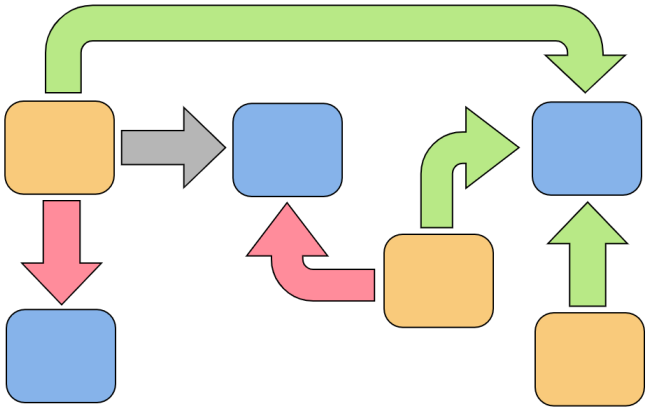
Market Equilibrium Model
for
LNG Shipments



Author
Justus van Kooi

Daily Supervisor
Dr. F. Beckebanze

Date
November 10, 2023

Supervisor, First Reader
Prof. Dr. C. W. Oosterlee

Second Reader
Prof. Dr. T. van Leeuwen

A thesis submitted to obtain the degree of Master of Science in Mathematical Sciences
at Utrecht University,
to be publicly defended on Friday November 17$^{\text{th}}$, 2023 at 13:00.

Mathematical Institute, Faculty of Science,
University Utrecht, The Netherlands.

Utrecht University

Abstract

We developed a market equilibrium model based on a constrained quadratic optimisation problem that maximizes the profit margins of liquefied natural gas (LNG) shipping companies. In 2022, the European LNG imports and prices have increased sharply due to the reduced pipeline gas from Russia. The market equilibrium model is calibrated to realistic trade volumes and prices for the most influential regions in the LNG market. We calibrate the market equilibrium model by solving a bilevel optimisation problem, to obtain ask and bid parameters, which cannot be collected directly from market data. We present a scenario analysis, in which we show that European prices would have increased by 5% if China had not been in lockdown during 2022. To illustrate the usability and limitations of our model, we also investigate the effects of the Panama Canal drought on trade flows and the impact of Australian port labour strikes on Asian imports based on the market data from August 2023.

Acknowledgements

I want to express my gratitude for the opportunity to complete this thesis during my internship at RaboResearch. I am very thankful for all the efforts of my daily supervisor Felix Beckebanze. He assisted in planning the project, gave insightful and detailed feedback in several revisions and motivated me when I felt challenged.

I want to thank my supervisor Kees Oosterlee for the many insightful comments and feedback during the thesis project. Kees has been a great supervisor and tutor to me. He has guided me in my decision making process during my master's programme. I want to express my gratitude for his guidance and insights.

I would like to thank my colleagues at RaboResearch who have welcomed me into the team. I'm thankful for the many opportunities granted to learn about global markets, the LNG market in particular and to learn about the skills necessary to work at a research department together with stakeholders. I want to especially thank Jeffrey Powell for his advising comments on this thesis and Joe DeLaura for providing his insights in the liquefied natural gas market.

Lastly, I want to thank my friends and family. Especially my parents, who have always supported me throughout my education.

Contents

List of Figures

List of Tables

List of Symbols

Overview of Variables in Market Equilibrium Model and Calibration

Description	Symbol
Set of regions	R
Number of regions	N
Subset of export regions	S
Number of export regions	M
Trade flows matrix	$X = (x_{ij})_{i \in S, j \in R}$
Trade flows vector between import and export regions	$\mathbf{x} = (\mathrm{x}_k)_{0 \leq k < M(N-M)}$
Panama Canal usage	$(d_{ij}^{(p)})_{i \in S, j \in R \setminus S}$
Suez Canal usage	$(d_{ij}^{(s)})_{i \in S, j \in R \setminus S}$
Cost of Panama Canal passage	$\pi^{(p)}$
Cost of Suez Canal passage	$\pi^{(s)}$
Travel distances	$T = (T_{ij})_{i \in S, j \in R \setminus S}$
Cost of transport per distance per unit of LNG	c
Transportations costs per unit of LNG	$(\tau_{ij})_{i \in S, j \in R \setminus S}$
Time horizon	H
Time-scales	$(\theta_{ij})_{i \in S, j \in R \setminus S}$
Passage time Panama Canal	$(t_{ij}^{(p)})_{i \in S, j \in R \setminus S}$
Passage time Suez Canal	$(t_{ij}^{(s)})_{i \in S, j \in R \setminus S}$
Average speed of ship	v_{avg}
Production limits	$\boldsymbol{L} = (L_i)_{i \in S}$
Port Capacities	$\boldsymbol{C} = (C_j)_{j \in R}$
Global shipping capacity	κ
Ask price slopes	$\boldsymbol{\eta}^{\mathrm{A}} = (\eta_i^{\mathrm{A}})_{i \in S}$
Ask price levels	$\boldsymbol{a} = (a_i)_{i \in S}$
Bid price slopes	$\boldsymbol{\eta}^{\mathrm{B}} = (\eta_j^{\mathrm{B}})_{j \in R}$
Bid price levels	$\boldsymbol{b} = (b_i)_{i \in S}$
Ask and bid parameters	$\mathbf{y} = (\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b})$
Matrix maximisation objective function	P
Vector maximisation objective function	$\mathbf{c}$
Matrix minimisation objective function	Q
Vector minimisation objective function	$\mathbf{q}$
Matrix linear constraints	D
Vector linear constraints	$\mathbf{v}$
Matrix and vectors depended on ask and bid parameters	$P_{\mathbf{y}}, \mathbf{c}_{\mathbf{y}}, \mathbf{v}_{\mathbf{y}}$
Identity matrix	$\mathbb{I}$
"All ones matrix"	$\mathcal{J}$
Feasible Region	$\mathcal{F}$

Overview of Functions in Market Equilibrium Model and Calibration

Description	Symbol
Imports in region $j \in R$	$I_j(X), I_j(\mathbf{x})$
Exports in region $i \in S$	$E_i(X), E_i(\mathbf{x})$
Total units bought in region $j \in R$	$m_j(X), m_j(\mathbf{x})$
Total units sold in region $i \in S$	$n_i(X), n_i(\mathbf{x})$
Bid price of regions $j \in R$	$\beta_j(m_j(X)), \beta_j(m_j(\mathbf{x}))$
Ask price of regions $i \in S$	$\alpha_i(n_i(X)), \alpha_i(n_i(\mathbf{x}))$
(Lower level) objective function of optimisation problem	$\varphi(X), \varphi(\mathbf{x})$
(Lower level) inequality constraint of optimisation problem	$\boldsymbol{\psi}(\mathbf{x})$
(Lower level) equality constraint of optimisation problem	$\boldsymbol{\xi}(\mathbf{x})$
Lagrange function of optimisation problem	$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$
Diagonal matrix formed from vector $\boldsymbol{\lambda}$	$\mathrm{Diag}(\boldsymbol{\lambda})$
Upper level objective function of MPEC and EPEC	$f(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu})$
Upper level inequality constraint of MPEC and EPEC	$\boldsymbol{g}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu})$
Upper level equality constraint of MPEC and EPEC	$\boldsymbol{h}(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu})$

List of Abbreviations

Abbriviation	Description
AU	Australia
CP	Complementarity problem
EC	Evolutionary Computing
EP	Equilibrium problem
EPEC	Equilibrium problem with equilibrium constraints
EU	Europe
GHz	Gigahertz
GLS	Genetic local search
HH	Henry Hub
ILS	Iterative local search
IPOPT	Interior point OPTimiser
IPOPT–C	Interior point OPTimiser for complementarity constraints
JKM	Japan/Korea Marker
KKT	Karush-Kuhn-Tucker
LNG	Liquefied natural gas
ME	The Middle East
MHz	Megahertz
MPEC	Mathematical problem with equilibrium constraints
MPCC	Mathematical problem with complementarity constraints
MSLS	Multi-start local search
MWh	Megawatt hour
NEA	North East Asia
NG	Natural gas
NP-hard	Non-deterministic polynomial-time hardness
NA	North America
NEA	North East Asia
OP	Optimisation problem
SA	South America
TTF	Title Transfer Facility

Chapter 1

Introduction

1.1 Problem Formulation

The natural gas market has changed significantly since the beginning of the Russo-Ukrainian war. In general – and especially in Europe – the reduced supply of Russian natural gas has lead to high energy prices, followed by rampant inflation. Amidst this crisis, Europe has also tried to shift energy composition, from climate-harming fossil fuels like oil and coal, to renewable energy sources. An important transitional energy source is natural gas (NG) because it is a relatively clean fossil fuel and can be used to generate electricity on demand. As such, NG can be used in gas turbines to generate electricity when electricity demand exceeds green energy production.

Since the start of the war, the Russian gas export to Europe have fallen by 80% [70]. As a consequence, European prices sharply increased. Europe sharply increased the import of NG from overseas in the form of *liquefied natural gas* (LNG). The liquefication, shipping and regasification necessary to transport LNG increase costs. In addition LNG could not fully supplement the shortfall of Russian gas, causing uncertainty whether the NG storage volumes could be filled sufficiently before the winter 2022-2023. This led to record high NG prices on the spot market during the summer of 2022 [49]. One of the major reasons why LNG could not fully compensate the reduced pipelined NG was the availability of LNG ships - about 700 world-wide [66]. This allowed the LNG shipping companies to sell their shipping capacity at a premium. If more LNG ships had been available, the profit margins of the LNG shipping companies would have been lower. In this thesis, we build a global LNG trade model in which the LNG shipping companies are treated as deal-makers between exporting and importing regions. To our knowledge, we are the first to build such a market model with a cap on the global LNG shipping capacity. The market power of the LNG shipping companies will be fundamental to our market model.

