{\sum\limits_{r \in \mathcal{RS}} assignSTCS_{r,rrr}}{\sum\limits_{r \in \mathcal{RF}} assignRFCS_{r,rr}} = \frac{stCluster_{rrr}}{rfCluster_{rr}},\ \forall\ (rr, rrr) \in \mathcal{N} : rr \in \mathcal{RFCS}. \tag{112}$$

By introducing the supporting variable $proportion_{rr,rrr}$, the third term of the objective can be cast as:

$$\sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{RFCS}}} \sum_{\substack{(rrrr,rrrrr) \in \mathcal{N}: \\ rrrr \in \mathcal{RFCS}}} \left| proportion_{rr,rrr} - proportion_{rrrr,rrrrr} \right|. \tag{113}$$

In order to linearize Eq. (113), the auxiliary variable $proportionDiff_{rr,rrr}^{rrrr,rrrrr} \in \mathbb{R} \geq 0$, $rr,\ rrrr \in \mathcal{RFCS}$, $rrr,\ rrrrr \in \mathcal{STCS}$, and $(rr, rrr),\ (rrrr, rrrrr) \in \mathcal{N}$, is introduced as an upper bound on the value of term $\left| proportion_{rr,rrr} - proportion_{rrrr,rrrrr} \right|$. Eqs. (114) and (115) ensure the consistency of the upper bound induced by $proportionDiff_{rr,rrr}^{rrrr,rrrrr}$ and the third term of the objective can be replaced by Eq. (116).

$$proportion_{rr,rrr} - proportion_{rrrr,rrrrr} \leq proportionDiff_{rr,rrr}^{rrrr,rrrrr},$$
$$\forall\ (rr, rrr) \in \mathcal{N},\ (rrrr, rrrrr) \in \mathcal{N} : rr \in \mathcal{RFCS}, rrrr \in \mathcal{RFCS}. \tag{114}$$

$$-proportion_{rr,rrr} + proportion_{rrrr,rrrrr} \leq proportionDiff_{rr,rrr}^{rrrr,rrrrr},$$
$$\forall\ (rr, rrr) \in \mathcal{N},\ (rrrr, rrrrr) \in \mathcal{N} : rr \in \mathcal{RFCS}, rrrr \in \mathcal{RFCS}. \tag{115}$$

$$\sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{RFCS}}} \sum_{\substack{(rrrr,rrrrr) \in \mathcal{N}: \\ rrrr \in \mathcal{RFCS}}} proportionDiff_{rr,rrr}^{rrrr,rrrrr}. \tag{116}$$

Notice that Eq. (112) is non-linear and can be reformulated as Eq. (117), which has the bilinear term $proportion_{rr,rrr}\ rfCluster_{rr}$.

$$proportion_{rr,rrr}\ rfCluster_{rr} = stCluster_{rrr},\ \forall\ (rr, rrr) \in \mathcal{N} : rr \in \mathcal{RFCS}. \quad (117)$$

In order to linearize the bilinear term $proportion_{rr,rrr}\ rfCluster_{rr}$, consider the following elements:

- From previous definitions, $proportion_{rr,rrr} \in \mathbb{R} \geq 0$. Also, the total number of platforms assigned to platform-cluster rr ($rfCluster_{rr}$) and the total number of storage tanks assigned to st-cluster rrr ($stCluster_{rrr}$) both a natural numbers (i.e., $rfCluster_{rr},\ stCluster_{rrr} \in \mathbb{N}$).

- Set $\mathcal{J} = \{1, ..., (|\mathcal{RF}| - |\mathcal{RFCS}| + 1)\}$. Cardinality $|\mathcal{J}|$ is equal to the maximum number of platforms that can be assigned to a platform cluster. For instance, if there are six platforms (i.e., $|\mathcal{RF}| = 6$) and two platform-clusters (i.e., $|\mathcal{RFCS}| = 2$), at least one platform must be assigned to each platform-cluster, and a maximum of five platforms (i.e., $|\mathcal{RF}| - |\mathcal{RFCS}| + 1 = 5$) can be assigned to a platform-cluster.

- Binary variable $z_{j,rr} \in \mathbb{B},\ j \in \mathcal{J}, rr \in \mathcal{RFCS}$ is 1 if integer value j in set $\mathcal{J}$, that represents the number of platforms $rfCluster_{rr}$ assigned to platform-cluster rr, is selected. As stated by Eq. (118), for each platform-cluster rr, only one integer value j in $\mathcal{J}$ can be selected. Then, as defined in Eq. (119), the integer variable $rfCluster_{rr}$ can be stated as a sum of binary variables multiplied by the integer value j.

$$\sum_{j \in \mathcal{J}} z_{j,rr} = 1,\ \forall\ rr \in \mathcal{RFCS}. \quad (118)$$

$$rfCluster_{rr} = \sum_{j \in \mathcal{J}} j z_{j,rr},\ \forall\ rr \in \mathcal{RFCS}. \quad (119)$$

  Variable $rfCluster_{rr}$ can be replaced in Eq. (117) leading to Eq. (120). Notice that in Eq. (120), the bilinear term $z_{j,rr}\ proportion_{rr,rrr}$ has a binary and a real variable, which can be easily linearized.

$$(\sum_{j \in \mathcal{J}} j z_{j,rr}) proportion_{rr,rrr} = stCluster_{rrr},\ \forall\ (rr, rrr) \in \mathcal{N} : rr \in \mathcal{RFCS}. \quad (120)$$

- Auxiliary variable $\theta_{j,rr,rrr} \in \mathbb{R} \geq 0,\ j \in \mathcal{J},\ (rr, rrr) \in \mathcal{N} : rr \in \mathcal{RFCS}$ can be defined such that the bilinear term $z_{j,rr}\ proportion_{rr,rrr} = \theta_{j,rr,rrr}$. Eq. (121) addresses the linearization of bilinear term $\theta_{j,rr,rrr}$, valid for all $j \in \mathcal{J}$ and $(rr, rrr) \in$

$\mathcal{N} : rr \in \mathcal{RFCS}$.

$$\begin{cases} \theta_{j,rr,rrr} & \geq 0, \\ \theta_{j,rr,rrr} & \leq proportion_{rr,rrr}, \\ \theta_{j,rr,rrr} & \leq z_{j,rr}|\mathcal{RS}|, \\ \theta_{j,rr,rrr} & \geq proportion_{rr,rrr} - |\mathcal{RS}|(1 - z_{j,rr}). \end{cases} \tag{121}$$

Variable $z_{j,rr} = 0$ drives $\theta_{j,rr,rrr} = 0$. However, if $z_{j,rr} = 1$, $\theta_{j,rr,rrr}$ gets bounded by the cardinality of the set of storage tanks $|\mathcal{RS}|$ and assumes the value of $proportion_{rr,rrr}$.

After replacing $\theta_{j,rr,rrr}$ in Eq. (120), it can be then reformulated as Eq. (122).

$$\sum_{j \in \mathcal{J}} j\theta_{j,rr,rrr} = stCluster_{rrr}, \ \forall \ (rr, rrr) \in \mathcal{N} : rr \in \mathcal{RFCS}. \tag{122}$$

Consequently, the objective can be reformulated as the linear function given by Eq. (123), with parameter *blendWeight* being a weighting factor for the second term of the objective.

$$\begin{aligned} C: \ \max \ f = & \left(\sum_{i \in \mathcal{T}} \sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{NRC}}} \widetilde{vt}_{i,rr,rrr} \right) / \overline{\widetilde{DEM}}_{CDUGroup} \\ & - blendWeight \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \sum_{rr \in \mathcal{STCS}} \widehat{deviation}_{k,c,rr} \\ & - \sum_{\substack{(rr,rrr) \in \mathcal{N}: \\ rr \in \mathcal{RFCS}}} \sum_{\substack{(rrrr,rrrrr) \in \mathcal{N}: \\ rrrr \in \mathcal{RFCS}}} proportionDiff_{rr,rrr}^{rrrr,rrrrr} \end{aligned} \tag{123}$$

Having introduced the definitions above, the MILP formulation for the clustering problem consists in minimizing the objective (123) subject to the constraints (78)-(93), Eqs. (95), (97)-(104), (106)-(107), (109), (110)-(111), (114)-(115), (118)-(119), and (121)-(122).

4.4.8 Remarks

There are two main consequences of clustering crudes.

- First, by restricting the crudes allowed in each st-cluster rr, lower and upper bounds $[\underline{PRST}_{r,k}, \ \overline{PRST}_{r,k}]$ can be induced on the value of property k for the storage tank r assigned to st-cluster rr.

$$\underline{PRST}_{k,r} = \min\{PR_{k,c} : crudeSTCS_{c,rr} assignRFCS_{r,rr} = 1, \ \forall \ c \in \mathcal{C}, \ rr \in \mathcal{STCS}\}, \quad k \in \mathcal{K}, r \in \mathcal{RS}. \tag{124}$$

$$\overline{PRST}_{k,r} = \max\{PR_{k,c} : crudeSTCS_{c,rr} assignRFCS_{r,rr} = 1, \ \forall \ c \in \mathcal{C}, \ rr \in \mathcal{STCS}\}, \quad k \in \mathcal{K}, r \in \mathcal{RS}. \tag{125}$$

- And second, the assignment of a platform cluster to feed a given st-cluster restricts a vessel to unload in specific storage tanks, and travel only between platforms and storage tanks within these clusters. The full development of these bounds and their use on the operational management of crude oil supply will be the focus of Section 4.5. The computational gains arising for this new solution methodology will be analyzed in Section 4.7.

4.5 LINEAR APPROXIMATION OF BILINEAR TERMS

The solution of the OMCOS MINLP formulation shown in Section 3.5 consists of a MILP-NLP decomposition scheme. The MILP is a relaxation of the MINLP, in which the blending constraints are relaxed by McCormick envelopes (MCCORMICK, 1976; CASTRO, 2015). Further, the logistics decisions (i.e., binary variables) obtained by solving the MILP are fixed into the MINLP, yielding a NLP which is then solved to obtain a primal solution. Despite generating tight relaxations, the use of McCormick envelopes (e.g., univariate or bivariate partitioning) increase the number of binary variables that lead to a significant impact on the solution time of the MILP. This section presents an alternative way to handle blending constraints, which takes advantage of the structure imposed by solving the clustering formulation.

When a blend of crudes is present in a storage or charging tank $r \in \mathcal{RS} \cup \mathcal{RC}$, the total level of crude $lr_{i,r}$ in tank r in period i can also be seen as the sum of volumes of each crude c (i.e., $lr_{i,r} = \sum_{c \in \mathcal{C}} lcr_{i,r,c}$). Variable $epr_{i,r,k}$, which denotes the value of property $k \in \mathcal{K}$ of the blend of crudes in a storage or charging tank r in period i, can be defined by the following non-linear equation:

$$epr_{i,r,k} = \sum_{c \in \mathcal{C}} PR_{k,c} \frac{lcr_{i,r,c}}{lr_{i,r}}, \ \forall \ i \in \mathcal{T}, \ r \in \mathcal{RS} \cup \mathcal{RC}, \ k \in \mathcal{K}, \tag{126}$$

where parameter $PR_{k,c}$ is the value of property k in crude oil c, and non-linear term $\frac{lcr_{i,r,c}}{lr_{i,r}}$ is the volume fraction of crude c in a storage or charging tank r in time i.

As in the model presented in Section 3.4, variables analogous to the total and individual level of crudes in a tank r in period i (i.e., $lcr_{i,r,c}$ and $lr_{i,r}$) can be introduced to track the flow of crudes between resources. While $vt_{i,v}$ is the total volume of crude oil transferred in period i by operation v, $vct_{i,v,c}$ is the volume of crude c transferred in period i by operation v (i.e., $vt_{i,v} = \sum_{c \in \mathcal{C}} vct_{i,v,c}$). The blending constraint defined in Eq. (66) states that the proportion of crude c inside a storage or charging tank r, defined by $\frac{lcr_{i,r,c}}{lr_{i,r}}$, must hold whenever there is a flow operation v out of resource r (i.e., operation $v \in \mathcal{O}_r$). This means that $\frac{vct_{i,v,c}}{vt_{i,v}} = \frac{lcr_{i,r,c}}{lr_{i,r}}$, which can be rewritten as:

$$vct_{i,v,c} = vt_{i,v} \frac{lcr_{i,r,c}}{lr_{i,r}}, \ \forall \ i \in \mathcal{T}, \ r \in \mathcal{RS} \cup \mathcal{RC}, \ v \in \mathcal{O}_r, \ c \in \mathcal{C}. \tag{127}$$

Equation (127) can be incorporated in Eq. (126) by multiplying both sides by $vt_{i,v}$ to obtain Eq. (128) and replacing $\frac{lcr_{i,r,c}}{lr_{i,r}} vt_{i,v}$ by $vct_{i,v,c}$ to obtain Eq. (129). Notice that the right-hand side of Eq. (129) is linear, while the left-hand side is non-linear.

$$epr_{i,r,k} vt_{i,v} = \sum_{c \in \mathcal{C}} PR_{k,c} \frac{lcr_{i,r,c}}{lr_{i,r}} vt_{i,v}, \tag{128}$$

$$epr_{i,r,k} vt_{i,v} = \sum_{c \in \mathcal{C}} PR_{k,c} vct_{i,v,c}, \ \forall \ i \in \mathcal{T}, \ r \in \mathcal{RS} \cup \mathcal{RC}, \ v \in \mathcal{O}_r, \ k \in \mathcal{K}. \tag{129}$$

This is valid for all periods of time $i \in \mathcal{T}$, storage and charging tanks $r \in \mathcal{RS} \cup \mathcal{RC}$, crude properties $k \in \mathcal{K}$ and transfer operations $v \in \mathcal{O}_r$ leaving resource r. Equation (129) balances, for every period i, the total ($vt_{i,v}$) and individual ($vct_{i,v,c}$) volumes of crude flowing out from storage or charging tanks with the overall value of property k in tank r ($epr_{i,r,k}$) and the individual property k of each crude c ($PR_{k,c}$).

Every feeding operation $v \in \mathcal{WD}$ between charging tanks and CDUs is bounded by lower and upper bounds [$\underline{DEMC}_{v,k}, \ \overline{DEMC}_{v,k}$] on the value of property k. This means that variable $epr_{i,r,k}$, which is the value of property k in charging tank r in period i, must be between these bounds when there is a transfer of crudes to a CDU. Eq. (130) takes advantage of lower and upper bounds [$\underline{DEMC}_{v,k}, \ \overline{DEMC}_{v,k}$] to define a linearization for Eq. (129).

$$\underline{DEMC}_{v,k} vt_{i,v} \leq \sum_{c \in \mathcal{C}} PR_{k,c} vct_{i,v,c} \leq \overline{DEMC}_{v,k} vt_{i,v}, \ \forall \ i \in \mathcal{T}, \ r \in \mathcal{RC}, \ v \in \mathcal{O}_r, \ k \in \mathcal{K}. \tag{130}$$

For storage tanks, vessels may unload different types of crudes during the planning horizon, making it difficult to derive bounds on properties and consequently linearizations such as Eq. (130). Nevertheless, as stated at the end of Section 4.4, the solution of the clustering problem restricts the crudes present in each st-cluster rr, and consequently in the storage tanks assigned to cluster rr. As a consequence, Eqs. (124) and (125) define lower and upper bounds [$\underline{PRST}_{k,r}, \ \overline{PRST}_{k,r}$] on the crude property k for storage tank r. Similarly as in Eq. (130), one can make advantage of these bounds to linearize Eq. (129) for the case of storage tanks, as indicated by Eq. (131).

$$\underline{PRST}_{k,r} vt_{i,v} \leq \sum_{c \in \mathcal{C}} PR_{k,c} vct_{i,v,c} \leq \overline{PRST}_{k,r} vt_{i,v}, \ \forall \ i \in \mathcal{T}, \ r \in \mathcal{RS}, \ v \in \mathcal{O}_r, \ k \in \mathcal{K}. \tag{131}$$

4.6 SOLUTION STRATEGY

In general terms, the combination of the clustering recommendation with the two-step MILP-NLP solution strategy (see Fig. 19) consists of the following steps:

- First, the clustering formulation (123) of a problem instance is solved. Then, the original problem instance is restricted according to the solution of the clustering problem (123). This means: (a) constrain the domain of the vessel's trips variable[1] $s_{i,r,v,u}$ to consider travels only among the platforms and storage tanks that belong to the same (platform-cluster, st-cluster) pair, and (b) constrain the domain of logistics decisions variable[2] $z_{i,v}$ for vessels to only consider offloading crudes from platforms and unloading them into storage tanks that belong to the same (platform-cluster, st-cluster) pair.

