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

Leandro Pohlmann Rocha

Linear Relaxations of Bilinear Terms for the Operational Management of Crude Oil Supply

Florianópolis
2021

Leandro Pohlmann Rocha

Linear Relaxations of Bilinear Terms for the Operational Management of Crude Oil Supply

Dissertação submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas da Universidade Federal de Santa Catarina para a obtenção do título de Mestre em Engenharia de Automação e Sistemas

Advisor: Prof. Eduardo Camponogara, Dr.
Co-advisor: Prof. Laio Oriel Seman, Dr.

Florianópolis
2021

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

Rocha, Leandro Pohlmann
Linear Relaxations of Bilinear Terms for theOperational Management of Crude Oil Supply / Leandro Pohlmann Rocha ; orientador, Eduardo Camponogara, coorientador, Laio Oriel Seman, 2021.
92 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, Florianópolis, 2021.

Inclui referências.

1. Engenharia de Automação e Sistemas. 2. Linear Relaxation. 3. McCormick Envelopes. 4. Piecewise McCormick Envelopes. 5. Multiparametric Disaggregation. I. Camponogara, Eduardo. II. Seman, Laio Oriel. III. Universidade Federal de Santa Catarina. Programa de Pós Graduação em Engenharia de Automação e Sistemas. IV. Título.

Leandro Pohlmann Rocha

Linear Relaxations of Bilinear Terms for the Operational Management of Crude Oil Supply

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

Prof. Felipe Gomes de Oliveira Cabral, Dr.
Universidade Federal de Santa Catarina

Prof. Marcus Rolf Peter Ritt, Dr.
Universidade Federal do Rio Grande do Sul

Prof. Pedro Augusto Munari Junior, Dr.
Universidade Federal de São Carlos

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

ICP-Edu
Documento assinado digitalmente
Werner Kraus Junior
Data: 12/07/2021 16:53:54-0300
CPF: 531.085.239-53
Verifique as assinaturas em https://v.ufsc.br

Prof. Werner Kraus Junior, Dr.
Coordenador do Programa

ICP-Edu
Documento assinado digitalmente
Eduardo Camponogara
Data: 12/07/2021 12:06:47-0300
CPF: 513.988.180-20
Verifique as assinaturas em https://v.ufsc.br

Prof. Eduardo Camponogara, Dr.
Orientador

Florianópolis, 3 de Julho de 2021.

RESUMO

O gerenciamento da cadeia de suprimento de petróleo não refinado envolve o gerenciamento de operações de descarga e transferência em terminais, controle de estoque e mistura de petróleo bruto para atender às demandas da refinaria. O planejamento das operações torna-se mais desafiador, pois as viagens dos navios devem ser programadas com precisão para dar vazão às plataformas de produção do petróleo. Tradicionalmente, métodos de otimização matemática são utilizados para auxiliar na gestão operacional através de programação não linear inteira mista (MINLP). Indiscutivelmente, a dificuldade computacional do problema surge devido ao tamanho e à combinação de decisões discretas com restrições não lineares, constituídas por termos bilineares que modelam as operações de mistura do petróleo. No que diz respeito às funções não lineares, este trabalho contribui com a avaliação de técnicas distintas de aproximação linear dos termos bilineares, especificamente: *McCormick envelopes, univariate* e *bivariate piecewise McCormick, multiparametric disaggregation* e *normalized multiparametric disaggregation*. Os métodos de relaxação geram um problema de programação linear inteira mista (MILP), que pode ser combinado com um algoritmo de programação não linear local (PNL) para atingir um cronograma de operações viável. Concluímos com uma comparação entre essas abordagens de relaxação juntamente com abordagens MINLP usuais, e demonstramos resultados computacionais em instâncias do problema. A relaxação utilizando *multiparametric disaggregation* produz tempos de solução menores para resultados similares comparativamente aos métodos de otimização global comumente utilizados.

Palavras-chaves: Relaxação Linear, McCormick Envelopes, Piecewise McCormick, Multiparametric Disaggregation, Normalized Multiparametric Disaggregation.

RESUMO EXPANDIDO

Introdução

O gerenciamento operacional da cadeia de suprimento de petróleo não refinado envolve o sequenciamento de operações de descarga e transferência em terminais, controle de estoque e mistura de petróleo bruto para atender às demandas da refinaria. Tradicionalmente, métodos de otimização matemática são utilizados para auxiliar na gestão operacional através de programação não linear inteira mista (MINLP). Indiscutivelmente, a dificuldade computacional do problema surge devido ao tamanho e à combinação de decisões discretas com restrições não lineares, constituídas por termos bilineares que modelam as operações de mistura do petróleo. No que diz respeito às funções não lineares, este trabalho contribui com a avaliação de técnicas distintas de aproximação linear dos termos bilineares, especificamente: *McCormick envelopes, univariate* e *bivariate piecewise McCormick, multiparametric disaggregation* e *normalized multiparametric disaggregation*. Os métodos de relaxação geram um problema de programação linear inteira mista (MILP), que pode ser combinado com um algoritmo de programação não linear local (PNL) para atingir um cronograma de operações viável.

Objetivos

O objetivo desta dissertação consiste em avaliar métodos de relaxação para termos bilineares em um estudo de caso do Gerenciamento Operacional da Cadeia de Suprimento de Petróleo Bruto (OMCOS). Além disso, os objetivos específicos são os seguintes: (a) Identificar e discutir o estado-da-arte em métodos de relaxação, capazes de lidar com o problema em questão; (b) Aplicar e analisar cada método aplicado ao estudo de caso, avaliando a qualidade da sua solução e o tempo computacional; (c) Selecionar o melhor método ou combinação de métodos, considerando a estrutura do problema e comparando seus resultados com *solvers* globais.

Metodologia

Os termos bilineares que aparecem nas restrições de mistura são relaxados por meio das técnicas de relaxação propostas. As instâncias MINLP originais do OMCOS são resolvidas até obter uma solução ótima ou até atingir o limite de 10h de execução, utilizando um *solver* global (Gurobi e SCIP). Estes servem como referência para comparar aos métodos de relaxação aplicados em um algoritmo de decomposição MILP-PNL. Este algoritmo é avaliado em três cenários distintos: (a) aplica-se a técnica de relaxação em ambos os termos bilineares; (b) a técnica de relaxação é aplicada apenas no termo bilinear do lado esquerdo da igualdade, enquanto um envelope *McCormick* padrão substitui o termo bilinear do lado direito da igualdade; (c) a técnica de relaxação é aplicada apenas no termo bilinear do lado direito da igualdade, enquanto um envelope *McCormick* padrão substitui o termo bilinear do lado esquerdo da igualdade. As técnicas são aplicadas com variação de parâmetros que permitem verificar a solução com relaxações mais e menos apertadas. Em uma relaxação mais apertada é esperado obter melhores resultados mas com um esforço computacional maior. Para os experimentos computacionais, foram criadas três instâncias distintas do problema.

Resultados e Discussão

A restrição de mistura do OMCOS possui uma estrutura única que consiste em dois termos bilineares conectados por uma restrição de igualdade. Assim, os resultados mostraram que foi possível melhorar o tempo computacional restringindo apenas um dos termos bilineares, enquanto o outro permaneceu limitado por um simples Envelope *McCormick*. A decomposição do MILP-PNL, quando a relaxação foi aplicada em apenas um dos termos bilineares, produziu soluções comparáveis às obtidas pelo problema original utilizando solvers globais. Examinando o *Piecewise McCormick*, embora o PNL possa obter um limite inferior ligeiramente melhor aumentando o número de partições, o esforço computacional é muito alto. Isso sugere que 4 e 8 são os números de partições recomendados, o que vai de encontro à literatura encontrada. Já para o *Multiparametric Disaggregation*, os resultados numéricos mostraram melhores resultados com um número maior de partições. A melhor solução encontrada para duas instâncias, por exemplo, foi obtida aplicando este método de relaxação com 500 partições equivalentes. Em relação às diferentes estratégias de relaxação para termos bilineares aplicadas às instâncias neste estudo, os resultados sugerem que o *Multiparametric Disaggregation* é capaz de obter bons resultados para as instâncias em um tempo computacional mais rápido. O método também produziu resultados semelhantes às soluções ótimas encontradas resolvendo o MINLP diretamente com o solver Gurobi, e com melhor performance quando comparado ao solver SCIP.

Considerações Finais

Esta dissertação contribui com a literatura ao trazer novos *insights* para a solução do Gerenciamento Operacional da Cadeia de Suprimento de Petróleo Bruto com termos bilineares. Ao apresentar diferentes abordagens para a solução do problema, espera-se que resultados utilizando *Multiparametric Disaggregation* possam atingir melhor qualidade e tempo computacional em comparação a *solvers* de otimização global. Os resultados também indicam que trabalhos futuros podem tirar proveito da formulação relaxada para aplicar métodos iterativos buscando a redução do domínio das variáveis.

Palavras-chaves: Relaxação Linear, McCormick Envelopes, Piecewise McCormick, Multiparametric Disaggregation, Normalized Multiparametric Disaggregation.

ABSTRACT

The operational management of crude oil supply entails solving large-scale mixed-integer nonlinear programming (MINLP) problems, accounting for unloading and transfer operations in terminals, inventory control, and blending of crude oils to meet the demands from the refinery. In offshore oil assets, the planning of operations becomes more challenging because vessel trips should be scheduled to relieve production platforms from crudes which are transferred to the terminals. Arguably, the problem's computational hardness emerges from its size and the combination of discrete decisions with nonlinear constraints, consisting of bilinear terms that model blending operations. Concerning the nonlinear functions, this work contributes by evaluating distinct linear approximation techniques of the bilinear terms, namely: standard McCormick envelopes, univariate and bivariate piecewise McCormick, multiparametric disaggregation, and normalized multiparametric disaggregation. The methods yield a mixed-integer linear programming (MILP) relaxation, which can be combined with a local nonlinear programming (NLP) algorithm to reach a feasible schedule of operations. We conclude with a comparison among these relaxation approaches along with common MINLP approaches and report computational results on instances of the problem. The relaxation derived using the multiparametric disaggregation technique is shown to yield faster solution times for similar optimality gaps comparatively to general global optimization solvers.

Key-words: Linear Relaxation, McCormick Envelopes, Piecewise McCormick Envelopes, Multiparametric Disaggreagation, Normalized Multiparametric Disaggreagation, Resource Constraints.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

CDU	Crude Distillation Unit
COS	Crude Oil Scheduling
CT	Charging Tank
FPSO	Floating Production, Storage, and Offloading unit
LP	Linear Programming
MCK	McCormick Envelopes
MDT	Multiparametric Disaggregation Technique
MILP	Mixed-Integer Linear Programming
MINLP	Mixed-Integer Nonlinear Programming
MIR	Maritime Inventory Routing
NLP	Non-Linear Programming
NMDT	Normalized Multiparametric Disaggregation Technique
OMCOS	Operational Managment of Crude Oil Supply
ST	Storage Tank

CONTENTS

1 INTRODUCTION

1.1 MOTIVATION

The petrochemical industry is arguably strategic for our society. Crude oil is transformed into valuable products present in our everyday life, such as plastics, oils, fuels, asphalt, and many chemicals utilized as raw material across various industries. To achieve that, vertically integrated oil companies must deal with a broad range of complex physical and chemical processes, from production and transportation of crude oil to storage and refining. They operate in an extremely competitive and dynamic market where substantial variations in product demand and crude oil prices occur frequently. Consequently, their profit margins are comparatively small regarding other industries, which raises a necessity to realize significant cost savings. Another critical factor comes from the increased demand for the optimal use of natural resources, with operations being regulated under strict safety, environmental, and governmental rules. The management of this intricate supply chain has proven to be a considerable challenge faced by these companies (ROCHA; GROSSMANN; ARAGÃO, 2009).

As illustrated in Figure 1, the crude oil supply chain starts with the extraction performed by floating production, storage, and offloading units (FPSOs). After the production stage, vessels transfer the crude oil from the deep-water offshore units to onshore terminals since oil pipelines are not technically feasible or economically viable in most cases. After arriving in a terminal, vessels unload through a pipeline to storage tanks (STs). Since different types of crude oil exist, mixtures are expected at this point. The crude oil reaches the refinery stage through a pipeline network connected from the storage tanks to charging tanks (CTs) responsible for continuously feeding the crude oil distillation units (CDUs). As becomes evident, production planning and scheduling represent a fundamental and prevalent tool in managing the crude oil supply chain.

Historically, the oil industry has been extensively studying and implementing mathematical programming techniques to address management issues (BODINGTON; BAKER, 1990). Particularly for planning and scheduling operations, Linear Programming (LP) and Mixed-Integer Linear Programming (MILP) have been employed since they are relatively straightforward to model and solve. However, considering that the crude oil production involves units operating in both batch and continuous modes, and the mixing of crudes has to be handled simultaneously, modeling is not a trivial task (KARUPPIAH; FURMAN; GROSSMANN, 2008). Nonlinear equations are often required when modeling continuous-time representations (KARUPPIAH; FURMAN; GROSSMANN, 2008; MOURET; GROSSMANN; PESTIAUX, 2011) or blending equations for mass balance. The blending or mixing equations are the most common type of constraints in Chemical Engineering (CASTRO, 2015). They contain the product of two decision variables forming bilinear terms, which, when formulated for optimization problems, give rise to

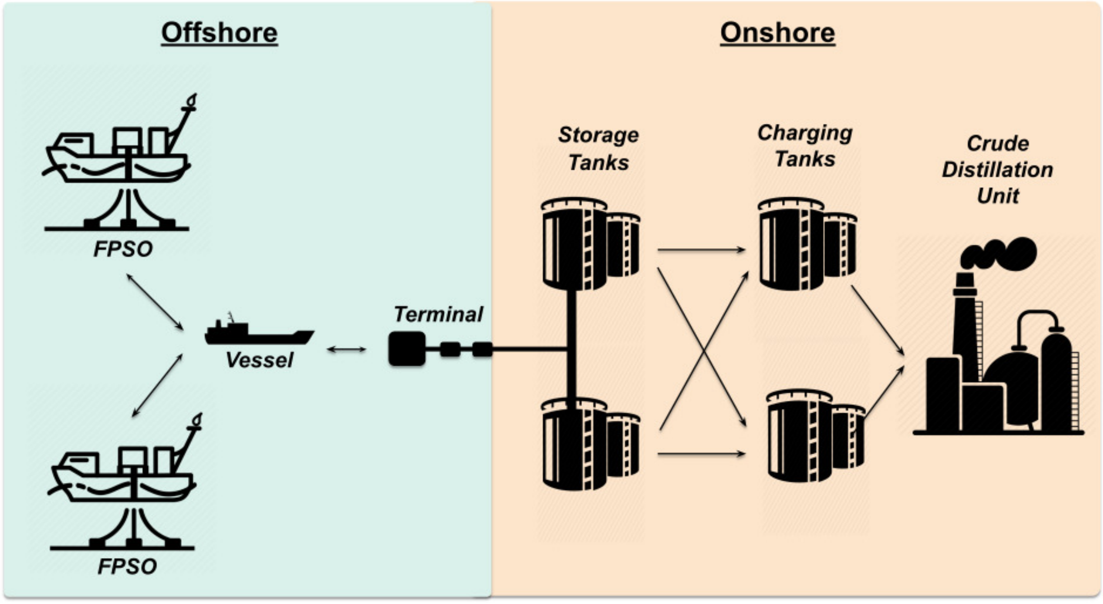

Figure 1 – The crude oil supply chain.

Mixed-Integer Nonlinear Programming (MINLP) formulations. In turn, these formulations lead to multiple local solutions, which gradient-based solvers are unable to guarantee optimality and are extremely hard to solve (QUESADA; GROSSMANN, 1995).

Such is the case of the MINLP model proposed by Assis et al. (2019) for the operational management of crude oil supply, where blending constraints containing bilinear terms are the only nonlinear nonconvex constraint in the formulation. Efforts on handling the difficulty of solving this type of MINLP problems, with nonconvex bilinear equations, motivate the study of strategies to obtain significant savings in the computational effort while handling large instances.

Usually, global optimization solvers have in common the generation of linear (LP) or mixed-integer linear (MILP) relaxations of the original problem (SAHINIDIS, 1996). One way of relaxing a nonconvex function is to obtain the convex underestimator and concave overestimator of the function over its domain known as McCormick envelopes (MCCORMICK, 1976). McCormick envelopes coupled with spatial branch and bound search frameworks have been the basis for many global optimization techniques. A tighter relaxation is crucial to obtain a fast convergence, and this is highly dependent on the bounds of the bilinear terms, improving as their domain is reduced (CASTRO, 2016). Simultaneous variable partitioning using piecewise McCormick envelopes (BERGAMINI; AGUIRRE; GROSSMANN, 2005), multiparametric disaggregation (KOLODZIEJ; CASTRO; GROSSMANN, 2013), or normalized multiparametric disaggregation (CASTRO, 2016) can provide better approximations. Still, since additional binary variables are necessary, it may become too hard computationally. Therefore, each application requires specific verifications.

Motivated to assess the application of piecewise McCormick, multiparametric disaggregation and normalized multiparametric disaggregation relaxions for the bilinear terms present in the operational management of crude oil supply proposed by Assis et al. (2019), this work aims to evaluate the quality of solution and computational effort when using such a technique against common MINLP approaches. The evaluation strategy yields a mixed-integer linear programming relaxation, which can be combined with a local nonlinear programming (NLP) algorithm to reach a feasible schedule of operations. We conclude with a comparison among these relaxation approaches along with common MINLP approaches, reporting computational results on instances of the problem.

1.2 OBJECTIVES

The objective of the present thesis consists in evaluating relaxation methods for bilinear terms in a Operational Management of Crude Oil supply case study. The problem of concern entails solving large-scale mixed-integer nonlinear programming problems. Since it is arguably computational hard to obtain solutions considering their size, discrete variables, and nonlinear constraints consisting of bilinear terms, it is interesting to assess the performance and quality of results achieved with existing methods. Specifically:

- Identify and discuss state-of-the-art relaxation methods that are capable of dealing with the problem of concern;
- Apply and analyze each identified method in the problem of concern, assessing their solution quality and computational time;
- Select the best method or combination of methods, considering the structure of the problem, by comparing their outcomes with global solvers.

Therefore, this thesis contributes to the literature by bringing new insights into solving the operational management of crude oil supply problem with bilinear terms. By presenting different approaches to the problem's solution, sufficiently good results are expected to be reached with a trade-off between quality and computational time.

1.3 OUTLINE OF THE THESIS

The thesis contains six chapters, of which the first is this introduction providing the main motivation for carrying out the work, along with the objectives.

In Chapter 2, the Operational Management of Crude Oil supply is presented. A short account of previous related works is introduced before stating the problem and the mathematical formulation.

Next, the necessary background for relaxation methods of bilinear terms can be found in Chapter 3. It comprises the important concepts relevant for the application of such methods, namely the McCormick envelopes, Piecewise McCormick Envelops with univariate and bivariate partitions, Multiparametric Disaggregation, and Normalized Multiparametric Disaggregation.

Further, Chapter 4 outlines the strategy and algorithm employed to analyze the application of each method on the problem of concern. The models generated when remodeling the bilinear terms of the blending constraint are exposed.

In Chapter 5, the results yielded by each relaxation approach, including performance and quality evaluation are discussed.

Finally, Chapter 6 concludes and provides reflections on the thesis, in addition to suggestions for future works.

2 OPERATIONAL MANAGEMENT OF CRUDE OIL SUPPLY

In this chapter, the operational management of crude oil supply is presented in reference to the work of Assis et al. (2019). Section 2.1 brings a summarized overview and relevant works in the literature. Further in Section 2.2, the complete mathematical model is detailed in-depth to provide the reader with enough knowledge concerning the problem.

2.1 OVERVIEW

Supply chain management consists of planning, managing, and coordinating resources and operations along the chain. Minimizing the overall cost while satisfying the customer demands with regard to quantity and time is the major objective (SIMCHI-LEVI et al., 2008). As for oil companies, Sahebi, Nickel and Ashayeri (2014) highlight the relevance of adopting supply chain management practices and decision supporting tools based on mathematical optimization to attain operational efficiency. Optimizing the management of this large-scale logistic network has created new challenges for oil industry managers, and is of interest to both academics and practitioners.

According to Sahebi, Nickel and Ashayeri (2014), operations in the petroleum chain can be divided and classified according to their position along the process into upstream, midstream and downstream as detailed in Table 1.

Table 1 – Segments 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)

Typical operations regarding the crude oil supply segment inside the entire petroleum chain, which will be addressed in the present work, integrates items of both upstream and midstream segments. They can be summarized as: offshore oil extraction and production, crude oil transportation and supply, crude oil storage, and finally the feed of CDUs in refineries. The midstream comprises the refining of crude oil into more elaborated products, while the downstream segment defines storage, distribution, and retail market of refined products (LIMA; RELVAS; BARBOSA-PÓVOA, 2016).

As seen in Fig. 2, a typical supply chain follows a hierarchical structure divided into three levels: Strategic, Tactical, and Operational (BARBOSA-PÓVOA, 2014). Focusing on the crude oil supply part, at the strategic level, decisions are made considering long-term (years) investments such as: infrastructure location and capacity; fleet sizing; and pipeline network design. Tactical arrangements cover refining these decisions in a medium-term perspective (months), considering in greater detail the production and distribution planning, inventory management, and inventory allocation (SAHEBI; NICKEL; ASHAYERI, 2014; BARBOSA-PÓVOA, 2014). Further, management at the operational level entails activities in a daily basis such as routing, scheduling of vessels, and scheduling of operations in terminals. In summary, the crude oil supply chain at the operational level is concerned with what is known as the Maritime Inventory Routing (MIR) (ASSIS; CAMPONOGARA, 2016) and Crude Oil Scheduling (COS) (MOURET; GROSSMANN; PESTIAUX, 2009) problems.

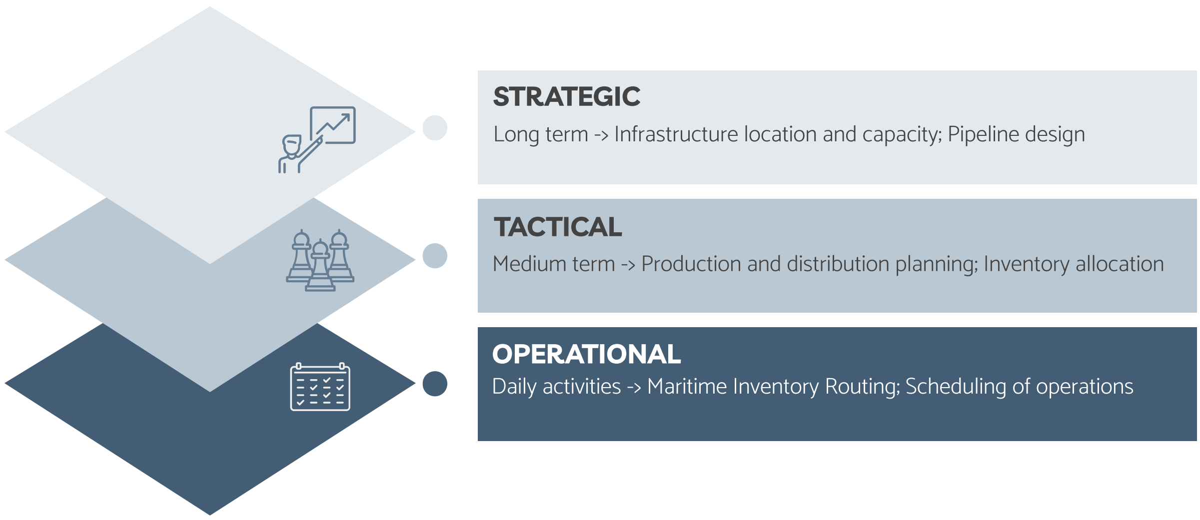

Figure 2 – A typical supply chain hierarchical structure

Maritime Inventory Routing comprises scheduling vessel trips between ports in order to meet product demands within limits of inventory at production and consumption ports. The work of Ronen (1983) was the first review on this subject. Further models began to combine inventory control in ports such as in Ronen (2002) and Camponogara and Plucenio (2014). A more detailed review on MIR problems is found in Christiansen et al. (2013). In Crude Oil Scheduling problems, the main goal is to meet CDU's demands in terms of 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 transfer between storage and charging tanks, and the feed of CDUs performed by charging tanks. The first work to address the COS problem is presented in Lee et al. (1996). The authors proposed and solved a discrete-time MILP model where blending constraints were not considered and replaced by a linear approximation. To the best of our knowledge, Aires et al. (2004) were the first to address the integrated problem of supplying crude oil from FPSOs to CDUs through a MILP formulation for strategic/tactical level decisions. The work tackled the allocation of crude oil from a set of platforms to a set of terminals in order to satisfy demands from the refinery in terms of volume and quality.

The solution of maritime inventory routing and crude oil scheduling when obtained separately may be unable to manage the access to common resources, such as storage tanks.

More precisely, the MIR generally assumes storage tanks are free for taking crudes from vessels that arrive periodically at the terminal. However, the availability of storage tanks are managed by the COS problem, resulting in a mismatch between the volumes and composition of crudes in those tanks. Furthermore, operational constraints that prohibits simultaneous inlet and outlet operations in tanks may not be satisfied when solving the MIR and COS problems independently. Such limitations motivated Assis et al. (2019) to propose the integration of the operational management of crude oil supply, taking into account elements of the MIR and COS problems at the operational level (i.e., from FPSOs to CDUs). The novel work took 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. The problem incorporates elements of maritime inventory routing and crude oil scheduling through an MINLP discrete-time formulation, named as the operational management of crude oil supply (OMCOS). Thus, the model stated by Assis et al. (2019) is utilized in this master's thesis.