1.1.1 Changing Energy Markets

Energy comes in many forms. Energy-dense physical commodities that power our world include oil, coal, (L)NG, biofuel, and hydrogen. These commodities can be used as fuels for combustion engines or to power electricity plants. In addition, green energy sources such as hydropower, solar and wind increasingly contribute to the electricity energy mix. Energy commodities are traded through long-term contracts, on the short-term spot markets and on the futures market, where prices for future delivery amounts and quality are determined.

The environmental goals of Europe call for a sharp reduction in CO_2 emission [8]. The usage of fossil fuels needs to be reduced to meet these goals. To secure future European energy demand, alternative energy sources will need to be put in place. Currently, there is not enough green electricity production combined with battery capacity to meet electricity demand at all times. Fossil fuel electricity generators are used to close the gap between supplied green electricity and demand. Natural gas is the preferred fossil fuel for this purpose, as it has the least environmental impact among fossil fuels [3]. It is therefore likely that (L)NG will remain an important energy commodity for at least the next decade [1]. In addition, (L)NG is used as fuel for high heat industrial processes, as major resource in the production of fertilizer and as fuel for transportation. Some of

these high heat applications depend on alternatives such as hydrogen to replace the necessity of NG. Commodities, such as LNG, will remain relevant until this transition is finalized.

The majority of NG is transported via pipelines from sourcing facilities to the consumer markets. The remaining transportation is covered by shipments for which the NG needs to be liquefied such that its volume is reduced by a factor 600. Pipeline NG is significantly cheaper than LNG because the transport, liquefaction and re-gasification are relatively expensive. As a result, importing regions typically try to match their NG demand through pipelined NG, and only rely on LNG if the demand exceeds the available pipelined NG. Thus, the LNG market is more sensitive to price changes than the pipeline NG market. The reduction of Russian NG exports to Europe means that the reduced supply of NG needs to be replaced with the more expensive LNG to meet Europe's (winter 2022-2023) demand.

The United States of America is a major producer of NG, historically with little export to other continents. However, the rising global demand for LNG (mainly from Europe) has created an incentive to transport the NG across the Atlantic. While the NG price in North America will remain significantly lower than in LNG importing regions, the increased LNG import will expose the local North-American market price to the volatility of the global LNG market, as LNG producers can now serve both the local demand as well as demand in other continents. The significant increase in LNG demand in Europe as a result of the Russo-Ukrainian war has accelerated the opening of the North-American NG market.

1.1.2 Research Objective

Our objective is to model the interaction between N number of regions partitioned in M number of export regions and $N - M$ number of import regions. We simplify the LNG market to export regions, import regions and LNG shipping companies to find a market equilibrium. We investigate if reducing the LNG market model to these components is sufficient to capture the dynamics of the trade flows and prices of the LNG market in times of high shipment demand. Our market equilibrium model maximizes the objective function $\varphi_{\mathbf{y}}(\mathbf{x})$ representing the profit of the LNG shipping companies, under consideration of their operational costs. The market equilibrium model can be defined as a quadratic optimisation problem, given by

$$\begin{aligned} \max_{\mathbf{x}} \ & \varphi_{\mathbf{y}}(\mathbf{x}) = \mathbf{x}^{\intercal} P_{\mathbf{y}} \mathbf{x} + \mathbf{c}_{\mathbf{y}}^{\intercal} \mathbf{x}, \\ & \text{subject to} \\ & D\mathbf{x} \leq \mathbf{v}_{\mathbf{y}}, \end{aligned} \tag{1.1}$$

where vector $\mathbf{x} \in \mathbb{R}^{M(N-M)}$ represents the time-averaged trade flows between the export and import regions, i.e., vector component x_k for $k = 0, ..., M(N-M)-1$ is expressed as a unit of LNG per time unit, e.g., MWh per day. As said, the objective function $\varphi_{\mathbf{y}}$ represents the profit of the LNG shipping companies, which is the difference between the revenue and the transportation costs. The revenue depends on the prices in the regions. The prices depend on linear ask and bid functions where the parameter vector $\mathbf{y} \in \mathbb{R}^{2M+2N}$ quantifies how the price responds to supply and demand. All components mentioned above, which describe the price dynamics are represented by the coefficient matrix $P_{\mathbf{y}} \in \mathbb{R}^{M(N-M)\times M(N-M)}$ and vector $\mathbf{c}_{\mathbf{y}} \in \mathbb{R}^{M(N-M)}$.

We incorporate operational constraints, such as port capacity and production limits, in the constraint $\mathbf{v}_{\mathbf{y}}$. Unique to our model is the inclusion of a global LNG transportation capacity constraint in $\mathbf{v}_{\mathbf{y}}$, which mimics the LNG market conditions from 2022 where global shipping capacity is close to the maximum capacity. We denote the coefficients by $P_{\mathbf{y}}, \mathbf{c}_{\mathbf{y}}, \mathbf{v}_{\mathbf{y}}$ because they depend on the ask and bid parameters $\mathbf{y}$. Currently, there are no market equilibrium models to our knowledge which focus on market scenarios in which the LNG shipping companies hold significant market power, such that only their profits form the objective function of a market equilibrium problem. By assuming that the LNG shipping companies have significant market power and combining their profits, the market can be interpreted as monopolistic, i.e., only one firm (representing all LNG shipping companies) holds market power. Finding the global solution of (1.1) leads to a market equilibrium based on supply and demand of LNG. The optimisation of quadratic optimisation problem is efficient when $\varphi_{\mathbf{y}}$ is convex.

Inconveniently, there is no reliable source for parameters $\mathbf{y}$. However, we do have data for LNG trade flows $\mathbf{x}$ and prices for the regions. To use the model for realistic scenarios, we therefore need to calibrate to acquire the parameters $\mathbf{y}$. Our calibration process is performed by optimizing a bilevel optimisation problem which separates the calibration process into two task formulated as an upper level and lower level optimisation problem. The lower level problem describes the market equilibrium problem (1.1). The upper level problem describes the minimisation between the market equilibrium output and the realistic market data. The bilevel optimisation problem for the calibration is given by

$$\min_{\mathbf{x},\mathbf{y}} f(\mathbf{x},\mathbf{y}), \tag{1.2a}$$

$$\text{subject to}$$

$$\boldsymbol{g}(\mathbf{x},\mathbf{y}) \leq \mathbf{0}, \tag{1.2b}$$

$$\max_{\mathbf{x}} \varphi_{\mathbf{y}}(\mathbf{x}),$$

$$\text{subject to} \tag{1.2c}$$

$$D\mathbf{x} \leq \mathbf{v}_{\mathbf{y}},$$

where distance function $f : \mathbb{R}^{M(N-M)} \times \mathbb{R}^{2M+2N} \to \mathbb{R}$, in (1.2a), quantifies the accuracy of the calibration with respect to the realistic market data. Vector function $\boldsymbol{g} : \mathbb{R}^{M(N-M)} \times \mathbb{R}^{2M+2N} \to \mathbb{R}^{L}$, in (1.2b), describes the L number of constraints present in the calibration process. We use $\boldsymbol{g}$ to put simple lower and upper bounds on the parameters $\mathbf{v}_{\mathbf{y}}$. The lower level optimisation problem, in (1.2c), is the same quadratic optimisation problem as (1.1). Solving a bilevel optimisation problem (1.2) is non-trivial, because even simple forms, e.g. both levels are linear optimisation problem, are non-convex [20]. As far as we know, calibration of the ask and bid parameters by solving a bilevel optimisation problem is not used in other natural gas market models. In this thesis, we explore methods of simplifying the bilevel optimisation under convexity assumptions of the lower level optimisation problem in (1.2). However, this simplified method still relies on an accurate choice of the initial solution. To improve the calibration process, we explore several meta-heuristics, such as multi-start local search (MSLS), iterative local search (ILS) and genetic local search (GLS).

In Figure 1.1, we illustrate how the *baseline step* and the *perturbation step*, which correspond to solving (1.2) and (1.1), respectively. In the baseline step, a market equilibrium model is calibrated by altering the inverse supply and demand functions of all regions until a market equilibrium is found which is as close as possible to a realistic scenario. Next, in the perturbation step, one (or several) parameter(s) can be perturbed to model a certain scenario. The objective is to compare the trade flows and prices between the baseline scenario and the perturbed scenario. We study the effects of increased Asian LNG demands on European prices in 2022, the impact of the Panama droughts on trade flows and the impact of Australian port labour strikes on Asian imports in 2023.

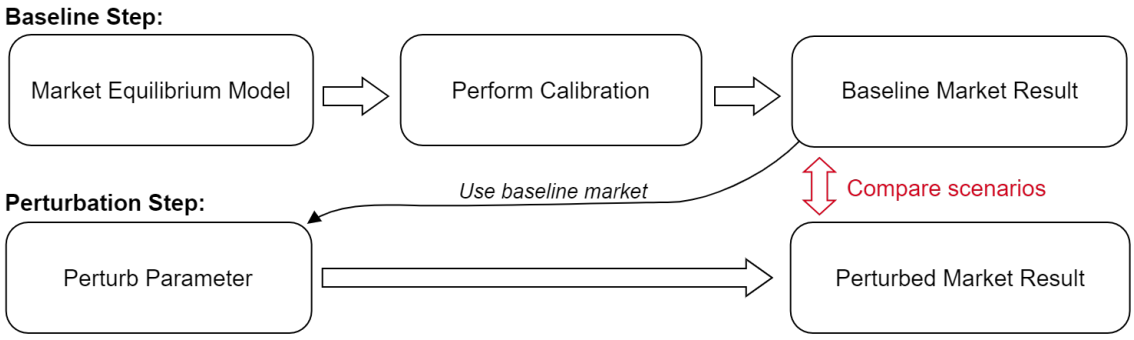


Figure 1.1: General model framework for scenario analysis.