- Following, an MILP linearization of problem (71) is constructed considering all of its equations except for the blending constraint (66), which is linearized by Eqs. (130) and (131).

- Finally, the solution of the MILP is used as an initial point and its logistics decisions $z_{i,v}$ and $s_{i,r,v,u}$ (binary variables) are fixed into the MINLP, resulting in a non-linear programming (NLP) problem, which is solved to obtain the final solution.

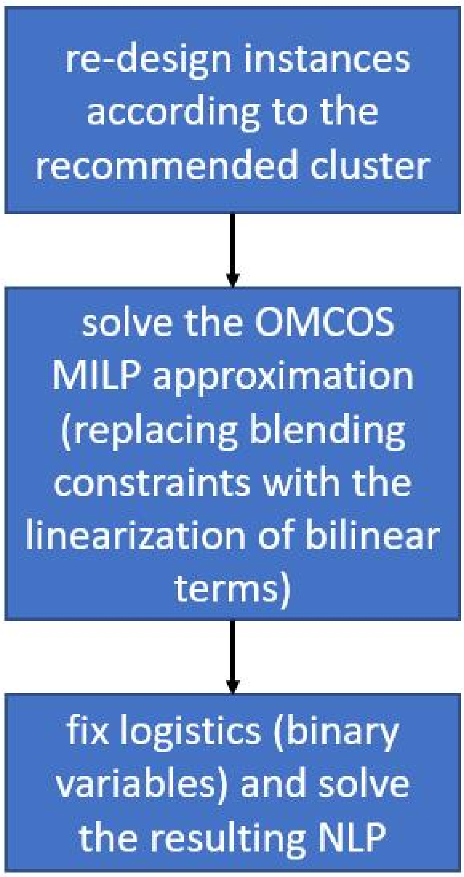


Figure 19 – Solution strategy diagram.

1 Binary variable $s_{i,r,v,u}$ takes on value 1 if, after performing an operation v in period i, vessel r performs an operation u in period $i+1$.
2 Logistic variable $z_{i,v}$ assumes value 1 if operation v is executed in period i.

4.7 ANALYSIS

The goal of this section is to (a) analyze the results given by the clustering formulation proposed in Section 4.4; (b) understand how the clustering of resources, and consequently crudes, affects the solution of the three instances considered for OMCOS; and (c) propose and solve new instances. These instances have already been defined in Section 3.6, which are illustrated by Figures 22, 23 and 24 in Appendix B.

First, the clustering formulation defined in Section 4.4 is run to obtain different clustering schemes for the instances 2F-2V-2ST-2CT-1CDU-2C-1P-15D, 4F-4V-6ST-4CT-3CDU-8C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D. For example, instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D means: 2 FPSOS, 2 vessels, 2 storage tanks, 2 charging tanks, 1 distillation column, 2 crude oils, 1 crude property, and 15 days for planning. Next, the clustering schemes are applied to the original instances, resulting into clustered instances which are solved according to the MILP-NLP solution strategy proposed in Section 4.6. Finally, larger instances are proposed and solved using the same MILP-NLP strategy.

The mathematical programming models and solution strategy were implemented in AMPL and solved in a computer with two Intel Core Xeon E5-2630 v4 Processor (2.20 GHz), totaling 20 cores of 2 threads, 64 GB of RAM and a Ubuntu environment. The MILP model is solved with CPLEX (IBM, 2013) and the NLP formulation with CONOPT (DRUD, 1985).

4.7.1 Clusterization of Instances

Figure 20 illustrates the clustering scheme obtained for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D considering 2 clusters. In this case, the (platform-cluster1, st-cluster1) pair is assigned to platform $FPSO1$, which produces crude cA with sulfur content of 0.01, and storage tank $ST1$. This means that the only platform allowed to supply tank $ST1$ is $FPSO1$, which implies that $ST1$ will only store crude cA. Further, vessels allocated to this cluster can only travel between the terminal and platform $FPSO1$, and unload crude cA into $ST1$.

Similarly, platforms $FPSO2$, $FPSO3$ and $FPSO4$ are assigned to supply storage tanks $ST2$ to $ST6$. These platforms produce respectively crudes cB, cC, and cD, with sulfur content of 0.03, 0.045 and 0.06. Vessels assigned to cover the routes in the (platform-cluster2, st-cluster2) pair will only travel between the terminal and platforms $FPSO2$ to $FPSO4$, and unload crudes cB, cC and cD into tanks $ST2$ to $ST6$.

After defining which resources are assigned to the (platform-cluster1, st-cluster1) and (platform-cluster2, st-cluster2) pairs, bounds can be derived on the crude property for each st-cluster and consequently for each of its storage tank. For instance, st-cluster1 and its tank $ST1$ will receive only crude cA, and therefore, the lower and upper bounds on the sulfur property in tank $ST1$ are $[0.01, 0.01]$. The same implication is also valid for st-cluster2, whose tanks $ST2$-$ST6$ will have $[0.03, 0.06]$ as bounds for the sulfur content. As given by Section 4.5 by Eq. (131), parameters $\underline{PRST}_{k,r}$ and $\overline{PRST}_{k,r}$ make use of these bounds to linearize the blending constraint (66), and thereby obtain the MILP linearization of the MINLP formulation (71).

At the end of the chain, the st-cluster 1 and st-cluster 2 (of storage tanks) feed the CT Group (group of charging tanks), which has the combined storage and transfer capacity of the hole group of charging tanks. Finally, the CT Group satisfies the crude demand of the CDUs, which are combined in the CDU Group.

Notice that, according to the objective (123), the cluster scheme illustrated in Figure 20 is such that: (a) the flow of crudes to the CDU Group is maximized; (b) the crude property deviation in a st-cluster is minimized; and (c) the difference of resource proportion (i.e., the ratio between the number of storage tanks and platforms in a cluster) among all (platform-cluster, st-cluster) pairs is minimized.

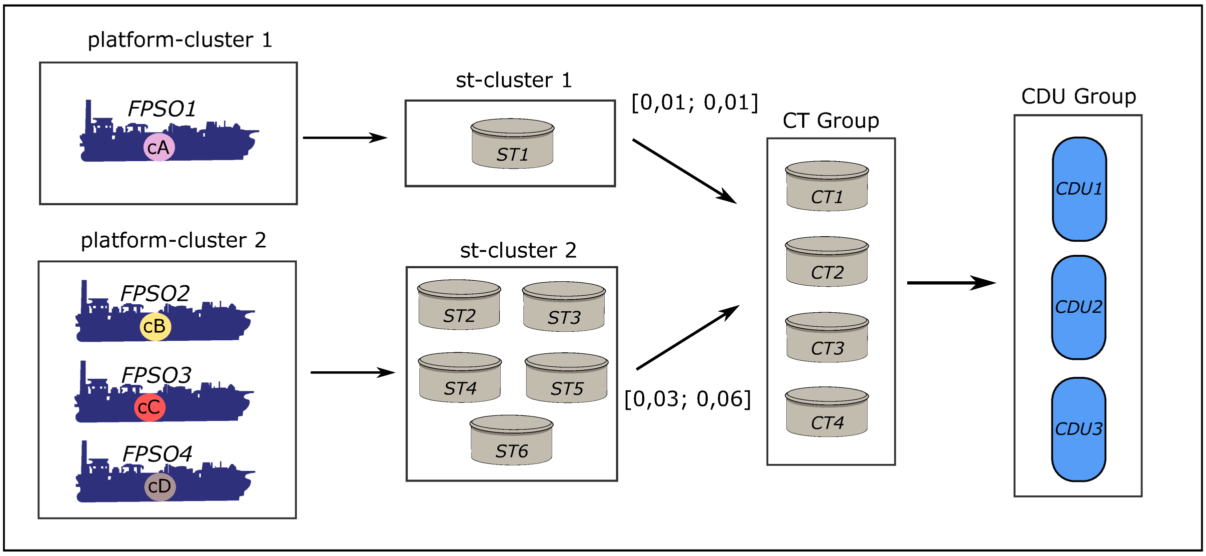


Figure 20 – Clustering for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D considering 2 clusters.

Tables 20, 21 and 22 summarize the clustering schemes for instances 2F-2V-2ST-2CT-1CDU-2C-1P-15D, 4F-4V-6ST-4CT-3CDU-8C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D considering different numbers of cluster. They show the number of clusters, the assignment of platforms and storage tanks to each (platform-cluster, st-cluster) pair and the resulting crudes in each st-cluster. CPU time is neglected since the solution of the clustering problem takes seconds and therefore is not a computational burden. These results are illustrated in Figures 25 to 31 in Appendix C.

Another important result from solving the clustering problem defined in Section 4.4 is the value of the target variable $tg_{k,rr}$ and the resulting property deviation in each st-cluster. Notice that for all instances, target variable $tg_{k,rr}$ assumes the value of property k of one of the crudes supplied to st-cluster rr. If a st-cluster receives only one type of crude oil (e.g., instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D in Table 20), the variable $tg_{k,rr}$ will assume the value of property k of this unique crude and the deviation in that st-cluster will be 0. On the other hand, as shown in Table 21, the deviation considering 3 clusters is 0.3 in st-cluster 2, which is selected to receive crudes cB ($PR_{S,cB}$ = 0.03) and cC ($PR_{S,cB}$ = 0.045). From the objective function, this deviation is computed as $\frac{|PR_{k,c}-tg_{k,rr}|}{\overline{PR}_k-\underline{PR}_k} = \frac{|0.03-0.045|}{(0.06-0.01)} = 0.3$.

Table 20 – Cluster schemes for instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D.

Number Clusters	(platform-cluster, st-cluster) pair	Crudes st-cluster	Target $tg_{k,rr}$ st-cluster	Deviat. st-cluster
2	pair1 = ({fpso1}, {st1})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2}, {st2})	st-cluster2 = {cB}	0.03	0

Table 21 – Cluster schemes for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D.

Number Clusters	(platform-cluster, st-cluster) pair	Crudes st-cluster	Target $tg_{k,rr}$ st-cluster	Deviat. st-cluster
2	pair1 = ({fpso1}, {st1})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2, fpso3, fpso4}, {st2, st3, st4, st5, st6})	st-cluster2 = {cB, cC, cD}	0.045	0.6
3	pair1 = ({fpso1}, {st1, st2})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2, fpso3}, {st3, st4, st5})	st-cluster2 = {cB, cC}	0.045	0.3
	pair3 = ({fpso4}, {st6})	st-cluster3 = {cD}	0.06	0
4	pair1 = ({fpso1}, {st1, st2})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2}, {st3, st4})	st-cluster2 = {cB}	0.03	0
	pair3 = ({fpso3}, {st5})	st-cluster3 = {cC}	0.045	0
	pair4 = ({fpso4}, {st6})	st-cluster4 = {cD}	0.06	0

Table 22 – Cluster schemes for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D.

Number Clusters	(platform-cluster, st-cluster) pair	Crudes st-cluster	Target $tg_{k,rr}$ st-cluster	Deviat. st-cluster
2	pair1 = ({fpso1}, {st1, st2})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2, fpso3, fpso4}, {st3, st4, st5, st6, st7, st8, st9, st10})	st-cluster1 = {cB, cC, cD}	0.045	0.6
3	pair1 = ({fpso1}, {st1, st2})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2}, {st3, st4, st5})	st-cluster2 = {cB}	0.03	0
	pair3 = ({fpso3, fpso4}, {st6, st7, st8, st9, st10})	st-cluster3 = {cC, cD}	0.045	0.3
4	pair1 = ({fpso1}, {st1, st2})	st-cluster1 = {cA}	0.01	0
	pair2 = ({fpso2}, {st3, st4})	st-cluster2 = {cB}	0.03	0
	pair3 = ({fpso3}, {st5, st6, st7})	st-cluster3 = {cC}	0.045	0
	pair4 = ({fpso4}, {st8, st9, st10})	st-cluster4 = {cD}	0.06	0

4.7.2 Solution of Clustered Instances

Here, an analysis is conducted based on the statistics and solution of the instances 2F-2V-2ST-2CT-1CDU-2C-1P-15D, 4F-4V-10ST-6CT-5CDU-8C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D that were re-defined according with the clusters defined in Section 4.7.1. As advocated above, the use of a clustering scheme for the original instance will produce a new instance with a more restricted set of possible operations on the offshore side.

Figure 21 illustrates the comparison between the set of all possible operations and clustered operations in the offshore side of instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D. In the original instance (Figure 21 (a)), the 2 vessels are allowed to travel between the terminal and the 2 FPSOs, and unload crudes *cA* and *cB* in any of storage tanks $ST1$ and $ST2$. By applying the clustering scheme with 2 clusters (see Figure 25) in the original instance (see Figure 22), a restricted set of offshore operations is derived (Figure 21 (b)). In this case, *Vessel1* can travel only between $FPSO1$ and the terminal, and unload crude oil from $FPSO1$ only into storage tank $ST1$. The same holds for the second vessel regarding $FPSO2$ and $ST2$.

For the example in Figure 21 (b), crudes from $FPSO1$ and $FPSO2$ will never get mixed in the storage tanks $ST1$ and $ST2$. Nevertheless, since the connections between storage and charging tanks remain the same, crudes can be normally mixed in the charging tanks to reach the demanded crude specification. Notice that one of the main consequences of using clusters will be the decrease on the number of variables and constraints.

Table 23 presents the number of clusters, the MILP MIPGAP used in the MILP-NLP solution strategy, the number of variables and constraints, the best known solution (taken from Section 3.6), the solution of the clustered instance, the GAP from the best known solution, and CPU time.

Table 23 – Statistics and solution for clustered instances.

			MILP Stat.			NLP Stat.	Best	MILP-NLP Solution		
Instances	Num. Clus.	MILP MIPGAP	Total Vars.	Total Cons.	Binary Vars.	Non-Linear Cons.	Solution [10^3, s]	Sol. [10^3]	[1]GAP	CPU Time [s]
2F-2V-2ST 2CT-1CDU 2C-1P-15D	2	0%	1 380	1 875	531	180	22 800, 8s	22 711	0.4%	0.25
4F-4V-6ST 4CT-3CDU 8C-1P-15D	2	0%	12 900	8 311	4 351	1 920	38 563, 446s	inf	-	-
	3	0%	8 895	6 815	2 098	1 920		38 500	0.16%	17
	4	0%	7 380	6 087	1 346	1 920		37 589	2.5%	11
4F-4V-10ST 6CT-5CDU 8C-1P-15D	2	0.5%	19 650	11 444	7 272	2 880	56 167, 8 843s	55 181	1.75%	300
	3	0.5%	13 335	9 534	3 301	2 880		54 681	2.64%	181
	4	0.5%	10 920	8 558	1 987	2 880		54 285	3.35%	17

[1] GAP = $\frac{BestSol.-Sol.}{BestSol.} 100$.

In addition, Table 24 depicts the best results found for the considered instances making use of the solution strategy proposed in Section 3.5. Also, the table presents the number of variables and constraints, the considered MILP MIPGAP used in the MILP-NLP solution strategy and the CPU time. Notice that for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D, experiments were conducted considering both MILP MILPGAPs of 0% and 1.5%, while for instances 2F-2V-2ST-2CT-1CDU-2C-1P-15D and 4F-4V-10ST-6CT-5CDU-8C-1P-15D it was considered a MILP MILPGAP = 0% and MIPGAP = 3% respectively.