2.2 PROBLEM STATEMENT

Fig.3 represents an instance of the Operational Management of Crude Oil Supply, composed by the following set of resources: FPSOs (*FPSO1* and *FPSO2*), vessels (*Vessel1*), storage tanks (*ST1* and *ST2*), charging tanks (*CT1* and *CT2*) and crude oil distillation units (*CDU1*). Moreover, the green arrows represent all operations that resources are capable to perform composed by WL (offloading operations), WU (unloading operations), WW (wait operations), WT (travel operations), WF (tank-to-tank feed operations), and WD (distillation operations).

In Fig. 3, *Vessel1* offloads crude oil from FPSOs through operations *WL*. Traveling operations of *Vessel1* between the crude oil terminal and FPSOs are coordinated by operations *WT*, which explains why the arrows are bidirectional.

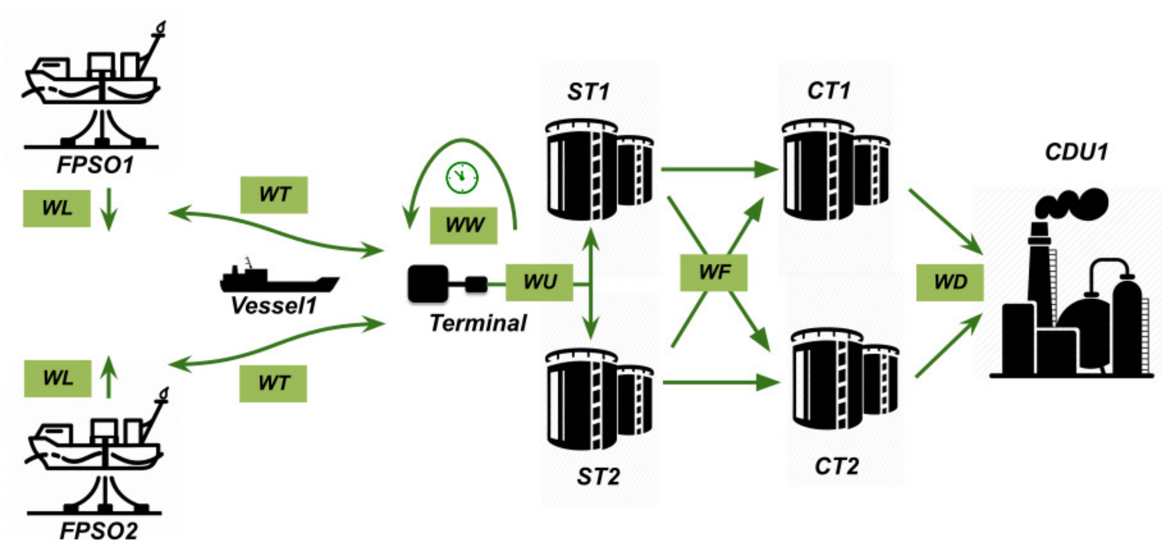

Figure 3 – The operational management of crude oil supply.

Approaching at the terminal, vessels can decide between directly unloading crude oil into storage tanks or waiting in case there is some restriction. *Vessel1* wait operations at the terminal are *WW*. Unloading into storage tanks *ST1* and *ST2* are performed by *Vessel1* through operations *WU*.

Since different types of crude oil exist and can be mixed in storage tanks, transfer operations *WF* to charging tanks are ruled by blending constraints. Charging tanks *CT1* and *CT2* will feed *CDU1* through operations *WD*. Transfer operations *WD* to CDUs are also regulated by blending constraints.

In addition to the operations performed by the resources, operational rules and constraints must be respected to assure feasibility given a real scenario. Also, the feed of crude oil delivered to CDUs over the planning horizon must satisfy a certain demand of total volume, as much as its composition bounded to a given range.

Briefly, the optimization problem consists of determining, for the desired planning horizon, the optimal schedule of operations associated with all resources to satisfy the demands of CDUs (i.e., both in terms of quality and quantity) while maximizing the gross margin. To this end, Assis et al. (2019) proposed a discrete-time MINLP model whose major decisions consist in selecting which operations take place at each time, the level of crudes in each resource, and the volumes of crude oil transferred between resources.

2.3 MATHEMATICAL MODEL

Before proposing the discrete-time MINLP model, the sets, parameters, variables, constraints, and objective function are presented below.

2.3.1 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 $\mathcal{RD}$: sets of resources, 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 $\mathcal{WD}$: sets of operations, 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 pair of consecutive 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 the different crude oils presented in the supply chain.
- $\mathcal{K}$: set of crude oil properties.
- $\mathcal{CL}$: set of cliques of conflicting operations. Let $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$ be an undirected 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 that ensure a coverage of all conflicting constraints.
- $\mathcal{WCL}_{cl}$: set of operations in a clique cl. $\mathcal{WCL}_{cl} \subset \mathcal{W}$ 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.

- $[\underline{CAP}_r,\ \overline{CAP}_r]$: lower and upper bounds on the volume stored in 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].
- $PR_{k,c}$: 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, after executing operation v in period i, performs 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.

2.3.2 Constraints

2.3.2.1 Material Balance and Resource Capacity

Considering all resources except CDUs ($r \in \mathcal{R} \setminus \mathcal{RD}$), for each period i there is an associated total level of crude oil ($lr_{i,r}$) and a specific level of each crude oil $c \in \mathcal{C}$ ($lcr_{i,r,c}$). These levels are obtained by Equations (2.1) - (2.8), through the difference between inlet and outlet flow of crudes on each resource (i.e., operations v in sets $\mathcal{I}_r$ and $\mathcal{O}_r$). The total volume and the specific volume of each crude oil c transferred between resources are defined, respectively, by $vt_{i,v}$ and $vct_{i,v,c}$.

Additionally, the initial levels (i.e., when $i = 1$) are handled by specific Equations, using parameters $CIL_{r,c}$ and TIL_r. These parameters correspond to the initial volume of crude oil c in resource r and the total initial volume of crude oil in resource r, respectively. Since it is assumed that CDUs have a daily flow of crude oil to the distillation units according to their processing capacity, the inventory tracking of this resource is neglected.

The FPSO inventory control is determined by Equations (2.1) through (2.4). Since FPSOs have no inlet operations but produce crude oil, the volume of crudes c, associated to a particular FPSO r, correspond to the daily production parameter $PROD_{r,c}$. The volume of crude transferred from this particular resource by output operations $v \in O_r$ is determined by variables $vt_{i,v}$ and $vct_{i,v,c}$. Note that $PROD_{r,c} = 0$ if FPSO r does not produce crude oil $c \in \mathcal{C}$.

$$lcr_{i,r,c} = CIL_{r,c} + PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c} \qquad r \in \mathcal{RF}, i \in \mathcal{T}, c \in \mathcal{C}, i = 1, \tag{2.1}$$

$$lcr_{i,r,c} = lcr_{i-1,r,c} + PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c} \qquad r \in \mathcal{RF}, i \in \mathcal{T}, c \in \mathcal{C}, i \neq 1, \tag{2.2}$$

$$lr_{i,r} = TIL_r + \sum_{c \in \mathcal{C}} PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vt_{i,v} \qquad r \in \mathcal{RF}, i \in \mathcal{T}, i = 1, \tag{2.3}$$

$$lr_{i,r} = lr_{i-1,r} + \sum_{c \in \mathcal{C}} PROD_{r,c} - \sum_{v \in \mathcal{O}_r} vt_{i,v} \qquad r \in \mathcal{RF}, i \in \mathcal{T}, i \neq 1. \tag{2.4}$$

Following the same reasoning, Eqs. (2.5) through (2.8) track the level of crudes in vessels, storage tanks and charging tanks. The main difference from Eqs. (2.1) to (2.4) is that these resources receive flows of crude oil from other resources, rather than producing it. More specifically, the levels of crude oil c are associated with the input flow (i.e., operations $v \in I_r$, $r \in \mathcal{RF} \cup \mathcal{RS} \cup \mathcal{RC}$), rather than a fixed daily production rate. The volume of crude oil c transferred by operations v in each period i is determined by $vct_{i,v,c}$ and $vt_{i,v}$.

$$lcr_{i,r,c} = CIL_{r,c} + \sum_{v \in \mathcal{I}_r} vct_{i,v,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}$$
$$r \in \mathcal{RV} \cup \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, c \in \mathcal{C}, i = 1, \tag{2.5}$$

$$lcr_{i,r,c} = lcr_{i-1,r,c} + \sum_{v \in \mathcal{I}_r} vct_{i,v,c} - \sum_{v \in \mathcal{O}_r} vct_{i,v,c}$$
$$r \in \mathcal{RV} \cup \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, c \in \mathcal{C}, i \neq 1, \tag{2.6}$$

$$lr_{i,r} = TIL_r + \sum_{v \in \mathcal{I}_r} vt_{i,v} - \sum_{v \in \mathcal{O}_r} vt_{i,v} \qquad r \in \mathcal{RV} \cup \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, i = 1 \tag{2.7}$$

$$lr_{i,r} = lr_{i-1,r} + \sum_{v \in \mathcal{I}_r} vt_{i,v} - \sum_{v \in \mathcal{O}_r} vt_{i,v} \qquad r \in \mathcal{RV} \cup \mathcal{RS} \cup \mathcal{RC}, i \in \mathcal{T}, i \neq 1 \tag{2.8}$$

The total level of crude oil in a resource r is equal to the sum of the levels of each crude oil c in the same resource, which is stated by the Ineq. (2.9). In addition, each resource is limited on its capacity by the bounds $\underline{CAP}_r$ and $\overline{CAP}_r$, imposed by Eq. (2.10).

$$lr_{i,r} = \sum_{c \in \mathcal{C}} lcr_{i,r,c} \qquad i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD} \tag{2.9}$$

$$\underline{CAP}_r \leq lr_{i,r} \leq \overline{CAP}_r \qquad i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD} \tag{2.10}$$

2.3.2.2 Vessel Operations Scheduling

All operations a vessel can perform is determined by the set of nodes $\mathcal{N}_r = \mathcal{I}_r \cup \mathcal{O}_r \cup \mathcal{D}_r \cup \mathcal{TR}_r$. Thus, a graph $\mathcal{G}_r = (\mathcal{N}_r, \mathcal{E}_r)$ can determine the flow between operations associated to vessel $r \in \mathcal{RV}$. In this case, $(v,u) \in \mathcal{VD}_r \subset \mathcal{E}_r = \mathcal{N}_r \times \mathcal{N}_r$ would be the edges of the graph denoting the allowed flow between operations.

The initial conditions for the flow of operations is defined in Eq. (2.11), being v the initial operation performed by vessel r at time $i = 1$, which is defined in the set $\mathcal{IOP}_r$. Intuitively, allowed operations for the period $i = 1$ are set to zero in Eq.(2.12)

$$\sum_{(v,u) \in \mathcal{VD}_r} s_{1,r,v,u} = 1, \qquad r \in \mathcal{RV}, v \in \mathcal{IOP}_r, \tag{2.11}$$

$$s_{1,r,v,u} = 0, \qquad r \in \mathcal{RV}, v \in \mathcal{N}_r \setminus \mathcal{IOP}_r, (v,u) \in \mathcal{VD}_r. \tag{2.12}$$

The Eq. (2.13) simply state the conservation of flow with respect to the graph, while Eq. (2.14) enforces the flow from operation v in period i to operation u in $i+1$ if $s_{i,r,v,u} = 1$.

$$\sum_{(v,u) \in \mathcal{VD}_r} s_{i,r,v,u} = \sum_{(u,v) \in \mathcal{VD}_r} s_{i+1,r,u,v}, \qquad i \in \mathcal{T}, r \in \mathcal{RV}, u \in \mathcal{N}_r, i \neq PH. \tag{2.13}$$

$$\sum_{(v,u) \in \mathcal{VD}_r} s_{i,r,v,u} = z_{i,v}, \qquad i \in \mathcal{T},\ r \in \mathcal{RV}, v \in \mathcal{N}_r. \tag{2.14}$$

2.3.2.3 Vessel Travel Times

Vessels are required to travel between crude oil terminals and FPSOs to transfer the oil produced offshore. Before any traveling to FPSOs, operational rules determine that vessels must always be empty. The travel operations are represented by $u \in \mathcal{TR}_r$ for each vessel r. For each travel operation u, there is an associated parameter $VTT_{r,u}$ defining the duration required for the travel operation to be performed by vessel r. The travel times are expressed in terms of discrete periods. The dynamics of offshore trips of a vessel r (i.e., from the terminal to the FPSOs) is given by Ineq. (2.15).

To illustrate, consider a certain vessel r performing an unloading or a waiting operation ($v \in \mathcal{O}_r \cup \mathcal{D}_r$) at the terminal at the period i. If the vessel r will start a new travel operation

$u \in \mathcal{TR}_r$ in the next period of time $(i+1)$, the vessel must arrive $VTT_{r,u}$ periods later at the FPSO associated with operation u, in order to start the offloading of crude oil.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r:\\ z\in\mathcal{I}_r}} s_{i+VTT_{r,u},r,u,z}$$

$$i \in \mathcal{T}, r \in \mathcal{RV}, u \in \mathcal{TR}_r, i \leq PH - VTT_{r,u}. \quad (2.15)$$

Conversely, once the offloading operation $v \in \mathcal{I}_r$ was performed in an FPSO at period i, suppose that vessel r will start a new travel operation $u \in \mathcal{TR}_r$ at the next period of time $(i+1)$, as indicated by $s_{i,r,v,u} = 1$. In this case, the vessel is expected to complete the trip, arriving at the terminal $VTT_{r,u}$ periods later in order to able to unload or wait (operation $z \in \mathcal{O}_r \cup \mathcal{D}_r$) at period $(VTT_{r,u} + 1)$, which is defined by Ineq. (2.16).

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r:\\ v\in\mathcal{I}_r}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r:\\ z\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i+VTT_{r,u},r,u,z}$$

$$i \in \mathcal{T}, r \in \mathcal{RV}, u \in \mathcal{TR}_r, i \leq PH - VTT_{r,u}. \quad (2.16)$$

2.3.2.4 Vessel Loading and Unloading Rules

A vessel must offload an FPSO to fill its storage tanks until no residual capacity is left. Effectively, once a vessel begins to offload crude oil from an FPSO, the vessel loading operation must continue until its full storage capacity is reached. The maximum flow rate must also be enforced as much as possible, determined by the flow rate upper bound. This rule is stated by Ineq. (2.17).

For instance, suppose vessel r is executing a travel operation $v \in \mathcal{TR}_r$ at time i, and starts the loading operation $u \in \mathcal{I}_r$ from an FPSO at time $(i+1)$, indicated by $s_{i,r,v,u} = 1$. When modeling this dynamic behavior to Ineq. (2.17), the right-hand side of the constraint will assume the value $\overline{CAP}_r$. This condition will force 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 operations (v,u) then Ineq. (2.17) becomes innocuous.

$$\sum_{c\in\mathcal{C}} \sum_{t=(i+1)}^{i+\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \rceil} vct_{t,u,c} \geq \overline{CAP}_r - \overline{CAP}_r(1 - \sum_{\substack{(v,u)\in\mathcal{VD}_r\\ v\in\mathcal{TR}_r}} s_{i,r,v,u})$$

$$i \in \mathcal{T},\ r \in \mathcal{RV},\ \ u \in \mathcal{I}_r, i \leq PH - \left\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \right\rceil \quad (2.17)$$

Moreover, after $\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \rceil$ periods of vessel r loading, its maximum storage capacity will be reached, forcing vessel r to start a travel operation to the terminal. This event is determined by Ineq. (2.18). Supposing vessel r arrived at an FPSO at time i and started loading at time

$(i+1)$, which is indicated by $s_{i,r,v,u} = 1$, then the loading operation must be finished at time $(i + \lceil \overline{CAP}_r / \overline{FR}_u \rceil)$, and the vessel must start a travel operation z at the next period of time.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r: \\ v\in\mathcal{TR}_r}} s_{i,r,v,u} \leq \sum_{\substack{(u,z)\in\mathcal{VD}_r: \\ z\in\mathcal{TR}_r}} s_{i+\lceil\frac{\overline{CAP}_r}{\overline{FR}_u}\rceil,r,u,z}$$

$$i \in \mathcal{T},\ r \in \mathcal{RV}, u \in \mathcal{I}_r, i \leq PH - \left\lceil \frac{\overline{CAP}_r}{\overline{FR}_u} \right\rceil \quad (2.18)$$

If a vessel r starts to unload crude oil in the terminal, it must keep unloading until it becomes empty. Only after the full unloading, a waiting operation $u \in \mathcal{D}_r$ or a travel operation $u \in \mathcal{TR}_r$ from the terminal are allowed, as imposed by constraint (2.19). Therefore, the total level of crude oil in the vessel is $lr_{i,r} = 0$ at time i, so the right-hand side of Ineq. (2.19) becomes $\overline{CAP}_r / \overline{CAP}_r = 1$, which allows $s_{i,r,v,u}$ to assume value 1 for a travel or waiting operation u at time $(i+1)$. It can also be inferred that if there is crude oil in the vessel (i.e., $lr_{i,r} > 0$), then the right-hand side will be less than 1, forcing $s_{i,r,v,u} = 0$ for all variables on the left-hand side.

$$\sum_{\substack{(v,u)\in\mathcal{VD}_r: \\ v\in(\mathcal{O}_r\cup\mathcal{D}_r)}} s_{i,r,v,u} \leq \frac{\overline{CAP}_r - lr_{i,r}}{\overline{CAP}_r} \qquad i \in \mathcal{T},\ r \in \mathcal{RV}, u \in \mathcal{TR}_r \cup \mathcal{D}_r \quad (2.19)$$

2.3.2.5 Transfer Constraints

Transfer operations are defined as operations where crude oil is being transferred between resources. Concretely, it involves all operations except waiting and travel operations $v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT})$. Ineq. (2.20) determines that the bounds on the flowrate of crude oil must be respected every period of time i when a transfer operation $v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT})$ is being executed, which is indicated by $z_{i,v} = 1$. The flow of crude oil between resources is bounded by the lower ($\underline{FR}_v$) and upper ($\overline{FR}_v$) bounds of each transfer operation v.

$$z_{i,v}\underline{FR}_v \leq vt_{i,v} \leq \overline{FR}_v z_{i,v} \qquad i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}) \quad (2.20)$$

As discussed in Section 2.2, during any transfer operation v executing in a period of time i, a blending of different types of crudes c can occur. For this reason, Eq. (2.21) assures that the total volume of crude oil $vt_{i,v}$ matches the sum of the volumes $vct_{i,v,c}$ of all crudes c transferred in the same operation.

$$vt_{i,v} = \sum_{c\in\mathcal{C}} vct_{i,v,c} \qquad i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}) \quad (2.21)$$

Additionally, storage and charging tanks ($r \in [\mathcal{RS} \cup \mathcal{RC}]$) have a total level $lr_{i,r}$ of crude oil composed by specific levels of $lcr_{i,r,c}$ for each crude type c. Effectively, each blending proportion of crude type c inside tanks must hold when crude oil is outlet from these tanks. More specifically, 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, as imposed by Eq. (2.22). This requirement enforces composition consistency since crude oil compositions inside tanks remain the same when crude oil batches are transferred between them. For this work, it is essential to emphasize that the blending condition in Eq. (2.22) is composed by bilinear terms and will be further explored in details for the application of the proposed linear relaxation methods.

$$\frac{vct_{i,v,c}}{vt_{i,v}} = \frac{lcr_{i,r,c}}{lr_{i,r}} \Rightarrow vct_{i,v,c} \cdot lr_{i,r} = vt_{i,v} \cdot lcr_{i,r,c} \qquad i \in \mathcal{T}, r \in \mathcal{RS} \cup \mathcal{RC}, v \in \mathcal{O}_r, c \in \mathcal{C} \quad (2.22)$$

2.3.2.6 CDUs

Distillation is constrained by operating ranges of crude oil composition, which corresponds concretely to a feasible range for each property k of the crude oil transferred to the CDU. In other terms, when a charging tank r outlet operation $v \in \mathcal{WD}$ is ongoing, the flow that reaches the target CDU must have its property k within the bounds $[\underline{DEMC}_{v,k}, \overline{DEMC}_{v,k}]$. This requirement is imposed by Ineq. (2.23). Parameter $PR_{k,c}$ defines the weight fraction of property k associated to crude c and together with bounds $\underline{DEMC}_{v,k}$ and $\overline{DEMC}_{v,k}$ are specified for each CDU according to the distillation demand by the user.

$$\underline{DEMC}_{v,k} vt_{i,v} \leq \sum_{c \in \mathcal{C}} vct_{i,v,c} PR_{k,c} \leq \overline{DEMC}_{v,k} vt_{i,v} \qquad i \in \mathcal{T}, v \in \mathcal{WD}, k \in \mathcal{K} \quad (2.23)$$

Additionally, over the planning horizon stipulated, the total volume of crude oil delivered to be distilled in each CDU r is bounded by $[\underline{DEM}_r, \overline{DEM}_r]$ as states Ineq. (2.24).

$$\underline{DEM}_r \leq \sum_{i \in \mathcal{T}} \sum_{v \in \mathcal{O}_r} vt_{i,v} \leq \overline{DEM}_r \qquad r \in \mathcal{RC} \quad (2.24)$$

Finally, operational rules for CDUs determine that one and only one distillation operation must occur in a period.

$$\sum_{v \in \mathcal{I}_r} z_{i,v} = 1, \qquad i \in \mathcal{T}, r \in \mathcal{RD} \quad (2.25)$$

2.3.2.7 Conflicting Operations

Due to logistic rules inherent to the problem, some operations cannot be performed simultaneously, denominated as conflicting operations. Such operations are aggregated in sets, as follows.

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

The operations within each set cannot be performed in the same period. In other words, *FPSO1* has only one output to transfer crude oil to a single vessel, and *CDU1* can only receive streams of oil from one charging tank at a time. In such case, let the set of cliques be $\mathcal{CL}$ = {FPSO-OFFLOAD, CDU1-INPUT} and the set $\mathcal{WCL}_{cl}$ contain the operations of each clique $cl \in \mathcal{CL}$. To prevent conflicting operations from occurring in a time period, Ineq. (2.26) ensures that at most one operation $v \in \mathcal{WCL}_{cl}$ will be performed:

$$\sum_{v \in \mathcal{WCL}_{cl}} z_{i,v} \leq 1, \qquad i \in \mathcal{T},\ cl \in \mathcal{CL} \tag{2.26}$$

2.3.3 Objective Function

The optimization objective consists of maximizing the sum of gross margin (G_c) over the total volume. This goal is achieved by setting the optimal schedule for all operations in a given planning horizon. These operations must satisfy the demands of CDUs both in terms of quality and quantity. To this end, we propose a discrete-time MINLP model, whose major 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.

2.3.4 Nonconvex Discrete Time MINLP Formulation

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

$$P:\ \max\ f = \sum_{i \in \mathcal{T}} \sum_{r \in \mathcal{RD}} \sum_{v \in \mathcal{I}_r} \sum_{c \in \mathcal{C}} G_c vct_{i,v,c} \tag{2.27a}$$

$$\text{s.t.}: \text{Eqs. (2.1)-(2.26)}, \tag{2.27b}$$

$$z_{i,v} \in \{0,1\},\ i \in \mathcal{T}, v \in \mathcal{W}, \tag{2.27c}$$

$$s_{i,r,v,u} \in \{0,1\},\ i \in (\mathcal{T} \setminus \{PH\}), r \in \mathcal{RV}, (v,u) \in \mathcal{VD}_r, \tag{2.27d}$$

$$vt_{i,v} \geq 0,\ i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}) \tag{2.27e}$$

$$vct_{i,v,c} \geq 0,\ i \in \mathcal{T}, v \in \mathcal{W} \setminus (\mathcal{WW} \cup \mathcal{WT}), c \in \mathcal{C}, \tag{2.27f}$$

$$lr_{i,r} \geq 0,\ i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}, \tag{2.27g}$$

$$lcr_{i,r,c} \geq 0, i \in \mathcal{T}, r \in \mathcal{R} \setminus \mathcal{RD}, c \in \mathcal{C}. \tag{2.27h}$$

2.4 ILLUSTRATION OF A SOLUTION

In this section, a solution for the MINLP problem is presented to illustrate the schedule of operations, and the level of the resources, along the planning horizon. Here it was considered a problem containing 2 FPSOs, 2 vessels, 2 storage tanks, 2 charging tanks, 1 CDU, 2 types of crude oil, and a planning horizon of 15 days. A more detailed discussion on designing instances for the OMCOS will be addressed in chapter 5.

For each resource in the supply chain, Fig. 4 displays from top to bottom: the total level of crude oil in the resource ($lr_{i,r}$) in a gray bar; the total volume of crude oil transferred if some operation is being performed ($vt_{i,v}$); a chart in blue showing the possible operations and when they are being executed. In order to simplify the illustration, the different crude types mixed in each operation are neglected.

In summary, FPSOs continuously produce crude oil from extraction in a rate of 130 $10^3 bbl$ during each time interval. For that reason, an ongoing raise in the crude oil level is sustained through the planning horizon. When reaching the capacity limit, the offloading of crude oil to vessels starts until filling up the vessel's reservoir completely, as shown during periods 7-8 for *FPSO1* and 11-12 for *FPSO2*.

Vessels initial operations are set before hand: *Vessel1* begins unloading to *ST1* while *Vessel2* waits at the terminal. *Vessel1* unloads until emptying its reservoir at period 3, and then awaits at the terminal until period 6 when it starts a travelling operation back to *FPSO1*. At period 9, upon completing offloading, *Vessel1* travels to the terminal and alternates unloading crude into *ST1* and *ST2*. Similar dynamic unfolds over the horizon with *Vessel2*. The diagram shows that the proposed model yields a feasible schedule that effectively integrates the operations related to maritime inventory routing and crude oil scheduling. Moreover, vessels always travel to FPSOs with empty tanks and return to the terminal loaded to its full capacity, as required by the problem statement.

Storage tanks transfer oil to charging tanks following the rules for the bounds on flow of crude, capacity, demand and crude oil composition. Charging tanks follow the same rules, plus they must feed continuously crude oil to *CDU1*, but only one at a time. Additionally, storage tanks and charging tanks cannot perform inlet and outlet operations occurring at the same period. Regarding the *CDU1*, crude oil is constantly being transferred from both charging tanks (but never overlapping in time). It can be noticed that the operational rules for the problem are being met altogether.