1.1.3 Thesis Structure

For the remainder of this chapter, we introduce the market agents in the NG market, followed by an overview of optimisation models used to capture the interaction between market agents and a

literature overview. In Chapter 2, we define the market equilibrium model based on a quadratic optimisation (1.1) of the LNG shipping companies. In Chapter 3, we define the bilevel optimisation (1.2) which is used to calibrate the market equilibrium model based on realistic data. Additionally, we perform an accuracy analysis of the calibration w.r.t. the number of regions in the model. In Chapter 4, we introduce three studies which show realistic scenario analysis based on events in 2022 and 2023. Lastly, in Chapter 5, we provide a conclusion, discussion and outlook on future work.

1.2 Models of the Natural Gas Market

In the upcoming sections, we introduce the mathematical methods used to model the interactions of the NG market. In Section 1.2.1, we cover the roles of the agents in the NG market. In addition, we discuss the different competition forms used to model the market interaction, based on [30]. In Section 1.2.2, we cover the mathematical formulation of optimisation methods used to model the interaction of all NG market agents, in increasing order of complexity. In Section 1.2.3, a literature survey is given, where we reference the scope of the study, the mathematical model used, and the noteworthy results.

1.2.1 Natural Gas Market Description

The supply chain of the NG market can be explained using agents. Agents are an entity which fulfil a specific role inside the NG market. In Table 1.1, we give an overview of all the agents which are used to model the NG market, together with their inclusion in NG market modelling literature, and a short description. A summary of the NG supply chain can be described as follows: NG is pumped up by producers from underground reserves, which is then sold to traders. The traders sell NG to marketeers who distribute the NG to several agents: consumers, storage operators, and liquefiers. Consumers are the last stage of the supply chain, and are usually divided into residential, industrial, commercial and power generation sectors. Storage operators receive NG to store for later use. Liquefiers cool down the NG to create LNG. LNG shipping companies transport the LNG over sea. At their destination, the LNG shipping companies offload their LNG to the regasifier who heats the LNG to obtain NG to sell to the marketeer in the destination region. NG is also transported by pipeline operators between regions. The expansion of pipeline infrastructure is managed by investment companies.

Focusing on the LNG market, there are a number of characteristics. In the global LNG market, 2/3 of the LNG is traded on the long-term (between 10–25 years [2]) contract market, while 1/3 is traded via the spot market. LNG shipping companies act as deal makers between importing and exporting regions. In the event that LNG demand suddenly increases, prices in the spot market will rise. High gas prices create profitable positions for LNG shipping companies who sell their transportation capacity to the highest bidder [4].

Now that the agents are defined, we illustrate how the supply and demand dynamics are characterized. Each agent i (from Table 1.1) can have a supply quantity $x_A^{(i)}(p)$, which is monotonically increasing in ask price p and demand quantity $x_B^{(i)}(p)$, which is monotonically decreasing in bid price p. Typical choices are linear or exponential functions. In addition, supply and demand functions can depend on more variables than just the price p, for example, the temperature, the season, the storage available. However, the exact parameters in the demand and supply functions which quantify the influence of each variable on the total demand (or supply) are usually unknown. Statistical methods, such as regression can be used to estimate the relation between independent variable(s), e.g. price, and the dependent variable, e.g. supply or demand. This is the field of econometrics, and lies outside the scope of this thesis. We will estimate the parameters for linear supply and demand functions through a calibration given in Chapter 3.

In the following, we assume the supply and demand functions depend on the price p only. A market equilibrium is defined by an equality between supply and demand. The supply and demand $x_A^{(i)}(p), x_B^{(j)}(p)$ between agents i and j, respectively, are set equal by

$$x_A^{(i)}(p) = x_B^{(j)}(p),$$

resulting in a price p for which the supply and demand are matched. This equilibrium is a fundamental economical concept, and will be present at each stage of the supply chain, e.g., between producers selling to traders, between traders selling to marketeers, etcetera. However, an equilibrium can also be achieved when inverse supply and demand functions are used, which are given by

$$\alpha^{(i)}(x) := x_A^{-1}(p), \tag{1.3a}$$

$$\beta^{(j)}(x) := x_B^{-1}(p), \tag{1.3b}$$

and are used for the equation

$$\alpha^{(i)}(x) = \beta^{(j)}(x). \tag{1.4}$$

The equality (1.4) leads to a quantity x for which the supply and demand sides of the market agree on a price, which we call the clearing price. A high demand leads to a relatively high clearing price, while high supply leads to a relatively low clearing price. Solving equality (1.4) at each step in the supply chain leads to a market equilibrium.

To model the interactions of the NG market, a type of competitiveness is assumed. The type describes the difference in market power between the agents in Table 1.1. This market power translates to how much knowledge an agent has about: its own (inverse) supply and (inverse) demand functions, and how much knowledge it has about (inverse) supply and (inverse) demand functions from other agents. Naturally, a powerful agent has more knowledge than a weaker agent. Agents can also be modelled to have the ability to anticipate a market equilibrium, and subsequently, alter their strategy. The types of competition used to model the NG market are given by perfect, Cournot, Bertrand, monopolistic and Stackelberg competitions.

A perfect competitive market, as formalized in [13], assumes that a market can perfectly equate the quantity demanded to the quantity supplied. A perfect market, briefly, is on in which there are infinite number of buyers and sellers, so no one agent can affect prives, and all agents in this system are price-takers, i.e., they have no influence on the price of a commodity through the use of inverse supply and demand functions. Transportation agents make very limited profits because they cannot influence the supply and demand functions. All competition forms which do not distribute the market power equally are considered imperfect competitions.

Cournot competition is a type of oligopolistic competition which assumes that a handful of firms hold market power. This makes these firms price-makers, i.e., they have (almost) full control over the price of a commodity. Cournot agents differentiate based on the supply, instead of price. They have knowledge of the inverse demand functions, i.e., they know how their supply will effect the price response of the demand side.

Bertrand competition can be interpreted as another form of oligopolistic competition, where the powerful agents compete based on price instead of supply. Bertrand firms have knowledge on the demand function, i.e., they know how their price will effect the demand. As we show in Section 1.2.3, Bertrand competition is not used in studies covering the NG market.

Monopolistic competition is an extreme form of competition, where we assume that only one firm holds all market power, i.e, the absence of competition. This situation gives an agent full control of the price. Monopolistic modelling of the NG market is uncommon, because there are no individual companies which determine the whole market (or part of the supply chain). Note that even a monopolistic (L)NG market will have a market equilibrium, because the supply and demand sides still need to equate while the middle-men, e.g., the LNG shipping companies, can be modelled as one firm maximizing their profit on every trade route.

Stackelberg competition, introduced in [72], is a hierarchical form of competition where, in constrast to Cournot competition, a firm can also influences the decisions of other firms. Agents who are assumed to be Stackelberg leaders are able to influence their supply, and anticipate the market reaction. This means that Stackelberg leaders have knowledge on their own inverse supply and demand, as well as the supply and demand functions of other agents. The Stackelberg followers are the weaker agent in the system, because they cannot anticipate the market. However, the followers

can still be part of any of the competition forms mentioned previously.

Realistically, any agent lies somewhere between the extremes of price-taker and price-maker. Other forms of competition, such as cartel competition exist. However, this competition form is (generally) not found in the mathematical descriptions capturing the (L)NG market models, and thus will not be discussed here. For more information about these forms of competition, we refer to [30].

Agent type:	Occurrence in paper:	Description:
Producer (Region)[1]	[55, 37, 27, 16, 31, 44, 54, 42, 53, 24, 33, 25]	Produces a quantity of NG until a production limit. Agent can represent a region.
Consumer	[37, 27, 16, 31, 44, 54, 42, 53, 24, 33, 25]	Consumes NG. Sector divided into: residential, industrial, commercial and power generation.
Consumer Region[1]		Region containing one consumer agent representing the total consumption i.e. all sectors.
Marketers	[31, 24, 25]	Distributes the NG within a region to different agents within the region.
Traders	[37, 16, 44, 27, 24]	Represents a subsidiary company of a producing agent focused on exporting NG.
Pipeline operators	[37, 27, 16, 54, 31, 42, 53, 24, 25]	Focus only on the operation of NG pipeline transport between regions.
Storage operators	[27, 31, 53, 24, 25]	Stores (L)NG within a region.
Liquefier	[27, 42, 54, 25]	Liquefies NG to LNG for transportation on LNG ships.
Regasifier	[27, 54, 25]	Regasifies LNG to NG for transportation through pipelines or for storage.
LNG shipping company[1]	[27, 25]	Transports LNG between regions via ships.
Investment company	[24, 33, 53]	Invests in the expansion of the pipeline infrastructure of a region.
Peak gas operators	[31]	Provides regions with gas in times of high demand.

Table 1.1: Overview of agents used in the references.

[1]This agent is incorporated in our model

1.2.2 Optimisation Methods to model NG Markets

Various types of optimisation problems are used to model the interaction of the NG market agents. All of the models use a clearing mechanism, which equates supply to demand, as given in (1.4). Optimisation methods are used to find a market equilibrium, and are given by a single optimisation problem (OP) or multiple optimisation problems which are optimised simultaneously, called equilibrium problem (EP). Maximizing the profit of a subset of agents requires anticipation of the outcoming market equilibrium, which can be formulated in hierarchical optimisation problems. Here, a mathematical program with equilibrium constraints (MPEC) is used, where the upper level optimisation describes the optimisation of a single type of agent, and the market equilibrium is given by the lower level optimisation. For multiple agents who optimise their behaviour in a market equilibrium, an equilibrium problem with equilibrium constraints (EPEC) is used, where the hierarchy is similar as in MPECs. Next, we will give an overview of each mathematical formulation which is used to analyse the gas market. The overview is a summary based on [30, 17].