The results on Table 23 suggest the following remarks:

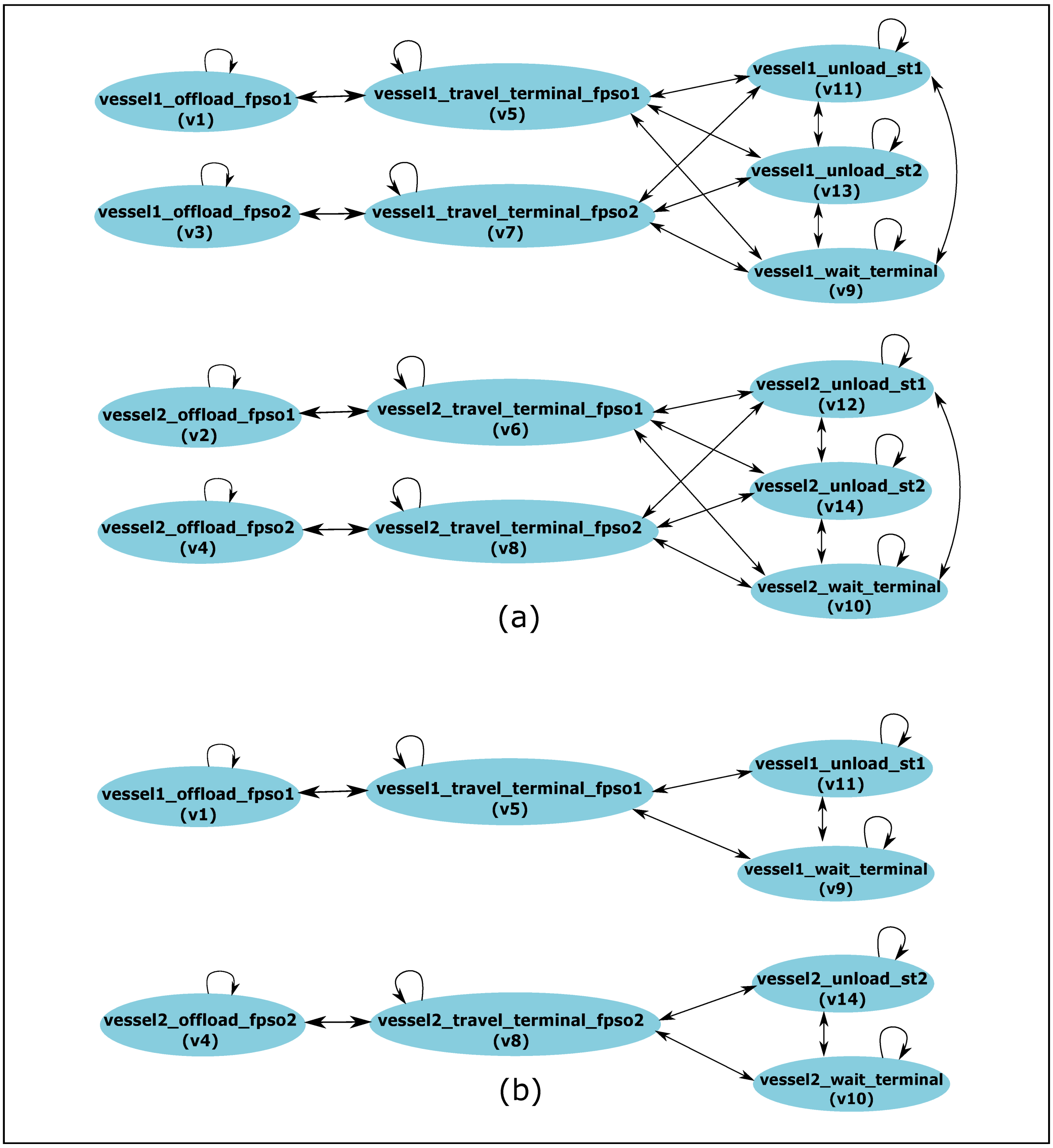


Figure 21 – (a) Graph for the flow of offshore operations in the original instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D; (b) Graph for the flow of offshore operations in the clustered instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D.

Table 24 – Statistics and solution for the original instances considering the MILP-NLP solution strategy presented in Section 3.5.

Instances	Total Vars.	Total Cons.	Binary Vars.	Non-Linear Cons.	MILP MIPGAP	Best Solution	CPU Time [s]
2F-2V-2ST-2CT-1CDU-2C-1P-15D	2 160	2 483	1 170	180	0%	22 800	8
4F-4V-6ST-4CT-3CDU-8C-1P-15D	17 490	11 593	8 310	1 920	0%	38 460.5	2 018
					1.5%	38 563.6	446
4F-4V-10ST-6CT-5CDU-8C-1P-15D	29 175	15 954	15 000	2 880	3%	56 167.9	8 843

1. **Number of Clusters, Problem Size and CPU Time**. As the number of clusters increase, the instance gets more restricted, and fewer are the routes covered by

the vessels assigned to handle the flow of crudes in a (platform-cluster, st-cluster) pair. Further, by clustering the instance, there is a limitation of the offloading and unloading operations.

The restrictions on vessels' trips, offloading and unloading operations have an impact on the number of variables and constraints when solving the clustered instances.

By comparing results in Tables 24 and 23 it is possible to see in numbers the decrease on problem size. For example, the original instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D goes from 29 175 to 10 920 variables in the version considering 4 clusters. From this total number of variables, the number of binaries drops from 15 000 to 1 987, which is a decrease of more than 7.5 times. Likewise, the overall number of constraints decreases from 15 954 to 8 558. The number of non-linear constraints remain the same since the STs-CTs and CTs-CDUs connections do not change.

CPU time follows the same trend as the number of constraints and variables. For a lower number of clusters, the clustered instance gets closer to the original one, and therefore solution time is higher. As the number of clusters increases, the number of variables and constraints decrease and so the solution time. The highest drop happens for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D, with solution time going from more than 8 000 seconds for the original instance to less than 20 seconds when considering 4 clusters.

2. **Number of Clusters and MILP-NLP Solution**.

   Table 23 shows the GAP between the clustered solution and the best known solution found for the original instance.

   Results show that the use of clusters affects the solution, besides a computational gain with the reduction of variables, constraints and CPU time. In addition, the GAP tends to increase with the number the clusters, reaching maximum values close to 3% in the worst case scenarios (i.e., instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D with 4 clusters) and lower than 1% in the best ones (i.e., 4F-4V-6ST-4CT-3CDU-8C-1P-15D with 3 clusters).

   The effect on results can be explained by the fact that (a) the linear approximation for the blending constraints, proposed in Section 4.5 is not as strong as the use of McCormick envelopes (MCCORMICK, 1976) to relax the same constraints; (b) the use of clusters restricts the problem and potentially excludes feasible solutions.

4.7.3 Solution of New Instances

As mentioned in the previous section, one can take advantage of the computational gains of using clusterization for solving larger instances.

Table 25 presents the number of variables and constraints for new instances 4F-4V-10ST-6CT-5CDU-8C-4P-15D and 8F-8V-10ST-6CT-5CDU-8C-4P-15D. Instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D (see Figure 32) considers 8 crude and 4 crude properties (i.e., S, T, U and V with values described in Figure 32) and all possible connections between storage and charging tanks. On the other hand, instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D (see Figure 36), takes into account the same 4 crude properties and connections between tanks, but also extends the number of platforms, vessels, the demand of the CDUs and the storage capacity of the FPSOs.

Table 26 presents the statistics and solution of instances 4F-4V-10ST-6CT-5CDU-8C-4P-15D and 8F-8V-10ST-6CT-5CDU-8C-4P-15D for different clustering schemes. For instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D, the clustering schemes for 2, 3, and 4 clusters are illustrated by Figures 33-35. Likewise, Figures 37 and 38 illustrate for instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D the clustering schemes considering 4 and 6 clusters.

Table 25 – Statistics for new instances.

Instances	Total Vars.	Total Cons.	Binary Vars.	Non-Linear Cons.
4F-4V-10ST-6CT-5CDU-8C-4P-15D	33 075	23 544	15 390	8 400
8F-8V-10ST-6CT-5CDU-8C-4P-15D	68 040	39 814	31 590	8 400

Tables 25 and 26 show a significant decrease on the number of variables and constraints when comparing the original instances and their clustered versions. This decrease has a direct effect on the CPU time. As mentioned in Section 3.6.2, for instances 4F-4V-6ST-4CT-3CDU-8C-1P-15D (see Figure 23) and 4F-4V-10ST-6CT-5CDU-8C-1P-15D (see Figure 24), off-the-shelf solvers were not able to find a feasible solution within a maximum CPU time of 10 hours. The same happens with new instances 4F-4V-10ST-6CT-5CDU-8C-4P-15D and 8F-8V-10ST-6CT-5CDU-8C-4P-15D.

Table 26 – Statistics and solution for new instances with clusterization.

			MILP Stat.			NLP Stat.	MILP-NLP Solution		
Instances	Num. Clus.	MILP MIPGAP	Total Var.	Total Cons.	Binary Vars.	Non-Linear Cons.	MILP Sol. [10^3]	NLP Sol. [10^3]	CPU Time [h]
4F-4V-10ST-6CT 5CDU-8C-4P-15D	2	1%	25 050	20 793	7 107	8 400	56 564	53 758	0.24
	3	1%	19 785	19 059	3 946	8 400	56 452	55 661	0.14
	4	1%	17 370	18 083	2 640	8 400	56 407	53 085	0.05
8F-8V-10ST-6CT 5CDU-8C-4P-15D	4	3%	28 530	24 261	7 811	8 400	59 589	59 230	2.28
	6	3%	22 440	21 739	4 527	8 400	58 250	56 590	0.90

Depending on the clustering scheme, instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D presents a decrease on the number of binary variables from 31 590 to 4 527, mainly resulting from the restriction on vessel trips, and offloading and unloading operations when using clusterization. When comparing the statistics of instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D (see Table 26) for clustering schemes with 4 and 6 clusters, the number of binary variables decreases to almost half, which is reflected in the CPU time that drops from 2.28 hours to 0.9 hour. Although not providing the same solution quality (i.e., from 59 230 to 56 590), there are clear computational gains on using clusters.

4.8 CONCLUSION

As highlighted in the introduction, the main goal of OMCOS is to coordinate the activities of vessels' trips and crude oil operations in the terminal in order to supply crudes to the CDUs. Nevertheless, without clear rules or constraints, all mixtures of crudes are allowed in storage tanks. This may lead to mixtures of crudes with opposite properties, which might be non-desirable.

With the goal of coordinating how crudes can be mixed inside storage tanks, this chapter proposed a MILP formulation to define the optimal cluster of crudes and resources, such that the difference among their properties is as low as possible. The use of clusters offers the following benefits: (a) reduces the number of routes for the vessels; (b) simplifies offloading and unloading operations; and (c) imposes rules for crude mixtures in clusters of storage tanks that minimize property variations.

The solution of the clustering formulation produces: (a) more restricted problem instances and (b) lower and upper bounds on crude properties inside each ST. These bounds are used to linearize the blending constraints and derive an MILP linearization of the original MINLP, which is used in the MILP-NLP solution strategy.

Although possibly eliminating feasible solutions, the use of clusters allows to reach solutions with a compatible quality, but with far fewer variables and constraints, and at much less computational cost.

5 CONCLUDING REMARKS AND FUTURE WORK

The *Management of Crude Oil Supply* consists in coordinating the supply of crude oil from offshore platforms to CDUs in refineries. As highlighted in Chapter 1, this problem is faced by vertically integrated oil companies which control production, transportation, storage and refining.

To the best of our knowledge, Aires et al. (2004), and then Rocha et al. (2009) were the first ones to address the problem of managing crude oil supply in an integrated fashion (i.e., from FPSOs to CDUs). They focused on strategic/tactical decision levels and proposed an MILP formulation to allocate the crude oil produced by platforms to onshore terminals and subsequently to refineries in order to satisfy their demands (i.e., both in terms total volume and quality of crude oil). In addition, their formulation considered crude oil import, inventory control over the planning horizon and vessel fleet sizing decisions. Nevertheless, operational level decisions such as: the limited number of vessels, scheduling of vessels, scheduling of operations in terminals and non-linearities due to blending were not addressed.

With the goal of advancing the state of the art on the management of crude oil supply, this thesis proposes mathematical programming models and solution strategies for the *Operational Management of Crude Oil Supply* (OMCOS), which considers elements of the operational decision level in an integrated fashion. OMCOS comprises both the upstream (i.e., platforms, vessels and terminal) and the midstream (i.e., CDUs at the refinery) segments. In relation to the technical literature, OMCOS combines elements of Maritime Inventory Routing (MIR) with Crude Oil Scheduling (COS) by considering decisions at the operational level (i.e., scheduling and crude oil blending) and tactical level (i.e., inventory control and resource allocation).

Such an integration leads to a non-convex Mixed Integer Non-Linear Programming (MINLP) model composed by an expressive number of variables and constraints. As highlighted by Floudas and Lin (2004) and Castro et al. (2018), scheduling problems with discrete decisions have a combinatorial nature, which when combined with non-linear constraints become challenging from the computational point of view. Therefore, this thesis also proposes solution strategies that exploited problem structure in order to decompose the problem and decrease the computation burden while maintaining solution quality.

The main contributions and remarks are the following:

- **Chapter 2**. An iterative two-step MILP-NLP decomposition algorithm, which implements a domain-reduction strategy for handling bilinear terms in the scheduling of crude oil operations (COS).

  It was shown that on small instances for which an optimal solution is known, the proposed strategy consistently finds optimal or near-optimal solutions. The strategy also solves larger instances which are, in some cases, intractable by a global optimization solver and the MILP linearization solution proposed by Zimberg et al. (2015).

By solving several instances of the scheduling problem, it has been shown that the bivariate partitioning scheme usually provides a stronger relaxation than univariate, leading to better results in fewer iterations. On the other hand, the CPU time is usually higher, which can be explained by the increase on the number of binary variables.

Finally, it is possible to observe that domain partitioning decisions should prioritize to have more, or at least equal, domain partitions for variables that track the amount of crude oil that flows between resources when compared to the number of domain partitions for variables that track the inventory of crude oil inside resources.

- **Chapter 3**. A non-convex MINLP model for OMCOS that brings elements of the operational level into the management of crude oil supply, thereby incorporating elements of maritime inventory routing and crude oil scheduling. Further, an iterative MILP-NLP decomposition is presented to tackle the MINLP problem that relies on bivariate piecewise McCormick envelopes (to yield an MILP relaxation), domain reduction (to reduce complexity), and a NLP solver (to reach feasible solutions).

  The results showed that the new solution strategy is able to solve the small instances almost to optimality in few minutes, while solvers like BARON and SCIP take more than 2 hours.

  Similarly to Chapter 2, bivariate partitioning schemes usually requires more CPU time since a number of new variables and constraints are added to the MILP problem.

  For larger instances, off-the-shelf solver like BARON and SCIP were not able to find a feasible solution in less than 10 hours. On the other hand, the proposed solution strategy is able to find solutions that can vary their quality according to the chosen relaxation scheme.

  Although a wide set of relaxation schemes is tested, on average, relaxations that favor domain partitioning on flow variables produce improved NLP bounds. This point was also observed in the previous chapter and can be explained by the fact that the flow variables have tighter bounds.

- **Chapter 4**. A Mixed Integer Linear Programming (MILP) clustering formulation for OMCOS that offers the following benefits: (a) reduces the number of routes for the vessels; (b) simplifies offloading and unloading operations; (c) imposes rules for crude mixtures in clusters of storage tanks that minimize property variations; and (d) produces bounds on crude properties inside storage and charging tanks that are used to linearize the bilinear terms in blending constraints.

Although possibly eliminating feasible solutions, through the combination of clusters and a MILP-NLP decomposition, good solutions were obtained for a set of representative instances of OMCOS at a reduced computational cost.

The work developed in this thesis also motivates future research in the following areas:

- According to the results presented in this thesis, most of the computational cost can be attributed to solving the MILP relaxation (or approximation in Chapter 4) in MILP-NLP decomposition strategies.

  Future research could pursue methods to find symmetry-breaking constraints in order to exclude symmetric solutions (MOURET et al., 2009; MARGOT, 2010), which are commonly found in scheduling problems and potentially increase computation time.

  An alternative can be in the use of temporal decomposition strategies such as rolling-horizon and relax-and-fix (ASSIS, Leonardo Salsano de; CAMPONOGARA, 2016) to solve the MILP problem.

  Also, continuous-time models (MOURET et al., 2011b) could be conceived and its performance evaluated in relation to the discrete-time approach presented in this work.

  Better formulations of inventory constraints can also be perused as in Rocha et al. (2013), Aizemberg et al. (2014) and Yifu Chen and Maravelias (2020). These works show that tighter formulations can improve the relaxation of MIP models. One can take advantage of improving the MILP relaxation of the MINLP formulation and consequently increase the performance of the MILP-NLP decomposition strategies described in this thesis.

- The operational management problem for crude oil supply can be structured in layers, which consist of the layer of platforms, storage tanks, charging tanks, and CDUs. This structure suggests a Lagrangean decomposition (KARUPPIAH et al., 2008) obtained by duplicating variables on the interface of neighboring layers (e.g., flow variables $vt_{i,v}$, $vct_{i,v,c}$, $vt'_{i,v}$ and $vct'_{i,v,c}$) and introducing equality constraints to ensure their consistency (e.g., $vt_{i,v} = vt'_{i,v}$ and $vct_{i,v,c} = vct'_{i,v,c}$). The dualization of these equality constraints with Lagrange multipliers renders a dual problem that, besides producing upper bounds, can yield good starting points for the search of primal solutions with an NLP solver. A feature of the Lagrange decomposition is that the dual function is decomposed in subproblems, one for each layer, which can be solved in parallel.

- As highlighted in Chapter 4, some cluster schemes may lead to infeasible solutions. Therefore, improvements in the clustering MILP formulation can be in the sense of considering in more details how resources are connected (i.e., instead of considering a CT Group and a CDU Group) and taking into account the specification for crude blends defined for the charging tanks.