2.5 SUMMARY

The operational management of crude oil supply problem is a discrete-time mixed-integer non-linear program of considerable complexity. The model incorporates the schedule of all resource operations from the production in FPSOs to distillation in CDUs. In a given planning horizon, the solver must select which operations take place at each time, the level of crudes in each resource, and the volume of crude oil transferred between resources. The objective is to satisfying the demands of CDUs both in terms of quality and quantity, while maximizing the gross margin over the total volume distilled. It is essential to notice that all constraints in the model are linear, except for the nonlinear nonconvex blending of crudes (2.22).

In Assis et al. (2019), the authors proposed to solve the MINLP using an iterative

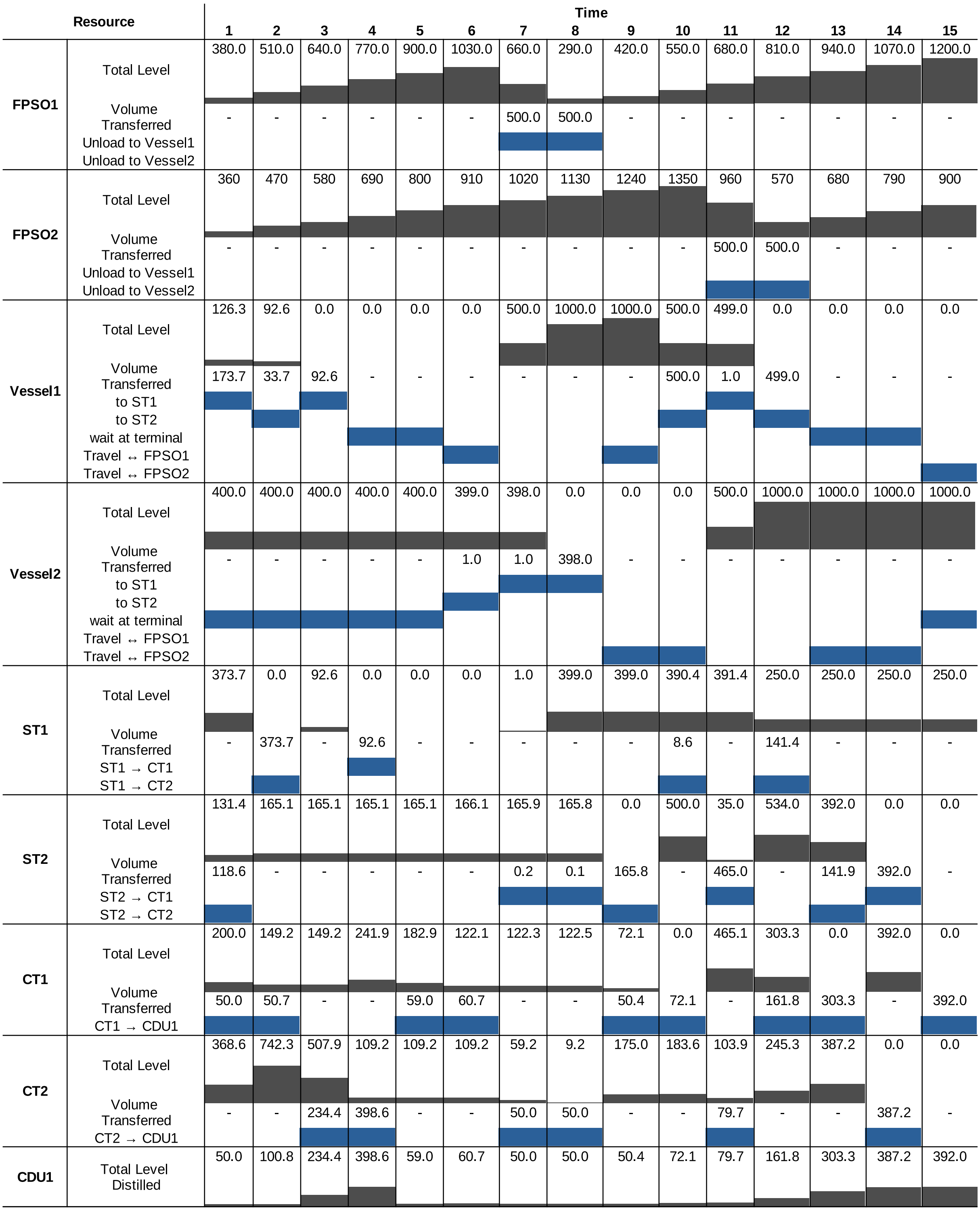

Figure 4 – Illustration of the solution computed for INS1.

MILP-NLP decomposition scheme with domain reduction. The approach obtained solutions in a reasonable CPU time with reduced gap on small and medium size instances and found good enough solutions for the larger instance. Despite that, it was clear the difficulty of solving the non-convex MINLP scheduling problem specially because of the blending equation involving bilinear terms, which renders the problem computationally hard. For this reason, relaxing these bilinear terms has a potential to reduce significantly the effort to solve this problem.

3 RELAXATIONS FOR BILINEAR PROGRAMS

From the previous chapter, it could be noticed that the OMCOS contains a blending constraint involving bilinear terms. The current work seeks to solve the problem more efficiently by relying on relaxation methods for bilinear terms. For this reason, in this chapter the necessary theoretical background on relaxations for bilinear programs is provided.

3.1 OVERVIEW

The management of crude oil supply problem at the operational level is a nonconvex nonlinear problem where nonconvexities arise due to bilinear terms. The work of Assis et al. (2019) proposed to handle the nonconvex bilinear terms associated with the blending of crudes. Blending constraints arise in crude oil operations in refineries (LEE et al., 1996; JIA; IERAPETRITOU; KELLY, 2003; YADAV; SHAIK, 2012) and are required to model the mixing of various streams, being the most common type of constraint in Chemical Engineering systems. They are known for creating bilinear terms that are nonconvex and which give rise to multiple local solutions.

For the purpose of finding rigorous global optimal solutions to bilinear problems, which can be of the nonlinear or mixed-integer nonlinear type, alternative algorithms have been proposed to generate linear or mixed-integer linear relaxations of the original problem. Therefore, it is critical to achieve tight relaxations in order to obtain better solutions. Such bilinear terms appear in distillation of diesel and gasoline when addressing the blending of different fractions (MORO; ZANIN; PINTO, 1998; JIA; IERAPETRITOU, 2003; KOLODZIEJ et al., 2013), in mass and property integration networks (NÁPOLES-RIVERA et al., 2010), in problems associated with operations of hydroelectric power systems (CATALÃO; POUSINHO; MENDES, 2011; CASTRO; GROSSMANN, 2014), and in the design of electric power converters (CAMPONOGARA; SEMAN; GILI, 2019), among others.

The most simple strategy corresponds to applying the standard McCormick relaxation (MCCORMICK, 1976), where a new variable substitutes the bilinear term along with four sets of linear constraints for the lower and upper bounds. The convex McCormick envelopes coupled with spatial branch-and-bound search frameworks have been the basis for many global optimization techniques.

From the standard McCormick relaxation, further reduction in the feasible space of the relaxed problem is possible, by partitioning the domain of one of the variables present in the bilinear term. This approach known as univariate piecewise McCormick was proposed by Bergamini, Aguirre and Grossmann (2005), Karuppiah and Grossmann (2006), being widely studied in the work of Misener, Thompson and Floudas (2011), Hasan and Karimi (2010), and Faria and Bagajewicz (2012). Intending to improve partitioning, Wicaksono and Karimi (2008) were the first to propose the domain partitioning of both variables, a strategy called bivariate piecewise McCormick. Bivariate partitioning yielded a stronger relaxation comparably to univariate partitioning in moderate-size problems in Wicaksono and Karimi (2008) and later in Hasan and Karimi (2010). A more in-depth study on piecewise McCormick relaxation for bilinear terms can be found in (GOUNARIS; MISENER; FLOUDAS, 2009).

Further, Teles, Castro and Matos (2012) have introduced a technique to approximate polynomial constraints through discretization of a subset of variables. Applying this technique to bilinear terms, Kolodziej, Castro and Grossmann (2013) have shown that the mixed-integer constraints of the multiparametric disaggregation technique of Teles, Castro and Matos (2012) can be derived from disjunctive programming and convex hull reformulation. This approximation technique proved to drive the upper and lower bounding formulations to convergence as the original nonlinear formulation does, considering an infinite number of discretization intervals. In short, they can replace the standard McCormick relaxation to provide stronger bounds, at the cost of higher computational effort from the solution of multiple MILPs instead of LPs (CASTRO; GROSSMANN, 2014).

The comparison analysis between piecewise McCormick and multiparametric disaggregation depends highly on each application and its characteristics. Some notable comparative studies have shown that the former is tighter when having quadratic terms, but produces significantly larger MILPs for the same number of partitions (KOLODZIEJ; CASTRO; GROSSMANN, 2013). In other words, the number of binary variables grows linearly in piecewise McCormick, whereas for multiparametric disaggregation the growth is only logarithmic. This behavior allows the solver to reach lower optimality gaps with multiparametric disaggregation due to a faster computational performance as noticed by Castro and Teles (2013). In contrast, piecewise McCormick notably allows for more freedom to choose the number of partitions, since accuracy in multiparametric disaggregation can only change by one order of magnitude. Contributions on bringing multiparametric disaggregation closer to piecewise McCormick, regarding how the number of partitions can be handled, were made by Castro (2016). He proposed the introduction of a normalized parameter in $[0, 1]$ to be discretized in the range between the lower and upper bound, instead of all possible values that a variable can assume. As a result, normalized multiparametric disaggregation became directly related to the number of partitions in piecewise McCormick, allowing for a better comparison between the relaxation methods. Global optimization algorithms for mixed-integer nonlinear programming (MINLP) in oil refinery planning are making use of piecewise McCormick and Normalized Multiparametric Disaggregation as in (CASTILLO; CASTRO; MAHALEC, 2017). This work showed that bound tightening using such methods is essential for large-scale problems, even though it is computationally expensive.

3.2 BILINEAR PROGRAMS

As seen, problems with bilinear terms have been studied by many researchers because of their commercial importance and nonconvex characteristics, which makes this subject suitable for evaluating distinct relaxation methods and algorithmic approaches.

From the OMCOS formulation, one can observe that the bilinear terms appear on both sides of the equality and are of the form $x_i \cdot x_j = y_i \cdot y_j$. In this section, the properties of commonly used relaxations for such bilinear problems are reviewed.

We consider the class of nonconvex, non-linear problems where all nonlinear terms are of the bilinear type $x_i \cdot x_j$, the continuous variables are denoted by x and the binary variables by γ. lx is the length of vector x, and $l\gamma$ is the length of vector γ. In **(P)**, x is an m-dimensional vector of non-negative variables that lie between given lower x^L and upper x^U bounds. Set Q collects all functions f_q , including the objective function f_0 and all the constraints. BL is an (i, j)-index set that defines the bilinear terms $x_i \cdot x_j$ present in the problem and a_{ijq} is a scalar. Note that $i = j$ can be allowed to accommodate quadratic problems.

(**P**)

$$\max \quad f_0(x, \gamma) \tag{3.1a}$$

s.t. :

$$f_q(x, \gamma) \leq 0 \qquad \forall q \in Q \setminus \{0\} \tag{3.1b}$$

$$f_q(x, \gamma) = \sum_{(i,j)\in BL} a_{ijq} x_i x_j + B_q x + C_q \gamma + d_q \qquad \forall q \in Q \tag{3.1c}$$

$$x^L \leq x \leq x^U \tag{3.1d}$$

$$x \in \mathbb{R}^{lx}, \gamma \in \{0, 1\}^{l\gamma} \tag{3.1e}$$

A relaxation of **(P)** can be achieved by **(PR)**, where the bilinear terms $x_i \cdot x_j$ are replaced by variables w_{ij}, thus linearizing $f_q(x, \gamma)$ into $f_q^R(x, \gamma)$. A set of linear constraints is added to determine the values of the w_{ij} variables, while the feasible region of such constraints is represented by W.

(**PR**)

$$\max \quad f_0^R(x, \gamma) \tag{3.2a}$$

s.t. :

$$f_q^R(x, \gamma) \leq 0 \qquad \forall q \in Q \setminus \{0\} \tag{3.2b}$$

$$f_q^R(x, \gamma) = \sum_{(i,j)\in BL} a_{ijq} w_{ij} + B_q x + C_q \gamma + d_q \qquad \forall q \in Q \tag{3.2c}$$

$$x^L \leq x \leq x^U \tag{3.2d}$$

$$x \in \mathbb{R}^{lx}, \gamma \in \{0, 1\}^{l\gamma}, w \in W \subset \mathbb{R}^{|BL|} \tag{3.2e}$$

Because solving a non convex problem is a complicated task, relaxing the bounds as performed in (3.2), transforms the baseline problem into a convex relaxation, decreasing the computational difficulty at the cost of introducing solutions that do not correspond to the original objective function.

The feasible region W of this baseline problem will be reformulated in the next sections according to each linear relaxation method employed.

3.3 STANDARD MCCORMICK ENVELOPES

An established method to solving bilinear programs of type (P) is to apply a type of convex relaxation to bound the bilinear terms using McCormick envelopes (MCCORMICK, 1976). In the standard approach, each bilinear term $x_i \cdot x_j$ is replaced by a new variable $w_{ij} = x_i x_j$ and a set with four linear inequality constraints is added to the formulation, two representing the overestimators and two underestimators (see Fig. 5).

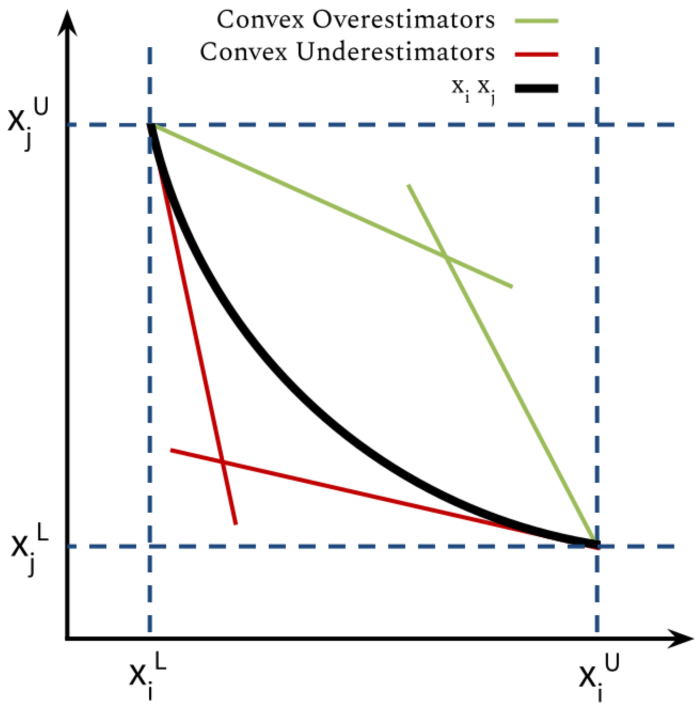

Figure 5 – Graphical interpretation of standard McCormick relaxation.

In this case, the feasible region W will be given by inequations (3.3) as follows:

$$w_{i,j} \geq x_i \cdot x_j^L + x_i^L \cdot x_j - x_i^L \cdot x_j^L \tag{3.3a}$$
$$w_{i,j} \geq x_i \cdot x_j^U + x_i^U \cdot x_j - x_i^U \cdot x_j^U \tag{3.3b}$$
$$w_{i,j} \leq x_i \cdot x_j^L + x_i^U \cdot x_j - x_i^U \cdot x_j^L \tag{3.3c}$$
$$w_{i,j} \leq x_i \cdot x_j^U + x_i^L \cdot x_j - x_i^L \cdot x_j^U \tag{3.3d}$$

Here inequation (3.3) defines the convex envelope of $w_{ij} = x_i x_j$, commonly referred to as the McCormick Envelope. One can also observe that envelope induced by inequation (3.3) can be obtained from the following four valid multiplications:

$$(x_i - x_i^L)(x_j - x_j^L) \geq 0, \qquad (x_i - x_i^U)(x_j - x_j^U) \geq 0 \tag{3.4a}$$

$$(x_i - x_i^U)(x_j - x_j^L) \leq 0, \qquad (x_i - x_i^L)(x_j - x_j^U) \leq 0 \tag{3.4b}$$

Using the McCormick envelope formulation relaxes a nonconvex problem into a convex problem, resulting in an upper bounding LP if the only nonlinearities are bilinear. By making a maximization problem convex, the maximum solution found will be a global maximum for the relaxed problem. This solution is then an upper bound solution for the original problem **(P)**. A lower bound can be obtained by solving the original non convex problem using values obtained from the relaxed problem and then checking for feasibility. McCormick Envelopes provide an envelope that retains convexity while minimizing the size of the new feasible region. This allows the lower bound solutions obtained by using these envelopes to be closer to the true solution than if other convex relaxations were used. However, this lower bound can be weak depending on the bounds on the bilinear terms.

3.4 PIECEWISE MCCORMICK ENVELOPES

The strength of the McCormick envelopes seen in Section 3.3 for a single bilinear term is strongly dependent on the variable bounds. One can notice that tighter bounds could lead to even stronger relaxations. Hence, instead of simply including equations with global bounds for the entire interval, partitioning the intervals of variables and then constructing McCormick envelopes in each interval leads to a much stronger relaxation. Considering that a bilinear term has two variables, three possible options for partitioning are evident. Two options consists in partitioning only one of the two variables, which is called uni variate partitioning. The third choice is to partition both the variables, known as bivariate partitioning (see Fig.6). The number of partitions will determine the strength of this new relaxation.

To enforce consistence of this relaxation, auxiliary binary variables must be added in order to select the appropriate partition. In other words, the new binary variables turn on/off each partition, with exactly one partition being activated. This gives rise to a MILP relaxation, referred to as the Piecewise McCormick Relaxation.

Because this method requires the introduction of binary variables, there is a trade-off between the quality of the relaxation and the computation effort required. In another words, is is known that better relaxation quality implies more partitions which leads to more binary variables. Given the maturity of MILP solvers, the current trend for solving nonconvex NLP and MINLP problems with only bilinear and quadratic terms is to employ piecewise linear relaxations.

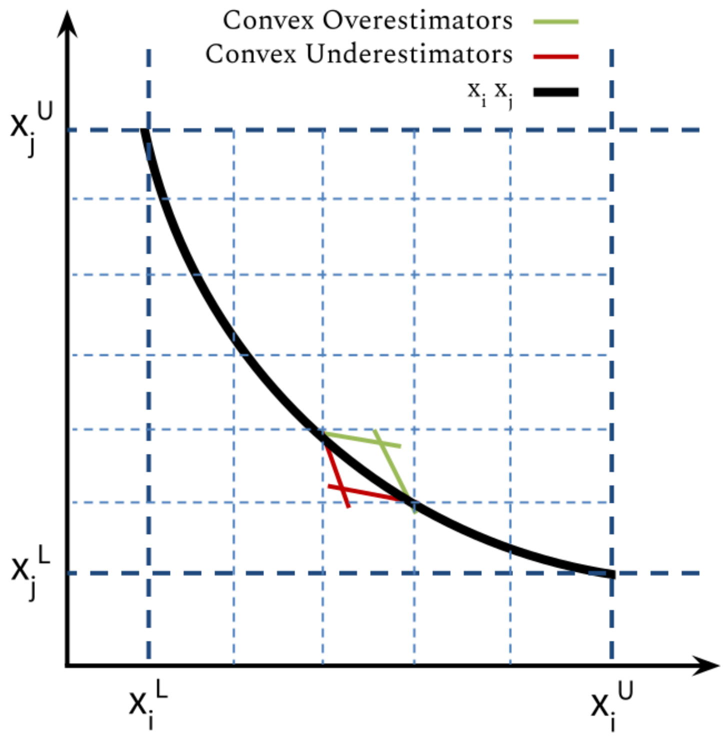

Figure 6 – Graphical interpretation of piecewise McCormick relaxation for the bivariate case.

3.4.1 Univariate Piecewise McCormick Envelopes

A tighter MILP relaxation can be constructed by partitioning the domain of one of the variables (x_j) of the bilinear term into n disjoint regions, with new binary variables being added to the formulation to select the optimal partition for x_j.

In the seminal works by Bergamini, Aguirre and Grossmann (2005), Karuppiah and Grossmann (2006), partitions are generated uniformly and there is a linear increase in problem size with the number of partitions. A different uniform partitioning scheme featuring a logarithmic growth in the number of binary variables was proposed by Misener, Thompson and Floudas (2011) but the results failed to show major benefits. Other substantial evaluations involving several piecewise under and overestimators have been performed by Wicaksono and Karimi (2008), Gounaris, Misener and Floudas (2009).

The quality of the relaxation is influenced by the choice of variables x_j selected for partitioning. In general, there may exist an optimal set of variables that would lead to the best relaxation, but such an analysis is beyond the scope of this study. In most engineering problems as in OMCOS, bilinear terms involve two different sets of variables and so there are just a couple of obvious choices. For instance, Hasan and Karimi (2010) chose to partition flow variables in problems involving column sequencing for distillation and integrated water use and treatment systems.

Let x_{jn}^L and x_{jn}^U represent respectively the lower and upper bounds of variable x_j for partition n. If the value of x_j does belong to such a partition, then binary variable $y_{jn} = 1$ and the respective McCormick envelope holds. The Piecewise McCormick Relaxation can be formulated as a Generalized Disjunctive Program (RAMAN; GROSSMANN, 1994), which is tighter due to the use of the partition-dependent parameters x_{jn}^L and x_{jn}^U in the four constraints inside the disjunction, instead of the global bounds x_j^L and x_j^U. The feasible region W will be defined by Eqs. (3.5).

(**PR-UV-GDP**)

$$\bigvee_{n=1}^{N} \left[\begin{array}{c} y_{jn} \\ \left\{ \begin{array}{l} w_{i,j} \geq x_i \cdot x_{jn}^L + x_i^L \cdot x_j - x_i^L \cdot x_{jn}^L \\ w_{i,j} \geq x_i \cdot x_{jn}^U + x_i^U \cdot x_j - x_i^U \cdot x_{jn}^U \\ w_{i,j} \leq x_i \cdot x_{jn}^L + x_i^U \cdot x_j - x_i^U \cdot x_{jn}^L \\ w_{i,j} \leq x_i \cdot x_{jn}^U + x_i^L \cdot x_j - x_i^L \cdot x_{jn}^U \end{array} \right\} \forall \{i \mid (i,j) \in BL\} \\ x_{jn}^L \leq x_j \leq x_{jn}^U \end{array} \right] \forall \{j \mid (i,j) \in BL\} \quad (3.5a)$$

$$x_i^L \leq x_i \leq x_i^U \qquad \forall \{i \mid (i,j) \in BL\} \quad (3.5b)$$

$$\left\{ \begin{array}{l} x_{jn}^L = x_j^L + \frac{(x_j^U - x_j^L)\cdot(n-1)}{N} \\ x_{jn}^U = x_j^L + \frac{(x_j^U - x_j^L)\cdot n}{N} \end{array} \right\} \qquad \forall \{j \mid (i,j) \in BL\}, n \in \{1, ..., N\} \quad (3.5c)$$

$$y_{jn} \in \{0, 1\} \qquad \forall \{j \mid (i,j) \in BL\}, n \in \{1, ..., N\} \quad (3.5d)$$

Since the linear GDP needs to be reformulated into a MILP, one alternative consists in applying the big-M technique. However, this approach was proven to yield a poor relaxation, specially when dealing with piecewise partitioning (WICAKSONO; KARIMI, 2008). For this reason, the reformulation is carried out through a convex hull relaxation based on the works of Karuppiah and Grossmann (2006), Castro and Teles (2013) as follows.

(**PR-UV-MILP**)

$$w_{i,j} \geq \sum_{n=1}^{N} \left(\widehat{x}_{ijn} x_{jn}^L + \widehat{x}_{jn} x_i^L - y_{jn} x_i^L x_{jn}^L \right) \qquad \forall (i,j) \in BL \quad (3.6a)$$

$$w_{i,j} \geq \sum_{n=1}^{N} \left(\widehat{x}_{ijn} x_{jn}^U + \widehat{x}_{jn} x_i^U - y_{jn} x_i^U x_{jn}^U \right) \qquad \forall (i,j) \in BL \quad (3.6b)$$

$$w_{i,j} \leq \sum_{n=1}^{N} \left(\widehat{x}_{ijn} x_{jn}^L + \widehat{x}_{jn} x_i^U - y_{jn} x_i^U x_{jn}^L \right) \qquad \forall (i,j) \in BL \quad (3.6c)$$

$$w_{i,j} \leq \sum_{n=1}^{N} \left(\widehat{x}_{ijn} x_{jn}^U + \widehat{x}_{jn} x_i^L - y_{jn} x_i^L x_{jn}^U \right) \qquad \forall (i,j) \in BL \quad (3.6d)$$

$$x_i = \sum_{n=1}^{N} \widehat{x}_{ijn} \qquad \forall (i,j) \in BL \quad (3.6e)$$

$$x_j = \sum_{n=1}^{N} \widehat{x}_{jn} \qquad \forall j \mid (i,j) \in BL \quad (3.6f)$$

$$\sum_{n=1}^{N} y_{jn} = 1 \qquad \forall j \mid (i,j) \in BL \tag{3.6g}$$

$$x_i^L y_{jn} \le \widehat{x}_{ijn} \le x_i^U y_{jn} \qquad \forall (i,j) \in BL, n \in \{1,...,N\} \tag{3.6h}$$

$$x_{jn}^L y_{jn} \le \widehat{x}_{jn} \le x_{jn}^U y_{jn} \qquad \forall j \mid (i,j) \in BL, n \in \{1,...,N\} \tag{3.6i}$$

$$\left\{ \begin{array}{l} x_{jn}^L = x_j^L + \frac{(x_j^U - x_j^L)(n-1)}{N} \\ x_{jn}^U = x_j^L + \frac{(x_j^U - x_j^L)n}{N} \end{array} \right\} \qquad \forall j \mid (i,j) \in BL, n \in \{1,...,N\} \tag{3.6j}$$

$$y_{jn} \in \{0,1\} \qquad \forall j \mid (i,j) \in BL, n \in \{1,...,N\} \tag{3.6k}$$

Notice that if $y_{\widehat{j}n} = 1$ for some $\widehat{j}$, then $\widehat{x}_{\widehat{j}n} \in [x_{\widehat{j}n}^L, x_{\widehat{j}n}^U]$ and $\widehat{x}_{jn} = 0$ for all $j \ne \widehat{j}$. Further, $\widehat{x}_{i\widehat{j}n} \in [x_i^L, x_i^U]$ whereas $\widehat{x}_{ijn} = 0$ for all $j \ne \widehat{j}$.