Optimisation Problems (OP) The first set of models, OPs, typically are characterised by a minimization of the costs of production while supplying the demand side of the market, or maximisation of total profits based on (inverse) supply and demand. These models, and especially the linear models, can deal with many variables because of the relatively low complexity. OPs are used to model markets with perfect, oligopolistic or monopolistic competition. Any market power is then incorporated via parameters acting as weights in the objective function. In addition, the oligopolistic (or monopolistic) agents have their profit depend on inverse supply and demand functions, while weaker agents do not. The general formulation of OPs is as follows:

$$\begin{aligned} &\max_{\mathbf{x}} \ \varphi(\mathbf{x}), \\ &\quad \text{subject to} \\ &\quad \boldsymbol{\psi}(\mathbf{x}) \leq \mathbf{0}, \\ &\quad \boldsymbol{\xi}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{1.5}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the optimised variable, usually indicating trade flows. The function φ is the objective function, which can be used to maximize the overall profits, or to minimize cost. The vector function $\boldsymbol{\psi}$ represents the inequality constraints. The vector function $\boldsymbol{\xi}$ represents the equality constraints, which include the market clearing equation (1.4). If the objective function φ is concave and the feasible region $\mathcal{F} = \{\mathbf{x} \ : \ \boldsymbol{\psi}(\mathbf{x}) \leq \mathbf{0}, \boldsymbol{\xi}(\mathbf{x}) = \mathbf{0}\}$ is convex, then a local solution will also be the global solution of (1.5). A function φ is concave when it is twice differentiable, and the Hessian of φ is negative semi-definite, i.e., $\mathbf{x}^\intercal P \mathbf{x} \leq 0$ for all $\mathbf{x} \in \mathbb{R}^n$. The feasible region $\mathcal{F}$ is convex if a line segment between two points in $\mathcal{F}$ lies in $\mathcal{F}$, i.e., for $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \mathcal{F}$ it holds that

$$t\mathbf{x}^{(1)} + (1-t)\mathbf{x}^{(2)} \in \mathcal{F},$$

for $t \in [0, 1]$. Note that typically, the literature (such as [17, 58]) describe OPs which minimize the objective function $\Phi := -\varphi$, which needs to be convex to guarantee a global optimum. Either way, for a maximization problem with concave objective function, or a minimisation function with convex optimisation function, we will refer to both as convex OPs. A convex OP has an optimal solution when it satisfies meaningful Karush-Kuhn-Tucker (KKT) conditions. The KKT conditions of (1.5) can be expressed via a complementarity problem (CP), as follows:

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0}, \tag{1.6a}$$
$$\boldsymbol{\xi}(\mathbf{x}) = \mathbf{0}, \tag{1.6b}$$
$$\mathbf{0} \leq \boldsymbol{\lambda} \perp \boldsymbol{\psi}(\mathbf{x}) \leq \mathbf{0}, \tag{1.6c}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^k$ and $\boldsymbol{\mu} \in \mathbb{R}^l$ represent the Lagrange multiplier vectors and the Lagrangian function, $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$, is given by

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) := \varphi(\mathbf{x}) + \boldsymbol{\lambda}^\intercal \boldsymbol{\psi}(\mathbf{x}) + \boldsymbol{\mu}^\intercal \boldsymbol{\xi}(\mathbf{x}).$$

The complementarity given in (1.6c) is defined by the following (in)equalities

$$\boldsymbol{\psi}(\mathbf{x}) \leq \mathbf{0}, \tag{1.7a}$$
$$\text{Diag}(\boldsymbol{\lambda})\boldsymbol{\psi}(\mathbf{x}) = \mathbf{0}, \tag{1.7b}$$
$$\boldsymbol{\lambda} \geq \mathbf{0}, \tag{1.7c}$$

where $\text{Diag}(\boldsymbol{\lambda}) \in \mathbb{R}^{k\times k}$ defines a diagonal matrix of $\boldsymbol{\lambda}$ on the diagonal, as seen in [58]. The complementarity symbol $\perp$ is a standard notation for a complementarity problem. Constraint (1.6a) is the gradient of the Lagrangian $\mathcal{L}$, which needs to be zero, such that the optimisation problem (1.5) returns an optimal solution $\mathbf{x}$. The constraints (1.6b) and (1.6c) enforce the equality and inequality constraint of optimisation problem (1.5). Constraint (1.6c) forms a complementarity problem consisting of two orthogonal vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\psi}(\mathbf{x})$. Finding a solution to the system of equations (1.6) is equivalent to finding the solution to the optimisation problem (1.5) [17].

Equilibrium Model (EP) The second set of models, EPs, form a collection of OPs used to optimise multiple optimisation problems simultaneously. EPs are used to model the same competition forms as OPs, but the difference in market power is not modelled via weights, but separate OPs are used to optimise for each agent (type) individually. Consider problem i, $i = 1, ..., m$ given by

$$\begin{aligned} &\max_{\mathbf{x}^{(i)}} \ \varphi^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}), \\ &\quad \text{subject to} \\ &\quad \boldsymbol{\psi}^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}) \leq \mathbf{0}, \\ &\quad \boldsymbol{\xi}^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}) = \mathbf{0}, \end{aligned} \tag{1.8}$$

where $\mathbf{x}^{(i)} \in \mathbb{R}^{n^{(i)}}$ is the optimised variable for problem i and $n^{(i)} \in \mathbb{N}$. The functions $\varphi^{(i)}, \boldsymbol{\psi}^{(i)}$ and $\boldsymbol{\xi}^{(i)}$ represent the objective function, inequality constraints and equality constraints of problem i, respectively, similar to the OP in (1.5). Note that each function $\varphi^{(i)}, \boldsymbol{\psi}^{(i)}, \boldsymbol{\xi}^{(i)}$ considers the variables of all problems $i = 1, ..., m$. This connects the different problems, which allows for the definition of an equilibrium. For each problem, $i = 1, ..., m$, it is possible to define the KKT conditions as given by (1.6), with functions $\varphi^{(i)}, \boldsymbol{\psi}^{(i)}, \boldsymbol{\xi}^{(i)}$ for problem i. Rewriting to KKT-conditions introduces the need of $\lambda^{(i)}, \mu^{(i)}$ for problem i. It should be noted that some EPs – through the KKT conditions – can be written as OPs.

Mathematical Program with Equilibrium Constraints (MPEC) The third set of models, MPECs, form an optimisation problem where the constraints consist of an EP, i.e., the constraints are made up by connected optimisation problems. MPECs include the hierarchical structure between the upper level optimisation problem and the lower level optimisation (or equilibrium) problem. MPECs are used to model two-stage markets, which are based on Stackelberg competition with one single Stackelberg leader, i.e., the upper level optimisation is an OP. For example, the OP maximizes the profit margins of a producer agent, which is modelled to be the Stackelberg leader. The upper and lower level problems can model perfect, oligopolistic or monopolistic competitions, as they are an OP and EP, respectively. The levels together model the Stackelberg competition. The mathematical formulation of an MPEC, with m optimisation problems as constraints, is given by

$$\max_{\substack{\mathbf{y},\mathbf{x}^{(1)},...,\mathbf{x}^{(m)} \\ \boldsymbol{\lambda}^{(1)},...,\boldsymbol{\lambda}^{(m)},\boldsymbol{\mu}^{(1)},...,\boldsymbol{\mu}^{(m)}}} \quad f(\mathbf{y}, \mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}, \boldsymbol{\lambda}^{(1)}, ..., \boldsymbol{\lambda}^{(m)}, \boldsymbol{\mu}^{(1)}, ..., \boldsymbol{\mu}^{(m)}), \tag{1.9a}$$

subject to

$$g(\mathbf{y}, \mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}, \boldsymbol{\lambda}^{(1)}, ..., \boldsymbol{\lambda}^{(m)}, \boldsymbol{\mu}^{(1)}, ..., \boldsymbol{\mu}^{(m)}) \leq \mathbf{0}, \tag{1.9b}$$

$$h(\mathbf{y}, \mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}, \boldsymbol{\lambda}^{(1)}, ..., \boldsymbol{\lambda}^{(m)}, \boldsymbol{\mu}^{(1)}, ..., \boldsymbol{\mu}^{(m)}) = \mathbf{0}, \tag{1.9c}$$