  Also, as discussed in Kelly et al. (2017a), one might define two layers of clusters. The first layer would be more restricted and the second one would allow a less restrictive clustering scheme in case the first one produces an infeasible solution.

  Finally, in relation to the MILP-NLP solution strategy, one may propose an iterative approach where at the end of each iteration i the linear approximation of the blending constraints is refined using the information obtained in the NLP solution, and thereby improving the MILP linearization of the OMCOS MINLP model to be used in iteration $i+1$.

REFERENCES

ACHTERBERG, Tobias. SCIP: solving constraint integer programs. **Mathematical Programming Computation**, v. 1, n. 1, p. 1–41, 2009. ISSN 1867-2957. DOI: 10.1007/s12532-008-0001-1.

AGRA, Agostinho; CHRISTIANSEN, Marielle; DELGADO, Alexandrino. Discrete time and continuous time formulations for a short sea inventory routing problem. **Optimization and Engineering**, v. 18, n. 1, p. 269–297, 2017. ISSN 1573-2924. DOI: 10.1007/s11081-016-9319-0.

AIRES, Mariza; LUCENA, Abílio; ROCHA, Roger; SANTIAGO, Cláudio; SIMONETTI, Luidi. Optimizing the petroleum supply chain at Petrobras. In: COMPUTER Aided Chemical Engineering. [S.l.]: Elsevier, 2004. v. 18. P. 871–876.

AIZEMBERG, Luiz; KRAMER, Hugo Harry; PESSOA, Artur Alves; UCHOA, Eduardo. Formulations for a problem of petroleum transportation. **European Journal of Operational Research**, v. 237, n. 1, p. 82–90, 2014.

ASSIS, Leonardo S.; CAMPONOGARA, Eduardo; MENEZES, Brenno C.; GROSSMANN, Ignacio E. An MINLP formulation for integrating the operational management of crude oil supply. **Computers & Chemical Engineering**, v. 123, p. 110–125, 2019. ISSN 0098-1354.

ASSIS, Leonardo Salsano de; CAMPONOGARA, Eduardo. A MILP model for planning the trips of dynamic positioned tankers with variable travel time. **Transportation Research Part E: Logistics and Transportation Review**, v. 93, p. 372–388, 2016. ISSN 1366-5545. DOI: http://dx.doi.org/10.1016/j.tre.2016.06.009.

ASSIS, Leonardo Salsano de; CAMPONOGARA, Eduardo; ZIMBERG, Bernardo; FERREIRA, Enrique; GROSSMANN, Ignacio E. A piecewise McCormick relaxation-based strategy for scheduling operations in a crude oil terminal. **Computers & Chemical Engineering**, v. 106, p. 309–321, 2017. ISSN 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2017.06.012.

BALAS, Egon. Disjunctive Programming and a Hierarchy of Relaxations for Discrete Optimization Problems. **SIAM Journal on Algebraic Discrete Methods**, v. 6, n. 3, p. 466–486, 1985. DOI: 10.1137/0606047.

BARBOSA-PÓVOA, Ana Paula. Process Supply Chains Management-Where are We? Where to Go Next? **Frontiers in Energy Research**, v. 2, p. 23, 2014. ISSN 2296-598X. DOI: 10.3389/fenrg.2014.00023.

BARBOSA-PÓVOA, Ana Paula; PINTO, José Mauricio. Process supply chains: Perspectives from academia and industry. **Computers & Chemical Engineering**, v. 132, p. 106606, 2020. ISSN 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2019.106606. Available from: http://www.sciencedirect.com/science/article/pii/S0098135419301899.

BERGAMINI, Maria Lorena; AGUIRRE, Pio; GROSSMANN, Ignacio. Logic-based outer approximation for globally optimal synthesis of process networks. **Computers & Chemical Engineering**, v. 29, n. 9, p. 1914–1933, 2005. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2005.04.003.

BP. **BP Statistical Review of World Energy**. [S.l.: s.n.], 2019. Available from: http://www.bp.com/statiscalreview.

CAMPONOGARA, Eduardo; PLUCENIO, Agustinho. Scheduling dynamically positioned tankers for offshore oil offloading. **International Journal of Production Research**, v. 52, n. 24, p. 7251–7261, 2014. DOI: 10.1080/00207543.2014.916828.

CAPGEMINI. **Creating the Integrated Value Chain for Downstream Oil**. [S.l.: s.n.], 2008.

CASTILLO CASTILLO, Pedro; CASTRO, Pedro M.; MAHALEC, Vladimir. Global Optimization Algorithm for Large-Scale Refinery Planning Models with Bilinear Terms. **Industrial & Engineering Chemistry Research**, v. 56, n. 2, p. 530–548, 2017. DOI: 10.1021/acs.iecr.6b01350.

CASTRO, Pedro M. Tightening piecewise McCormick relaxations for bilinear problems. **Computers & Chemical Engineering**, v. 72, p. 300–311, 2015. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2014.03.025.

CASTRO, Pedro M.; GROSSMANN, Ignacio E. Global Optimal Scheduling of Crude Oil Blending Operations with RTN Continuous-time and Multiparametric Disaggregation. **Industrial & Engineering Chemistry Research**, v. 53, n. 39, p. 15127–15145, 2014. DOI: 10.1021/ie503002k.

CASTRO, Pedro M.; GROSSMANN, Ignacio E.; ZHANG, Qi. Expanding scope and computational challenges in process scheduling. **Computers & Chemical Engineering**, v. 114, p. 14–42, 2018. FOCAPO/CPC 2017. ISSN 0098-1354. DOI: `https://doi.org/10.1016/j.compchemeng.2018.01.020`.

CERDÁ, Jaime; PAUTASSO, Pedro C.; CAFARO, Diego C. Efficient Approach for Scheduling Crude Oil Operations in Marine-Access Refineries. **Industrial & Engineering Chemistry Research**, v. 54, n. 33, p. 8219–8238, 2015. DOI: `10.1021/acs.iecr.5b01461`.

CERDÁ, Jaime; PAUTASSO, Pedro C.; CAFARO, Diego C. Optimization Approaches for Efficient Crude Blending in Large Oil Refineries. **Industrial & Engineering Chemistry Research**, v. 57, n. 25, p. 8484–8501, 2018. DOI: `10.1021/acs.iecr.8b01008`.

CHEN, Xuan; GROSSMANN, Ignacio; ZHENG, Li. A comparative study of continuous-time models for scheduling of crude oil operations in inland refineries. **Computers & Chemical Engineering**, v. 44, p. 141–167, 2012. DOI: `http://dx.doi.org/10.1016/j.compchemeng.2012.05.009`.

CHEN, Xuan; HUANG, Simin; CHEN, Dejing; ZHANG, Zhihai; ZHENG, Li; GROSSMANN, Ignacio; CHEN, Shihui. Hierarchical Decomposition Approach for Crude Oil Scheduling: A SINOPEC Case. **Interfaces**, v. 44, n. 3, p. 269–285, 2014. DOI: `10.1287/inte.2014.0744`.

CHEN, Yifu; MARAVELIAS, Christos T. Preprocessing algorithm and tightening constraints for multiperiod blend scheduling: cost minimization. **Journal of Global Optimization**, Springer, p. 1–23, 2020. DOI: `https://doi.org/10.1007/s10898-020-00882-3`.

CHRISTIANSEN, Marielle; FAGERHOLT, Kjetil; NYGREEN, BjÃžrn; RONEN, David. Ship routing and scheduling in the new millennium. **European Journal of Operational Research**, v. 228, n. 3, p. 467–483, 2013.

DEMPSTER, M A H; HICKS PEDRÓN, N.; MEDOVA, E A; SCOTT, J E; SEMBOS, A. Planning logistics operations in the oil industry. **Journal of the Operational Research Society**, v. 51, n. 11, p. 1271–1288, 2000. ISSN 1476-9360. DOI: `10.1057/palgrave.jors.2601043`.

DONDO, Rodolfo; CERDÁ, Jaime. A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. **European Journal of Operational Research**, v. 176, n. 3, p. 1478–1507, 2007. ISSN 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2004.07.077.

DRUD, Arne. CONOPT: A GRG code for large sparse dynamic nonlinear optimization problems. **Mathematical Programming**, Springer, v. 31, n. 2, p. 153–191, 1985.

ESCUDERO, Laureano F.; QUINTANA, Francisco J.; SALMERÓN, Javier. CORO, a modeling and an algorithmic framework for oil supply, transformation and distribution optimization under uncertainty. **European Journal of Operational Research**, v. 114, n. 3, p. 638–656, 1999. ISSN 0377-2217. DOI: http://dx.doi.org/10.1016/S0377-2217(98)00261-6.

FLOUDAS, Christodoulos A.; LIN, Xiaoxia. Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. **Computers & Chemical Engineering**, v. 28, n. 11, p. 2109–2129, 2004. DOI: http://dx.doi.org/10.1016/j.compchemeng.2004.05.002.

FOURER, R.; GAY, D.M.; KERNIGHAN, B.W. **AMPL: a modeling language for mathematical programming**. [S.l.]: Thomson/Brooks/Cole, 2003.

FRAGA, C. T. C.; CAPELEIRO PINTO, A. C.; BRANCO, C. C. M.; PIZARRO, J. O. S.; PAULO, C. A. S. Brazilian Pre-Salt: An Impressive Journey from Plans and Challenges to Concrete Results. In: PROCEEDINGS of the Offshore Technology Conference. [S.l.: s.n.], 2009.

GANESH, K.; NARENDRAN, T.T. CLOVES: A cluster-and-search heuristic to solve the vehicle routing problem with delivery and pick-up. **European Journal of Operational Research**, v. 178, n. 3, p. 699–717, 2007. ISSN 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2006.01.037.

GHIANI, G.; LAPORTE, G.; MUSMANNO, R. **Introduction to Logistics Systems Planning and Control**. [S.l.]: Wiley, 2004. (Wiley Interscience Series in Systems and Optimization).

GILLETT, Billy E.; MILLER, Leland R. A Heuristic Algorithm for the Vehicle-Dispatch Problem. **Operations Research**, v. 22, n. 2, p. 340–349, 1974. DOI: 10.1287/opre.22.2.340.

GOUNARIS, Chrysanthos E.; MISENER, Ruth; FLOUDAS, Christodoulos A. Computational Comparison of Piecewise-Linear Relaxations for Pooling Problems. **Industrial & Engineering Chemistry Research**, v. 48, n. 12, p. 5742–5766, 2009. DOI: 10.1021/ie8016048.

GROSSMANN, Ignacio E.; TRESPALACIOS, Francisco. Systematic modeling of discrete-continuous optimization models through generalized disjunctive programming. **AIChE Journal**, v. 59, n. 9, p. 3276–3295, 2013. ISSN 1547-5905. DOI: 10.1002/aic.14088.

GUROBI OPTIMIZATION, Inc. **Gurobi Optimizer Reference Manual**. [S.l.: s.n.], 2016. Available from: http://www.gurobi.com.

GUYONNET, Pierre; GRANT, F. Hank; BAGAJEWICZ, Miguel J. Integrated Model for Refinery Planning, Oil Procuring, and Product Distribution. **Industrial & Engineering Chemistry Research**, v. 48, n. 1, p. 463–482, 2009. DOI: 10.1021/ie701712z.

HAMISU, Aminu A.; KABANTIOK, Stephen; WANG, Meihong. Refinery scheduling of crude oil unloading with tank inventory management. **Computers & Chemical Engineering**, v. 55, p. 134–147, 2013. DOI: http://dx.doi.org/10.1016/j.compchemeng.2013.04.003.

HASAN, M. M. Faruque; KARIMI, I.A. Piecewise linear relaxation of bilinear programs using bivariate partitioning. **AIChE Journal**, v. 56, n. 7, p. 1880–1893, 2010. DOI: 10.1002/aic.12109.

IBM. **IBM ILOG CPLEX Optimization Studio**. [S.l.: s.n.], 2013. Available from: http://www.cplex.com.

KARUPPIAH, Ramkumar; FURMAN, Kevin C.; GROSSMANN, Ignacio E. Global optimization for scheduling refinery crude oil operations. **Computers & Chemical Engineering**, v. 32, n. 11, p. 2745–2766, 2008. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2007.11.008.

KAUFMAN, L.; ROUSSEEUW, P.J. Statistical Data Analysis Based on the L_1-Norm and Related Methods. In: [s.l.]: North-Holland, 1987. Clustering by means of Medoids, p. 405–416.

KELLY, Jeffrey D; MENEZES, Brenno C; ENGINEER, Faramroze; GROSSMANN, Ignacio E. Crude-oil blend scheduling optimization of an industrial-sized refinery: a discrete-time benchmark. **Foundations of Computer Aided Process Operations**, 2017a.

KELLY, Jeffrey D; MENEZES, Brenno C; GROSSMANN, Ignacio E; ENGINEER, Faramroze. Feedstock storage assignment in process industry quality problems. **Foundations of Computer Aided Process Operations**, 2017b.

LEE, Heeman; PINTO, Jose M.; GROSSMANN, Ignacio E.; PARK, Sunwon. Mixed-Integer Linear Programming Model for Refinery Short-Term Scheduling of Crude Oil Unloading with Inventory Management. **Industrial & Engineering Chemistry Research**, v. 35, n. 5, p. 1630–1641, 1996. DOI: 10.1021/ie950519h.

LEE, In Gyu; CHUNG, Sung Hoon; YOON, Sang Won. Two-stage storage assignment to minimize travel time and congestion for warehouse order picking operations. **Computers & Industrial Engineering**, v. 139, p. 106129, 2020. ISSN 0360-8352. DOI: https://doi.org/10.1016/j.cie.2019.106129.

LI, Jie; MISENER, Ruth; FLOUDAS, Christodoulos A. Scheduling of crude oil operations under demand uncertainty: A robust optimization framework coupled with global optimization. **AIChE Journal**, v. 58, n. 8, p. 2373–2396, 2012. DOI: 10.1002/aic.12772.

LIMA, Camilo; RELVAS, Susana; BARBOSA-PÓVOA, Ana Paula F.D. Downstream oil supply chain management: A critical review and future directions. **Computers & Chemical Engineering**, v. 92, p. 78–92, 2016. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2016.05.002.

LIU, Chiun-Ming. Clustering techniques for stock location and order-picking in a distribution center. **Computers & Operations Research**, v. 26, n. 10, p. 989–1002, 1999. ISSN 0305-0548. DOI: https://doi.org/10.1016/S0305-0548(99)00026-X.

MARGOT, François. Symmetry in integer linear programming. In: 50 Years of Integer Programming 1958-2008. [S.l.]: Springer, 2010. P. 647–686.

MARQUES, Catarina M.; MONIZ, Samuel; SOUSA], Jorge Pinho [de; BARBOSA-PÓVOA, Ana Paula; REKLAITIS, Gintaras. Decision-support challenges in the chemical-pharmaceutical industry: Findings and future research directions.

Computers & Chemical Engineering, v. 134, p. 106672, 2020. ISSN 0098-1354. DOI: `https://doi.org/10.1016/j.compchemeng.2019.106672`.

MCCORMICK, Garth P. Computability of global solutions to factorable nonconvex programs: Part I — Convex underestimating problems. **Mathematical Programming**, v. 10, n. 1, p. 147–175, 1976. DOI: `10.1007/BF01580665`.

MIRÓ, Anton; POZO, Carlos; GUILLÉN-GOSÁLBEZ, Gonzalo; EGEA, Jose A; JIMÉNEZ, Laureano. Deterministic global optimization algorithm based on outer approximation for the parameter estimation of nonlinear dynamic biological systems. **BMC bioinformatics**, Springer, v. 13, n. 1, p. 90, 2012.

MOGALE, D. G.; GHADGE, Abhijeet; KUMAR, Sri Krishna; TIWARI, Manoj Kumar. Modelling supply chain network for procurement of food grains in India. **International Journal of Production Research**, Taylor & Francis, v. 0, n. 0, p. 1–20, 2019. DOI: `10.1080/00207543.2019.1682707`.

MOURET, Sylvain; GROSSMANN, Ignacio E.; PESTIAUX, Pierre. A new Lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling. **Computers & Chemical Engineering**, v. 35, n. 12, p. 2750–2766, 2011a. ISSN 0098-1354. DOI: `http://dx.doi.org/10.1016/j.compchemeng.2011.03.026`.

MOURET, Sylvain; GROSSMANN, Ignacio E.; PESTIAUX, Pierre. A Novel Priority-Slot Based Continuous-Time Formulation for Crude-Oil Scheduling Problems. **Industrial & Engineering Chemistry Research**, v. 48, n. 18, p. 8515–8528, 2009. DOI: `10.1021/ie8019592`.