3.4.2 Bivariate Piecewise McCormick Envelopes

the domain of both variables bilinear term domain of both variables forming the bilinear term is known a priori, leading to a relaxation that is usually tighter

In bivariate partitioning, both variables are partiotinng to obtain a tighter relaxation (HASAN; KARIMI, 2010). The binary variable responsible to select the active partition will now be $y_{injn'}$. The variable x_i is bounded by x_{in}^L and x_{in}^U) and the variable x_j is bounded by $x_{jn'}^L$ and $x_{jn'}^U$. The bivariate partitioning for the piecewise McCormick relaxation is formulated as a Generalized Disjunctive Program **(PR-BV-GDP)**.

If the Bivariate Piecewise McCormick Relaxation is used, then the feasible region W will be given by Eqs. (3.7).

(**PR-BV-GDP**)

$$\bigvee_{n=1}^{N} \bigvee_{n'=1}^{N} \left[\begin{array}{c} y_{injn'} \\ w_{i,j} \ge x_i \cdot x_{jn'}^L + x_{in}^L \cdot x_j - x_{in}^L \cdot x_{jn'}^L \\ w_{i,j} \ge x_i \cdot x_{jn'}^U + x_{in}^U \cdot x_j - x_{in}^U \cdot x_{jn'}^U \\ w_{i,j} \le x_i \cdot x_{jn'}^L + x_{in}^U \cdot x_j - x_{in}^U \cdot x_{jn'}^L \\ w_{i,j} \le x_i \cdot x_{jn'}^U + x_{in}^L \cdot x_j - x_{in}^L \cdot x_{jn'}^U \\ x_{in}^L \le x_i \le x_{in}^U \\ x_{jn'}^L \le x_j \le x_{jn'}^U \end{array} \right] \qquad \forall (i,j) \in BL \tag{3.7a}$$

$$\left\{ \begin{array}{l} x_{in}^L = x_i^L + \frac{(x_i^U - x_i^L)\cdot(n-1)}{N} \\ x_{in}^U = x_i^L + \frac{(x_i^U - x_i^L)\cdot n}{N} \end{array} \right\} \qquad \forall \{i \mid (i,j) \in BL\}, n \in \{1,...,N\} \tag{3.7b}$$

$$\left\{ \begin{array}{l} x_{jn'}^L = x_j^L + \frac{(x_j^U - x_j^L)\cdot(n'-1)}{N} \\ x_{jn'}^U = x_j^L + \frac{(x_j^U - x_j^L)\cdot n'}{N} \end{array} \right\} \qquad \forall \{j \mid (i,j) \in BL\}, n' \in \{1,...,N\} \tag{3.7c}$$

$$y_{injn'} \in \{0,1\} \qquad \forall (i,j) \in BL,$$

$$n, n' \in \{1, ..., N\} \quad (3.7d)$$

Applying the reformulation of the linear GDP into a MILP using the convex hull technique leads to **(PR-BV-MILP)**. The variable $\widehat{x}_{iinjn'}$ corresponds to the value that x_i assumes considering the bilinear term $x_i \cdot x_j$, partition n of variable x_i, and partition n' of variable j. The constraints below ensure that $\widehat{x}_{iinjn'}$ will be nonzero only if the bivariate partition (n, n') is selected for the bilinear term $x_i \cdot x_j$. Similar reasoning applies for $\widehat{x}_{jinjn'}$ regarding the second variable x_j of the bilinear term. Notice that the variables x_i assume consistent values over all bilinear terms where they appear.

(**PR-BV-MILP**)

$$w_{i,j} \geq \sum_{n=1}^{N} \sum_{n'=1}^{N} \left(\widehat{x}_{iinjn'} \cdot x^L_{jn'} + \widehat{x}_{jinjn'} \cdot x^L_{in} - x^L_{in} \cdot x^L_{jn'} \cdot y_{injn'} \right) \qquad \forall (i,j) \in BL \quad (3.8a)$$

$$w_{i,j} \geq \sum_{n=1}^{N} \sum_{n'=1}^{N} \left(\widehat{x}_{iinjn'} \cdot x^U_{jn'} + \widehat{x}_{jinjn'} \cdot x^U_{in} - x^U_{in} \cdot x^U_{jn'} \cdot y_{injn'} \right) \qquad \forall (i,j) \in BL \quad (3.8b)$$

$$w_{i,j} \leq \sum_{n=1}^{N} \sum_{n'=1}^{N} \left(\widehat{x}_{iinjn'} \cdot x^L_{jn'} + \widehat{x}_{jinjn'} \cdot x^U_{in} - x^U_{in} \cdot x^L_{jn'} \cdot y_{injn'} \right) \qquad \forall (i,j) \in BL \quad (3.8c)$$

$$w_{i,j} \leq \sum_{n=1}^{N} \sum_{n'=1}^{N} \left(\widehat{x}_{iinjn'} \cdot x^U_{jn'} + \widehat{x}_{jinjn'} \cdot x^L_{in} - x^L_{in} \cdot x^U_{jn'} \cdot y_{injn'} \right) \qquad \forall (i,j) \in BL \quad (3.8d)$$

$$x_i = \sum_{n=1}^{N} \sum_{n'=1}^{N} \widehat{x}_{iinjn'} \qquad \forall (i,j) \in BL \quad (3.8e)$$

$$x_j = \sum_{n=1}^{N} \sum_{n'=1}^{N} \widehat{x}_{jinjn'} \qquad \forall (i,j) \in BL \quad (3.8f)$$

$$\sum_{n=1}^{N} \sum_{n'=1}^{N} y_{injn'} = 1 \qquad \forall (i,j) \in BL \quad (3.8g)$$

$$x^L_{in} \cdot y_{injn'} \leq \widehat{x}_{iinjn'} \leq x^U_{in} \cdot y_{injn'} \qquad \forall (i,j) \in BL, n, n' \in \{1, ..., N\} \quad (3.8h)$$

$$x^L_{jn'} \cdot y_{injn'} \leq \widehat{x}_{jinjn'} \leq x^U_{jn'} \cdot y_{injn'} \qquad \forall (i,j) \in BL, n, n' \in \{1, ..., N\} \quad (3.8i)$$

$$\left\{ \begin{array}{l} x^L_{in} = x^L_i + \frac{(x^U_i - x^L_i) \cdot (n-1)}{N} \\ x^U_{in} = x^L_i + \frac{(x^U_i - x^L_i) \cdot n}{N} \end{array} \right\} \qquad \forall \{i \mid (i,j) \in BL\}, n \in \{1, ..., N\} \quad (3.8j)$$

$$\left\{ \begin{array}{l} x^L_{jn'} = x^L_j + \frac{(x^U_j - x^L_j) \cdot (n'-1)}{N} \\ x^U_{jn'} = x^L_j + \frac{(x^U_j - x^L_j) \cdot n'}{N} \end{array} \right\} \qquad \forall \{j \mid (i,j) \in BL\}, n' \in \{1, ..., N\} \quad (3.8k)$$

$$y_{injn'} \in \{0, 1\} \qquad \forall (i,j) \in BL, n, n' \in \{1, ..., N\} \quad (3.8l)$$

Here we briefly discuss the convex-hull reformulation above of the GDP (3.7). Let us consider a bilinear term $(i, j) \in BL$. From Eq. (3.8g), precisely one partition $\widehat{n}$ is selected for variable x_i, and one partition $\widehat{n}'$ is selected for variable x_j. Then Eq. (3.8h) ensures that $\widehat{x}_{ii\widehat{n}j\widehat{n}'} \in [x^L_{i\widehat{n}}, x^U_{i\widehat{n}}]$, while the remaining variables $\widehat{x}_{iinjn'} = 0$ for all $(n, n') \neq (\widehat{n}, \widehat{n}')$. Likewise Eq. (3.8i) ensures that $\widehat{x}_{ji\widehat{n}j\widehat{n}'} \in [x^L_{j\widehat{n}'}, x^U_{j\widehat{n}'}]$, while the remaining variables $\widehat{x}_{jinjn'} = 0$ for all $(n, n') \neq (\widehat{n}, \widehat{n}')$. Now it can be seen that Eqs. (3.8e) and (3.8f) enforce $x_i = \widehat{x}_{ii\widehat{n}j\widehat{n}'}$ and $x_j = \widehat{x}_{ji\widehat{n}j\widehat{n}'}$ which establish the consistency of the values of the variables x_i and x_j, over all bilinear terms where they appear and their respective partitions.

3.5 MULTIPARAMETRIC DISAGGREGATION

Based on the general problem **(P)**, with a nonconvex bilinear term $w_{ij} = x_i \cdot x_j$, the multiparametric disaggregation described by Teles, Castro and Matos (2013) can be used to obtain an upper bound on problem **(P)**. Multiparametric disaggregation is a technique for generating a mixed-integer linear relaxation of a bilinear problem by discretizing the domain of one of the variables in the bilinear term according to a numeric representation system.

The method works by discretizing one of the bilinear terms over a set of powers $l \in \{p, ..., P\}$, where $P = \lceil \log_{10} x^U_j \rceil$ (in case the discretized variable is x_j) and p is selected by the user for accuracy purposes. The formulation is obtained by deriving first a generalized disjunctive programming (GDP) model followed by a convex hull reformulation and exact linearization. Simplifying the notation, we assume the bilinear product $w_{ij} = x_i \cdot x_j$ as a single bilinear term $w = u \cdot v$. This product can be represented exactly with the following constraints and disjunction:

$$w = u \cdot v \tag{3.9a}$$

$$v = \sum_{l \in \mathbb{Z}} \lambda_l \tag{3.9b}$$

$$\bigvee_{d=0}^{9} \left[\lambda_l = 10^l \cdot d\right] \qquad \forall l \in \mathbb{Z} \tag{3.9c}$$

The term v is discretized through the disjunction in (3.9c) that selects one digit $d \in D = \{0, 1, ..., 9\}$ for each power in $\mathbb{Z}$. Here we assume a basis of 10, although other bases can be selected (MISENER; GOUNARIS; FLOUDAS, 2010). Note that since (3.9c) is defined over the domain of all the integer numbers, this implies an infinite number of disjunctions. Furthermore, v can represent any positive real number.

First, let us consider the convex hull reformulation of the disjunction in (3.9c) after which the disaggregated variables will be introduced,

$$\lambda_l = \sum_{d=0}^{9} \widehat{\lambda}_{d,l} \qquad \forall l \in \mathbb{Z} \tag{3.10a}$$

$$\widehat{\lambda}_{d,l} = 10^l \cdot d \cdot z_{d,l} \qquad \forall l \in \mathbb{Z}, d \in D \tag{3.10b}$$

$$\sum_{d=0}^{9} z_{d,l} = 1 \qquad \forall l \in \mathbb{Z} \tag{3.10c}$$

$$z_{d,l} \in \{0,1\} \qquad \forall l \in \mathbb{Z}, d \in D \tag{3.10d}$$

Substituting (3.10b) into (3.10a) and then into (3.9b) leads to the fully discretized (but still exact) representation of v:

$$v = \sum_{l\in\mathbb{Z}} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{d,l} \tag{3.11}$$

Considering the product $w = u \cdot v$ by substituting (3.11) into (3.9a) leads to (3.12) which involves nonlinear terms $u \cdot z_{d,l}$,

$$w = u \cdot \left[\sum_{l\in\mathbb{Z}} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{d,l}\right] \tag{3.12}$$

Carrying out an exact linearization, additional continuous variables $\widehat{u}_{d,l} = u \cdot z_{d,l}$ are introduced so that:

$$w = \sum_{l\in\mathbb{Z}} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{u}_{d,l} \tag{3.13}$$

Since,

$$u \cdot z_{d,l} = \begin{cases} 0, & \text{if } z_{d,l} = 0 \\ u, & \text{if } z_{d,l} = 1 \end{cases} \tag{3.14}$$

and $\widehat{u}_{d,l}$ is non-negative, the following lower and upper bounding constraints are inserted, being u^U and u^L the non-negative upper and lower bounds on u respectively.

$$u^L \cdot z_{d,l} \le \widehat{u}_{d,l} \le u^U \cdot z_{d,l} \qquad \forall l \in \mathbb{Z}, d \in D \tag{3.15}$$

Finally, multiplying equation (3.10c) by u and replacing the bilinear terms by the recent added continuous variables, results in (3.16).

$$u = \sum_{d=0}^{9} \widehat{u}_{d,l} \qquad \forall l \in \mathbb{Z} \tag{3.16}$$

The full set of mixed integer linear constraints for the exact representation of the bilinear product $w = u \cdot v$ is thus given by Eqs. (3.10c)–(3.11) and (3.13)–(3.16). For convenience, these equations are brought together below:

$$\sum_{d=0}^{9} z_{d,l} = 1 \qquad \forall l \in \mathbb{Z} \tag{3.17a}$$

$$z_{d,l} \in \{0,1\} \qquad \forall l \in \mathbb{Z}, d \in D \tag{3.17b}$$

$$v = \sum_{l\in\mathbb{Z}} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{d,l} \tag{3.17c}$$

$$w = \sum_{l \in \mathbb{Z}} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{u}_{d,l} \tag{3.17d}$$

$$u^L \cdot z_{d,l} \leq \widehat{u}_{d,l} \leq u^U \cdot z_{d,l} \qquad \forall l \in \mathbb{Z}, d \in D \tag{3.17e}$$

$$u = \sum_{d=0}^{9} \widehat{u}_{d,l} \qquad \forall l \in \mathbb{Z} \tag{3.17f}$$

3.5.1 Lower Bounding Formulation

Since it is impractical to compute an infinite sum over all positive integers like in the previous formulation, now v is limited to a finite level of representation v', resulting into w' as an approximate continuous form of the bilinear term. This approximation is reached by setting p and P as a minimum and maximum power of 10, respectively.

Further, the constraints in (3.11) and (3.13) are adjusted in (3.18a)-(3.18b) in order to limit the maximum power of 10 (P) and the minimum power of 10 (p). For the remaining constraints (3.10c)-(3.10d) and (3.15)-(3.16), it is sufficient to replace $l \in \mathbb{Z}$ with $l \in L = \{p, p+1, ..., P\}$. This set of constraints correspond to the equations proposed by (TELES; CASTRO; MATOS, 2013).

$$v' = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{d,l} \tag{3.18a}$$

$$w' = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{u}_{d,l} \tag{3.18b}$$

Fig. 7 illustrates the lower bounding formulation using multiparametric disaggregation. The solid curve denotes the feasible region for the bilinear term $u \cdot v = 0.1$ while the dots represent the discretization considering parameters $p = P = -1$.

Now incorporating them into the problem **(P)**, being x_j the variable to be discretized and redefining $w_{ij} = x_i \cdot x_j$, the feasible region W will be given by the resulting equations **(MDT-MILP)**.

(**MDT-MILP**)

$$w_{ij} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl}, \qquad \forall (i,j) \in BL \tag{3.19a}$$

$$x_j = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} \qquad \forall j \in \{j|(i,j) \in BL\} \tag{3.19b}$$

$$x_i = \sum_{d=0}^{9} \widehat{x}_{ijdl}, \qquad \forall (i,j) \in BL, l \in \{p, p+1, ..., P\} \tag{3.19c}$$

$$x_i^L \cdot z_{jdl} \leq \widehat{x}_{ijdl} \leq x_i^U \cdot z_{jdl}, \qquad \forall (i,j) \in BL,\, d \in \{0, ..., 9\},\; l \in \{p, p+1, ..., P\} \tag{3.19d}$$

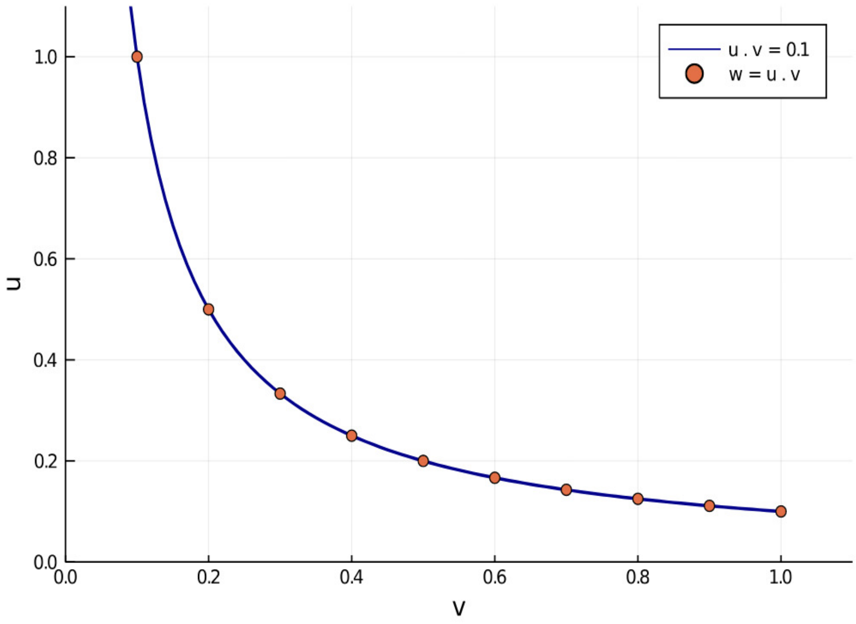

Figure 7 – Graphical interpretation of the multiparametric disaggregation relaxation with parameters $p = P = -1$.

$$\sum_{d=0}^{9} z_{jdl} = 1, \qquad \forall j \in \{j|(i,j) \in BL\},\ l \in \{p, p+1, ..., P\} \tag{3.19e}$$

$$z_{jdl} \in \{0,1\}, \qquad \forall j \in \{j|(i,j) \in BL\},\ d \in \{0,...,9\}, \quad l \in \{p, p+1, ..., P\} \tag{3.19f}$$

When we incorporate the MDT modeling (3.19) into problem **(P)**, by redefining $w_{ij} = x_i \cdot x_j$, and selecting x_j as the variable on which discretization is performed, the resulting problem **(MDT-P)** represents a mixed-integer approximation to the original problem. Further, note that problem **(MDT-P)** is a restricted version of problem **(P)**, or equivalently problem **(P)** is a relaxation of problem **(MDT-P)**. Notice that if **(MDT-P)** is a feasible problem, then the resulting solution is a lower bound.

Although the user can freely set the parameters p and P, some common instructions must be followed for ensuring feasibility of **(MDT-P)**. The largest power of 10 (P) must be sufficient large to allow 10^P to represent the upper bound on x_j, precisely $P = \lceil \log_{10} x_j^U \rceil$. In regards to p, the guideline is to let it be small enough so that at least one discretization point lies between the lower and upper bounds for x_j. Thus, $p \leq P$ is the absolute minimum requirement, but better results and feasibility are more likely obtained decreasing p. However, even respecting these rules, one cannot guarantee feasibility of **(MDT-P)** in all cases.

3.5.2 Upper Bounding Formulation

In the discretized approximation problem **(MDT-P)** illustrated by Fig. 7, one can notice the existence of a gap between discretization points. This will always exists for a finite p. Thus, in order to obtain an upper bounding problem using multiparametric disaggregation, a slack variable Δx_j can be introduced such that $x_j^R = x_j{}' + \Delta x_j$, where $x_j{}'$ is the discretized representation of x_j, x_j^R is the continuous representation of x_j, and the slack variable Δx_j is bounded between 0 and 10^p. Again switching to the notation $w = u \cdot v$ for the bilinear term, we have for the continuous representation of v, denoted as v^R:

$$v^R = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{d,l} + \Delta v \tag{3.20a}$$

$$0 \leq \Delta v \leq 10^p \tag{3.20b}$$

For the continuous representation of the bilinear term, w^R , note that:

$$w^R = u \cdot v^R = u \cdot (v' + \Delta v) = w' + u \cdot \Delta v = w' + \Delta w \tag{3.21}$$

where v' and w' are given by (3.18a)-(3.18b) respectively. The slack variable Δw replaces the bilinear term $u \cdot \Delta v$ that can be relaxed using the McCormick envelope (3.22a)-(3.22b).

$$u^L \cdot \Delta v \leq \Delta w \ \leq u^U \cdot \Delta v \tag{3.22a}$$

$$(u - u^U) \cdot 10^p + u^U \cdot \Delta v \leq \Delta w \ \leq (u - u^L) \cdot 10^p + u^L \cdot \Delta v \tag{3.22b}$$

Introducing these constraints into Problem **(P)**, and expressing the variables in terms of the original variables $w_{ij} = x_i \cdot x_j$, the feasible region W will be determined by **(MDT-MILP-UB)** for all $d \in D, l \in L$.

(MDT-MILP-UB)

$$w_{ij} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} + \Delta w_{ij} \qquad \forall (i,j) \in BL \tag{3.23a}$$

$$x_j = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} + \Delta x_j \qquad \forall \{j|(i,j) \in BL\} \tag{3.23b}$$

$$x_i = \sum_{d=0}^{9} \widehat{x}_{ijdl} \qquad \forall (i,j) \in BL, l \in \{p, p+1, ..., P\} \tag{3.23c}$$

$$x_i^L \cdot z_{jdl} \leq \widehat{x}_{ijdl} \leq x_i^U \cdot z_{jdl} \qquad \forall (i,j) \in BL, d \in \{0, ..., 9\} \ \{l \in \{p, p+1, ..., P\} \tag{3.23d}$$

$$\sum_{d=0}^{9} z_{jdl} = 1 \qquad \forall \{j|(i,j) \in BL\}, l \in \{p, p+1, ..., P\} \tag{3.23e}$$

$$\left\{ \begin{array}{l} x_i^L \cdot \Delta x_j \leq \Delta w_{ij} \leq x_i^U \cdot \Delta x_j \\ \Delta w_{ij} \leq (x_i - x_i^L) \cdot 10^p + x_i^L \cdot \Delta x_j \\ \Delta w_{ij} \geq (x_i - x_i^U) \cdot 10^p + x_i^U \cdot \Delta x_j \end{array} \right\} \qquad \forall (i,j) \in BL \tag{3.23f}$$

$$0 \leq \Delta x_j \leq 10^p \qquad \forall \{j | (i,j) \in BL\} \tag{3.23g}$$

$$z_{jdl} \in \{0,1\} \qquad \forall \{j | (i,j) \in BL\}, d \in \{0,...,9\}$$
$$l \in \{p, p+1, ..., P\} \tag{3.23h}$$

3.6 NORMALIZED MULTIPARAMETRIC DISAGGREGATION

Given the Multiparametric Disaggregation presented before, a normalized version of it can be derived. This is obtained by discretizing $\lambda_j \in [0,1]$, an auxiliary variable that serves to compute x_j as a linear combination of its lower x_j^L and upper x_j^U bounds:

$$x_j = x_j^L + \lambda_j (x_j^U - x_j^L), \qquad \forall j \tag{3.24}$$

The exact representation of λ_j can be achieved by considering an infinite number of positions $l \in \mathbb{Z}^-$ in the decimal system,

$$\lambda_j = \sum_{l \in \mathbb{Z}^-} \lambda_{jl} \tag{3.25}$$

and by picking the appropriate digit $d \in \{0, 1, ..., 9\}$ for each power l. This can be developed as a disjunction, using binary variables z_{jdl} to take the value of one if digit d is selected for position l for discretized λ_j:

$$\bigvee_{d=0}^{9} \begin{bmatrix} z_{jdl} \\ \lambda_{jl} = 10^l \cdot d \end{bmatrix} \qquad \forall j, l \in \mathbb{Z}^- \tag{3.26}$$

3.6.1 Lower Bounding Formulation

However, because it is impracticable to compute the infinite sums over all negative integers l, we define a finite precision level by substituting $l \in \mathbb{Z}^-$ with $l \in \{p, p+1, ..., -1\}$, where p is a negative integer chosen by the user.

$$\lambda_j = \sum_{l=p}^{-1} \lambda_{jl} \qquad \forall j \tag{3.27}$$

For choosing the appropriate digit $d \in \{0, 1, ..., 9\}$ for each power l, a disjunction is stated as before, where binary variables z_{jdl} decide whether or not digit d is selected for position l of the discretized variable λ_j:

$$\bigvee_{d=0}^{9} \begin{bmatrix} z_{jdl} \\ \lambda_{jl} = 10^l \cdot d \end{bmatrix} \qquad \forall j \tag{3.28}$$

The convex hull reformulation of the disjunction in (3.28) can be simplified in order to generate a sharp formulation without disaggregated variables.

$$\lambda_j = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} \qquad \forall j \qquad (3.29a)$$

$$\sum_{d=0}^{9} z_{jdl} = 1 \qquad \forall j, l \in \mathbb{Z}^- \qquad (3.29b)$$

Multiplying variable x_i by (3.24) and substituting $x_i \cdot x_j$ and $x_i \lambda_j$ with bilinear variables w_{ij} and v_{ij} leads to,

$$w_{ij} = x_i x_j^L + v_{ij}(x_j^U - x_j^L) \qquad \forall (i,j) \qquad (3.30)$$

Substituting (3.29a) into the definition of v_{ij} leads to the appearance of bilinear terms involving the product of a continuous and a binary variable.

$$v_{ij} = x_i \cdot \lambda_j$$
$$v_{ij} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot x_i \cdot z_{jdl} \qquad \forall (i,j) \qquad (3.31)$$

An exact linearization can be performed by introducing additional continuous variables $\widehat{x}_{ijdl} = x_i \cdot z_{jdl}$, resulting in:

$$v_{ij} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} \qquad \forall (i,j) \qquad (3.32a)$$

$$z_{jdl} \cdot x_i^L \leq \widehat{x}_{ijdl} \leq z_{jdl} \cdot x_i^U \qquad \forall (i,j), d \in \{0,...,9\}, l \in \{p,...,-1\} \qquad (3.32b)$$

Finally, multiplying (3.29b) by x_i and replacing the bilinear terms by the recent added continuous variables leads to,

$$x_i = \sum_{d=0}^{9} \widehat{x}_{ijdl} \qquad \forall (i,j), l \in \{p,...,-1\} \qquad (3.33)$$

The full set of mixed integer linear constraints for the exact representation of bilinear terms $w_{ij} = x_i x_j$ is thus given by Eqs. (3.24), (3.29b)-(3.30) and (3.32)-(3.33), leading to the following optimization problem **(NMDT-MILP-LB)**.