For each $i = 1, ..., m$**:**

$$\begin{aligned} &\max_{\mathbf{x}^{(i)}} \ \varphi^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}), \\ &\quad \text{subject to} \\ &\quad \boldsymbol{\psi}^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}) \leq \mathbf{0}, \\ &\quad \boldsymbol{\xi}^{(i)}(\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m)}) = \mathbf{0}, \end{aligned} \tag{1.9d}$$

where $\mathbf{y} \in \mathbb{R}^n$ is the optimised (upper level) variable, which is optimised in the upper optimisation problem. Variables $\mathbf{x}^{(i)} \in \mathbb{R}^{n^{(i)}}$ for $i = 1, ..., m$ are optimised in the upper level and lower level

problem(s). Variables $\boldsymbol{\lambda}^{(i)}, \boldsymbol{\mu}^{(i)}$ represent the Lagrange multiplier vectors of the i-th lower level problem, which the upper level optimisation problem also optimises. The functions $f, \boldsymbol{g}, \boldsymbol{h}$ are the upper level objective function, inequality constraints and equality constraints, respectively, which describe the Stackelberg leader. The lower level optimisation problem(s) in (1.9d), are the same formulations as in (1.8). If the EPs in (1.9d) are convex, then (1.9d) can be replaced by the KKT-conditions. The simplest non-trivial MPEC is the bilevel optimisation problem, which has one ($m = 1$) optimisation problem as constraint. In Chapter 3, we define a bilevel optimisation to calibrate the market equilibrium model, which shows that MPECs are also applicable in other contexts. A bilevel optimisation with the convex lower level optimisation replaced by KKT-conditions is referred to as a mathematical program with complementarity constraint (MPCC), and is given by

$$
\begin{aligned}
&\max_{\mathbf{y},\mathbf{x}} f(\mathbf{y}, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}), && \text{(1.10a)}\\
&\quad \text{subject to}\\
&\quad \boldsymbol{g}(\mathbf{y}, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq \mathbf{0}, && \text{(1.10b)}\\
&\quad \boldsymbol{h}(\mathbf{y}, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0}, && \text{(1.10c)}\\
&\quad \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{0},\\
&\quad \boldsymbol{\xi}(\mathbf{x}) = \mathbf{0}, && \text{(1.10d)}\\
&\quad \mathbf{0} \leq \boldsymbol{\lambda} \perp \boldsymbol{\psi}(\mathbf{x}) \leq \mathbf{0},
\end{aligned}
$$

where the problem description can be simplified to the variables $\mathbf{y}, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}$ and the lower level optimisation in (1.9d) is replaced by (1.6) to obtain (1.10d). This leader-follower (upper-lower) hierarchy is present in several modelling problems, for example, military defence applications, electricity markets, highway network design, resource allocation, supply chain configurations, support vector machines, meta-learning, portfolio optimisation, and many more, described in [20, 65].

Equilibrium Problems with Equilibrium Constraints (EPEC) The last set of models, EPECs, are a collection of MPECs, similar to how EPs are a collection of OPs. As such, EPECs are also separated in an upper and lower level. Again, the hierarchical structure of EPECs is used to model Stackelberg competition, but now, the upper level optimisation is an EP. Here, we optimise for multiple Stackelberg leaders. We present the simplified formulation of EPECs, as in [30], although more generalized formulations exist. The mathematical formulation of an EPEC, made up of K MPECS, is given by

$$
\begin{aligned}
&\textbf{For each } j = 1, ..., K\textbf{:}\\
&\quad \max_{X^{(j)}} f^{(j)}(X^{(j)}),\\
&\quad \text{subject to}\\
&\quad g^{(j)}(X^{(j)}) \leq \mathbf{0},\\
&\quad h^{(j)}(X^{(j)}) = \mathbf{0},\\
&\quad \textbf{For each } i = 1, ..., m^{(j)}\textbf{:}\\
&\qquad \max_{\mathbf{x}^{(j,i)}} \varphi^{(j,i)}(\mathbf{x}^{(j,1)}, ..., \mathbf{x}^{(j,m^{(j)})}),\\
&\qquad \text{subject to}\\
&\qquad \boldsymbol{\psi}^{(j,i)}(\mathbf{x}^{(j,1)}, ..., \mathbf{x}^{(j,m^{(j)})}) \leq \mathbf{0},\\
&\qquad \boldsymbol{\xi}^{(j,i)}(\mathbf{x}^{(j,1)}, ..., \mathbf{x}^{(j,m^{(j)})}) = \mathbf{0},
\end{aligned}
\qquad \text{(1.11)}
$$

where $X^{(j)} := \left\{\mathbf{y}^{(j)}, \mathbf{x}^{(j,1)}, ..., \mathbf{x}^{(j,m)}, \boldsymbol{\lambda}^{(j,1)}, ..., \boldsymbol{\lambda}^{(k)}, \boldsymbol{\mu}^{(j,1)}, ...\boldsymbol{\mu}^{(j,l)}\right\}$ is the set of optimisation variables used in the i-th MPEC. It should be noted that the feasible regions of MPECs and EPECs are typically non-convex. This makes finding an optimal solution difficult. For a more extensive coverage of the mathematical structures mentioned above, we refer to [30].

1.2.3 Literature Review of Natural Gas Market Studies

In this section, we discuss multiple studies that have modelled the NG market. We provide an overview of economical and mathematical methods used. The focus of these studies is either a single (economically important) region such as North America [33], Europe [32] or globally, such as in [54, 24].

The theory of markets dates back to be end of the 19th century in Marshall [56]. The (nowadays) fundamental principle of Marshall is the concept of a market equilibrium found between a supply and demand side via utility functions. The majority of the market equilibrium models still relies on this fundamental principle. The studies covering the NG market can be categorized according to three types of methods. These methods may have similar objective goals, but generally have distinct applications. The models which are used for the NG market are:

- The traditional economic models;
- The (constrained) optimisation models;
- The agent-based systems.

First, we cover the traditional economic studies, because chronologically this type precedes the other models. The results regarding price elasticity and cost functions from these traditional studies usually are inserted in the latter models. The optimisation-based methods range on a complexity scale, as seen in Section 1.2.2, for which we cover the studies corresponding to each optimisation method. Lastly, a recent development including agent-based systems is touched upon.

Traditional Economical Model Most of the studies discussed in this section rely on methodologies originating from economics such as equilibrium modelling, time series analysis and net present value computation. Therefore, these models are labelled traditional economical models.

The study [55] looks into the security of supply in the Western European NG market using equilibrium modelling, i.e., equating simple supply and demand functions, with a focus on the probability of disruption. These models study whether quotas and tariffs can be used to guarantee the security of supply in Northwest Europe. They conclude that depending on the probability of disruption which is acceptable by policy makers either quotas or tariffs can be used to guarantee a secure supply, but this depends on the desired probability.

Another paper is [37] which introduced several functions, most notably the *Golombek production cost function* which is present in numerous follow up research studies (containing relevant optimisation models). This marginal production cost function, $g(x, C)$, is characterized by a combination of a linear part which describes the cost whenever the production capacity is not met, and a logarithmic part, which is most prominent close to the capacity quantity $C \in \mathbb{R}$ of a producer. The function is given by

$$g(x, C) = a + bx + \lambda \cdot \log\left(\frac{1-x}{C}\right),$$

where $x \in \mathbb{R}$ is the quantity of the product, $a, b \in \mathbb{R}$ are parameters determining the linear supply, $C \in \mathbb{R}$ is the production capacity and $\lambda \in \mathbb{R}$ is a scaling factor. The authors in [37] study the long-run impact of liberalizing the NG markets in Western Europe. The model starts with a baseline where there are limited producers able to sell and traders cannot exploit arbitrage possibilities. It studies three scenarios built on different assumptions which, in increasing order, would simulate an increasingly liberalizing market. The paper concludes that increasing the liberalization in Western markets will decrease the social welfare.

A study performed in [50] provides a global analysis of the crude oil and NG prices over an extensive period (1918-1999) in which the oil crisis in 1973 is seen as a significant disruption of the dynamics. Prices were stable before the shock, and became volatile afterwards. Later, the authors in [50] used the supply and demand elasticity to explain the price volatility after a shock and the market power of oil producers. They used statistical linear regression based on time-series data.

Optimisation Problems (OP) The study in [33] introduced the Gas Systems Analysis Model (GSAM) which was used to study the effects from Canadian carbon stabilization on North American gas markets. A temporal integrating linear program was used to approximate the non-linear demand, supply and transportation functions. Here, the objective function optimises the operating costs of the producer, consumer, and investment costs of the infrastructure. The other participants of the gas market do not have their utilities optimized in this paper.

The study [54] looked at the costs of the NG supply for Europe, the United States and Japan in a perfectly competing globalising market. This work is based on the MAGELAN model which was developed in [63]. The objective function in this model includes capital and operating costs for production and transport around the world. The costs are minimised, while subject to technical restrictions for the production and transport. The major inputs are demand developments, reserves, production costs per region, existing infrastructure and cost parameters.

The linear optimisation model TIGER developed in [53] performs an analysis of the impact of the Nord Stream pipeline on the European Gas Transmission System. The objective function is based on the total costs of the gas supply (production) and transportation while meeting demand in each region. The goal is to minimize the costs, under the constraint of storage limits, supply and demand equations, pipeline operation, LNG import storage and regasification capacity. In [21], the effects of crises on the Nabucco and South Stream pipelines are studied via the TIGER model. They concluded that the expansion would increase the security of supply for Europe. In later development, [52] used an enhanced version of the TIGER model, which was more precise with respect to the temporal granularity and geographic coverage, to study congestion in the European NG network. Lastly, the paper [22] analysed a highly granulated European NG infrastructure model using TIGER to study the gas flows and possible congestion in Western Europe.

However, there is critique on this relatively simple optimisation structure. In [30], the following is commented on the design of the TIGER model: "Since it is a linear program, it can handle many variables and constraints from a computational perspective. However, there is the overriding assumption that the market will be efficient so that cost-minimizing behaviour is appropriate to characterize it. Given recent strategic behaviour in European gas markets as mentioned above, other economic paradigms may also be appropriate depending on the questions being asked.". The assumption mentioned would also in the period of interest for our project (2022-2023) be undesired.

Equilibrium Model (EP) The study [42] used an EP to develop the Baker Institute World Gas Trade Model (BIWGTM), which is a dynamic spatial equilibrium model with an elastic demand based on economic theory. The model estimates the coefficients of several regression functions. BIWGTM is used in [43] to perform a study on several scenarios. Two main implications from their research are: Russia might have less ability to negatively affect the West European gas markets as previously was thought and the Middle Eastern gas supply is a possible counterweight to Russia. They conclude that coordinated plans by Russia and the Middle East to disrupt the West European market might be more devastating.