MOURET, Sylvain; GROSSMANN, Ignacio E.; PESTIAUX, Pierre. Time representations and mathematical models for process scheduling problems. **Computers & Chemical Engineering**, v. 35, n. 6, p. 1038–1063, 2011b. ISSN 0098-1354. DOI: `http://dx.doi.org/10.1016/j.compchemeng.2010.07.007`.

MULVEY, John M.; BECK, Michael P. Solving capacitated clustering problems. **European Journal of Operational Research**, v. 18, n. 3, p. 339–348, 1984. ISSN 0377-2217. DOI: `https://doi.org/10.1016/0377-2217(84)90155-3`.

NAKANO, C. M. F.; CAPELEIRO PINTO, A. C.; MARCUSSO, J. L.; MINAMI, K. Pre-Salt Santos Basin - Extended Well Test and Production Pilot in the Tupi Area - The

Planning Phase. In: OFFSHORE TECHNOLOGY CONFERENCE. PROCEEDINGS of the Offshore Technology Conference. [S.l.: s.n.], 2009.

NAMBIRAJAN, Ramkumar; MENDOZA, Abraham; PAZHANI, Subramanian; NARENDRAN, T.T.; GANESH, K. CARE: Heuristics for two-stage multi-product inventory routing problems with replenishments. **Computers & Industrial Engineering**, v. 97, p. 41–57, 2016. ISSN 0360-8352. DOI: https://doi.org/10.1016/j.cie.2016.04.004. Available from: http://www.sciencedirect.com/science/article/pii/S0360835216301073.

NEGREIROS, Marcos; PALHANO, Augusto. The capacitated centred clustering problem. **Computers & Operations Research**, v. 33, n. 6, p. 1639–1663, 2006. ISSN 0305-0548. DOI: https://doi.org/10.1016/j.cor.2004.11.011.

NEIRO, Sérgio M. S.; PINTO, José M. Multiperiod Optimization for Production Planning of Petroleum Refineries. **Chemical Engineering Communications**, v. 192, n. 1, p. 62–88, 2005. DOI: 10.1080/00986440590473155.

NEIRO, Sérgio M.S.; PINTO, José M. A general modeling framework for the operational planning of petroleum supply chains. **Computers & Chemical Engineering**, v. 28, 67, p. 871–896, 2004. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2003.09.018.

NEIRO, Sergio Mauro da Silva; MURATA, Valéria Viana; JAHN, Bárbara; ROLAND SEIXAS, Raiana; HORN HOLLMANN, Estefane; SALGADO PEREIRA, Cristiane. Dealing with Multiple Tank Outflows and In-Line Blending in Continuous-Time Crude Oil Scheduling Problems. **Industrial & Engineering Chemistry Research**, v. 58, n. 11, p. 4495–4510, 2019. DOI: 10.1021/acs.iecr.8b03749.

NEMHAUSER, GL; WOLSEY, LA. **Integer and Combinatorial Optimization John Wiley & Sons, Inc**. [S.l.]: Hoboken, New Jersey, 1988.

OLIVEIRA, F.; NUNES, P.M.; BLAJBERG, R.; HAMACHER, S. A framework for crude oil scheduling in an integrated terminal-refinery system under supply uncertainty. **European Journal of Operational Research**, v. 252, n. 2, p. 635–645, 2016. ISSN 0377-2217. DOI: http://dx.doi.org/10.1016/j.ejor.2016.01.034.

AL-OTHMAN, Wafa B.E.; LABABIDI, Haitham M.S.; ALATIQI, Imad M.; AL-SHAYJI, Khawla. Supply chain optimization of petroleum organization under uncertainty in market demands and prices. **European Journal of Operational Research**, v. 189, n. 3, p. 822–840, 2008. ISSN 0377-2217. DOI: `http://dx.doi.org/10.1016/j.ejor.2006.06.081`.

PAPAGEORGIOU, Dimitri J; TRESPALACIOS, Francisco. Pseudo basic steps: bound improvement guarantees from Lagrangian decomposition in convex disjunctive programming. **EURO Journal on Computational Optimization**, Springer, v. 6, n. 1, p. 55–83, 2018.

PAPAGEORGIOU, Dimitri J.; CHEON, Myun-Seok; HARWOOD, Stuart; TRESPALACIOS, Francisco; NEMHAUSER, George L. Recent Progress Using Matheuristics for Strategic Maritime Inventory Routing. In: **Modeling, Computing and Data Handling Methodologies for Maritime Transportation**. Ed. by Charalampos Konstantopoulos and Grammati Pantziou. [S.l.]: Springer International Publishing, 2018. P. 59–94. ISBN 978-3-319-61801-2. DOI: `10.1007/978-3-319-61801-2_3`.

PAPAGEORGIOU, Lazaros G. Supply chain optimisation for the process industries: Advances and opportunities. **Computers & Chemical Engineering**, v. 33, n. 12, p. 1931–1938, 2009. ISSN 0098-1354. DOI: `http://dx.doi.org/10.1016/j.compchemeng.2009.06.014`.

PINTO, J.M.; JOLY, M.; MORO, L.F.L. Planning and scheduling models for refinery operations. **Computers & Chemical Engineering**, v. 24, 90, p. 2259–2276, 2000. ISSN 0098-1354. DOI: `http://dx.doi.org/10.1016/S0098-1354(00)00571-8`.

QI, Mingyao; LIN, Wei-Hua; LI, Nan; MIAO, Lixin. A spatiotemporal partitioning approach for large-scale vehicle routing problems with time windows. **Transportation Research Part E: Logistics and Transportation Review**, v. 48, n. 1, p. 248–257, 2012. Select Papers from the 19th International Symposium on Transportation and Traffic Theory. ISSN 1366-5545. DOI: `https://doi.org/10.1016/j.tre.2011.07.001`.

RAMKUMAR, N. Mathematical models and heuristics for a class of inventory routing problems. **Master's thesis. Department of Management Studies, Indian Institute of Technology Madras**, 2011.

REDDY, P. Chandra Prakash; KARIMI, I. A.; SRINIVASAN, R. Novel solution approach for optimizing crude oil operations. **AIChE Journal**, v. 50, n. 6, p. 1177–1197, 2004. ISSN 1547-5905. DOI: 10.1002/aic.10112.

REDDY, P.Chandra Prakash; KARIMI, I.A.; SRINIVASAN, R. A new continuous-time formulation for scheduling crude oil operations. **Chemical Engineering Science**, v. 59, n. 6, p. 1325–1341, 2004. ISSN 0009-2509. DOI: http://dx.doi.org/10.1016/j.ces.2004.01.009.

REJOWSKI, R.; PINTO, J.M. Efficient MILP formulations and valid cuts for multiproduct pipeline scheduling. **Computers & Chemical Engineering**, v. 28, n. 8, p. 1511–1528, 2004. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2003.12.001.

ROCHA, Roger; GROSSMANN, Ignacio E; POGGI DE ARAGÃO, Marcus VS. Petroleum allocation at PETROBRAS: Mathematical model and a solution algorithm. **Computers & Chemical Engineering**, v. 33, n. 12, p. 2123–2133, 2009.

ROCHA, Roger; GROSSMANN, Ignacio E.; ARAGÃO, Marcus V. S. Poggi de. Petroleum supply planning: reformulations and a novel decomposition algorithm. **Optimization and Engineering**, v. 18, n. 1, p. 215–240, 2017. ISSN 1573-2924. DOI: 10.1007/s11081-017-9349-2.

ROCHA, Roger; GROSSMANN, Ignacio E.; POGGI DE ARAGÃO, Marcus V. S. Cascading Knapsack Inequalities: reformulation of a crude oil distribution problem. **Annals of Operations Research**, v. 203, n. 1, p. 217–234, 2013. ISSN 1572-9338. DOI: 10.1007/s10479-011-0857-8.

RODRIGUES, Filipe; AGRA, Agostinho; CHRISTIANSEN, Marielle; HVATTUM, Lars Magnus; REQUEJO, Cristina. Comparing techniques for modeling uncertainty in a maritime inventory routing problem. **European Journal of Operational Research**, v. 277, n. 3, p. 831–845, 2019. ISSN 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2019.03.015.

RONEN, D. Marine inventory routing: shipments planning. **Journal of the Operational Research Society**, v. 53, n. 1, p. 108–114, 2002. ISSN 1476-9360. DOI: 10.1057/palgrave.jors.2601264.

RONEN, David. Cargo ships routing and scheduling: Survey of models and problems. **European Journal of Operational Research**, v. 12, n. 2, p. 119–126, 1983.

SAHEBI, Hadi; NICKEL, Stefan; ASHAYERI, Jalal. Strategic and tactical mathematical programming models within the crude oil supply chain context - A review. **Computers & Chemical Engineering**, v. 68, p. 56–77, 2014. ISSN 0098-1354. DOI: http://dx.doi.org/10.1016/j.compchemeng.2014.05.008.

SAHINIDIS, N. V. **BARON 14.3.1: Global Optimization of Mixed-Integer Nonlinear Programs, *User's Manual***. [S.l.], 2014.

SHAH, N. Mathematical programming techniques for crude oil scheduling. **Computers & Chemical Engineering**, v. 20, s1227–s1232, 1996. ISSN 0098-1354. DOI: https://doi.org/10.1016/0098-1354(96)00212-8.

SILVA, Thiago Lima; CAMPONOGARA, Eduardo. A computational analysis of multidimensional piecewise-linear models with applications to oil production optimization. **European Journal of Operational Research**, v. 232, n. 3, p. 630–642, 2014. ISSN 0377-2217. DOI: http://dx.doi.org/10.1016/j.ejor.2013.07.040.

SIMCHI-LEVI, D.; KAMINSKY, P.; SIMCHI-LEVI, E. **Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies**. [S.l.]: McGraw-Hill/Irwin, 2008.

SONG, Jin-Hwa; FURMAN, Kevin C. A maritime inventory routing problem: Practical approach. **Computers & Operations Research**, v. 40, n. 3, p. 657–665, 2013. ISSN 0305-0548. DOI: http://dx.doi.org/10.1016/j.cor.2010.10.031.

SONG, Yingkai; MENEZES, Brenno C.; GARCIA-HERREROS, Pablo; GROSSMANN, Ignacio E. Scheduling and Feed Quality Optimization of Concentrate Raw Materials in the Copper Refining Industry. **Industrial & Engineering Chemistry Research**, v. 57, n. 34, p. 11686–11701, 2018. DOI: 10.1021/acs.iecr.8b01512.

TRESPALACIOS, Francisco; GROSSMANN, Ignacio E. Review of Mixed-Integer Nonlinear and Generalized Disjunctive Programming Methods. **Chemie Ingenieur Technik**, v. 86, n. 7, p. 991–1012, 2014. ISSN 1522-2640. DOI: 10.1002/cite.201400037.

WICAKSONO, Danan Suryo; KARIMI, IA. Modeling piecewise under- and overestimators for bilinear process network synthesis via mixed-integer linear programming. In: 18TH European Symposium on Computer-Aided Process Engineering-ESCAPE18, Lyon, France. [S.l.: s.n.], 2008.

YANG, Haokun; BERNAL, David E.; FRANZOI, Robert E.; ENGINEER, Faramroze G.; KWON, Kysang; LEE, Sechan; GROSSMANN, Ignacio E. Integration of crude-oil scheduling and refinery planning by Lagrangean Decomposition. **Computers & Chemical Engineering**, v. 138, p. 106812, 2020. ISSN 0098-1354. DOI: `https://doi.org/10.1016/j.compchemeng.2020.106812`.

YÜZGEÇ, Uğur; PALAZOGLU, Ahmet; ROMAGNOLI, Jose A. Refinery scheduling of crude oil unloading, storage and processing using a model predictive control strategy. **Computers & Chemical Engineering**, v. 34, n. 10, p. 1671–1686, 2010. ISSN 0098-1354. DOI: `https://doi.org/10.1016/j.compchemeng.2010.01.009`.

ZIMBERG, Bernardo; CAMPONOGARA, Eduardo; FERREIRA, Enrique. Reception, mixture, and transfer in a crude oil terminal. **Computers & Chemical Engineering**, v. 82, p. 293–302, 2015. ISSN 0098-1354. DOI: `http://dx.doi.org/10.1016/j.compchemeng.2015.07.012`.

APPENDIX A – PIECEWISE MCCORMICK ENVELOPES REFORMULATION

This appendix describes the MILP reformulation of the disjunctive set of equations given by Eqs. (72)-(76), which approximate the bilinear terms using bivariate piecewise McCormick envelopes. Before discussing each equation, the complete MILP formulation is given, which consists of the following equations for all $i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}$:

$$\eta^{\text{RHS}}_{i,r,v,c} \geq \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} (\underline{VT}_{i,v,q} lcr_{i,r,v,c,q,p} + \underline{LCR}_{i,r,c,p} vt_{i,r,v,c,q,p} - \underline{VT}_{i,v,q}\underline{LCR}_{i,r,c,p} y^{\text{RHS}}_{i,r,v,c,q,p}), \tag{132}$$

$$\eta^{\text{RHS}}_{i,r,v,c} \geq \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} (\overline{VT}_{i,v,q} lcr_{i,r,v,c,q,p} + \overline{LCR}_{i,r,c,p} vt_{i,r,v,c,q,p} - \overline{VT}_{i,v,q}\overline{LCR}_{i,r,c,p} y^{\text{RHS}}_{i,r,v,c,q,p}), \tag{133}$$

$$\eta^{\text{RHS}}_{i,r,v,c} \leq \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} (\overline{VT}_{i,v,q} lcr_{i,r,v,c,q,p} + \underline{LCR}_{i,r,c,p} vt_{i,r,v,c,q,p} - \overline{VT}_{i,v,q}\underline{LCR}_{i,r,c,p} y^{\text{RHS}}_{i,r,v,c,q,p}), \tag{134}$$

$$\eta^{\text{RHS}}_{i,r,v,c} \leq \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} (\underline{VT}_{i,v,q} lcr_{i,r,v,c,q,p} + \overline{LCR}_{i,r,c,p} vt_{i,r,v,c,q,p} - \underline{VT}_{i,v,q}\overline{LCR}_{i,r,c,p} y^{\text{RHS}}_{i,r,v,c,q,p}), \tag{135}$$

$$\begin{cases} lcr_{i,r,c} = \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} lcr_{i,r,v,c,q,p} \\ vt_{i,v} = \sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} vt_{i,r,v,c,q,p} \end{cases} \tag{136}$$

$$\begin{cases} y^{\text{RHS}}_{i,r,v,c,q,p}\underline{LCR}_{i,r,c,p} \leq lcr_{i,r,v,c,q,p} \leq \overline{LCR}_{i,r,c,p} y^{\text{RHS}}_{i,r,v,c,q,p} \\ y^{\text{RHS}}_{i,r,v,c,q,p}\underline{VT}_{i,v,q} \leq vt_{i,r,v,c,q,p} \leq \overline{VT}_{i,v,q} y^{\text{RHS}}_{i,r,v,c,q,p} \end{cases} \quad q \in \mathcal{Q}, p \in \mathcal{P}, \tag{137}$$

$$\sum_{q\in\mathcal{Q}} \sum_{p\in\mathcal{P}} y^{\text{RHS}}_{i,r,v,c,q,p} = 1, \tag{138}$$

$$y^{\text{RHS}}_{i,r,v,c,q,p} \in \{0,1\},\ q \in \mathcal{Q}, p \in \mathcal{P}, \tag{139}$$

$$\underline{LCR}_{i,r,c} \leq lcr_{i,r,c} \leq \overline{LCR}_{i,r,c}, \tag{140}$$

$$\underline{VT}_{i,v} \leq vt_{i,v} \leq \underline{VT}_{i,v}, \tag{141}$$

$$\text{Eqs. (74) } \textit{and} \text{ (75).} \tag{142}$$

Eq. (138) states that only one domain partition $[q,p]$ of the bilinear term $\eta^{\text{RHS}}_{i,r,v,c}$ must be selected. The selection of a domain partition ($y^{\text{RHS}}_{i,r,v,c,q,p} = 1$) activates the set of envelopes defined in Eqs. (132) to (135), which approximates the bilinear function $\eta^{\text{RHS}}_{i,r,v,c} = vt_{i,v} lcr_{i,r,c}$ over the domain partition $[q,p]$.

Notice that for a domain partition $[q,p]$ not selected, in which case $y^{\text{RHS}}_{i,r,v,c,q,p} = 0$, the constants on the right-hand side of Eqs. (132)-(135) will be driven to zero. Moreover, the bo und constraints (137) will bring the variables $lcr_{i,r,v,c,q,p}$ and $vt_{i,r,v,c,q,p}$ to zero, thereby forcing the bilinear term $\eta^{\text{RHS}}_{i,r,v,c}$ also to zero. For each partition $[q,p]$, Eqs. (137) ensure that the variables $lcr_{i,r,v,c,q,p}$ and $vt_{i,r,v,c,q,p}$ are bounded by the parameters $[\underline{LCR}_{i,r,c,p}, \overline{LCR}_{i,r,c,p}]$ and $[\underline{VT}_{i,v,q}, \overline{VT}_{i,v,q}]$. These parameters are defined respectively in Eqs. (74) and (75).