(NMDT-MILP-LB)

$$\left.\begin{aligned} & x_j = x_j^L + \lambda_j (x_j^U - x_j^L) \\ & \lambda_j = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} \end{aligned}\right\} \qquad \forall j \in \{j | (i,j) \in BL\} \qquad (3.34a)$$

$$\left.\begin{aligned} w_{ij} &= x_i x_j^L + v_{ij}(x_j^U - x_j^L) \\ v_{ij} &= \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} \end{aligned}\right\} \qquad \forall (i,j) \in BL \tag{3.34b}$$

$$x_i = \sum_{d=0}^{9} \widehat{x}_{ijdl} \qquad \forall (i,j) \in BL, l \in \{p, ..., -1\} \tag{3.34c}$$

$$\sum_{d=0}^{9} z_{jdl} = 1 \qquad \forall j \in \{j|(i,j) \in BL\}, l \in \{p, ..., -1\} \tag{3.34d}$$

$$z_{jdl} x_i^L \leq \widehat{x}_{ijdl} \leq z_{jdl} x_i^U \qquad \forall (i,j) \in BL, d \in \{0, ..., 9\}, l \in \{p, ..., -1\} \tag{3.34e}$$

$$z_{jdl} \in \{0,1\} \qquad \forall j \in \{j|(i,j) \in BL\}, d \in \{0, ..., 9\}, l \in \{p, ..., -1\} \tag{3.34f}$$

The problem that results by replacing the bilinear terms of **(P)** with the discretization given above, also referred to as **(NMDT-MILP-LB)**, is an approximation of the original problem. In other words, **(P)** is a relaxation of **(NMDT-MILP-LB)**. If **(NMDT-MILP-LB)** is feasible, then the resulting solution is lower bound for **(P)**.

3.6.2 Upper Bounding formulation

With the purpose of allowing λ_j to reach all possible values, it is required to close the gap between discretization points. For this reason, a slack variable $\Delta\lambda_j$ with bounds between 0 and 10^p is introduced. The continuous representation of λ_j is then given by:

$$\lambda_j = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} + \Delta\lambda_j \qquad \forall j \tag{3.35a}$$

$$0 \leq \Delta\lambda_j \leq 10^p \qquad \forall j \tag{3.35b}$$

Following the same reasoning, the continuous representation of the bilinear term $v_{ij} = x_i\lambda_j$ is therefore determined as:

$$v_{ij} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} + x_i \cdot \Delta\lambda_j \qquad \forall (i,j) \tag{3.36}$$

It can be noticed that an undesired bilinear term $x_i \cdot \Delta\lambda_j$ appears in Eq. (3.36), which is replaced by variable Δv_{ij} and the resulting equation is going to be relaxed using McCormick envelope as follows:

$$v_{ij} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} + \Delta v_{ij} \qquad \forall (i,j) \tag{3.37a}$$

$$x_i^L \cdot \Delta\lambda_j \leq \Delta v_{ij} \leq x_i^U \cdot \Delta\lambda_j \qquad \forall (i,j) \tag{3.37b}$$

$$(x_i - x_i^U) \cdot 10^p + x_i^U \cdot \Delta\lambda_j \leq \Delta v_{ij} \leq (x_i - x_i^L) \cdot 10^p + x_i^L \cdot \Delta\lambda_j \qquad \forall (i,j) \tag{3.37c}$$

Substituting Eq. (3.29a) by Eqs. (3.35), and Eq. (3.32a) by Eqs. (3.37), in **(NMDT-MILP-LB)**, a new optimization problem **(NMDT-MILP-UB)** is obtained, corresponding to a relaxation of **(P)**. In other words, **(NMDT-MILP-UB)** will be feasible for values of w_{ij}, x_i and x_j that not necessarily satisfy $w_{ij} = x_i \cdot x_j$. The objective value obtained by solving **(NMDT-MILP-UB)** will not be lower than the optimal value of **(P)**.

(NMDT-MILP-UB)

$$\left.\begin{array}{l} x_j = x_j^L + \lambda_j(u_j^U - u_j^L) \\ \lambda_j = \sum_{l=p}^{-1}\sum_{d=0}^{9} 10^l \cdot d \cdot z_{jdl} + \Delta\lambda_j \\ 0 \leq \Delta\lambda_j \leq 10^p \end{array}\right\} \qquad \forall j \in \{j|(i,j) \in BL\} \tag{3.38a}$$

$$\left.\begin{array}{l} w_{ij} = x_i x_j^L + v_{ij}(x_j^U - x_j^L) \\ v_{ij} = \sum_{l=p}^{-1}\sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{x}_{ijdl} + \Delta v_{ij} \\ x_i^L \cdot \Delta\lambda_j \leq \Delta v_{ij} \leq x_i^U \cdot \Delta\lambda_j \\ \Delta v_{ij} \leq (x_i - x_i^L)10^p + x_i^L \cdot \Delta\lambda_j \\ \Delta v_{ij} \geq (x_i - x_i^U)10^p + x_i^U \cdot \Delta\lambda_j \end{array}\right\} \qquad \forall (i,j) \in BL \tag{3.38b}$$

$$x_i = \sum_{d=0}^{9} \widehat{x}_{ijdl} \qquad \forall (i,j) \in BL, l \in \{p, ..., -1\} \tag{3.38c}$$

$$\sum_{d=0}^{9} z_{jdl} = 1 \qquad \forall j \in \{j|(i,j) \in BL\}, l \in \{p, ..., -1\} \tag{3.38d}$$

$$z_{jdl} x_i^L \leq \widehat{x}_{ijdl} \leq z_{jdl} x_i^U \qquad \forall (i,j) \in BL, d \in \{0, ..., 9\}, l \in \{p, ..., -1\} \tag{3.38e}$$

3.7 SUMMARY

McCormick Envelopes are a particular class of convex relaxations for bilinear problems. By maximizing a concave function subject to a convex set, the resulting optimal solution will be a global maximum for the relaxed problem. This solution is then an upper bound for the original problem **(P)**. Instead of applying only one envelope for the entire space, one can partition and apply a tighter envelope for each partition, using Piecewise McCormick. It involves partitioning the domain of variable x_j in **(P)** for the univariate case or x_i and x_j in **(P)** for the bivariate case. Although bivariate partitioning does increase the size of the model, this is not the only factor that affects the performance of a global optimization algorithm. The quality of results from a larger relaxation model may be better comparatively to a smaller model (HASAN; KARIMI, 2010). Since the number of partitions directly relates to the binary variables added, this is an important tuning parameter. Experiments by Misener and Floudas (2012) have shown that $N = 2, 4$ and 8 are a reasonable choice.

Multiparametric disaggregation is a conceptually distinct class of MILP relaxations. It operates by discretizing variable x_j in **(P)** to a certain accuracy level p and subsequently adding slack variables to achieve continuous domains. Accuracy in MDT can only change by one order of magnitude since partitioning is based on a numeric representation system. For this reason the number of added binary variables grows logarithmically, while with PMCK it grows linearly. For this reason, smaller optimality gaps are expected to be achieved with MDT, since computational performance is often orders of magnitude faster.

The normalized version of the MDT is known as Normalized Multiparamectric Disaggregation and it seeks to bring multiparametric disaggregation closer to uniform piecewise McCormick. This is done by considering a dimensionless domain for the discretized variables in MDT. In other words, instead of discretizing all values that a variable can assume, the discretization happens in a range [0, 1] between the lower and upper bound. Doing so, the accuracy level parameter p will be directly related to the number of partitions, allowing a more precise comparison between the two alternative relaxation methods. The important feature of scaling logarithmically with the number of partitions is kept.

4 ALGORITHMS, MODELS AND APPLICATION

4.1 STRATEGY OVERVIEW

This sections detail the proposed solution strategy, which consists of applying the relaxation methods discussed in this work to the Operational Management of Crude Oil Supply problem stated in Section 2.2.

The approach solves the MILP relaxation of formulation (2.27), in which Eq. (2.22) is dropped and replaced by the relaxation methods, which will provide an upper bound to the MINLP maximization problem. Recalling the blending constraint (2.22) imposed in storage and charging tanks.

$$\frac{vct_{i,v,c}}{vt_{i,v}} = \frac{lcr_{i,r,c}}{lr_{i,r}} \Rightarrow vct_{i,v,c}lr_{i,r} = vt_{i,v}lcr_{i,r,c}$$

$$i \in \mathcal{T}, r \in \mathcal{RS} \cup \mathcal{RC}, v \in \mathcal{O}_r, c \in \mathcal{C}. \quad \text{(2.22 revisited)}$$

It is important to notice that the two sides of the equation have bilinear terms. However, the MILP relaxation becomes hard to solve due to the large number of binary variables. Because the OMCOS blending constraint has a unique structure, consisting of two bilinear terms linked by an equality constraint, it is also proposed to tight only one of the bilinear terms, while the other remains bounded by a simple McCormick envelope.

Therefore, the same linear relaxation method must be applied for both of them. In the next sections, the left bilinear term will be referenced as Left-Hand Side with its respective variables containing a superscript LHS, and conversely the right bilinear term will be referenced as Right-Hand Side with its respective variables containing a superscript RHS.

By fixing the logistics decisions (i.e., binary variables) to the values obtained from the solution of the MILP relaxation, a continuous non-linear program is obtained. Its solution yields a lower bound for the MINLP problem.

The solution strategy adopted in this work is a MILP-NLP decomposition consisting of the following steps:

- **Step 1:** Set initial parameters required by each relaxation technique.
- **Step 2:** Solve the MILP relaxation of the crude oil operational management problem, obtained by relaxing the bilinear terms with the relaxation technique. Notice that an upper bound for the objective function f of OMCOS is obtained when the relaxation is solved to optimally.

- **Step 3:** Fix the binary decision variables of the OMCOS MINLP formulation as the values obtained from the MILP relaxation solution, namely the logistic decisions regarding the operations ($z_{i,v}$) and vessel trips ($s_{i,r,v,u}$). By fixing the binary variables, the OMCOS results in a continuous Non-Linear Programming (NLP) problem. A feasible solution to this NLP yields a lower bound for OMCOS.

If Step 3 does not produce a feasible solution or the quality of the solution is not satisfactory, the procedure can be repeated from Step 1 with new initial parameters. A key feature of the MILP-NLP decomposition is the simplicity and the use of MILP algorithms, which are more efficient and robust than MINLP algorithms. Such a strategy proved to be effective in other applications (ASSIS; CAMPONOGARA; GROSSMANN, 2021).

4.2 STANDARD MCCORMICK ENVELOPES

A linear relaxation using McCormick Envelopes is derived replacing the nonlinear blending constraints (2.22) by McCormick Envelopes as follows. To obtain the McCormick envelopes, the left term of the equality ($vct_{i,v,c} \cdot lr_{i,r}$) is replaced by $w^{\text{LHS}}_{i,r,v,c}$ and the right term ($vt_{i,v} \cdot lcr_{i,r,c}$) by $w^{\text{RHS}}_{i,r,v,c}$.

4.2.1 Left-Hand Side Term

For $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$, the McCormick envelope is derived from the bound constraints for the involved variables:

$$\underline{CAP}_r \leq lr_{i,r} \leq \overline{CAP}_r \tag{4.1a}$$

$$0 \leq vct_{i,v,c} \leq \overline{FR}_v \tag{4.1b}$$

Rearranging the first of these inequalities,

$$lr_{i,r} - \underline{CAP}_r \geq 0 \tag{4.2a}$$

$$\overline{CAP}_r - lr_{i,r} \geq 0 \tag{4.2b}$$

and the second,

$$vct_{i,v,c} \geq 0 \tag{4.3a}$$

$$\overline{FR}_v - vct_{i,v,c} \geq 0 \tag{4.3b}$$

and multiplying (4.2) with (4.3), we obtain the McCormick envelopes $w^{\text{LHS}}_{i,r,v,c}$ as follows:

$$w^{\text{LHS}}_{i,r,v,c} \geq \underline{CAP}_r vct_{i,v,c} \tag{4.4a}$$

$$w^{\text{LHS}}_{i,r,v,c} \geq \overline{CAP}_r vct_{i,v,c} + \overline{FR}_v lr_{i,r} - \overline{FR}_v \overline{CAP}_r \tag{4.4b}$$

$$w^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}_r vct_{i,v,c} \tag{4.4c}$$

$$w^{\text{LHS}}_{i,r,v,c} \leq \underline{CAP}_r vct_{i,v,c} + \overline{FR}_v lr_{i,r} - \overline{FR}_v \underline{CAP}_r \tag{4.4d}$$

4.2.2 Right-Hand Side Term

Similarly, for $w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r,c}$, the McCormick envelope results from the bound constraints on the involved variables:

$$0 \leq lcr_{i,r,c} \leq \overline{CAP}_r \tag{4.5a}$$

$$\underline{FR}_v z_{i,v} \leq vt_{i,v} \leq \overline{FR}_v z_{i,v} \tag{4.5b}$$

Rearranging the first of these inequalities,

$$lcr_{i,r,c} - 0 \geq 0 \tag{4.6a}$$

$$\overline{CAP}_r - lcr_{i,r,c} \geq 0 \tag{4.6b}$$

and the second,

$$vt_{i,v} - \underline{FR}_v z_{i,v} \geq 0 \tag{4.7a}$$

$$\overline{FR}_v z_{i,v} - vt_{i,v} \geq 0 \tag{4.7b}$$

and multiplying (4.6) with (4.7), we obtain the McCormick envelopes $w^{\text{RHS}}_{i,r,v,c}$ as follows:

$$w^{\text{RHS}}_{i,r,v,c} \geq \underline{FR}_v z_{i,v} lcr_{i,r,c} \tag{4.8a}$$

$$w^{\text{RHS}}_{i,r,v,c} \geq \overline{CAP}_r vt_{i,v} + \overline{FR}_v z_{i,v} lcr_{i,r,c} - \overline{FR}_v z_{i,v} \overline{CAP}_r \tag{4.8b}$$

$$w^{\text{RHS}}_{i,r,v,c} \leq \overline{FR}_v z_{i,v} lcr_{i,r,c} \tag{4.8c}$$

$$w^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r vt_{i,v} + \underline{FR}_v z_{i,v} lcr_{i,r,c} - \underline{FR}_v z_{i,v} \overline{CAP}_r \tag{4.8d}$$

It can be noticed that the bounds for variable $vt_{i,v}$ contain the terms $\underline{FR}_v z_{i,v}$ and $\overline{FR}_v z_{i,v}$ that involve the binary decision variable $z_{i,v}$, which add complicating bilinear terms in the envelopes. This issue can be fixed by formulating the McCormick envelope as a Generalized Disjunctive Program (GDP). For each $i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}$, the disjunction between the following two equations is defined:

$$\begin{bmatrix} z_{i,v} \\ w^{\text{RHS}}_{i,r,v,c} \geq \underline{FR}_v lcr_{i,r,c} \\ w^{\text{RHS}}_{i,r,v,c} \geq \overline{CAP}_r vt_{i,v} + \overline{FR}_v lcr_{i,r,c} - \overline{FR}_v \overline{CAP}_r \\ w^{\text{RHS}}_{i,r,v,c} \leq \overline{FR}_v lcr_{i,r,c} \\ w^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r vt_{i,v} + \underline{FR}_v lcr_{i,r,c} - \underline{FR}_v \overline{CAP}_r \\ 0 \leq lcr_{i,r,c} \leq \overline{CAP}_r \\ \underline{FR}_v \leq vt_{i,v} \leq \overline{FR}_v \end{bmatrix}, \tag{4.9a}$$

$$\begin{bmatrix} (1 - z_{i,v}) \\ w^{\text{RHS}}_{i,r,v,c} = 0 \\ 0 \leq lcr_{i,r,c} \leq \overline{CAP}_r \\ vt_{i,v} = 0 \end{bmatrix}, \tag{4.9b}$$

To solve a GDP problem using existing solvers, one has to first translate the problem into a MILP problem by reformulating all the logic expressions into algebraic forms.

For the GDP related to Eqs. (4.9), the convex hull reformulation is derived as follows,

$$\begin{cases} w1^{\text{RHS}}_{i,r,v,c} \geq \underline{FR}_v lcr1_{i,r,c} \\ w1^{\text{RHS}}_{i,r,v,c} \geq \overline{CAP}_r vt1_{i,v} + \overline{FR}_v lcr1_{i,r,c} - \overline{FR}_v \overline{CAP}_r \cdot z_{i,v} \\ w1^{\text{RHS}}_{i,r,v,c} \leq \overline{FR}_v lcr1_{i,r,c} \\ w1^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r vt1_{i,v} + \underline{FR}_v lcr1_{i,r,c} - \underline{FR}_v \overline{CAP}_r \cdot z_{i,v} \\ 0 \leq lcr1_{i,r,c} \leq \overline{CAP}_r \cdot z_{i,v} \\ \underline{FR}_v \cdot z_{i,v} \leq vt1_{i,v} \leq \overline{FR}_v \cdot z_{i,v} \\ w0^{\text{RHS}}_{i,r,v,c} = 0 \\ 0 \cdot (1 - z_{i,v}) \leq lcr0_{i,r,c} \leq \overline{CAP}_r \cdot (1 - z_{i,v}) \\ vt0_{i,v} = 0 \\ w^{\text{RHS}}_{i,r,v,c} = w0^{\text{RHS}}_{i,r,v,c} + w1^{\text{RHS}}_{i,r,v,c} \\ lcr_{i,r,c} = lcr0_{i,r,c} + lcr1_{i,r,c} \\ vt_{i,v} = vt0_{i,v} + vt1_{i,v} \end{cases} \tag{4.10a}$$

Alternatively, instead of using the GDP, we reformulated (4.9) introducing $wz_{i,r,v,c} = z_{i,v} \cdot lcr_{i,r,c}$ and further it is sufficient to apply the big-M strategy for simplicity as follows:

$$\begin{cases} w^{\text{RHS}}_{i,r,v,c} \geq \underline{FR}_v wz_{i,r,v,c} \\ w^{\text{RHS}}_{i,r,v,c} \geq \overline{CAP}_r vt_{i,v} + \overline{FR}_v wz_{i,r,v,c} - \overline{FR}_v \overline{CAP}_r \cdot z_{i,v} \\ w^{\text{RHS}}_{i,r,v,c} \leq \overline{FR}_v wz_{i,r,v,c} \\ w^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r vt_{i,v} + \underline{FR}_v wz_{i,r,v,c} - \underline{FR}_v \overline{CAP}_r \cdot z_{i,v} \\ 0 \leq wz_{i,r,v,c} \leq \text{bigM} \cdot z_{i,v} \\ -\text{bigM}(1 - z_{i,v}) + lcr_{i,r,c} \leq wz_{i,r,v,c} \leq lcr_{i,r,c} \end{cases} \tag{4.11}$$

Notice that it suffices to use $\overline{CAP}_r$ as the bigM value for a constraint on the variable $wz_{i,r,v,c}$. Additionally, the bound constraints on $vt_{i,v}$ is already part of the original problem, namely the constraint (2.20), $\underline{FR}_v z_{i,v} \leq vt_{i,v} \leq \overline{FR}_v z_{i,v}$, which was reproduced here for convenience.

Finally, after the bilinear terms are reformulated as $w^{\text{LHS}}_{i,r,v,c}$ and $w^{\text{RHS}}_{i,r,v,c}$, the following constraint must be added:

$$w^{\text{LHS}}_{i,r,v,c} = w^{\text{RHS}}_{i,r,v,c}, \ \forall i \in \mathcal{T}, r \in \mathcal{RC}, v \in \mathcal{O}_r, c \in \mathcal{C} \tag{4.12}$$

4.3 UNIVARIATE PIECEWISE MCCORMICK

To keep the presentation as short as possible, the univariate piecewise McCormick relaxation is not explicitly developed here. However, this relaxation is derived from the bivariate piecewise McCormick relaxation that follows below, by setting the number of partitions for one of the variables to be equal 1.

4.4 BIVARIATE PIECEWISE MCCORMICK ENVELOPES

A tighter mixed-integer linear programming relaxation will be constructed by partitioning the domain of both variables in each bilinear term.

4.4.1 Envelope for the Left-Hand Side Term

For $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$, let the new binary variable $p^{n,n'}_{i,r,v,c}$ indicate the active partition n for variable $lr_{i,r}$ and n' for variable $vct_{i,v,c}$. The bound constraints with partitions $n = \{1, ..., N\}$ and $n' = \{1, ..., N'\}$ for the respective involved variables will become,

$$\underline{CAP}^n_r \leq lr_{i,r} \leq \overline{CAP}^n_r \tag{4.13a}$$

$$\underline{FR}^{n'}_v \leq vct_{i,v,c} \leq \overline{FR}^{n'}_v \tag{4.13b}$$

The piecewise McCormick relaxation corresponding to the bivariate partitioning case can be formulated as the following Generalized Disjunctive Program.

The following equations are defined for all $i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}$.

$$\bigvee_{n=1}^{N} \bigvee_{n'=1}^{N'} \begin{bmatrix} p^{n,n'}_{i,r,v,c} \\ w^{\text{LHS}}_{i,r,v,c} \geq \underline{CAP}^n_r vct_{i,v,c} + \underline{FR}^{n'}_v lr_{i,r} - \underline{FR}^{n'}_v \underline{CAP}^n_r \\ w^{\text{LHS}}_{i,r,v,c} \geq \overline{CAP}^n_r vct_{i,v,c} + \overline{FR}^{n'}_v lr_{i,r} - \overline{FR}^{n'}_v \overline{CAP}^n_r \\ w^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}^n_r vct_{i,v,c} + \underline{FR}^{n'}_v lr_{i,r} - \underline{FR}^{n'}_v \overline{CAP}^n_r \\ w^{\text{LHS}}_{i,r,v,c} \leq \underline{CAP}^n_r vct_{i,v,c} + \overline{FR}^{n'}_v lr_{i,r} - \overline{FR}^{n'}_v \underline{CAP}^n_r \\ \underline{CAP}^n_r \leq lr_{i,r} \leq \overline{CAP}^n_r \\ \underline{FR}^{n'}_v \leq vct_{i,v,c} \leq \overline{FR}^{n'}_v \end{bmatrix} \tag{4.14a}$$

in which

$$\begin{cases} \underline{CAP}^n_r = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n-1)}{N} \\ \overline{CAP}^n_r = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n)}{N} \\ \underline{FR}^{n'}_r = \underline{FR}_r + \frac{(\overline{FR}_r - \underline{FR}_r)(n'-1)}{N'} \\ \overline{FR}^{n'}_r = \underline{FR}_r + \frac{(\overline{FR}_r - \underline{FR}_r)(n')}{N'} \end{cases} \tag{4.14b}$$

$$p^{n,n'}_{i,r,v,c} \in \{True, False\}, n \in \{1, ..., N\}, n' \in \{1, ..., N'\} \tag{4.14c}$$

Then, the linear GDP is reformulated using the convex-hull approach, leading to the following MILP formulation:

$$\begin{cases} w^{\text{LHS}}_{i,r,v,c} \geq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \underline{CAP}^n_r vct^{n,n'}_{i,r,v,c} + \underline{FR}^{n'}_v lr^{n,n'}_{i,r,v,c} - \underline{FR}^{n'}_v \underline{CAP}^n_r \cdot p^{n,n'}_{i,r,v,c} \\ w^{\text{LHS}}_{i,r,v,c} \geq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \overline{CAP}^n_r vct^{n,n'}_{i,r,v,c} + \overline{FR}^{n'}_v lr^{n,n'}_{i,r,v,c} - \overline{FR}^{n'}_v \overline{CAP}^n_r \cdot p^{n,n'}_{i,r,v,c} \\ w^{\text{LHS}}_{i,r,v,c} \leq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \overline{CAP}^n_r vct^{n,n'}_{i,r,v,c} + \underline{FR}^{n'}_v lr^{n,n'}_{i,r,v,c} - \underline{FR}^{n'}_v \overline{CAP}^n_r \cdot p^{n,n'}_{i,r,v,c} \\ w^{\text{LHS}}_{i,r,v,c} \leq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \underline{CAP}^n_r vct^{n,n'}_{i,r,v,c} + \overline{FR}^{n'}_v lr^{n,n'}_{i,r,v,c} - \overline{FR}^{n'}_v \underline{CAP}^n_r \cdot p^{n,n'}_{i,r,v,c} \\ lr_{i,r} = \sum_{n=1}^{N}\sum_{n'=1}^{N'} lr^{n,n'}_{i,r,v,c} \\ vct_{i,v,c} = \sum_{n=1}^{N}\sum_{n'=1}^{N'} vct^{n,n'}_{i,r,v,c} \\ \sum_{n=1}^{N}\sum_{n'=1}^{N'} p^{n,n'}_{i,r,v,c} = 1 \end{cases} \tag{4.15a}$$

$$\begin{cases} \underline{CAP}^n_r p^{n,n'}_{i,r,v,c} \leq lr^{n,n'}_{i,r,v,c} \leq \overline{CAP}^n_r p^{n,n'}_{i,r,v,c} \\ \underline{FR}^{n'}_v p^{n,n'}_{i,r,v,c} \leq vct^{n,n'}_{i,r,v,c} \leq \overline{FR}^{n'}_v p^{n,n'}_{i,r,v,c} \\ \underline{CAP}^n_r = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n-1)}{N} \\ \overline{CAP}^n_r = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n)}{N} \\ \underline{FR}^{n'}_v = \underline{FR}_v + \frac{(\overline{FR}_v - \underline{FR}_v)(n'-1)}{N'} \\ \overline{FR}^{n'}_v = \underline{FR}_v + \frac{(\overline{FR}_v - \underline{FR}_v)(n')}{N'} \end{cases} \tag{4.15b}$$

$$p^{n,n'}_{i,r,v,c} \in \{0,1\},\ n \in \{1,...,N\}, n' \in \{1,...,N'\} \tag{4.15c}$$