More recently, [26] has formulated a simplification of a previously developed Global Gas Model (which we will discuss later). This paper shows an equivalence of the convex problem formulation and a mixed complementarity problem by showing that the KKT conditions are essentially the same. This has resulted in a significant speed-up of the solution method, which allows for more scenarios and more granularity. This simplified model is used in [27] to perform a case study on exports from U.S. to Europe and how NG consumption in Europe is influenced by several policy scenarios. The objective function of the model maximizes the profit of the system i.e. the sales profit from all agents minus the production, transportation, infrastructure costs.

Mathematical Program with Equilibrium Constraints (MPEC) The study [16] constructed the model GASTALE which is a NG market model consisting of successive oligopolies which are markets with a handful of powerful agents. The market consists of two levels, the upstream and downstream market. The upstream market consists of producers which are Cournot producer agents, which hold significant market power. In addition, they act as Stackelberg leaders with respect to the trade agents. The trader agent can either be a price-taker or price-maker and

acts as the Stackelberg follower. The study concludes that this two level approach has led to higher prices and lower consumer welfare compared to the one level market from OPs and EPs. The study showed that when the traders are price-makers, then they have more impact on the clearing price, compared to the producers, who are also price makers. Interestingly, when the number of Cournot traders grows, then the market converges to a market with perfect competition i.e. the traders and transportation agents are unable to assert their market power because there are too many of them.

The method in [31] models the profit optimisation of producers, consumers, pipeline operators, investments and storage agents. Unique to this model is the inclusion of peak gas operators. The MPEC used is a mixed non-linear complementarity problem. The model is performed on three small scale base case markets. In addition, two assumptions are given such that a solution can be guaranteed. An extension to this model is the World Gas Model in [25], which is a large-scale multi-period complementarity model. The model includes liquefiers (cooling down the NG to LNG), LNG shipping companies and regasifiers (heating up the LNG to NG) which model the necessary components of the LNG market. These agents are necessary to model a global NG market.

The study by [19] applies a net present value method to determine whether there is an incentive to begin construction of the South-stream project which connects the supply from Asia to the demand in Europe through the Black Sea and the Balkans into Central Europe. The study can be interpreted as simulating the effect of an investment agent while optimizing for the profit of the whole system while studying the net present value of the project. They conclude that the economic value is negative in all three considered scenarios. The MPEC used is similar to the GASTALE model [30].

Equilibrium Problems with Equilibrium Constraints (EPEC) In [44] a static model of the European NG supply called GASMOD is discussed. The market is modelled as a two-stage game of a NG market which separates the market in an upstream (exports to Europe) and a downstream (trade within Europe) market. Trading is constrained by the infrastructure capacities, which are implemented via an upper bound for the amount being traded on each route. Three scenarios are studied: Perfect competition in both markets, Cournot competition in both markets and perfect competition in the upstream market together with Cournot competition in the downstream market. They concluded that a scenario by Cournot competition in both markets is the best representation.

In [10], an analysis of the impact of uncertain disruptions in gas supply is performed. The resulting price and welfare caused by the disruptions are characterized by long-term gas contracts. This model, called GaMMES, also uses an upstream market to model the producer agents and local marketeers, who manage the gas distribution within a region, connected to a downstream market to model the demand side via consumer agents. The model is performed on three scenarios: Supply security of the Western European market, German market (same scenario as is [55]) and the Bulgarian market.

The NATGAS model in [75] is a long-term projection model. The model includes producers, consumers, pipeline operators, storages, investments, traders and LNG shipping agents. The scope of this model is the European market. The mathematical formulation consists of first-order conditions which can be used to formulate a mixed complementarity problem. The model assumes imperfect competition, because of the restrictions on the supply side. The gas market has characteristics of Cournot and Stackelberg competition. Sensitivity testing is performed on the parameters which quantify the degree of imperfect competition. Several scenarios are run for base, low and high LNG prices where Dutch imports, Norwegian and Russian export and resulting prices are studied.

Agent-Based Systems In [46] an agent-based system is used to model the major NG markets: The European, American and Asian markets. Each market consists of seller and buyer agents. These agents perform the ask and bid mechanism to perform trading in each market. Each time step, a clearing price is determined for that market. Then, trading between markets is performed. This simulation is a timeline of market equilibria where for each time step the equilibrium is simulated using the agent-based system. This method drops the assumption that the market acts

completely rationally, which underlies much economical theory and are present in all the previously discussed models. As a consequence, the equilibrium found is (generally) not optimal. The study performed a long-term forecast of the NG market where prices and quantities are tracked. Both supply and demand forecasts per region are published.

Chapter 2

LNG Market Equilibrium Model

In this chapter, we describe the LNG market from the perspective of maximizing the LNG shipping companies' profit margins. In Section 2.1, we introduce individual components that play a role in the optimization of the LNG shipping companies' profit as well as the optimisation problem description. We assume that solving this optimisation problem leads to a market equilibrium by equating the supply and demand side of the market. Next, in Section 2.2, we express the optimisation using compact matrix notation. Lastly, using this matrix notation, we explore the mathematical properties of the market equilibrium model.

2.1 Mathematical Formulation of Key Market Components

In this section, we will cover the mathematical formulation of the supply and demand dynamics, the objective function which defines the traders profit margins and cover the constraints which are applicable to the LNG market. In Section 2.1.1, we introduce the definition of a region and the formulation of the inverse supply and demand as ask and bid functions. The ask and bid functions are combined with the transportation costs, canal passage costs and travel time to form the objective function which represents all LNG shipping companies as one firm. This results in a monopolistic LNG market model. As far as we know, a monopolistic OP approach has not yet been used to model the LNG market. In Section 2.1.2, we cover the formulation of the constraints of the LNG market and the local market clearing process. We show that regionally the import and export are constrained by the port capacity. Exporting regions have limited selling capability modelled via a production capacity. In addition to defining the objective function based on the LNG shipping companies' profit, our model distinguishes itself from the literature in Table 1.1 by incorporating a constraint on the global LNG shipping capacity.

2.1.1 Formulation of Objective Function

Global LNG trade happens between regions $R := \{0, ..., N-1\}$, each classified as an export region (seller) $S := \{0, ..., M-1\} \subset R$ or import region (buyer) $R \backslash S = \{M, ..., N-1\}$. Conceptually, one can view this trading network as a matrix $X \in \mathbb{R}^{M \times N}$, where the rows represent the sellers (S) and the columns represent the sellers and buyers, i.e., all regions (R). Each entry, $x_{ij} \in \mathbb{R}$, is a non-negative quantity being traded from export region $i \in S$ to region $j \in R$. All regions have a unique linear demand function from which the inverse is called the bid function, as in (1.3b). Export regions have an additional ask function, which is the inverse linear supply function, as in (1.3a). Export regions use pipeline trade within their own region which is assumed to be already priced into the supply price. Liquefication and regasification terminals usually add a flat rate to the costs of LNG. As such, we assume that liquefication and regasification are absent in the model because their costs can be incorporated in the prices of the regions. In addition, we assume that export regions do not trade with each other, but only allow for LNG trade to import regions. This can be expressed as

$$x_{ij} = 0, \text{for } i, j \in S,\ i \neq j. \tag{2.1}$$

Note that $x_{ii} \geq 0$, where $i \in S$ is still possible i.e. there is non-LNG tanker trade in an exporting region. In Figure 2.1, we illustrate the basic intuition behind the trade flow matrix X for six import (denoted by blue) and six export (denoted by orange) regions. The orange rectangle denotes the

diagonal of the matrix, which represent non-negative elements. The right hand side of matrix X denoted by the green square denotes the LNG trade flows between the export and import regions.

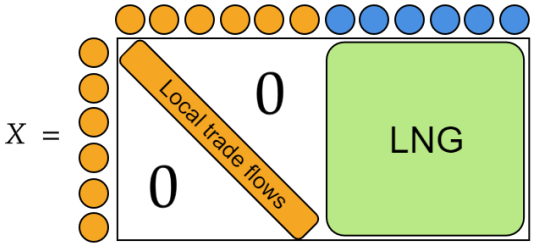


Figure 2.1: Illustrative example of matrix X.

The regional price is determined by the quantities matching regional supply and demand. Through the trade flows X, all regions are interconnected, which makes solving for the market equilibrium non-trivial. The connection defines the supply and demand dynamics, where the price (output) depends on the quantity traded (input). Modelling this interaction requires definitions for the quantities bought and sold. The import and export are given by

$$\text{export: } E_i(X) := \sum_{j=M}^{N-1} x_{ij}, \quad \text{for } i \in S, \tag{2.2a}$$

$$\text{import: } I_j(X) := \sum_{i=0}^{M-1} x_{ij}, \quad \text{for } j \in R \backslash S. \tag{2.2b}$$

In total, export region $i \in S$ sells quantity $n_i(X)$ units given by

$$n_i(X) := x_{ii} + E_i(X), \text{ for } i \in S, \tag{2.3}$$

where x_{ii} denotes the quantity that region $i \in S$ sold within the own region and $E_i(X)$ is the quantity that region i exported. Export and import regions $j \in R$ buy $m_j(X)$ units of LNG. However, the definition of m_i for $i \in S$ and m_j for $j \in R\backslash S$ are different, because we assume that export region $i \in S$ cannot buy from other export regions, i.e., export regions have no imports. Import regions cannot sell to themselves, thus they only buy LNG based on imports. The definitions for m_j, for $j \in R$, are given by

$$\text{export region: } m_i(X) := x_{ii}, \text{ for } i \in S, \tag{2.4a}$$

$$\text{import region: } m_j(X) := I_j(X), \text{ for } j \in R\backslash S, \tag{2.4b}$$

where $I_j(X)$ is the imported quantity of region j defined in (2.2b).