Eqs. (136) state that the variables $lcr_{i,r,v,c,q,p}$ and $vt_{i,r,v,c,q,p}$ will respectively assume the values of the original variables of the model $lcr_{i,r,c}$ and $vt_{i,v}$, provided that the partition $[q, p]$ is selected. For all other partitions, the former variables are forcibly set to zero.

Finally, overall bounds on variables $lcr_{i,r,c}$ and $vt_{i,v}$ are imposed by Eqs. (140) and (141).

APPENDIX B – ORIGINAL INSTANCES

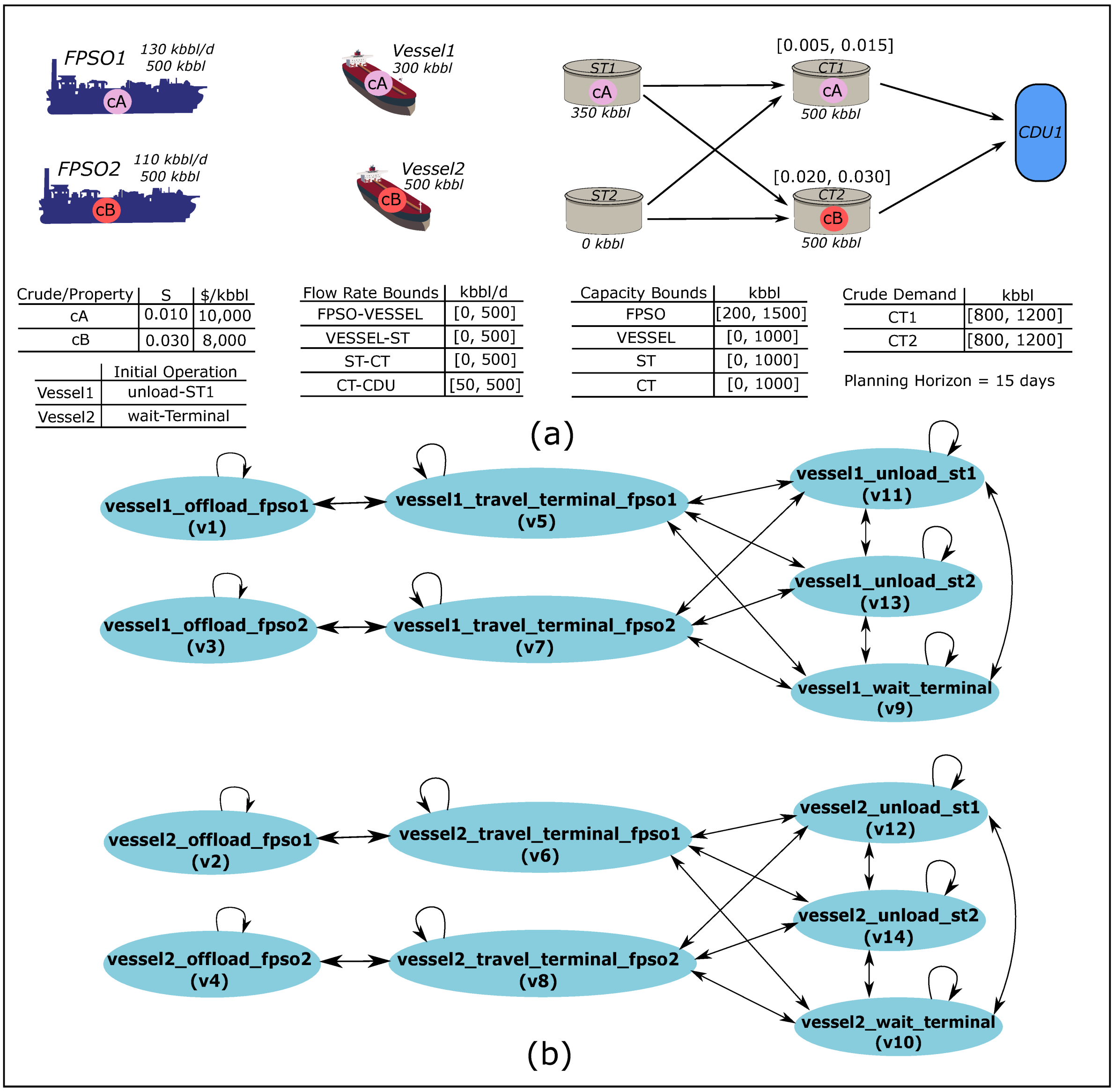
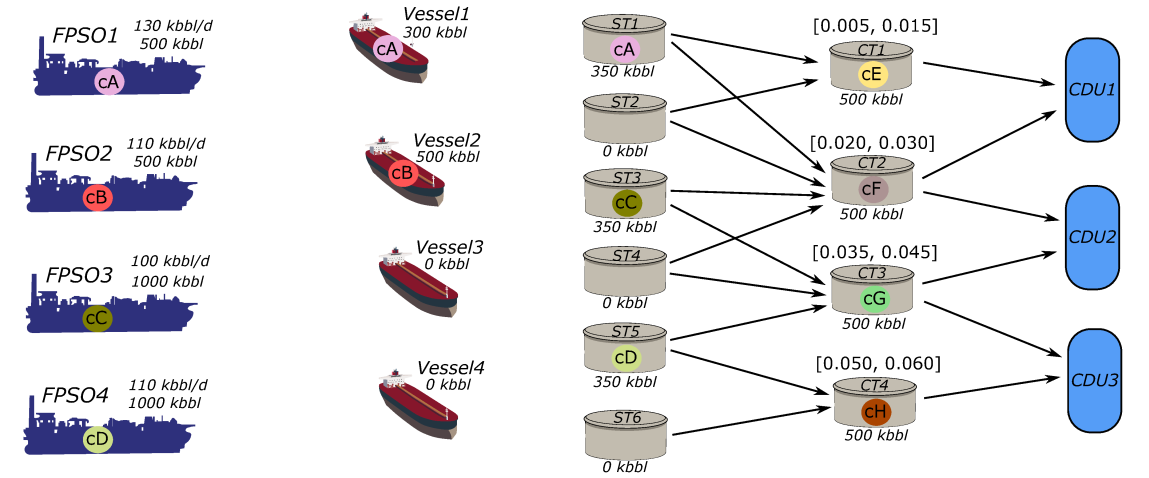


Figure 22 – Illustration of network and graph for the flow of offshore operations for original instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D.



Crude/Property	S	$/kbbl
cA	0.010	10,000
cB	0.030	8,000
cC	0.045	6,500
cD	0.060	5,000
cE	0.012	9,500
cF	0.026	8,500
cG	0.041	7,000
cH	0.056	5,500

Flow Rate Bounds	kbbl/d
FPSO-VESSEL	[0, 500]
VESSEL-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

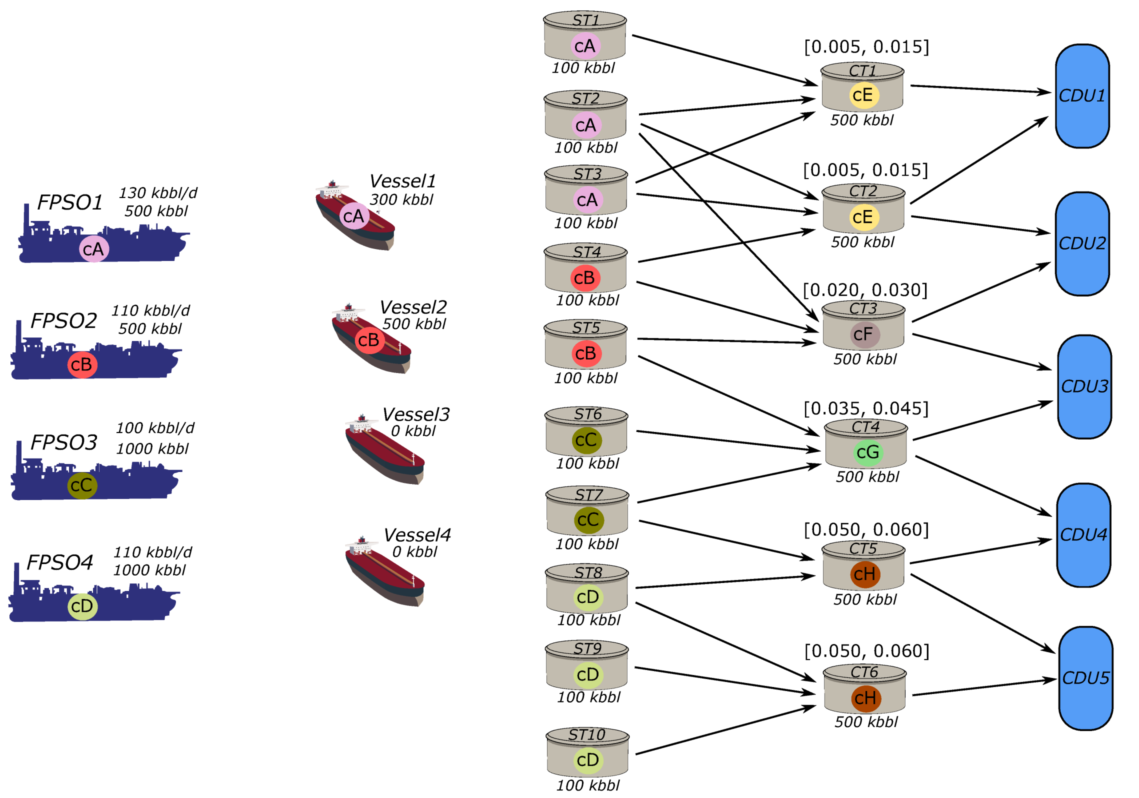
Capacity Bounds	kbbl
FPSO	[200, 1500]
VESSEL	[0, 1000]
ST	[0, 1000]
CT	[0, 1000]

	Initial Operation
Vessel1	unload-ST1
Vessel2	wait-Terminal
Vessel3	wait-Terminal
Vessel4	wait-Terminal

Crude Demand	kbbl
CT1	[800, 1200]
CT2	[800, 1200]
CT3	[800, 1200]
CT4	[800, 1200]

Planning Horizon = 15 days

Figure 23 – Illustration of network for original instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D.



Crude/Property	S	$/kbbl
cA	0.010	10,000
cB	0.030	8,000
cC	0.045	6,500
cD	0.060	5,000
cE	0.012	9,500
cF	0.026	8,500
cG	0.041	7,000
cH	0.056	5,500

Flow Rate Bounds	kbbl/d
FPSO-VESSEL	[0, 500]
VESSEL-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

Capacity Bounds	kbbl
FPSO	[200, 1500]
VESSEL	[0, 1000]
ST	[0, 1000]
CT	[0, 1000]

	Initial Operation
Vessel1	unload-ST1
Vessel2	wait-Terminal
Vessel3	wait-Terminal
Vessel4	wait-Terminal

Crude Demand	kbbl
CT1	[800, 1200]
CT2	[800, 1200]
CT3	[800, 1200]
CT4	[800, 1200]
CT5	[800, 1200]
CT6	[800, 1200]

Planning Horizon = 15 days

Figure 24 – Illustration of network for original instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D.

APPENDIX C – CLUSTERS OF INSTANCES

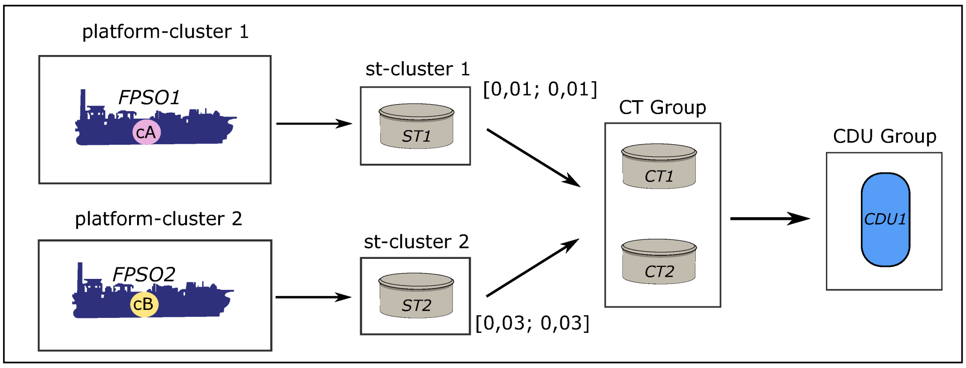


Figure 25 – Clustering for instance 2F-2V-2ST-2CT-1CDU-2C-1P-15D considering 2 clusters.

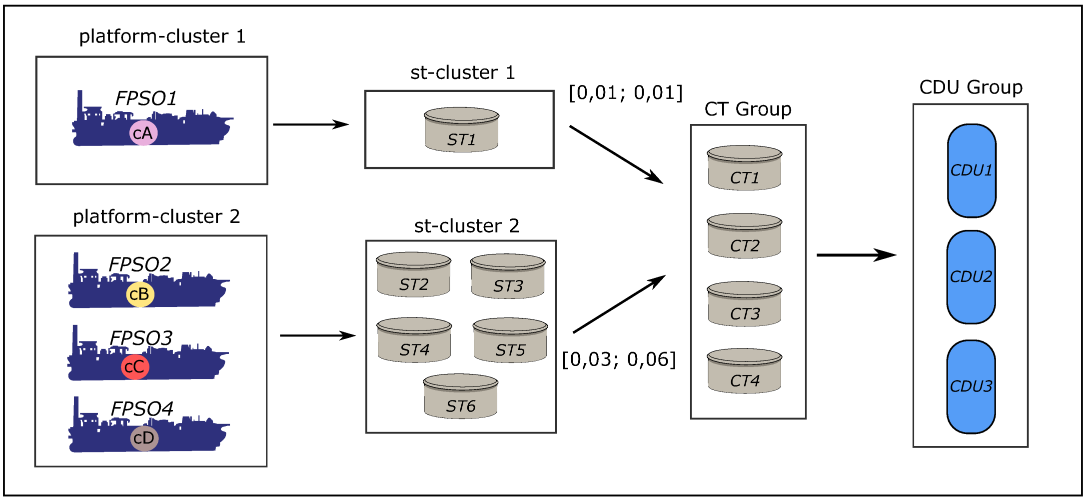


Figure 26 – Clustering for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D considering 2 clusters.

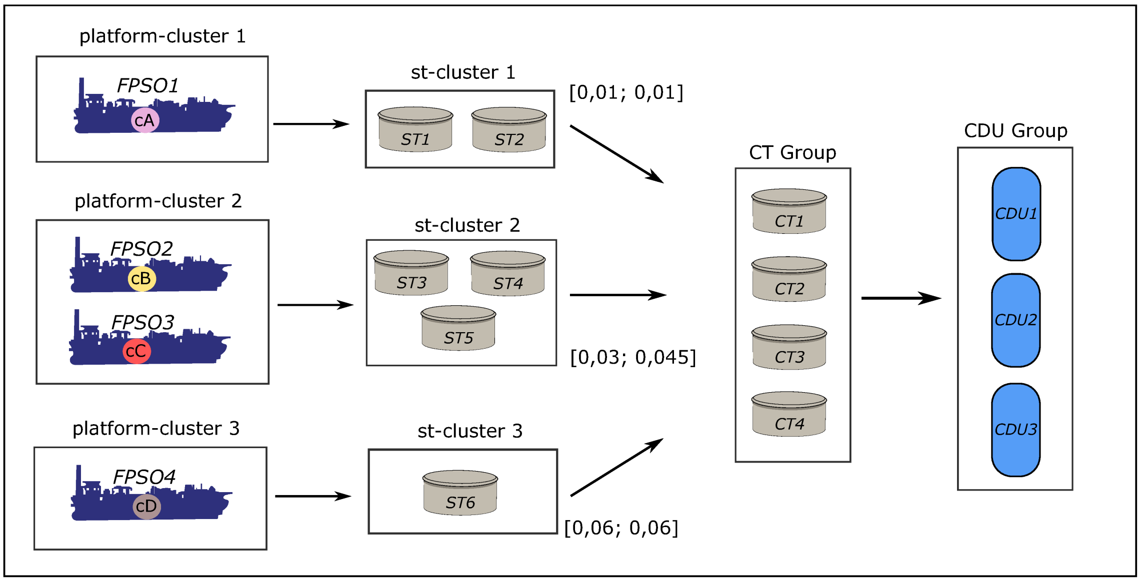


Figure 27 – Clustering for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D considering 3 clusters.