4.4.2 Envelope for the Right-Hand Side Term

For $w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r,c}$, let the new binary variable $q^{n,n'}_{i,r,v,c}$ indicate the active partition n for the variable $lcr_{i,r,c}$ and n' for the variable $vt_{i,v}$. When the variable $lcr_{i,r,c}$ is within a partition $n \in \{1,...,N\}$ and respectively $vt_{i,v}$ is within a partition $n' \in \{1,...,N'\}$, they will become bounded as follows:

$$\underline{CAP}^n_r \leq lcr_{i,r,c} \leq \overline{CAP}^n_r \tag{4.16a}$$

$$\underline{FR}^{n'}_v z_{i,v} \leq vt_{i,v} \leq \overline{FR}^{n'}_v z_{i,v} \tag{4.16b}$$

The following equations are defined for all $i \in \mathcal{T},\ r \in \mathcal{RS} \cup \mathcal{RC},\ v \in \mathcal{O}_r,\ c \in \mathcal{C}$.

$$\bigvee_{n=1}^{N}\bigvee_{n'=1}^{N'} \begin{bmatrix} q_{i,r,v,c}^{n,n'} \\ w_{i,r,v,c}^{\text{RHS}} \geq \underline{CAP}_r^n vt_{i,v} + \underline{FR}_v^{n'} lcr_{i,r,c} z_{i,v} - \underline{FR}_v^{n'} \underline{CAP}_r^n \cdot z_{i,v} \\ w_{i,r,v,c}^{\text{RHS}} \geq \overline{CAP}_r^n vt_{i,v} + \overline{FR}_v^{n'} lcr_{i,r,c} z_{i,v} - \overline{FR}_v^{n'} \overline{CAP}_r^n \cdot z_{i,v} \\ w_{i,r,v,c}^{\text{RHS}} \leq \underline{CAP}_r^n vt_{i,v} + \overline{FR}_v^{n'} lcr_{i,r,c} z_{i,v} - \overline{FR}_v^{n'} \underline{CAP}_r^n \cdot z_{i,v} \\ w_{i,r,v,c}^{\text{RHS}} \leq \overline{CAP}_r^n vt_{i,v} + \underline{FR}_v^{n'} lcr_{i,r,c} z_{i,v} - \underline{FR}_v^{n'} \overline{CAP}_r^n \cdot z_{i,v} \\ \underline{CAP}_r^n \leq lcr_{i,r,c} \leq \overline{CAP}_r^n \\ \underline{FR}_v^{n'} z_{i,v} \leq vt_{i,v} \leq \overline{FR}_v^{n'} z_{i,v} \end{bmatrix} \tag{4.17a}$$

$$\begin{cases} \underline{CAP}_r^n = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n-1)}{N} \\ \overline{CAP}_r^n = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n)}{N} \\ \underline{FR}_r^{n'} = \underline{FR}_r + \frac{(\overline{FR}_r - \underline{FR}_r)(n'-1)}{N'} \\ \overline{FR}_r^{n'} = \underline{FR}_r + \frac{(\overline{FR}_r - \underline{FR}_r)(n')}{N'} \end{cases} \tag{4.17b}$$

$$q_{i,r,v,c}^{n,n'} \in \{True, False\}, n \in \{1, ..., N\}, n' \in \{1, ..., N'\} \tag{4.17c}$$

By introducing the variables $wz_{i,r,v,c}^{n,n'} = lcr_{i,r,c} \cdot z_{i,v}$, this GDP is recast using the convex hull method to obtain the following MILP reformulation:

$$\begin{cases} w_{i,r,v,c}^{\text{RHS}} \geq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \underline{CAP}_r^n vt_{i,r,v,c}^{n,n'} + \underline{FR}_v^{n'} wz_{i,r,v,c}^{n,n'} - \underline{FR}_v^{n'} \underline{CAP}_r^n \cdot x_{i,r,v,c}^{n,n'} \\ w_{i,r,v,c}^{\text{RHS}} \geq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \overline{CAP}_r^n vt_{i,r,v,c}^{n,n'} + \overline{FR}_v^{n'} wz_{i,r,v,c}^{n,n'} - \overline{FR}_v^{n'} \overline{CAP}_r^n \cdot x_{i,r,v,c}^{n,n'} \\ w_{i,r,v,c}^{\text{RHS}} \leq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \overline{CAP}_r^n vt_{i,r,v,c}^{n,n'} + \underline{FR}_v^{n'} wz_{i,r,v,c}^{n,n'} - \underline{FR}_v^{n'} \overline{CAP}_r^n \cdot x_{i,r,v,c}^{n,n'} \\ w_{i,r,v,c}^{\text{RHS}} \leq \sum_{n=1}^{N}\sum_{n'=1}^{N'} \underline{CAP}_r^n vt_{i,r,v,c}^{n,n'} + \overline{FR}_v^{n'} wz_{i,r,v,c}^{n,n'} - \overline{FR}_v^{n'} \underline{CAP}_r^n \cdot x_{i,r,v,c}^{n,n'} \\ lcr_{i,r} = \sum_{n=1}^{N}\sum_{n'=1}^{N'} lcr_{i,r,v,c}^{n,n'} \\ vt_{i,v,c} = \sum_{n=1}^{N}\sum_{n'=1}^{N'} vt_{i,r,v,c}^{n,n'} \\ \sum_{n=1}^{N}\sum_{n'=1}^{N'} q_{i,r,v,c}^{n,n'} = 1 \\ \sum_{n=1}^{N}\sum_{n'=1}^{N'} x_{i,r,v,c}^{n,n'} = z_{i,v} \\ 0 \leq wz_{i,r,v,c}^{n,n'} \leq \text{bigM} \cdot z_{i,v} \\ -\text{bigM}(1 - z_{i,v}) + lcr_{i,r,v,c}^{n,n'} \leq wz_{i,r,v,c}^{n,n'} \leq lcr_{i,r,v,c}^{n,n'} \\ x_{i,r,v,c}^{n,n'} \leq q_{i,r,v,c}^{n,n'} \\ x_{i,r,v,c}^{n,n'} \leq z_{i,v} \\ x_{i,r,v,c}^{n,n'} \geq q_{i,r,v,c}^{n,n'} + z_{i,v} - 1 \end{cases} \tag{4.18a}$$

$$q_{i,r,v,c}^{n,n'} \in \{0,1\},\ n \in \{1,...,N\}, n' \in \{1,...,N'\} \tag{4.18b}$$

$$x_{i,r,v,c}^{n,n'} \in \{0,1\},\ n \in \{1,...,N\}, n' \in \{1,...,N'\} \tag{4.18c}$$

Notice that the big-M value for the constraint on $wz_{i,r,v,c}$ can be replaced by $\overline{CAP}_r$ in order to produce a tigher big-M.

$$\begin{cases} \underline{CAP}_r^n q_{i,r,v,c}^{n,n'} \leq lcr_{ir,v,c}^{n,n'} \leq \overline{CAP}_r^n q_{i,r,v,c}^{n,n'} \\ \underline{FR}_v^{n'} x_{i,r,v,c}^{n,n'} \leq vt_{i,r,v,c}^{n,n'} \leq \overline{FR}_v^{n'} x_{i,r,v,c}^{n,n'} \\ \underline{CAP}_r^n = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n-1)}{N} \\ \overline{CAP}_r^n = \underline{CAP}_r + \frac{(\overline{CAP}_r - \underline{CAP}_r)(n)}{N} \\ \underline{FR}_v^{n'} = \underline{FR}_v + \frac{(\overline{FR}_v - \underline{FR}_v)(n'-1)}{N'} \\ \overline{FR}_v^{n'} = \underline{FR}_v + \frac{(\overline{FR}_v - \underline{FR}_v)(n')}{N'} \end{cases} \tag{4.18d}$$

4.5 MULTIPARAMETRIC DISAGGREGATION

Here the Multiparametric Disaggregation Technique (MDT) is applied for the linear approximation and relaxation of the bilinear terms.

4.5.1 Lower Bounding Formulation

4.5.1.1 Left-Hand Side Term

The discretized term in the bilinear term $vct_{i,v,c} \cdot lr_{i,r}$ will be $vct_{i,v,c}$. The formulation is obtained deriving first a generalized disjunctive programming (GDP) model as follows:

$$w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r} \tag{4.19a}$$

$$vct_{i,v,c} = \sum_{l=p}^{P} \lambda^{\text{LHS}}_{i,v,c,l} \tag{4.19b}$$

$$\bigvee_{d=0}^{9} \left[\lambda^{\text{LHS}}_{i,v,c,l} = 10^l \cdot d\right] \qquad \forall l \in L \tag{4.19c}$$

where we discretize $vct_{i,v,c}$ through the disjunction in (4.19c) that selects one digit $d \in D = \{0, 1, ..., 9\}$ for each power $l \in L = \{p, p+1, ..., P\}$. First, we consider the convex hull reformulation of the disjunction in (4.19c), after which we introduce the disaggregated variables,

$$\lambda^{\text{LHS}}_{i,v,c,l} = \sum_{d=0}^{9} \widehat{\lambda}^{\text{LHS}}_{i,v,c,l,d} \qquad \forall l \in L \tag{4.20a}$$

$$\widehat{\lambda}^{\text{LHS}}_{i,v,c,l,d} = 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \qquad \forall l \in L, d \in D \tag{4.20b}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.20c}$$

$$y^{\text{LHS}}_{i,v,c,l,d} \in \{0, 1\} \qquad \forall l \in L, d \in D \tag{4.20d}$$

Substituting (4.20b) into (4.20a) and then into (4.19b) leads to the fully discretized (but still exact representation) of $vct_{i,v,c}$:

$$vct_{i,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \tag{4.21}$$

Considering the product $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$ by substituting (4.21) into (4.19a) leads to (4.22) which involves nonlinear terms $y^{\text{LHS}}_{i,v,c,l,d} \cdot lr_{i,r}$,

$$w^{\text{LHS}}_{i,r,v,c} = \left[\sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d}\right] \cdot lr_{i,r} \tag{4.22}$$

Performing an exact linearization, we introduce new continuous variables $\widehat{lr}_{i,r,v,c,l,d} = y^{\text{LHS}}_{i,v,c,l,d} \cdot lr_{i,r}$ so that:

$$w^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} \tag{4.23}$$

Since $y^{\text{LHS}}_{i,v,c,l,d} \cdot lr_{i,r} = \left\{ \begin{array}{ll} 0 & \text{if } y^{\text{LHS}}_{i,v,c,l,d} = 0 \\ lr_{i,r} & \text{if } y^{\text{LHS}}_{i,v,c,l,d} = 1 \end{array} \right\}$, $\widehat{lr}_{i,r,v,c,l,d}$ is non-negative, the lower bound of $lr_{i,r}$ is $\underline{CAP}_r$ and the upper bound is $\overline{CAP}_r$, we introduce the bounding constraints:

$$\underline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \leq \widehat{lr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \qquad \forall l \in L, d \in D \tag{4.24}$$

Finally, multiplying equation (4.20c) by $lr_{i,r}$ and replacing the bilinear terms by the new continuous variables, results in (4.25),

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d}, \qquad \forall l \in L \tag{4.25}$$

The full set of mixed integer linear constraints for the exact representation of the bilinear product $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$ is given by Eqs. (4.26).

(MDT-LHS-LB)

$$w^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} \tag{4.26a}$$

$$vct_{i,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \tag{4.26b}$$

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.26c}$$

$$\underline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \leq \widehat{lr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \qquad \forall l \in L, d \in D \tag{4.26d}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.26e}$$

$$y^{\text{LHS}}_{i,v,c,l,d} \in \{0, 1\} \qquad \forall l \in L, d \in D \tag{4.26f}$$

It can be noticed that the problem (MDT-LHS-LB) represents a mixed integer linear program (MILP), which is a restricted version of the original problem and therefore its solution yields a lower bound.

4.5.1.2 Right-Hand Side Term

The discretized variables in the bilinear term $vt_{i,v} \cdot lcr_{i,r,c}$ will be $vt_{i,v}$. The formulation is obtained by deriving first a generalized disjunctive programming (GDP) model as follows:

$$w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r,c} \tag{4.27a}$$

$$vt_{i,v} = \sum_{l=p}^{P} \lambda_{i,v,l}^{\text{RHS}} \tag{4.27b}$$

$$\bigvee_{d=0}^{9} \left[\lambda_{i,v,l}^{\text{RHS}} = 10^l \cdot d\right] \qquad \forall l \in L \tag{4.27c}$$

where we discretize $vt_{i,v}$ through the disjunction in (4.27c) that selects one digit $d \in D = \{0, 1, ..., 9\}$ for each power $l \in L = \{p, p+1, ..., P\}$. First, we consider the convex hull reformulation of the disjunction in (4.27c) after which we introduce the disaggregated variables,

$$\lambda_{i,v,l}^{\text{RHS}} = \sum_{d=0}^{9} \widehat{\lambda}_{i,v,l,d}^{\text{RHS}} \qquad \forall l \in L \tag{4.28a}$$

$$\widehat{\lambda}_{i,v,l,d}^{\text{RHS}} = 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} \qquad \forall l \in L, d \in D \tag{4.28b}$$

$$\sum_{d=0}^{9} y_{i,v,l,d}^{\text{RHS}} = 1 \qquad \forall l \in L \tag{4.28c}$$

$$y_{i,v,l,d}^{\text{RHS}} \in \{0, 1\} \qquad \forall l \in L, d \in D \tag{4.28d}$$

Substituting (4.28b) into (4.28a) and then into (4.27b) leads to the fully discretized (but still exact representation) of $vt_{i,v}$:

$$vt_{i,v} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} \tag{4.29}$$

Considering the product $w_{i,r,v,c}^{\text{RHS}} = vt_{i,v} \cdot lcr_{i,r,c}$ by substituting (4.29) into (4.27a) leads to (4.30) which involves the nonlinear terms $y_{i,v,l,d}^{\text{RHS}} \cdot lcr_{i,r,c}$,

$$w_{i,r,v,c}^{\text{RHS}} = \left[\sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}}\right] \cdot lcr_{i,r,c} \tag{4.30}$$

Performing an exact linearization, we introduce new continuous variables $\widehat{lcr}_{i,r,v,c,l,d} = y_{i,v,l,d}^{\text{RHS}} \cdot lcr_{i,r,c}$, so that:

$$w_{i,r,v,c}^{\text{RHS}} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} \tag{4.31}$$

Since $lcr_{i,r,c} \cdot y_{i,v,l,d}^{\text{RHS}} = \begin{Bmatrix} 0 & \text{if } y_{i,v,l,d}^{\text{RHS}} = 0 \\ lcr_{i,r,c} & \text{if } y_{i,v,l,d}^{\text{RHS}} = 1 \end{Bmatrix}$, $\widehat{lcr}_{i,r,v,c,l,d}$ is non-negative, the lower bound of $lcr_{i,r,c}$ is $\underline{CAP}_r$ and the upper bound is $\overline{CAP}_r$, we introduce the bounding constraints:

$$\underline{CAP}_r \cdot y_{i,v,l,d}^{\text{RHS}} \leq \widehat{lcr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y_{i,v,l,d}^{\text{RHS}} \qquad \forall l \in L, d \in D \tag{4.32}$$

Finally, multiplying equation (4.28c) by $lcr_{i,r,c}$ and replacing the bilinear terms by the new continuous variables, results in (4.33),

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d}, \qquad \forall l \in L \tag{4.33}$$

The full set of mixed integer linear constraints for the exact representation of the bilinear product $w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r,c}$ is given by Eqs. (4.34).

(MDT-RHS-LB)

$$w^{\text{RHS}}_{i,r,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} \tag{4.34a}$$

$$vt_{i,v} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{RHS}}_{i,v,l,d} \tag{4.34b}$$

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.34c}$$

$$\underline{CAP}_r \cdot y^{\text{RHS}}_{i,v,l,d} \leq \widehat{lcr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y^{\text{RHS}}_{i,v,l,d} \qquad \forall l \in L, d \in D \tag{4.34d}$$

$$\sum_{d=0}^{9} y^{\text{RHS}}_{i,v,l,d} = 1 \qquad \forall l \in L \tag{4.34e}$$

$$y^{\text{RHS}}_{i,v,l,d} \in \{0,1\} \qquad \forall l \in L, d \in D \tag{4.34f}$$

Similarly to the left-hand side, the problem **(MDT-RHS-LB)** represents a mixed integer linear program (MILP), which is a restricted version of the original problem and therefore its solution yields a lower bound.

4.5.2 Upper Bounding Formulation

4.5.2.1 Left-Hand Side Term

Now we introduce a slack variable $\Delta vct_{i,v,c}$ such that $vct^R_{i,r} = vct_{i,v,c}' + \Delta vct_{i,v,c}$, where $vct_{i,v,c}'$ is the discretized representation of the original bilinear term $vct_{i,v,c}$, $vct^R_{i,r}$ is the continuous representation, and the slack variable $\Delta vct_{i,v,c}$ is bounded between 0 and 10^p. Thus, the continuous representation $vct^R_{i,r}$ is denoted as:

$$vct^R_{i,r} = \underbrace{\sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d}}_{vct_{i,v,c}'} + \Delta vct_{i,v,c} \tag{4.35a}$$

$$0 \leq \Delta vct_{i,v,c} \leq 10^p \tag{4.35b}$$

For the continuous representation of the bilinear term, $w^{\text{LHS}}_{i,r,v,c}$, note that $w^{\text{LHS}}_{i,r,v,c} = vct^R_{i,r} \cdot lr_{i,r}$ so we can derive:

$$\begin{aligned} w^{\text{LHS}}_{i,r,v,c} &= vct^R_{i,r} \cdot lr_{i,r} = (vct_{i,v,c}' + \Delta vct_{i,v,c}) \cdot lr_{i,r} \\ &= w^{\text{LHS}}_{i,r,v,c}{}' + lr_{i,r} \cdot \Delta vct_{i,v,c} = w^{\text{LHS}}_{i,r,v,c}{}' + \Delta w^{\text{LHS}}_{i,r,v,c} \end{aligned} \tag{4.36}$$

The slack variable $\Delta w^{\text{LHS}}_{i,r,v,c}$ replaces the bilinear term $lr_{i,r} \cdot \Delta vct_{i,v,c}$ that can be relaxed using the McCormick envelope for:

$$\underline{CAP}_r \cdot \Delta vct_{i,v,c} \leq \Delta w^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta vct_{i,v,c} \tag{4.37a}$$

$$\Delta w^{\text{LHS}}_{i,r,v,c} \geq (lr_{i,r} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta vct_{i,v,c} \tag{4.37b}$$

$$\Delta w^{\text{LHS}}_{i,r,v,c} \leq (lr_{i,r} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta vct_{i,v,c} \tag{4.37c}$$

Introducing these constraints we obtain the new equations for the bilinear term:

(MDT-LHS-UB)

$$w^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} + \Delta w^{\text{LHS}}_{i,r,v,c} \tag{4.38a}$$

$$vct_{i,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} + \Delta vct_{i,v,c} \tag{4.38b}$$

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d}, \qquad \forall l \in L \tag{4.38c}$$

$$\underline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \leq \widehat{lr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y^{\text{LHS}}_{i,v,c,l,d} \qquad \forall l \in L, d \in D \tag{4.38d}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.38e}$$

$$\left\{ \begin{array}{l} \underline{CAP}_r \cdot \Delta vct_{i,v,c} \leq \Delta w^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta vct_{i,v,c} \\ \Delta w^{\text{LHS}}_{i,r,v,c} \geq (lr_{i,r} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta vct_{i,v,c} \\ \Delta w^{\text{LHS}}_{i,r,v,c} \leq (lr_{i,r} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta vct_{i,v,c} \end{array} \right\} \tag{4.38f}$$

$$0 \leq \Delta vct_{i,v,c} \leq 10^p \tag{4.38g}$$

$$y^{\text{LHS}}_{i,v,c,l,d} \in \{0,1\} \qquad \forall l \in L, d \in D \tag{4.38h}$$

4.5.2.2 Right-Hand Side Term

Now a slack variable $\Delta vt_{i,v}$ is introduced such that $vt^R_{i,v} = vt_{i,v}' + \Delta vt_{i,v}$, where $vt_{i,v}'$ is the discretized representation of the original bilinear term $w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r,c}$, $vt^R_{i,v}$ is the continuous representation, and the slack variable $\Delta vt_{i,v}$ is bounded between 0 and 10^p. Thus, the continuous representation $vt^R_{i,v}$ is denoted as:

$$vt^R_{i,v} = \underbrace{\sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{RHS}}_{i,v,l,d}}_{vt_{i,v}'} + \Delta vt_{i,v} \tag{4.39a}$$

$$0 \leq \Delta vt_{i,v} \leq 10^p \tag{4.39b}$$

Notice that for optimization purposes, we now consider that $vt_{i,v} = vt^R_{i,v}$.

For the continuous representation of the bilinear term, $w^{\text{RHS}}_{i,r,v,c}$, note that $w^{\text{RHS}}_{i,r,v,c} = vt^R_{i,v} \cdot lcr_{i,v,c}$ so we can derive:

$$\begin{aligned} w^{\text{RHS}}_{i,r,v,c} = vt^R_{i,v} \cdot lcr_{i,r,c} &= (vt_{i,v}{}' + \Delta vt_{i,v}) \cdot lcr_{i,r,c} \\ &= w^{\text{RHS}}_{i,r,v,c}{}' + lcr_{i,r,c} \cdot \Delta vt_{i,v} = w^{\text{RHS}}_{i,r,v,c}{}' + \Delta w^{\text{RHS}}_{i,r,v,c} \end{aligned} \quad (4.40)$$

The slack variable $\Delta w^{\text{RHS}}_{i,r,v,c}$ replaces the bilinear term $lcr_{i,r,c} \cdot \Delta vt_{i,v}$ that can be relaxed using the McCormick envelope that follows:

$$\underline{CAP}_r \cdot \Delta vt_{i,v} \leq \Delta w^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta vt_{i,v} \quad (4.41a)$$

$$\begin{aligned} (lcr_{i,r,c} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta vt_{i,v} &\leq \Delta w^{\text{RHS}}_{i,r,v,c} \\ &\leq (lcr_{i,r,c} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta vt_{i,v} \end{aligned} \quad (4.41b)$$

Introducing these constraints, we obtain the new equations for the right-hand side bilinear term:

(MDT-RHS-UB)

$$w^{\text{RHS}}_{i,r,v,c} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} + \Delta w^{\text{RHS}}_{i,r,v,c} \quad (4.42a)$$

$$vt_{i,v} = \sum_{l=p}^{P} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{RHS}}_{i,v,l,d} + \Delta vt_{i,v} \quad (4.42b)$$

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d} \qquad \forall l \in L \quad (4.42c)$$

$$\underline{CAP}_r \cdot y^{\text{RHS}}_{i,v,l,d} \leq \widehat{lcr}_{i,r,v,c,l,d} \leq \overline{CAP}_r \cdot y^{\text{RHS}}_{i,v,l,d} \qquad \forall l \in L, d \in D \quad (4.42d)$$

$$\sum_{d=0}^{9} y^{\text{RHS}}_{i,v,l,d} = 1 \qquad \forall l \in L \quad (4.42e)$$

$$\left\{ \begin{array}{l} \underline{CAP}_r \cdot \Delta vt_{i,v} \leq \Delta w^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta vt_{i,v} \\ \Delta w^{\text{RHS}}_{i,r,v,c} \leq (lcr_{i,r,c} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta vt_{i,v} \\ \Delta w^{\text{RHS}}_{i,r,v,c} \geq (lcr_{i,r,c} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta vt_{i,v} \end{array} \right\} \quad (4.42f)$$

$$0 \leq \Delta vt_{i,v} \leq 10^p \quad (4.42g)$$

$$y^{\text{RHS}}_{i,v,l,d} \in \{0, 1\} \qquad \forall l \in L, d \in D \quad (4.42h)$$

4.6 NORMALIZED MULTIPARAMETRIC DISAGGREGATION

Now we derive a normalized version of the multiparametric disagreggation presented in previous section.