As stated in (1.3) α, β, the input for the ask and bid functions are the quantities sold $n_i(X)$ for $i \in S$ and quantities bought $m_j(X)$ for $j \in R$. As a result, both ask and bid functions depend on the trade flows X. We model the local prices with ask and bid functions which have n_i, m_j as input, respectively, i.e., $\alpha_i(n_i(X))$ and $\beta_j(m_j(X))$ for region $i \in S, j \in R\backslash S$, respectively. We define α_i, β_j by the inverse demand and inverse supply functions:

$$\text{ask function: } \alpha_i(n_i(X)) := \eta_i^{\mathrm{A}} n_i(X) + a_i, \quad \text{for } i \in S, \tag{2.5a}$$

$$\text{bid function: } \beta_j(m_j(X)) := -\eta_j^{\mathrm{B}} m_j(X) + b_j, \quad \text{for } j \in R, \tag{2.5b}$$

where $\eta_i^{\mathrm{A}}, \eta_j^{\mathrm{B}} \in R$ are the price slopes and $a_i, b_j \in \mathbb{R}$ are the price levels in region $i \in S, j \in R$. These parameters form the vectors $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$ and will be calibrated in Chapter 3, based on real market data. For the optimisation regarding the market equilibrium, they are considered to be constants.

Naturally, a shorter distance allows for quicker transportation. Hence the shorter the distance, the less ship capacity is needed to trade a fixed amount of LNG within a fixed time period. It is

possible, that it is more beneficial to trade multiple times on a less profitable but shorter route, than trading on a longer route with a higher profit per unit. We will approximate the number of times an LNG tanker can travel on the trade route by a time scale

$$\frac{H}{\theta_{ij}}, \tag{2.6}$$

where $H \in \mathbb{R}$ is the time horizon and $\theta_{ij} \in \mathbb{R}$ denotes the time it takes to travel from region $i \in S$ to region $j \in R \backslash S$. We define the travel time θ_{ij} by

$$\theta_{ij} := \frac{T_{ij}}{v_{\text{avg}}} + t^{(s)} d_{ij}^{(s)} + t^{(p)} d_{ij}^{(p)}, \tag{2.7}$$

for trade from region $i \in S$ to region $j \in R \backslash S$. Here, $T_{ij} \in \mathbb{R}$ gives the distance needed to travel to deliver a shipment from region i to region j and return back to region i, which we a symmetric matrix $T \in \mathbb{R}^{M(N-M) \times M(N-M)}$. Parameter v_{avg} is the average speed of an LNG tanker. Dividing the distance by the average speed gives the average travel time based on only the distance between two regions. Binary parameters $d_{ij}^{(s)}, d_{ij}^{(p)} \in \{0, 1\}$ and real parameters $\pi^{(s)}, \pi^{(p)} \in \mathbb{R}$ indicate the usage and costs of the Suez and Panama canals, respectively. Parameters $t^{(s)}, t^{(p)}$ are the passage times for the Suez- and Panama canal, respectively.

The transportation costs per unit of LNG are based on the distance the LNG tanker travels and the usage of a passage (Suez- or Panama canal), and are given by

$$\tau_{ij} := cT_{ij} + \frac{H}{\theta_{ij}} \big(\pi^{(s)} d_{ij}^{(s)} + \pi^{(p)} d_{ij}^{(p)} \big), \tag{2.8}$$

where $\frac{H}{\theta_{ij}}$ is the time scale from (2.6) and parameters $T_{ij}, d_{ij}^{(s)}, d_{ij}^{(p)}$ are the same as in (2.7). Parameter $c \in \mathbb{R}$ is the operation cost per distance unit per unit of LNG (in our case nautical mile per MWh of LNG), such that cT_{ij} denotes the operation costs per unit of LNG on the route from region i to region j. The cost of the canals are scaled by (2.6) because a shorter route will use the canal(s) more often than a longer route.

We capture the profit margins of the LNG traders by defining the quantities traded multiplied by the price difference (based on local prices from region i and j) minus the transportation costs to compute the profit of the LNG trader on route (i, j). The profit of all LNG traders combined form one firm which is given by an objective function φ, which is optimised for all routes, which is given by:

$$\varphi(X) := \sum_{\substack{i \in S \\ j \in R \backslash S}} \frac{H}{\theta_{ij}} \big[\beta_j(m_j(X)) - \alpha_i(n_i(X)) \big] x_{ij} - \tau_{ij} x_{ij}, \tag{2.9}$$

where $X = (x_{ij})_{i \in S, j \in R}$ represents the trade flows between export regions $i \in S$ and regions $j \in R$, α_i, β_j are the ask and bid functions in (2.5). Parameter τ_{ij} represents the transportation costs per unit of LNG given by (2.8), and $\frac{H}{\theta_{ij}}$ denotes the appropriate time scaling from (2.6) of the revenue $\beta_j - \alpha_i$ for trade between regions $i \in S$ and $j \in R \backslash S$.

2.1.2 Formulation of Constraints

In this section, we cover the definition of the regional and global constraints which are used to model the LNG market.

Starting with the clearing mechanism, we assume that the local ask and bid prices in the region must be equal, as in (1.4). This is expressed by the following equality constraint:

$$\beta_i(m_i(X)) = \alpha_i(n_i(X)), \qquad i \in S. \tag{2.10}$$

where demand price from the consumer is equated to the price of the producer. This constraint defines the market clearing process where prices are equated within the export region $i \in S$ such that the inverse demand and supply agree on one price. In export regions $i \in S$, we neglect the

costs of regional distributors and pipeline operations, which will not be part of the regional price.

There are two regional capacities which may limit the trade. Firstly, the port capacity $C_j \in \mathbb{R}$, for $j \in R$, which restricts import and export, and is given by

$$0 \leq I_j(X) \leq C_j, \qquad j \in R \backslash S; \tag{2.11}$$
$$0 \leq E_i(X) \leq C_i, \qquad i \in S, \tag{2.12}$$

where $I_j(X), E_i(X)$ are from (2.2). Secondly, the production limit $L_i \in \mathbb{R}$ for $i \in S$ is an upper limit denoting how much NG can be produced. The value L_i is set somewhere higher than the actual production limit to account for storage of NG within a region. This constraint is given by

$$0 \leq n_i(X) \leq L_i, \qquad i \in S. \tag{2.13}$$

An import region, $j \in R \backslash S$, has to deal with constraint (2.11), while an export region, $i \in S$, has to deal with constraints (2.12) and (2.13).

Next to regional constraints, we add a constraint on the global shipping capacity. We define maximal global LNG shipping capacity $\kappa \in \mathbb{R}$. We formulate this constraint by

$$\sum_{\substack{i \in S \\ j \in R \backslash S}} \frac{\theta_{ij}}{H} x_{ij} \leq \kappa, \tag{2.14}$$

where the ratio $\frac{\theta_{ij}}{H}$ incorporates a time-scaling component into the model. We refer to the left-hand side of (2.14) as the *utilized shipping capacity*. An example which illustrates the need for this ratio (which is the inverse of the time scaling (2.7)) is the following: Using one LNG tanker to trade one tanker-unit of LNG per week on a one-week-travel route is equivalent to seven tanker-units on a one-day-travel route, because in both cases one tanker-unit is being put to use. The consequence of adding this constraint, instead of a route specific constraint, is that the global capacity implies that when there is a business case on one side of the world, it might not be used because global capacity limits the total trade. This restriction is realistic for our period of interest because of the relative small number of LNG ships in global use. This also implies that a shock on one side of the world could have a ripple effect to the other side.

In summary, we give the overall optimisation problem which maximizes the time-scaled profit margins of all LNG shipping companies. We will refer to this program as the *sum notation*, based on the element-wise summation of equation (2.9).

$$\begin{aligned}
&\max_X \varphi(X) \\
&\text{subject to} \\
&x_{ij} \geq 0, && i \in S, j \in R && \text{(Non-negative traded amount)}, \\
&\sum_{\substack{i \in S \\ j \in R \backslash S}} \frac{\theta_{ij}}{H} x_{ij} \leq \kappa, && && \text{(Global shiping capacity)}, \\
&\beta_i(m_i(X)) = \alpha_i(n_i(X)), && i \in S && \text{(Local market clearing)}, \\
&0 \leq E_i(X) \leq C_i, && i \in S && \text{(Port capacity)}, \\
&0 \leq n_i(X) \leq L_i, && i \in S && \text{(Production limit)}, \\
&0 \leq I_j(X) \leq C_j, && j \in R \backslash S && \text{(Port capacity)}.
\end{aligned} \tag{2.15}$$

2.2 Standard Quadratic Optimisation Form

The optimisation problem (2.15) follows an intuitive notation which can be easily traced back to the different modelling components. However, implementing this optimisation in a programming language is not as convenient as a matrix-vector based notation which is commonly used for quadratic optimisation problems. Understanding, the quadratic structure of this function is non-trivial from the notation in Section 2.1. Lastly, understanding properties such as convexity is

well understood for standard quadratic matrix notation. For that reason, we transform (2.15) to a mathematical program which uses matrices and vectors to formulate the same problem, but now in standardized quadratic optimisation notation, which is in line with the theory of quadratic optimisation, discussed in [17, 58]. In Section 2.2.1, we rewrite the objective function. In Section 2.2.2, we rewrite the constraints. Rewriting will be based on a re-indexing which will be introduced before we show the details of the objective function and constraint.