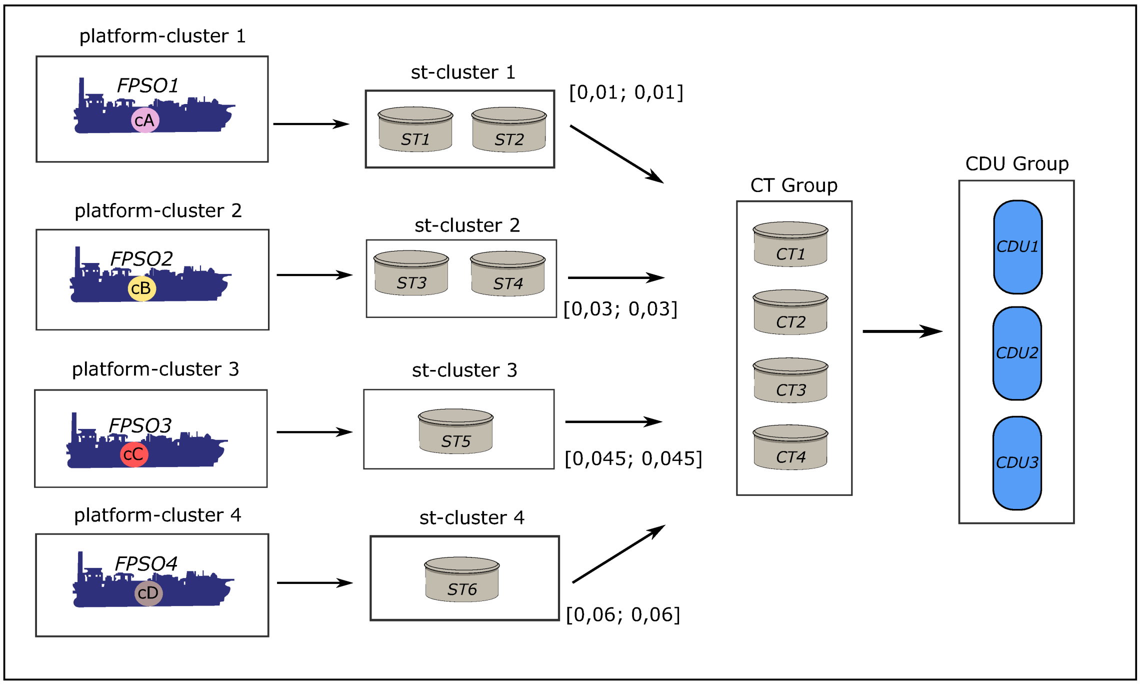


Figure 28 – Clustering for instance 4F-4V-6ST-4CT-3CDU-8C-1P-15D considering 4 clusters.

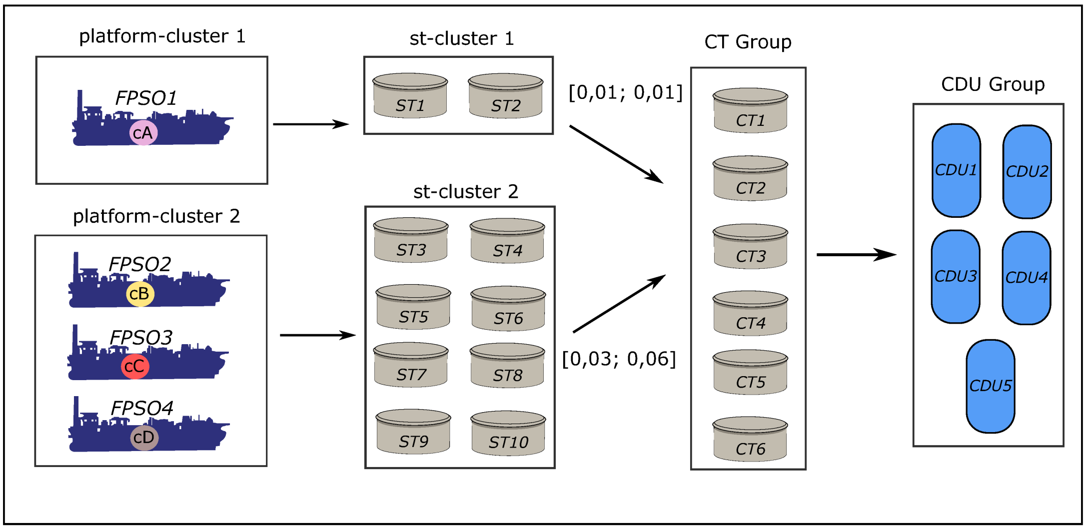


Figure 29 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D considering 2 clusters.

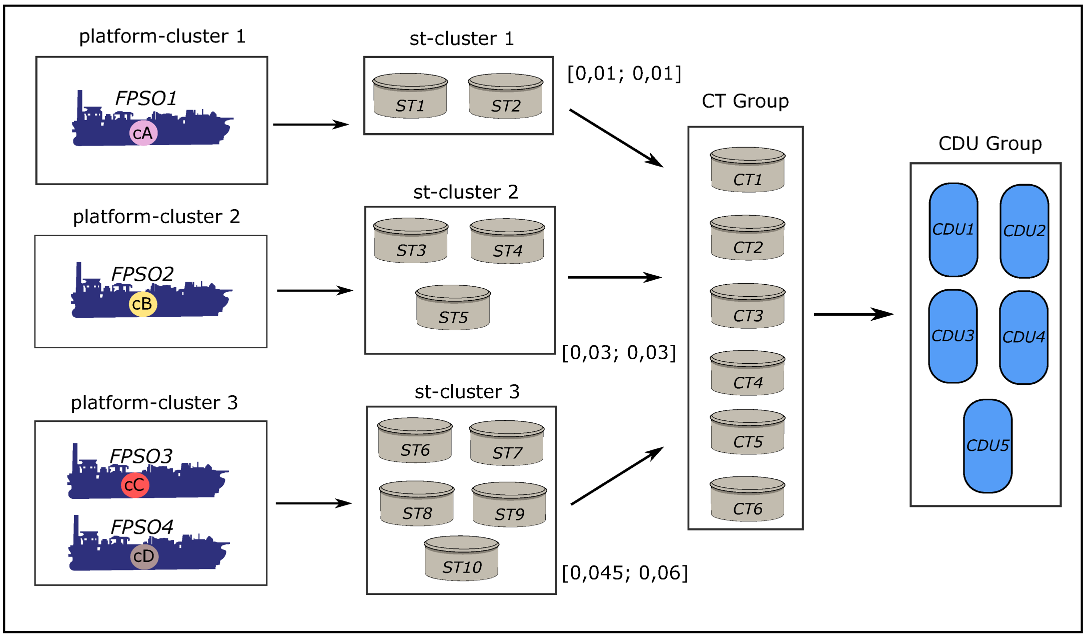


Figure 30 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D considering 3 clusters.

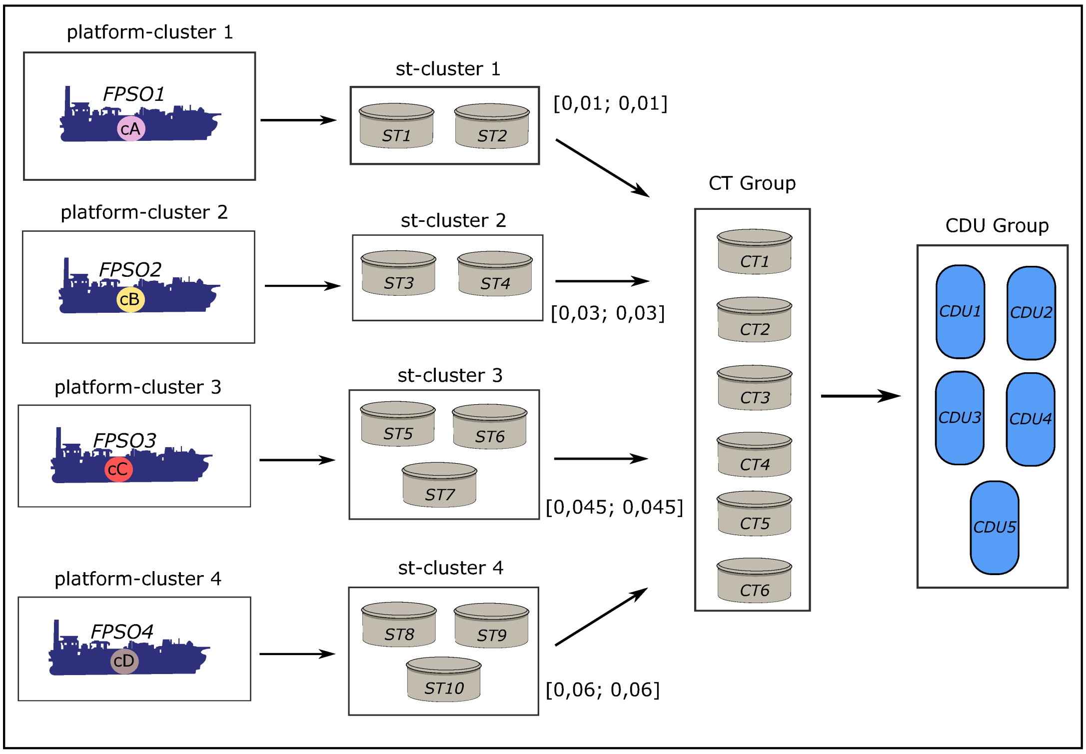
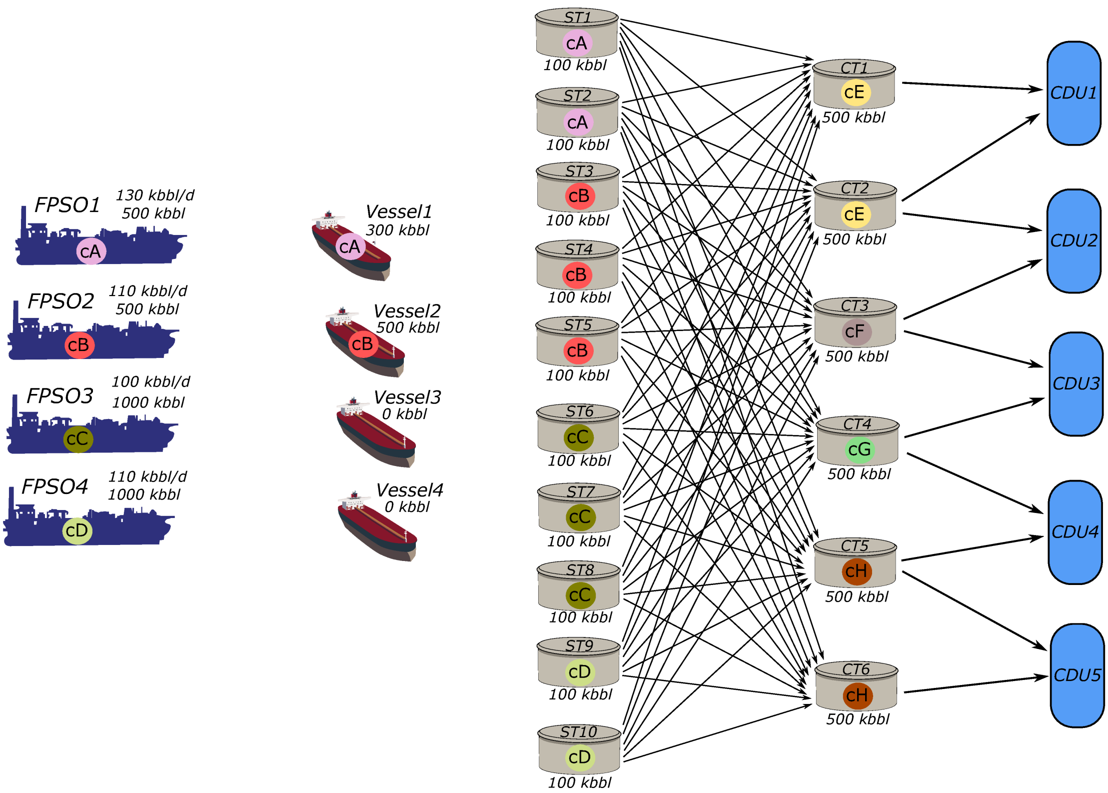


Figure 31 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-1P-15D considering 4 clusters.

APPENDIX D – NEW INSTANCES



Crude/Property	S	T	U	V	$/kbbl
cA	0.010	0.0050	0.200	0.150	10,000
cB	0.030	0.0150	0.067	0.050	8,000
cC	0.045	0.0225	0.044	0.033	6,500
cD	0.060	0.0300	0.033	0.025	5,000
cE	0.012	0.0060	0.166	0.125	9,500
cF	0.026	0.0130	0.077	0.057	8,500
cG	0.041	0.0205	0.049	0.036	7,000
cH	0.056	0.0280	0.035	0.026	5,500

Capacity Bounds	kbbl
FPSO	[200, 1500]
VESSEL	[0, 1000]
ST	[0, 1000]
CT	[0, 1000]

Flow Rate Bounds	kbbl/d
FPSO-VESSEL	[0, 500]
VESSEL-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

Crude Demand	kbbl
CT1	[800, 1200]
CT2	[800, 1200]
CT3	[800, 1200]
CT4	[800, 1200]
CT5	[800, 1200]
CT6	[800, 1200]

	Initial Operation
Vessel1	unload-ST1
Vessel2	wait-Terminal
Vessel3	wait-Terminal
Vessel4	wait-Terminal

Planning Horizon = 15 days

	S	T	U	V
CT1	[0.005, 0.015]	[0.0025, 0.075]	[0.133, 0.400]	[0.100, 0.300]
CT2	[0.005, 0.015]	[0.0025, 0.075]	[0.133, 0.400]	[0.100, 0.300]
CT3	[0.020, 0.030]	[0.0100, 0.0150]	[0.066, 0.100]	[0.050, 0.075]
CT4	[0.035, 0.045]	[0.0175, 0.0225]	[0.044, 0.057]	[0.033, 0.042]
CT5	[0.050, 0.060]	[0.0250, 0.0300]	[0.033, 0.040]	[0.025, 0.030]
CT6	[0.050, 0.060]	[0.0250, 0.0300]	[0.033, 0.040]	[0.025, 0.030]

Figure 32 – Illustration of network for original instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D considering all connections between STs and CTs.

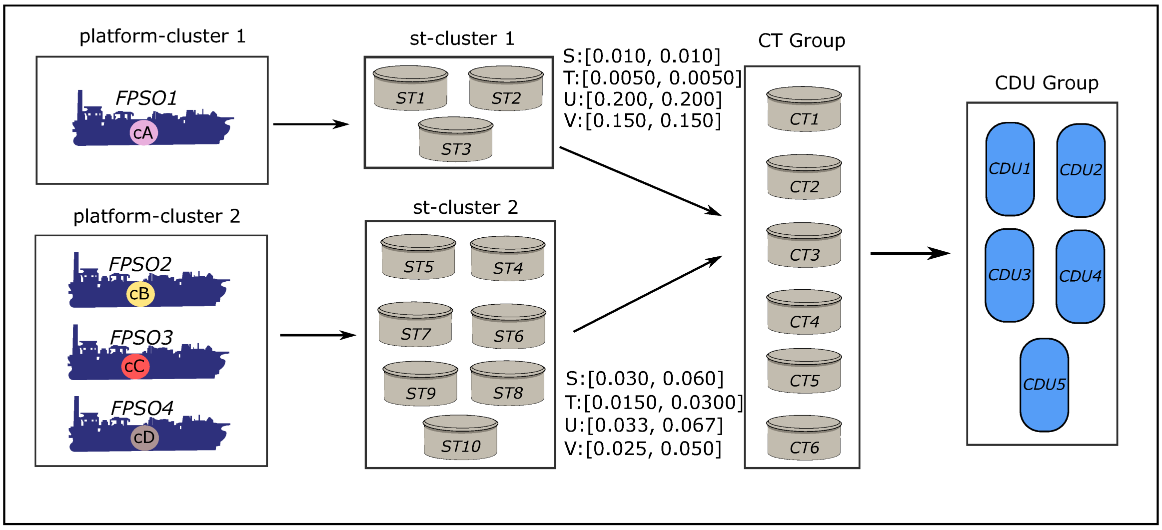


Figure 33 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D considering 2 clusters.