4.6.1 Left-Hand Side Term

For the left-hand side bilinear term $vct_{i,v,c} \cdot lr_{i,r}$, the NMDT formulation is obtained by discretizing $\lambda^{\text{LHS}}_{i,v,c} \in [0,1]$, an auxiliary variable that is employed to determine $vct_{i,v,c}$ as a linear combination of its lower bound $\underline{FR}_v$ and upper bound $\overline{FR}_v$:

$$vct_{i,v,c} = \underline{FR}_v + \lambda^{\text{LHS}}_{i,v,c}(\overline{FR}_v - \underline{FR}_v) \tag{4.43}$$

An approximate representation of $\lambda^{\text{LHS}}_{i,v,c}$ can be achieved by considering a finite number of positions $l \in \{p, p+1, ..., -1\}$, where p is a negative integer chosen by the operator.

$$\lambda^{\text{LHS}}_{i,v,c} = \sum_{l=p}^{-1} \lambda^{\text{LHS}}_{i,v,c,l} \tag{4.44}$$

For choosing the appropriate digit $d \in D = \{0, 1, ..., 9\}$ for each power $l \in L = \{p, p+1, \ldots, -1\}$, a disjunction is stated, where binary variables $y^{\text{LHS}}_{i,v,c,l,d}$ are set to 1 if digit d is selected for position l for the discretized variable $\lambda^{\text{LHS}}_{i,v,c}$:

$$\bigvee_{d=0}^{9} \begin{bmatrix} y^{\text{LHS}}_{i,v,c,l,d} \\ \lambda^{\text{LHS}}_{i,v,c,l} = 10^l \cdot d \end{bmatrix} \qquad \forall l \in L \tag{4.45}$$

The convex hull reformulation of the disjunction in (4.45) can be simplified in order to generate a sharp formulation without disaggregated variables.

$$\lambda^{\text{LHS}}_{i,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \tag{4.46a}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.46b}$$

Multiplying variable (4.43) by $lr_{i,r}$, substituting $vct_{i,v,c} \cdot lr_{i,r}$ with bilinear variable $w^{\text{LHS}}_{i,r,v,c}$, and replacing $\lambda^{\text{LHS}}_{i,v,c} \cdot lr_{i,r}$ with $\upsilon^{\text{LHS}}_{i,r,v,c}$ leads to,

$$w^{\text{LHS}}_{i,r,v,c} = lr_{i,r}\underline{FR}_v + \upsilon^{\text{LHS}}_{i,r,v,c}(\overline{FR}_v - \underline{FR}_v) \tag{4.47}$$

Substituting (4.46a) in $\upsilon^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot \lambda^{\text{LHS}}_{i,v,c}$, leads to the appearance of bilinear terms involving the product of a continuous and a binary variable.

$$\upsilon^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} \cdot 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \cdot lr_{i,r} \tag{4.48}$$

We can now perform an exact linearization by introducing new continuous variables as $\widehat{lr}_{i,r,v,c,l,d} = y^{\text{LHS}}_{i,v,c,l,d} \cdot lr_{i,r}$:

$$\upsilon^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} \tag{4.49a}$$

$$y^{\text{LHS}}_{i,v,c,l,d} \cdot \underline{FR}_v \leq \widehat{lr}_{i,r,v,c,l,d} \leq y^{\text{LHS}}_{i,v,c,l,d} \cdot \overline{FR}_v \qquad \forall d \in D, l \in L \tag{4.49b}$$

Finally, multiplying (4.46b) by $vct_{i,v,c}$ and replacing the bilinear terms by the new continuous variables results in,

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.50}$$

The full set of mixed integer linear constraints for the approximate representation of bilinear terms $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} lr_{i,r}$ is thus given by Eqs. (4.43), (4.46b)-(4.47) and (4.49)-(4.50), leading to the optimization problem:

(NMDT-LHS-MILP)

$$vct_{i,v,c} = \underline{FR}_v + \lambda^{\text{LHS}}_{i,v,c}(\overline{FR}_v - \underline{FR}_v) \tag{4.51a}$$

$$\lambda^{\text{LHS}}_{i,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} \tag{4.51b}$$

$$w^{\text{LHS}}_{i,r,v,c} = lr_{i,r}\underline{FR}_v + \upsilon^{\text{LHS}}_{i,r,v,c}(\overline{FR}_v - \underline{FR}_v) \tag{4.51c}$$

$$\upsilon^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} \tag{4.51d}$$

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.51e}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.51f}$$

$$y^{\text{LHS}}_{i,v,c,l,d} \cdot \underline{CAP}_r \leq \widehat{lr}_{i,r,v,c,l,d} \leq y^{\text{LHS}}_{i,v,c,l,d} \cdot \overline{CAP}_r \qquad \forall d \in D, l \in L \tag{4.51g}$$

With the purpose of allow $\lambda^{\text{LHS}}_{i,v,c}$ to reach all possible values, it is required to close the gap between discretization points. For this reason, a slack variable $\Delta\lambda^{\text{LHS}}_{i,v,c}$ with bounds between 0 and 10^p is introduced. The continuous representation of $\lambda^{\text{LHS}}_{i,v,c}$ is then given by:

$$\lambda^{\text{LHS}}_{i,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} + \Delta\lambda^{\text{LHS}}_{i,v,c} \tag{4.52a}$$

$$0 \leq \Delta\lambda^{\text{LHS}}_{i,v,c} \leq 10^p \tag{4.52b}$$

Following the same reasoning, the continuous representation of the bilinear term $\upsilon^{\text{LHS}}_{i,r,v,c} = lr_{i,r} \cdot \lambda^{\text{LHS}}_{i,v,c}$ is therefore determined as:

$$\upsilon^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} + lr_{i,r} \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \tag{4.53}$$

It can be noticed that a new complicating bilinear term $lr_{i,r} \cdot \Delta\lambda^{\text{LHS}}_{i,v,c}$ appears in Eq. (4.53), which is replaced by variable $\Delta\upsilon^{\text{LHS}}_{i,r,v,c}$ and the resulting equation is relaxed using McCormick envelopes as follows:

$$\underline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \leq \Delta\upsilon^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \tag{4.54a}$$

$$\Delta\upsilon^{\text{LHS}}_{i,r,v,c} \leq (lr_{i,r} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \tag{4.54b}$$

$$\Delta\upsilon^{\text{LHS}}_{i,r,v,c} \geq (lr_{i,r} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \tag{4.54c}$$

Substituting Eqs. (4.46a) and (4.49a) in **(NMDT-LHS-MILP)** by (4.52a) –(4.54c), a new optimization problem is obtained **(NMDT-LHS-UB)**, corresponding to a relaxation of the original problem. In other words, **(NMDT-UB-MILP)** will be feasible for values of $w^{\text{LHS}}_{i,r,v,c}$, $vct_{i,v,c}$ and $lr_{i,r}$ that not necessarily satisfy $w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$.

(NMDT-LHS-UB)

$$\left.\begin{aligned} & vct_{i,v,c} = 0 + \lambda^{\text{LHS}}_{i,v,c}(\overline{FR}_v - 0) \\ & \lambda^{\text{LHS}}_{i,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{LHS}}_{i,v,c,l,d} + \Delta\lambda^{\text{LHS}}_{i,v,c} \\ & 0 \leq \Delta\lambda^{\text{LHS}}_{i,v,c} \leq 10^p \end{aligned}\right\} \tag{4.55a}$$

$$\left.\begin{aligned} & w^{\text{LHS}}_{i,r,v,c} = lr_{i,r}0 + \upsilon^{\text{LHS}}_{i,r,v,c}(\overline{FR}_v - 0) \\ & \upsilon^{\text{LHS}}_{i,r,v,c} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lr}_{i,r,v,c,l,d} + \Delta\upsilon^{\text{LHS}}_{i,r,v,c} \\ & \underline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \leq \Delta\upsilon^{\text{LHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \\ & \Delta\upsilon^{\text{LHS}}_{i,r,v,c} \leq (lr_{i,r} - \underline{CAP}_r)10^p + \underline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \\ & \Delta\upsilon^{\text{LHS}}_{i,r,v,c} \geq (lr_{i,r} - \overline{CAP}_r)10^p + \overline{CAP}_r \cdot \Delta\lambda^{\text{LHS}}_{i,v,c} \end{aligned}\right\} \tag{4.55b}$$

$$lr_{i,r} = \sum_{d=0}^{9} \widehat{lr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.55c}$$

$$\sum_{d=0}^{9} y^{\text{LHS}}_{i,v,c,l,d} = 1 \qquad \forall l \in L \tag{4.55d}$$

$$y^{\text{LHS}}_{i,v,c,l,d}\underline{CAP}_r \leq \widehat{lr}_{i,r,v,c,l,d} \leq y^{\text{LHS}}_{i,v,c,l,d}\overline{CAP}_r \qquad \forall d \in D, l \in L \tag{4.55e}$$

4.6.2 Right-Hand Side Term

For the right-hand side bilinear term, the NMDT formulation is obtained by discretizing $\lambda^{\text{RHS}}_{i,v} \in [0, 1]$, an auxiliary variable that is employed to compute $vt_{i,v}$ as a linear combination of its lower bound $\underline{FR}_v$ and upper bound $\overline{FR}_v$:

$$vt_{i,v} = \underline{FR}_v + \lambda_{i,v}^{\text{RHS}}(\overline{FR}_v - \underline{FR}_v) \tag{4.56}$$

The approximate representation of $\lambda_{i,v}^{\text{RHS}}$ can be achieved by considering a finite number of positions $l \in L = \{p, p+1, ..., -1\}$, where p is a negative integer chosen by the operator.

$$\lambda_{i,v}^{\text{RHS}} = \sum_{l=p}^{-1} \lambda_{i,v,l}^{\text{RHS}} \tag{4.57}$$

For choosing the appropriate digit $d \in D = \{0, 1, ..., 9\}$ for each power l, a disjunction is introduced, where binary variables $y_{i,v,l,d}^{\text{RHS}}$ take the value of 1 if digit d is selected for position l of discretized variable $\lambda_{i,v}^{\text{RHS}}$:

$$\bigvee_{d=0}^{9} \begin{bmatrix} y_{i,v,l,d}^{\text{RHS}} \\ \lambda_{i,v,l}^{\text{RHS}} = 10^l \cdot d \end{bmatrix} \tag{4.58}$$

The convex hull reformulation of the disjunction in (4.58) can be simplified in order to generate a sharp formulation without disaggregated variables.

$$\lambda_{i,v}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} \tag{4.59a}$$

$$\sum_{d=0}^{9} y_{i,v,l,d}^{\text{RHS}} = 1 \qquad \forall l \in L \tag{4.59b}$$

Multiplying variable (4.56) by $lcr_{i,r,c}$, substituting $lcr_{i,r,c} \cdot vt_{i,v}$ with the bilinear variable $w_{i,r,v,c}^{\text{RHS}}$, and replacing $lcr_{i,r,c} \cdot \lambda_{i,v}^{\text{RHS}}$ with the variable $\upsilon_{i,r,v,c}^{\text{RHS}}$ leads to,

$$w_{i,r,v,c}^{\text{RHS}} = lcr_{i,r,c}\underline{FR}_v + \upsilon_{i,r,v,c}^{\text{RHS}}(\overline{FR}_v - \underline{FR}_v) \tag{4.60}$$

Substituting (4.59a) into the definition of $\upsilon_{i,r,v,c}^{\text{RHS}}$ leads to the appearance of bilinear terms involving the product of a continuous and a binary variable.

$$\begin{aligned} \upsilon_{i,r,v,c}^{\text{RHS}} &= lcr_{i,r,c} \cdot \lambda_{i,v}^{\text{RHS}} \\ &= lcr_{i,r,c} \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} \\ &= \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot lcr_{i,r,c} \cdot y_{i,v,l,d}^{\text{RHS}} \end{aligned} \tag{4.61}$$

We can now perform an exact linearization by introducing new continuous variables as $\widehat{lcr}_{i,r,v,c,l,d} = lcr_{i,r,c} \cdot y_{i,v,l,d}^{\text{RHS}}$:

$$\upsilon_{i,r,v,c}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} \tag{4.62a}$$

$$y_{i,v,l,d}^{\text{RHS}} \cdot \underline{FR}_v \leq \widehat{lcr}_{i,r,v,c,l,d} \leq y_{i,v,l,d}^{\text{RHS}} \cdot \overline{FR}_v \qquad \forall d \in D, l \in L \tag{4.62b}$$

Finally, multiplying (4.59b) by $lcr_{i,r,c}$ and replacing the bilinear terms by the new continuous variables results in,

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.63}$$

The full set of mixed integer linear constraints for the approximate representation of bilinear terms $w_{i,r,v,c}^{\text{RHS}} = lcr_{i,r,c} \cdot vt_{i,v}$ is thus given by Eqs. (4.56), (4.59b)-(4.60) and (4.62)-(4.63), leading to optimization problem **(NMDT-RHS-MILP)**.

(NMDT-RHS-MILP)

$$vt_{i,v} = \underline{FR}_v + \lambda_{i,v}^{\text{RHS}}(\overline{FR}_v - \underline{FR}_v) \tag{4.64a}$$

$$\lambda_{i,v}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} \tag{4.64b}$$

$$w_{i,r,v,c}^{\text{RHS}} = lcr_{i,r,c}\underline{FR}_v + \upsilon_{i,r,v,c}^{\text{RHS}}(\overline{FR}_v - \underline{FR}_v) \tag{4.64c}$$

$$\upsilon_{i,r,v,c}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} \tag{4.64d}$$

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.64e}$$

$$\sum_{d=0}^{9} y_{i,v,l,d}^{\text{RHS}} = 1 \qquad \forall l \in L \tag{4.64f}$$

$$y_{i,v,l,d}^{\text{RHS}} \cdot \underline{CAP}_r \leq \widehat{lcr}_{i,r,v,c,l,d} \leq y_{i,v,l,d}^{\text{RHS}} \cdot \overline{CAP}_r \qquad \forall d \in D, l \in L \tag{4.64g}$$

With the purpose of allowing $\lambda_{i,v}^{\text{RHS}}$ to reach all possible values, it is required to close the gap between the discretization points. For this reason, a slack variable $\Delta\lambda_{i,v}^{\text{RHS}}$ with bounds between 0 and 10^p is introduced. The continuous representation of $\lambda_{i,v}^{\text{RHS}}$ is then given by:

$$\lambda_{i,v}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot y_{i,v,l,d}^{\text{RHS}} + \Delta\lambda_{i,v}^{\text{RHS}} \tag{4.65}$$

$$0 \leq \Delta\lambda_{i,v}^{\text{RHS}} \leq 10^p \tag{4.66}$$

Following the same reasoning, the continous representation of the bilinear term $\upsilon_{i,r,v,c}^{\text{RHS}} = lcr_{i,r,c} \cdot \lambda_{i,v}^{\text{RHS}}$ is therefore determined as:

$$\upsilon_{i,r,v,c}^{\text{RHS}} = \sum_{l=p}^{-1} \sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} + lcr_{i,r,c} \cdot \Delta\lambda_{i,v}^{\text{RHS}} \tag{4.67}$$

It can be noticed that the new undesired bilinear term $lcr_{i,r,c} \cdot \Delta\lambda^{\text{RHS}}_{i,v}$ appears in Eq. (4.53), which is replaced by variable $\Delta\upsilon^{\text{RHS}}_{i,r,v,c}$ and the resulting equation is relaxed using a McCormick envelope as follows:

$$\underline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \leq \Delta\upsilon^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \tag{4.68a}$$

$$\Delta\upsilon^{\text{RHS}}_{i,r,v,c} \leq (lcr_{i,r,c} - \underline{CAP}_r) \cdot 10^p + \underline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \tag{4.68b}$$

$$\Delta\upsilon^{\text{RHS}}_{i,r,v,c} \geq (lcr_{i,r,c} - \overline{CAP}_r) \cdot 10^p + \overline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \tag{4.68c}$$

Substituting Eqs. (3.29a) and (3.32a) in **(NMDT-RHS-MILP)** by (4.65)–(4.68c), a new optimization problem is obtained **(NMDT-RHS-UB)**, corresponding to a relaxation of the original bilinear problem. In other words, **(NMDT-RHS-UB)** will be feasible for values of $w^{\text{RHS}}_{i,r,v,c}$, $lcr_{i,r,c}$ and $vt_{i,v}$ that not necessarily satisfy $w^{\text{RHS}}_{i,r,v,c} = lcr_{i,r,c} \cdot vt_{i,v}$.

(NMDT-RHS-UB)

$$\left.\begin{aligned} & vt_{i,v} = \underline{FR}_v + \lambda^{\text{RHS}}_{i,v}(\overline{FR}_v - \underline{FR}_v) \\ & \lambda^{\text{RHS}}_{i,v} = \sum_{l=p}^{-1}\sum_{d=0}^{9} 10^l \cdot d \cdot y^{\text{RHS}}_{i,v,l,d} + \Delta\lambda^{\text{RHS}}_{i,v} \\ & 0 \leq \Delta\lambda^{\text{RHS}}_{i,v} \leq 10^p \end{aligned}\right\} \tag{4.69a}$$

$$\left.\begin{aligned} & w^{\text{RHS}}_{i,r,v,c} = lcr_{i,r,c}\underline{FR}_v + \upsilon^{\text{RHS}}_{i,r,v,c}(\overline{FR}_v - \underline{FR}_v) \\ & \upsilon^{\text{RHS}}_{i,r,v,c} = \sum_{l=p}^{-1}\sum_{d=0}^{9} 10^l \cdot d \cdot \widehat{lcr}_{i,r,v,c,l,d} + \Delta\upsilon^{\text{RHS}}_{i,r,v,c} \\ & \underline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \leq \Delta\upsilon^{\text{RHS}}_{i,r,v,c} \leq \overline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \\ & \Delta\upsilon^{\text{RHS}}_{i,r,v,c} \leq (lcr_{i,r,c} - \underline{CAP}_r)10^p + \underline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \\ & \Delta\upsilon^{\text{RHS}}_{i,r,v,c} \geq (lcr_{i,r,c} - \overline{CAP}_r)10^p + \overline{CAP}_r \cdot \Delta\lambda^{\text{RHS}}_{i,v} \end{aligned}\right\} \tag{4.69b}$$

$$lcr_{i,r,c} = \sum_{d=0}^{9} \widehat{lcr}_{i,r,v,c,l,d} \qquad \forall l \in L \tag{4.69c}$$

$$\sum_{d=0}^{9} y^{\text{RHS}}_{i,v,l,d} = 1 \qquad \forall l \in L \tag{4.69d}$$

$$y^{\text{RHS}}_{i,v,l,d} \cdot \underline{CAP}_r \leq \widehat{lcr}_{i,r,v,c,l,d} \leq y^{\text{RHS}}_{i,v,l,d} \cdot \overline{CAP}_r \qquad \forall d \in D, l \in L \tag{4.69e}$$

5 ANALYSIS

This chapter presents the analysis of the previous presented relaxation techniques (see Chapter 4) when applied to the OMCOS problem. In this sense, the bilinear terms that appear in the blending constraints (2.22) are relaxed by means of Piecewise McCormick Envelops with univariate and bivariate partitions, Multiparametric Disaggregation, and Normalized Multiparametric Disaggregation.

The original MINLP instances of OMCOS were solved to optimality as a benchmark; this was done using a global solver (Gurobi and SCIP) with a maximum solving time of 10 hours. The benchmark results will provide a reference to compare with MILP-NLP decomposition, allowing the comparative analysis of the quality of the solutions and the computational time.

The analysis presented in this chapter employs the following steps:

(i) the relaxation technique is applied on both bilinear terms, that is the left-hand side ($w^{\text{LHS}}_{i,r,v,c} = vct_{i,v,c} \cdot lr_{i,r}$) and the right-hand side ($w^{\text{RHS}}_{i,r,v,c} = vt_{i,v} \cdot lcr_{i,r}$).

(ii) the relaxation technique is applied only on the left-hand side, while a standard McCormick envelope approximates the right-hand side.

(iii) the relaxation is applied only on the right-hand side, whereas a standard McCormick envelope replaces the bilinear term on the left-hand side.

For the MILP relaxations of the MILP-NLP decomposition algorithm, regarding the piecewise McCormick technique, the bilinear terms were partitioned in 2, 4, 8 and 10 partitions for the univariate and bivariate case. The number of partitions are set directly by parameter N when univariate partition is used and parameters N and N' when the bivariate relaxation is employed.

When applying MDT, its parameters are set as follows. Since the upper bound on the discretized variable is 500, we must set $P = 2$ to enable the selection of the hundredths, for all instances. As for the smallest power of ten, we selected the precision level $p = 2$ (coarse resolution), $p = 1$, $p = 0$ (highest resolution) for the MDT employed in the MILP. A greater computation effort is expected on the lower values of p. Notice that because the upper bound is 500, the discretized variable in the MDT will have 5 partitions with $p = 2$, 50 partitions with $p = 1$, and 500 partitions with $p = 0$.

The same can be inferred for NMDT, despite partitions not varying according to the upper and lower bounds as in MDT. The accuracy level parameter p now directly relates to the number of partitions N, regardless of the lower and upper bound values. For our analysis,

$p = 1$ is set to create 10 partitions, allowing for a better comparison with univariate PMCK when $N = 10$.

5.1 PROBLEM INSTANCES

For the computational experiments, three instances were designed regarding distinct scenarios for the operational management of crude oil supply. Designing the instances it is not a trivial task since it can easily lead to an infeasible problem. For this reason, they were similar to the instances used by Assis et al. (2019), changing only the initial conditions and the CDU constraints for the oil demand detailed in Table 3. The instances were named INS1, INS2 and INS3 and contain 2 FPSOs, 2 vessels, 2 storage tanks, 2 charging tanks, 1 CDU, 2 types of crude oil, and a planning horizon of 15 days. Although refineries usually contain far more resources, the instances were designed to capture the essential elements found in a real scenario.

Besides the number of resources, some data are required for the bounds, initial conditions, and distillation demands. Some key parameters will vary in the experiments, as discussed below.

Distillation is constrained by operating ranges of crude oil composition, which correspond concretely to a feasible range for each property k of the crude oil transferred to the CDU. Property k is defined by parameter $PR_{k,c}$ which regards the weight fraction associated to crude c. Together with the bounds $\underline{DEMC}_{v,k}$ and $\overline{DEMC}_{v,k}$, parameter $PR_{k,c}$ specifies the distillation demand by the user for each CDU. Tighter bounds inevitably result in a harder problem to solve.

Additionally, over the planning horizon PH stipulated, the total volume of crude oil delivered to be distilled in each CDU r is bounded by $[\underline{DEM}_r, \overline{DEM}_r]$. Adjusting these bounds also affects the difficulty of solving OMCOS when maximizing the total volume distilled.

Finally, the initial level of each crude c in every resource is part of the initial conditions. Since the blending constraints are the most complex constraints of the problem, and are affected by the initial mixture of crude types in the tanks, we consider varying initial conditions for the test scenarios.

The data regarding the crude types appear in Table 2, while the data with initial conditions and resource bounds appear in Table 3.

Table 2 – Crude Types

Crude Types (c)	Property Concentration ($PR_{k,c}$)
A	0.010
B	0.030

Table 3 – Data for Problem Instances

Instances	INS1	INS2	INS3
CDU bounds on property concentration [$\underline{DEMC}_{v,k}$ / $\overline{DEMC}_{v,k}$] Lower bound / Upper bound (weight fraction)			
CDU1	from CT1 0.005 / 0.015 from CT2 0.020 / 0.030	from CT1 0.005 / 0.025 from CT2 0.010 / 0.040	from CT1 0.005 / 0.015 from CT2 0.020 / 0.030
CDU bounds on distillation demand [$\underline{DEM}_r, \overline{DEM}_r$] Lower bound / Upper bound (10^3 bbl)			
CDU1	from CT1 800 / 1200 from CT2 800 / 1200	from CT1 800 / 1700 from CT2 800 / 1700	from CT1 1200 / 1800 from CT2 1200 / 1800
Initial Level Of Crudes (10^3 bbl) (Crude type A, Crude type B)			
FPSO1	(500 , 0)	(500 , 0)	(500 , 0)
FPSO2	(0 , 500)	(0 , 500)	(0 , 500)
Vessel1	(300 , 0)	(300 , 0)	(300 , 0)
Vessel2	(0 , 400)	(0 , 400)	(0 , 400)
ST1	(0 , 300)	(450 , 50)	(0 , 300)
ST2	(0, 0)	(400 , 100)	(0 , 0)
CT1	(500 , 0)	(500 , 0)	(500 , 0)
CT2	(0 , 500)	(100 , 400)	(0 , 500)

5.2 NUMERICAL RESULTS

The proposed mathematical programming models and algorithms were implemented in AMPL. All instances were solved on a Ubuntu 20.04.1 LTS server, with two Intel Core Xeon E5-2630 v4 Processors (2.20 GHz), adding up to 20 cores of 2 threads and 64 GB of RAM. Gurobi Optimizer version 9.0.0 build v9.0.0rc2 and SCIP version 7.0.2 were employed to solve the MINLPs, which establish the benchmark solution approach. Conversely, for the MILP-NLP decomposition, the Gurobi Optimizer was used for solving the MILPs, while Knitro 11.1.2 solved the NLP problems. The termination criteria for the MINLP problems is reached with an optimality gap of 0.01% or a time limit of 10 hours. MILP problems were set to an optimality gap of 0.01% or 2 hours. Gurobi parallelism was always active with a maximum number of threads equal to 20.

Table 4 shows the results regarding the solutions obtained by solving the MINLP directly, with Gurobi and SCIP. The optimal solution and the respective computation time are presented for each instance. In case the time limit of 10 hours is reached, the best feasible solution found is reported. It is visible that Gurobi performs better solving the MINLP as expected due to the recent improvement of its version 9, which features advanced techniques for solving MINLP problems with bilinear terms. Also, Gurobi enables parallelism while SCIP runs only in one core. Moreover, from the results one can observe that for INS2 it is easier to obtain an optimal solution, since the bounds on distillation demand and property concentration (see Table 3) are comparatively less restrictive.

Table 4 – Global Optimization results for the OMCOS

Solver	INS1	INS2	INS3
Gurobi	22 798 (22673.0s)	32 700 (150.1s)	34 045 (36000.0s)
SCIP	22 565 (36000.0s)	32 700 (6397.2s)	32 522 (36000.0s)

The complete results are in Tables 5, 6, 7 for INS1, INS2 and INS3 respectively. The two best results for each instance are highlighted in bold.

First we analyze whether the relaxation method returns better results when applied only to one side of the blending equation or both sides. The OMCOS blending constraint has a unique structure, consisting of two bilinear terms linked by an equality constraint. One could expect to obtain relaxation models that are harder to solve when relaxing both sides with refined approximations (e.g., piecewise McCormick and MDT), in part due to the large number of binary variables added, but this was not the case for the OMCOS. For all instances, the computational time required to solve the MILP relaxation obtained with tighter approximations on both sides was generally not comparatively greater than solving the relaxation with a refined model on just one side, while using a standard McCormick envelope on the other side. However, it is clear that the best lower bound found in the three instances did not come from the models with both sides relaxed with refined models (PMCK and MDT). Thus, it was possible to enhance the results and the computational time by tightening only one of the bilinear terms, while the other remained bounded by a simple McCormick envelope. The MILP-NLP decomposition, when obtained from refined relaxation methods applied to just one of the bilinear terms, produced solutions comparable to the ones obtained by solving the baseline MINLP.

Further, we examine the effects of increasing the number of partitions. As seen in Tables 5, 6, and 7, the computational time raises proportionally to the number of partitions regardless the relaxation method employed. In terms of quality of the solutions obtained, for univariate and bivariate PMCK, the MILP tends to achieve a slightly tighter upper bound with the increase of N. However, a better MILP result will not necessarily generate a better NLP primal solution in the MILP-NLP decomposition algorithm. The likely explanation is that the binary decision variables from the MILP are not always optimal and in some cases they are restricting the NLP when these variables are fixed. Thus, in this case, increasing the number of partitions will not bring significant improvements to the NLP. Although the NLP may obtain a slightly better lower bound when increasing the number of partitions, the computation effort spent for such small gain does not support the choice of a high number of partitions. None of the three instances have shown better results when $N = 10$, so the experiments suggest that $N = 4$ and 8 are preferred, which was also claimed by Misener and Floudas (2012) on their experiments.

For MDT and NMDT relaxations, the reader must recall that the number of partitions can only change by one order of magnitude and the number of added binary variables grows logarithmically, while with PMCK it grows linearly. For this reason, unlike PMCK, the numerical results have shown better bounds with a higher number of partitions. For example, the best solution found for INS2 and INS3 (see Tables 6 and 7) was obtained by applying the MDT relaxation with 500 partitions ($p = 0$). Here it should be mentioned that an attempt to increase the number of partitions was made by setting $p = -1$, resulting in 5000 partitions. However, such a high number of partitions resulted in an MILP relaxation that was excessively large for the solver, which failed to return a feasible solution as one would expect. For these reasons, we conclude that $p = 0$ is expected to be the optimal MDT accuracy parameter for the OMCOS problem based on the instances.

Regarding the different strategies of relaxation for bilinear terms applied in this study, we analyze and compare the quality of the results and the computational effort to assess whether one of the relaxation is more suitable for OMCOS. Comparing the univariate and bivariate PMCK, it is noticeable that bivariate produces tighter bounds due to the greater number of partitions. Yet, it introduces many binary variables to the model which requires more calculation. Clearly, there is a trade-off between the quality of the relaxation and the computation work required. Examining the results of INS1 in Table 5, when relaxing only the left side of the blending constraint and applying univariate PMCK with 10 partitions, a solution of $22\,749$ was obtained in 44.22s. A similar result could also be obtained by bivariate PMCK with 4x4 partitions, reaching a value of $22\,750$ but with a total time of 487.2s. Conversely, there are also examples of the contrary occurring, as in INS2 (see Table 6), when relaxing only the left-hand side of the blending constraint, the best solution found with univariate PMCK was $32\,697$ in 66.6s with 4 partitions. With bivariate PMCK, a similar result of $32\,698$ was reached in 34.4s. These findings indicate that a general rule for selecting the best relaxation method is not likely to be designed for the instances of OMCOS. Nevertheless, we can state that univariate PMCK generally yields good enough results in a shorter time, being the preferred choice. The bivariate PMCK approach would only be advised in a situation where the end-user needs to prioritize quality of solution regardless of computational effort.

From our current choice of univariate PMCK as the best relaxation method, we follow to evaluate and compare it to MDT. An exact comparison between them is not possible since MDT could only model 5, 50 or 500 partitions. In this study, we consider MDT with 5 partitions ($p = 2$) to compare with univariate PMCK containing 4 partitions. The reader will note that this approximation does not interfere on the conclusion, since MDT clearly outperforms PMCK under those conditions, even when considering a greater number of partitions on the MDT side. MDT has reached better solutions in the experiments with a faster computational time for most cases. This can be shown by examining the results when both sides are relaxed by MDT, against both sides relaxed with PMCK univariate. For INS1 (see Table 5), MDT returned a solution of $22\,652$ in 3.2s against $22\,068$ in 3.8s by PMCK. For INS2 (see Table 6) MDT obtained $32\,696$ in 76.5s versus $32\,300$ in 171.2s by PMCK. Lastly, for INS3 (see Table 7) MDT reached $33\,391$ in 77.9s, against $33\,139$ in 243.5s by PMCK. Moreover, MDT allowed to solve the instances within a reasonable time, even when modeling bilinear terms with a high number of partitions. It can be highlighted the optimal solution found in INS2 (see Table 6) using MDT only on the right-hand side of the blending constraint with $p = 0$, which is equivalent to 500 partitions. The result was similar to the optimal solution found by solving the MINLP directly, with Gurobi, within the same computational time. When comparing it to the the global solver SCIP, the solution obtained was considerably better and with a significant faster computational time.

Given that they are roughly the same method, NMDT is expected to perform similarly to MDT. In essence, NMDT provides only an alternative means for the end-user to adjust the accuracy parameter and consequently the number of partitions. For the experiments, the parameter adjustment enabled an exact comparison between NMDT and univariate PMCK with 10 partitions by setting $p = -1$, which is equivalent to 10 partitions. From the findings one can conclude that NMDT outperforms PMCK undoubtedly for all cases as expected. This conclusion is inferred contrasting the results from both sides relaxed by NMDT against both sides relaxed with PMCK univariate. For INS1 (see Table 5), NMDT returned a solution of $22\,154$ in 5.1s, against $22\,177$ in 18.6s by PMCK. For INS2 (see Table 6) NMDT obtained $32\,696$ in 220.1s versus $32\,327$ in 307.0s by PMCK. Lastly, for INS3 (see Table 7) NMDT reached $33\,072$ in 138.6s against $33\,079$ in 408.7s by PMCK.

Table 5 – Solutions for the instance INS1.

Method	Partitions	Both LHS+RHS MILP	Both LHS+RHS NLP	Only LHS MILP	Only LHS NLP	Only RHS MILP	Only RHS NLP
Piecewise Univariate	2	22 800 (1.7s)	21 946 (0.5s)	22 800 (0.8s)	22 325 (0.1s)	22 800 (4.0s)	22 291 (0.3s)
	4	22 800 (2.7s)	22 068 (1.1s)	22 800 (11.2s)	22 540 (0.2s)	22 800 (3.7s)	22 603 (0.1s)
	8	22 800 (17.9s)	22 595 (0.8s)	22 800 (12.9s)	22 456 (0.2s)	22 800 (100.7s)	22 379 (0.1s)
	10	22 800 (18.3s)	22 177 (0.3s)	**22 800 (43.8s)**	**22 750 (0.4s)**	22 798 (42.9s)	22 638 (0.5s)
Piecewise Bivariate	2x2	22 800 (3.3s)	22 282 (0.1s)	22 800 (3.3s)	22 338 (1.9s)	22 800 (11.0s)	22 275 (0.2s)
	4x4	22 799 (601.5s)	22 290 (0.7s)	**22 798 (486.9s)**	**22 750 (0.3s)**	22 799 (125.4s)	22 622 (2.3s)
	8x8	22 800 (155.7s)	22 189 (0.1s)	**22 798 (3690.3s)**	**22 796 (0.3s)**	22 800 (1899.0s)	22 687 (0.2s)
	10x10	22 800 (5871.1s)	22 299 (0.2s)	22 798 (6177.6s)	22 718 (0.2s)	22 798 (3480.8s)	22 718 (13.4s)
MDT	5 ($p = 2$)	22 800 (2.8s)	22 652 (0.4s)	22 800 (1.0s)	22 662 (0.1s)	22 800 (0.7s)	22 662 (0.1s)
	50 ($p = 1$)	22 800 (33.0s)	22 666 (0.6s)	22 800 (100.0s)	22 651 (0.2s)	22 800 (4.4s)	22 479 (0.2s)
	500 ($p = 0$)	22 612 (7206.9s)	22 200 (0.3s)	22 798 (299.6s)	22 479 (0.2s)	22 799 (15.0s)	22 666 (0.4s)
NMDT	10 ($p = 1$)	22 750 (4.4s)	22 154 (0.7s)	22 800 (11.8s)	22 632 (0.8s)	2250 (1.5s)	22 573 (0.2s)

Table 6 – Solutions for the instance INS2.

Method	Partitions	Both LHS+RHS MILP	Both LHS+RHS NLP	Only LHS MILP	Only LHS NLP	Only RHS MILP	Only RHS NLP
Piecewise Univariate	2	32 700 (112.7s)	32 209 (0.2s)	32 700 (34.2s)	32 209 (0.1s)	32 700 (68.6s)	32 300 (0.2s)
	4	32 700 (171.0s)	32 300 (0.2s)	32 700 (66.5s)	32 697 (0.1s)	32 700 (60.2s)	32 460 (0.1s)
	8	32 700 (260.9s)	32 300 (0.1s)	32 700 (54.7s)	32 300 (0.2s)	32 700 (220.3s)	32 587 (0.1s)
	10	32 700 (306.9s)	32 327 (0.1s)	32 700 (295.6s)	32 694 (0.1s)	32 700 (216.9s)	32 697 (0.2s)
Piecewise Bivariate	2x2	32 700 (69.1s)	32 695 (0.1s)	32 700 (34.4s)	32 698 (0.4s)	32 700 (200.2s)	32 695 (0.1s)
	4x4	32 700 (1628.9s)	32 256 (0.1s)	**32 700 (803.1s)**	**32 700 (0.1s)**	32 700 (322.6s)	32 299 (0.2s)
	8x8	32 700 (4608.9s)	32 256 (0.5s)	32 698 (7066.9s)	32 305 (0.2s)	32 700 (7201.0s)	32 379 (0.5s)
	10x10	32 700 (3216.8s)	32 295 (0.2s)	32 698 (7201.6s)	32 695 (0.2s)	x x	x x
MDT	5 ($p = 2$)	32 700 (76.3s)	32 696 (0.2s)	32 700 (97.6s)	32 695 (0.1s)	32 700 (44.8s)	32 300 (0.2s)
	50 ($p = 1$)	32 700 (3689.6s)	32 256 (0.2s)	32 700 (441.0s)	32 695 (0.2s)	32 700 (233.5s)	32 698 (0.1s)
	500 ($p = 0$)	x x	x x	32 694 (7200.0s)	32 695 (0.2s)	**32 700 (152.1s)**	**32 700 (0.2s)**
NMDT	10 ($p = 1$)	32 700 (220.0s)	32 696 (0.1s)	32 700 (103.3s)	32 578 (0.1s)	32 700 (55.0s)	32 697 (0.2s)

Table 7 – Solutions for the instance INS3.

Method	Partitions	Both LHS+RHS MILP	Both LHS+RHS NLP	Only LHS MILP	Only LHS NLP	Only RHS MILP	Only RHS NLP
Piecewise Univariate	2	34 200 (21.1s)	33 352 (0.5s)	34 200 (9.2s)	33 735 (0.4s)	34 200 (26.8s)	33 714 (0.5s)
	4	34 200 (242.9s)	33 149 (0.6s)	34 200 (119.2s)	33 751 (0.9s)	34 200 (133.1s)	33 777 (0.2s)
	8	34 200 (257.4s)	33 003 (0.3s)	**34 200 (258.8s)**	**33 976 (0.7s)**	34 200 (267.2s)	30 524 (0.7s)
	10	34 200 (407.6s)	33 079 (1.1s)	34 199 (632.5s)	32 799 (0.4s)	34 200 (328.0s)	33 771 (0.2s)
Piecewise Bivariate	2x2	34 200 (197.3s)	32 771 (0.5s)	34 199 (171.9s)	33 962 (0.3s)	34 200 (126.1s)	33 825 (0.6s)
	4x4	34 198 (1345.8s)	33 689 (0.2s)	34 200 (1348.9s)	33 914 (0.2s)	34 200 (2915.2s)	33 787 (2.1s)
	8x8	33 900 (7200.0s)	31 343 (1.6s)	33 853 (7200.0s)	33 697 (0.2s)	34 200 (7200.0s)	33 926 (0.2s)
	10x10	30 200 (7200.0s)	32 072 (0.7s)	33 579 (7200.6s)	33 550 (0.4s)	33 579 (7200.0s)	30 439 (0.4s)
MDT	5 ($p = 2$)	34 200 (76.3s)	33 391 (1.6s)	34 200 (96.9s)	33 993 (0.3s)	34 200 (19.6s)	33 209 (1.2s)
	50 ($p = 1$)	33 431 (7200.0s)	30 400 (1.3s)	34 200 (1139.2s)	33 863 (4.2s)	34 198 (38.5s)	33 396 (0.3s)
	500 ($p = 0$)	x x	x x	**34 198 (2611.5s)**	**34 002 (0.5s)**	34 200 (198.2s)	28 400 (0.1s)
NMDT	10 ($p = 1$)	34 150 (138.0s)	33 072 (0.6s)	34 200 (184.6s)	33 774 (0.3s)	34 150 (18.7s)	32 400 (0.1s)

6 FINAL REMARKS

This section presents a brief discussion and concluding remarks in Section 6.1. Future work appears in Section 6.2 and the resulting publications in Section 6.3

6.1 DISCUSSION AND CONCLUSION

The objective of this thesis was to evaluate the performance of different relaxation methods applied to bilinear terms that appear in the operational management of crude oil supply problem. The problem of concern entails solving a computationally hard mixed-integer nonlinear program which is faced by vertically integrated oil companies.

The evaluation strategy proposed to apply Piecewise McCormick Envelops with univariate and bivariate partitions, Multiparametric Disaggregation, and Normalized Multiparametric Disaggregation. Implementing these methods yield a mixed-integer linear programming relaxation, which was combined with a local nonlinear programming algorithm to reach a feasible schedule of operations. Instances for the problem were designed for comparison among these relaxation methods, along with common MINLP approaches, in order to measure their performance and quality of the results.

During the implementation of the relaxation methods, it was necessary to consider the unique characteristic of one of the bilinear terms, which has bounds subject to a decision variable. Thus, it was necessary to model modified McCormick envelopes considering this peculiarity in one of the equality constraint sides. This property was not an issue when employing Multiparametric or Normalized Multiparametric Disaggregation, given that only one of the variables in the bilinear term must be discretized, so it was straightforward to discretize only the variable with fixed bounds.

When analyzing the implemented methods, the choice whether to apply them only to one side of the blending equation or both sides has proved to be the major decision affecting the results. Since the Operational Management of Crude Oil Supply blending constraint has a unique structure, consisting of two bilinear terms linked by an equality constraint, it was possible to enhance the results and reduce the computational time by tightening only one of the bilinear terms, while the other remained bounded by a simple McCormick envelope. To the best of our knowledge, this is the first study to propose this alternative.

From the presented results, Multiparametric Disaggregation has reached better solutions in the experiments with a faster computational time for most cases, allowing to solve the instances within a reasonable time, even when employing a high number of partitions. For one instance of the problem, Multiparametric Disaggregation reached the optimal solution within the same computational time taken by Gurobi to solve the MINLP directly. When comparing Multiparametric Disaggregation to the the global solver SCIP, the solution obtained was considerably better and with a significant faster computational time.

Although a global optimum can be reached given sufficient computational resources and time, it was noticed that relaxation procedures can yield nearly-optimal solutions using less computational resources. Thus, the results have shown that the solution based on the proposed relaxation compares favorably with commercial global solvers.

6.2 FUTURE WORK

Besides showing that MDT can achieve tighter bounds than a global optimization solver within a given computational time, the results indicate that one can take advantage of the relaxed formulation for restrict the variable domains and further reduce the optimality gap as in Castro and Grossmann (2014).

6.3 PUBLICATIONS

This research led to the scientific article titled "Multiparametric Disaggregation Relaxation of Bilinear Terms for the Operational Management of Crude Oil Supply," which was accepted for presentation in the *17th IEEE International Conference on Automation Science and Engineering (CASE)*, to be held in Lyon, France, during August 23-27, 2021.

BIBLIOGRAPHY

AIRES, M. et al. Optimizing the petroleum supply chain at Petrobras. In: *Computer Aided Chemical Engineering*. [S.l.]: Elsevier, 2004. v. 18, p. 871–876.

ASSIS, L. S.; CAMPONOGARA, E.; GROSSMANN, I. E. A MILP-based clustering strategy for integrating the operational management of crude oil supply. *Computers & Chemical Engineering*, Elsevier, v. 145, p. 107161, 2021.

ASSIS, L. S. et al. An MINLP formulation for integrating the operational management of crude oil supply. *Computers & Chemical Engineering*, Elsevier, v. 123, p. 110–125, 2019.

ASSIS, L. S. de; 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*, Elsevier, v. 93, p. 372–388, 2016.

BARBOSA-PÓVOA, A. P. Process supply chains management–where are we? where to go next? *Frontiers in Energy Research*, Frontiers, v. 2, p. 23, 2014.

BERGAMINI, M. L.; AGUIRRE, P.; GROSSMANN, I. Logic-based outer approximation for globally optimal synthesis of process networks. *Computers & Chemical Engineering*, Elsevier, v. 29, n. 9, p. 1914–1933, 2005.

BODINGTON, C. E.; BAKER, T. E. A history of mathematical programming in the petroleum industry. *Interfaces*, INFORMS, v. 20, n. 4, p. 117–127, 1990.

CAMPONOGARA, E.; PLUCENIO, A. Scheduling dynamically positioned tankers for offshore oil offloading. *International Journal of Production Research*, Taylor & Francis, v. 52, n. 24, p. 7251–7261, 2014.

CAMPONOGARA, E.; SEMAN, L. O.; GILI, L. C. Mixed-integer bilinear and piecewise-linear models for designing switching strategies of multilevel power converters. *Computers & Electrical Engineering*, v. 77, p. 88–108, 2019.

CASTILLO, P. C.; CASTRO, P. M.; MAHALEC, V. Global optimization algorithm for large-scale refinery planning models with bilinear terms. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 56, n. 2, p. 530–548, 2017.

CASTRO, P. M. Tightening piecewise mccormick relaxations for bilinear problems. *Computers & Chemical Engineering*, Elsevier, v. 72, p. 300–311, 2015.

CASTRO, P. M. Normalized multiparametric disaggregation: an efficient relaxation for mixed-integer bilinear problems. *Journal of Global Optimization*, Springer, v. 64, n. 4, p. 765–784, 2016.

CASTRO, P. M.; GROSSMANN, I. E. Optimality-based bound contraction with multiparametric disaggregation for the global optimization of mixed-integer bilinear problems. *Journal of Global Optimization*, Springer, v. 59, n. 2-3, p. 277–306, 2014.

CASTRO, P. M.; TELES, J. P. Comparison of global optimization algorithms for the design of water-using networks. *Computers & Chemical Engineering*, Elsevier, v. 52, p. 249–261, 2013.

CATALÃO, J. P. d. S.; POUSINHO, H. M. I.; MENDES, V. M. F. Hydro energy systems management in Portugal: profit-based evaluation of a mixed-integer nonlinear approach. *Energy*, Elsevier, v. 36f, n. 1, p. 500–507, 2011.

CHRISTIANSEN, M. et al. Ship routing and scheduling in the new millennium. *European Journal of Operational Research*, Elsevier, v. 228, n. 3, p. 467–483, 2013.

FARIA, D. C.; BAGAJEWICZ, M. J. A new approach for global optimization of a class of MINLP problems with applications to water management and pooling problems. *AIChE Journal*, Wiley Online Library, v. 58, n. 8, p. 2320–2335, 2012.

GOUNARIS, C. E.; MISENER, R.; FLOUDAS, C. A. Computational comparison of piecewise-linear relaxations for pooling problems. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 48, n. 12, p. 5742–5766, 2009.

HASAN, M. F.; KARIMI, I. Piecewise linear relaxation of bilinear programs using bivariate partitioning. *AIChE Journal*, Wiley Online Library, v. 56, n. 7, p. 1880–1893, 2010.

JIA, Z.; IERAPETRITOU, M. Mixed-integer linear programming model for gasoline blending and distribution scheduling. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 42, n. 4, p. 825–835, 2003.

JIA, Z.; IERAPETRITOU, M.; KELLY, J. D. Refinery short-term scheduling using continuous time formulation: Crude-oil operations. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 42, n. 13, p. 3085–3097, 2003.

KARUPPIAH, R.; FURMAN, K. C.; GROSSMANN, I. E. Global optimization for scheduling refinery crude oil operations. *Computers & Chemical Engineering*, Elsevier, v. 32, n. 11, p. 2745–2766, 2008.

KARUPPIAH, R.; GROSSMANN, I. E. Global optimization for the synthesis of integrated water systems in chemical processes. *Computers & Chemical Engineering*, Elsevier, v. 30, n. 4, p. 650–673, 2006.

KOLODZIEJ, S.; CASTRO, P. M.; GROSSMANN, I. E. Global optimization of bilinear programs with a multiparametric disaggregation technique. *Journal of Global Optimization*, Springer, v. 57, n. 4, p. 1039–1063, 2013.

KOLODZIEJ, S. P. et al. A discretization-based approach for the optimization of the multiperiod blend scheduling problem. *Computers & Chemical Engineering*, Elsevier, v. 53, p. 122–142, 2013.

LEE, H. et al. Mixed-integer linear programming model for refinery short-term scheduling of crude oil unloading with inventory management. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 35, n. 5, p. 1630–1641, 1996.

LIMA, C.; RELVAS, S.; BARBOSA-PÓVOA, A. P. F. Downstream oil supply chain management: A critical review and future directions. *Computers & Chemical Engineering*, Elsevier, v. 92, p. 78–92, 2016.

MCCORMICK, G. P. Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems. *Mathematical Programming*, Springer, v. 10, n. 1, p. 147–175, 1976.

MISENER, R.; FLOUDAS, C. A. Global optimization of mixed-integer quadratically-constrained quadratic programs (miqcqp) through piecewise-linear and edge-concave relaxations. *Mathematical Programming*, Springer, v. 136, n. 1, p. 155–182, 2012.

MISENER, R.; GOUNARIS, C. E.; FLOUDAS, C. A. Mathematical modeling and global optimization of large-scale extended pooling problems with the (EPA) complex emissions constraints. *Computers & Chemical Engineering*, Elsevier, v. 34, n. 9, p. 1432–1456, 2010.

MISENER, R.; THOMPSON, J. P.; FLOUDAS, C. A. APOGEE: Global optimization of standard, generalized, and extended pooling problems via linear and logarithmic partitioning schemes. *Computers & Chemical Engineering*, Elsevier, v. 35, n. 5, p. 876–892, 2011.

MORO, L.; ZANIN, A.; PINTO, J. A planning model for refinery diesel production. *Computers & Chemical Engineering*, Elsevier, v. 22, p. S1039–S1042, 1998.

MOURET, S.; GROSSMANN, I. E.; PESTIAUX, P. A novel priority-slot based continuous-time formulation for crude-oil scheduling problems. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 48, n. 18, p. 8515–8528, 2009.

MOURET, S.; GROSSMANN, I. E.; PESTIAUX, P. Time representations and mathematical models for process scheduling problems. *Computers & Chemical Engineering*, v. 35, n. 6, p. 1038–1063, 2011.

NÁPOLES-RIVERA, F. et al. Global optimization of mass and property integration networks with in-plant property interceptors. *Chemical Engineering Science*, Elsevier, v. 65, n. 15, p. 4363–4377, 2010.

QUESADA, I.; GROSSMANN, I. E. Global optimization of bilinear process networks with multicomponent flows. *Computers & Chemical Engineering*, Elsevier, v. 19, n. 12, p. 1219–1242, 1995.

RAMAN, R.; GROSSMANN, I. E. Modelling and computational techniques for logic based integer programming. *Computers & Chemical Engineering*, Elsevier, v. 18, n. 7, p. 563–578, 1994.

ROCHA, R.; GROSSMANN, I. E.; ARAGÃO, M. V. P. de. Petroleum allocation at Petrobras: Mathematical model and a solution algorithm. *Computers & Chemical Engineering*, Elsevier, v. 33, n. 12, p. 2123–2133, 2009.

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

RONEN, D. Marine inventory routing: Shipments planning. *Journal of the Operational Research Society*, Taylor & Francis, v. 53, n. 1, p. 108–114, 2002.

SAHEBI, H.; NICKEL, S.; ASHAYERI, J. Strategic and tactical mathematical programming models within the crude oil supply chain context-—A review. *Computers & Chemical Engineering*, Elsevier, v. 68, p. 56–77, 2014.

SAHINIDIS, N. V. BARON: A general purpose global optimization software package. *Journal of Global Optimization*, Springer, v. 8, n. 2, p. 201–205, 1996.

SIMCHI-LEVI, D. et al. *Designing and managing the supply chain: concepts, strategies and case studies*. [S.l.]: Tata McGraw-Hill Education, 2008.

TELES, J. P.; CASTRO, P. M.; MATOS, H. A. Global optimization of water networks design using multiparametric disaggregation. *Computers & Chemical Engineering*, Elsevier, v. 40, p. 132–147, 2012.

TELES, J. P.; CASTRO, P. M.; MATOS, H. A. Multi-parametric disaggregation technique for global optimization of polynomial programming problems. *Journal of Global Optimization*, Springer, v. 55, n. 2, p. 227–251, 2013.

WICAKSONO, D. S.; KARIMI, I. A. Piecewise MILP under-and overestimators for global optimization of bilinear programs. *AIChE Journal*, Wiley Online Library, v. 54, n. 4, p. 991–1008, 2008.

YADAV, S.; SHAIK, M. A. Short-term scheduling of refinery crude oil operations. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 51, n. 27, p. 9287–9299, 2012.