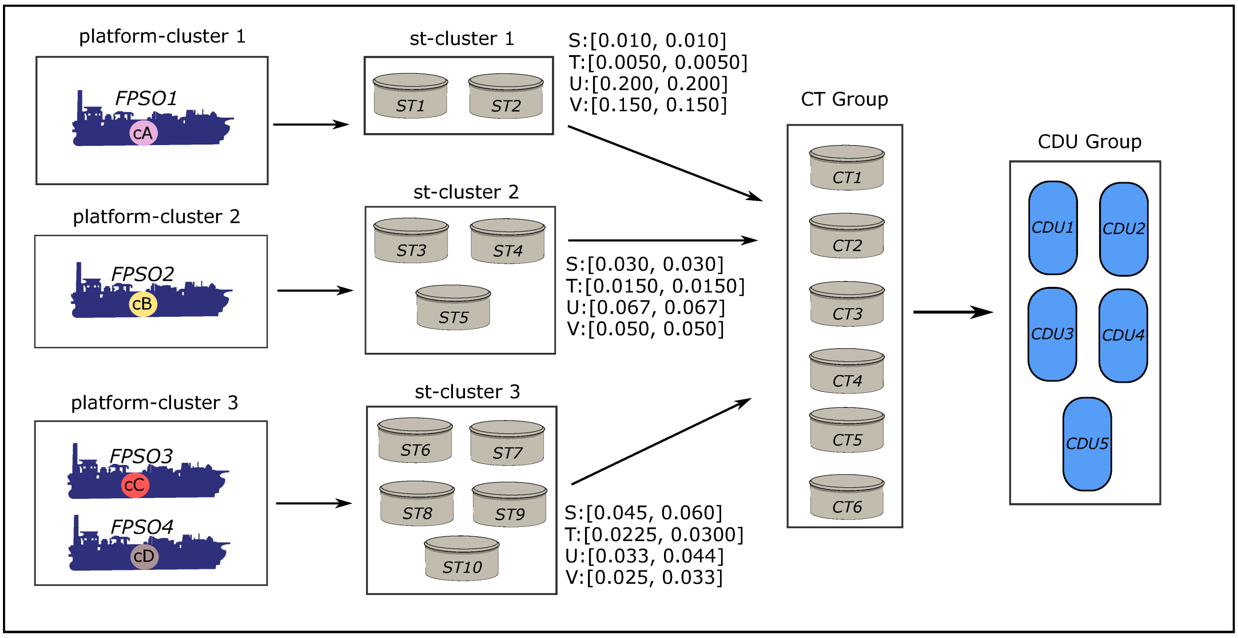


Figure 34 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D considering 3 clusters.

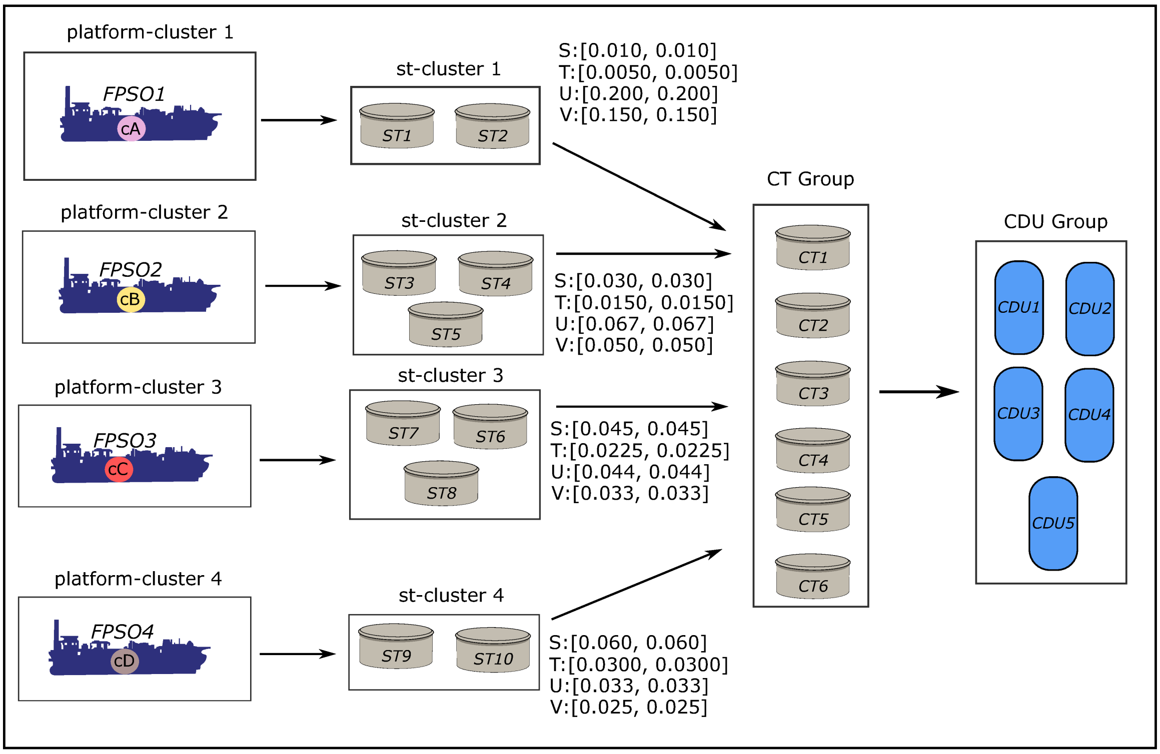
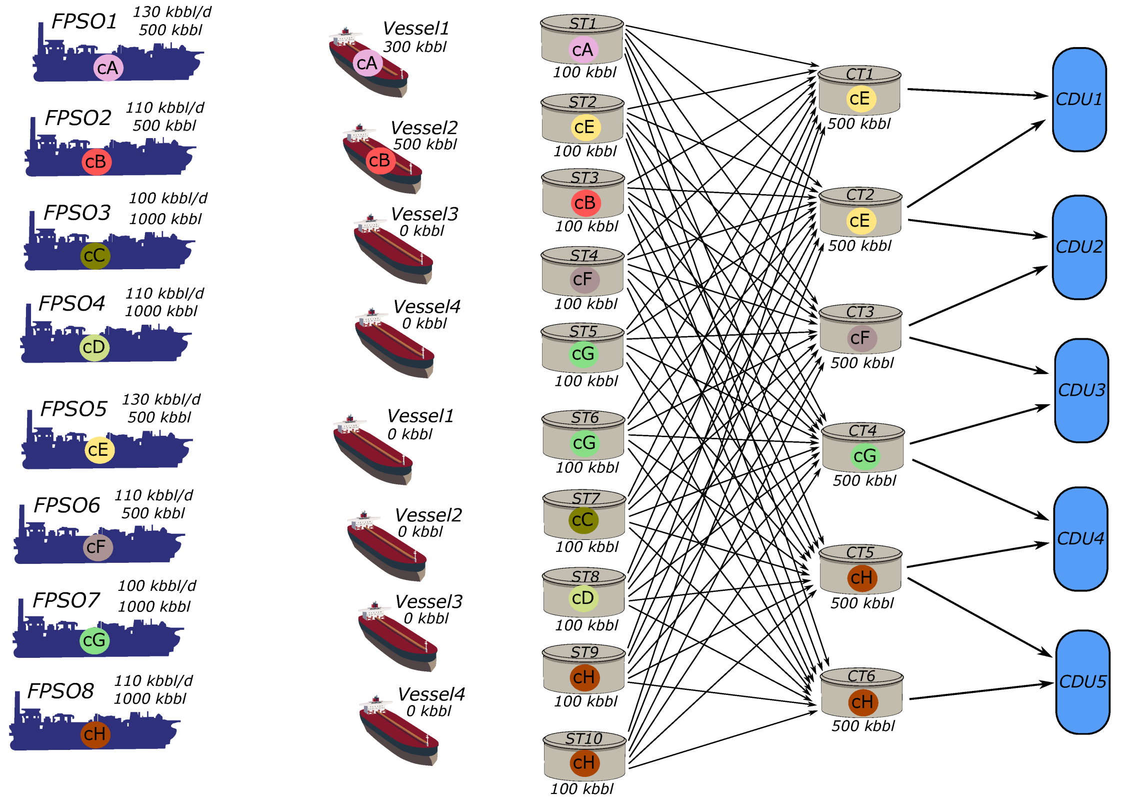


Figure 35 – Clustering for instance 4F-4V-10ST-6CT-5CDU-8C-4P-15D considering 4 clusters.



Crude/Property	S	T	U	V	$/kbbl
cA	0.010	0.0050	0.200	0.150	10,000
cB	0.030	0.0150	0.067	0.050	8,000
cC	0.045	0.0225	0.044	0.033	6,500
cD	0.060	0.0300	0.033	0.025	5,000
cE	0.012	0.0060	0.166	0.125	9,500
cF	0.026	0.0130	0.077	0.057	8,500
cG	0.041	0.0205	0.049	0.036	7,000
cH	0.056	0.0280	0.035	0.026	5,500

Capacity Bounds	kbbl
FPSO	[200, 2500]
VESSEL	[0, 1000]
ST	[0, 1000]
CT	[0, 1000]

Flow Rate Bounds	kbbl/d
FPSO-VESSEL	[0, 500]
VESSEL-ST	[0, 500]
ST-CT	[0, 500]
CT-CDU	[50, 500]

Crude Demand	kbbl
CT1	[500, 1500]
CT2	[500, 1500]
CT3	[500, 1500]
CT4	[500, 1500]
CT5	[500, 1500]
CT6	[500, 1500]

	S	T	U	V
CT1	[0.005, 0.015]	[0.0025, 0.075]	[0.133, 0.400]	[0.100, 0.300]
CT2	[0.005, 0.015]	[0.0025, 0.075]	[0.133, 0.400]	[0.100, 0.300]
CT3	[0.020, 0.030]	[0.0100, 0.0150]	[0.066, 0.100]	[0.050, 0.075]
CT4	[0.035, 0.045]	[0.0175, 0.0225]	[0.044, 0.057]	[0.033, 0.042]
CT5	[0.050, 0.060]	[0.0250, 0.0300]	[0.033, 0.040]	[0.025, 0.030]
CT6	[0.050, 0.060]	[0.0250, 0.0300]	[0.033, 0.040]	[0.025, 0.030]

	Initial Operation
Vessel1	unload-ST1
Vessel2	wait-Terminal
Vessel3	wait-Terminal
Vessel4	wait-Terminal
Vessel5	wait-Terminal
Vessel6	wait-Terminal
Vessel7	wait-Terminal
Vessel8	wait-Terminal

Planning Horizon = 15 days

Figure 36 – Illustration of network for original instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D considering all connections between STs and CTs.

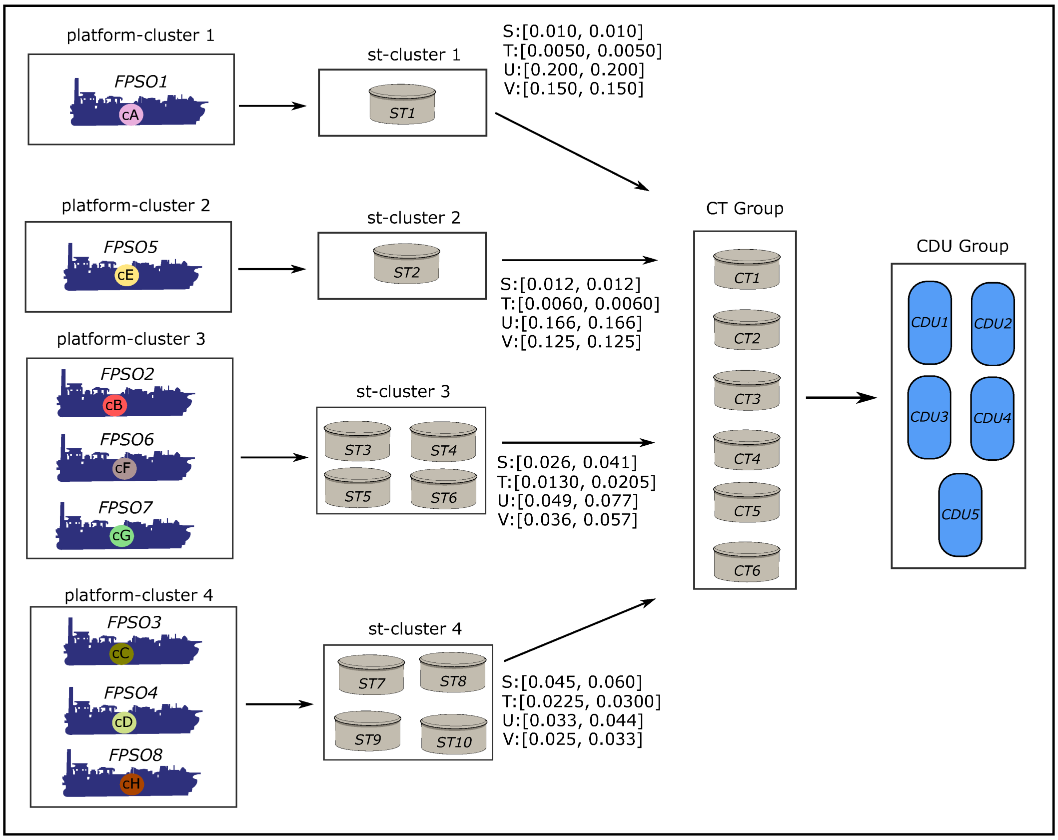


Figure 37 – Clustering for instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D considering 4 clusters.

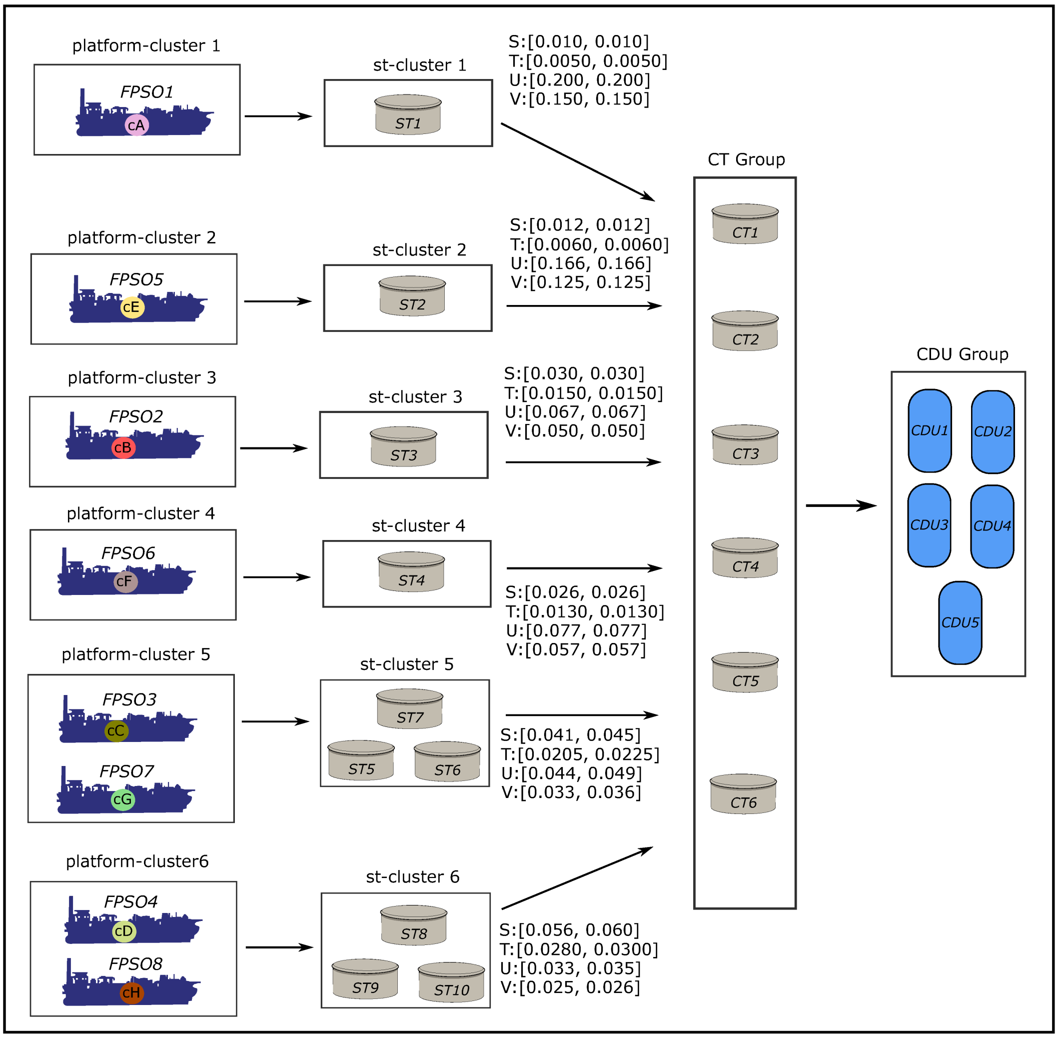


Figure 38 – Clustering for instance 8F-8V-10ST-6CT-5CDU-8C-4P-15D considering 6 clusters.