In the clearing price constraint (2.10), we substitute the definitions of $n_i(X), m_j(X)$ from (2.3) and (2.4), respectively, to reduce the number of variables present in the objective function. By rewriting (2.10), we obtain an expression for x_{ii}, as follows

$$x_{ii} = \frac{a_i - b_i}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} + \frac{\eta_i^{\mathrm{A}}}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} E_i(X), \tag{2.16}$$

for $0 \leq i < M$. By substituting (2.16) in (2.3), we find that $n_i(X)$ from (2.3) can be written as

$$n_i(X) = \frac{a_i - b_i}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} + \frac{-\eta_i^{\mathrm{B}}}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} E_i(X), \tag{2.17}$$

which can be substituted in the objective function $\varphi(X)$ in (2.9), to find that the multiplication $\alpha_i(n_i(X)) \cdot x_{ij}$ can be split in a quadratic and linear part, as follows

$$\alpha_i(n_i(X)) \cdot x_{ij} = \underbrace{\left[\eta_i^{\mathrm{A}} \frac{-\eta_i^{\mathrm{B}}}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} E_i(X)\right] x_{ij}}_{\text{quadratic}} + \underbrace{\left[\frac{\eta_i^{\mathrm{A}}(a_i - b_i)}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}} + a_i\right] x_{ij}}_{\text{linear}}. \tag{2.18}$$

As a consequence, we find that α in (2.18) and β from (2.5b), are only dependent on x_{ij} for $i \in S, j \in R \backslash S$, because $I_j(X), E_j(X)$ also only depends on x_{ij} for $i \in S, j \in R \backslash S$, and thus the objective function φ can be simplified.

In addition to the possible reduction of variables, we show the reformulation of (2.9) to matrix notation (2.20) by first rewriting the matrix variable $X \in \mathbb{R}^{M \times N-M}$ into a vector variable $\mathbf{x} \in \mathbb{R}^{M(N-M)}$. We rewrite $\varphi(X)$ into a quadratic matrix form via the column-wise matrix notation, where the index is

$$Mj + i := i, j + M, \tag{2.19}$$

such that we use vector $\mathbf{x} = (\mathrm{x}_k)_{0 \leq k \leq M(N-M)}$ for the objective function and constraints. We illustrate the translation between matrix X and vector $\mathbf{x}$ in Figure 2.2. The orange and blue circles represent the export and import regions, respectively. The Roman numerals are used to identify the arrangement of the columns.

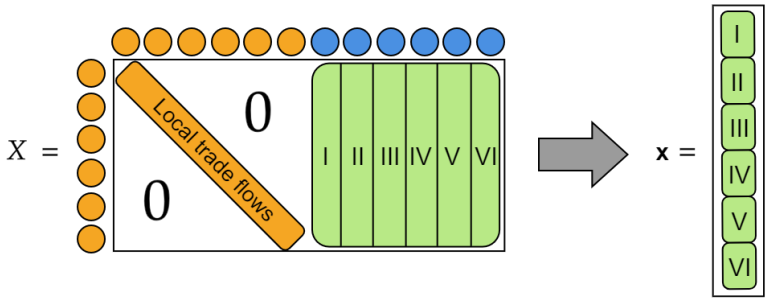


Figure 2.2: Illustrative example of the translation between matrix X and vector $\mathbf{x}$.

We define $P \in \mathbb{R}^{M(N-M) \times M(N-M)}$, $D \in \mathbb{R}^{K \times M(N-M)}$ and $\mathbf{c} \in \mathbb{R}^{M(N-M)}$ such that the optimisation (2.15) is given by quadratic optimisation problem

$$\begin{aligned} \max_{\mathbf{x}} \ & \varphi(\mathbf{x}) = \mathbf{x}^{\intercal} P \mathbf{x} + \mathbf{c}^{\intercal} \mathbf{x}, \\ & \text{subject to} \\ & D\mathbf{x} \leq \mathbf{v}, \end{aligned} \tag{2.20}$$

which is an elegant notation to provide insight into the convexity of φ by investigating whether or not the matrix P is positive semi-definite [17]. Note that the equality constraint (2.10) responsible for the market clearing is implicitly present in the optimisation, because of the substitution of (2.16) in the objective function φ. Thus, we can remove it from the constraints. in (2.20).

2.2.1 Objective Function in Matrix Form

In this section, we show the application of the re-indexing of (2.19). With this re-indexing, we rewrite the objective function φ in (2.9) to a quadratic objective function in matrix form as follows

$$\mathbf{x}^\intercal P\mathbf{x} + \mathbf{c}^\intercal\mathbf{x} = \sum_{\substack{i\in S\\ j\in R\setminus S}} \frac{H}{\theta_{ij}}\left[\beta_j(m_j(X)) - \alpha_i(n_i(X))\right] x_{ij} - \tau_{ij}x_{ij}.$$

To accomplish this, we define matrix $P \in \mathbb{R}^{M(N-M)\times M(N-M)}$, which corresponds to the quadratic part of (2.9). We use matrices $A, B \in \mathbb{R}^{M(N-M)\times M(N-M)}$ as auxiliary matrices to define matrix P, describing the ask functions α_i from (2.17) and the bid functions β_j from (2.5b), respectively. It is important to understand how the re-indexing from (2.19), combined with exploiting the structure of the summations $I_j(X)$ and $E_i(X)$, results in the expressions for the import and export,

$$I_j(\mathbf{x}) := (\mathcal{I}^{\mathrm{b}}\mathbf{x})_j = \sum_{i=0}^{M-1} \mathrm{x}_{Mj+i} = \sum_{i=0}^{M-1} x_{i,j+M} = I_{j+M}(X) \quad \text{for } 0 \le j < N-M, \tag{2.21a}$$

$$E_i(\mathbf{x}) := (\mathcal{I}^{\mathrm{c}}\mathbf{x})_i = \sum_{j=0}^{N-M-1} \mathrm{x}_{Mj+i} = \sum_{j=M}^{N-1} x_{i,j+M} = E_i(X) \quad \text{for } 0 \le i < M, \tag{2.21b}$$

where $(.)_i$ denotes the i-th component, and matrices $\mathcal{I}^{\mathrm{b}}, \mathcal{I}^{\mathrm{c}} \in \mathbb{R}^{M(N-M)\times M(N-M)}$ represent a *block-wise* and *cyclic* structure, respectively. With this block-wise and cyclic structure, we refer to the arrangement of non-zero elements in a matrix $\mathcal{I}^{\mathrm{b}}, \mathcal{I}^{\mathrm{c}}$, defined by

$$\mathcal{I}^{\mathrm{b}} := \begin{bmatrix} \mathbb{I} & \cdots & \mathbb{I} \\ \vdots & \ddots & \vdots \\ \mathbb{I} & \cdots & \mathbb{I} \end{bmatrix}, \qquad \mathcal{I}^{\mathrm{c}} := \mathrm{Diag}(\mathcal{J}, ..., \mathcal{J}), \tag{2.22}$$

where $\mathbb{I} \in \mathbb{R}^{M\times M}$ is the identity matrix and $\mathcal{J} \in \mathbb{R}^{N-M\times N-M}$ is the "all ones matrix". Matrices $\mathcal{I}^{\mathrm{c}}$ and $\mathcal{I}^{\mathrm{b}}$ are used to define the matrices A and B, respectively. The operator Diag(.) defines a diagonal block matrix, with zeros on the off-diagonal blocks. In addition, we define the time-scaling matrix

$$\Theta := \begin{bmatrix} \boldsymbol{\theta}^{(0)} & \cdots & \boldsymbol{\theta}^{(N-M)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\theta}^{(0)} & \cdots & \boldsymbol{\theta}^{(N-M)} \end{bmatrix}, \tag{2.23}$$

where vector $\boldsymbol{\theta}^{(i)}$ is the i-th column of matrix $\theta = (\theta_{ij})_{i\in S, j\in R\setminus S}$ from (2.7), which is used to define the block matrix Θ. Depending on the index which is used in the summation, the coefficients in P will be placed in a cyclic or block-wise manner. Matrix P is defined, in an element-wise fashion, as follows

$$P_{kl} := \frac{H}{\Theta_{kl}}(B_{kl} - A_{kl}), \qquad \text{for } 0 \le k, l < M(N-M), \tag{2.24}$$

where $\Theta_{k,l}$ is the travel time coefficient from (2.23). Next, we define matrix A as

$$A_{kl} := \eta^{\mathrm{A}}_{k \bmod M}\left(\frac{-\eta^{\mathrm{B}}_{k \bmod M}}{-\eta^{\mathrm{B}}_{k \bmod M} - \eta^{\mathrm{A}}_{k \bmod M}}\right)\cdot \mathcal{I}^{\mathrm{c}}_{kl}, \tag{2.25}$$

where $0 \le k, l < M(N-M)$, Alternatively, matrix A can be equivalently defined as

$$A := \begin{bmatrix} H^{\mathrm{A}} & \cdots & H^{\mathrm{A}} \\ \vdots & \ddots & \vdots \\ H^{\mathrm{A}} & \cdots & H^{\mathrm{A}} \end{bmatrix}, \tag{2.26}$$

where the auxiliary matrix H^{A} is defined as

$$H^{\mathrm{A}} = \operatorname{diag}\left(\eta_0^{\mathrm{A}}\left(\frac{-\eta_0^{\mathrm{B}}}{-\eta_0^{\mathrm{B}}-\eta_0^{\mathrm{A}}}\right), ..., \eta_{M-1}^{\mathrm{A}}\left(\frac{-\eta_{M-1}^{\mathrm{B}}}{-\eta_{M-1}^{\mathrm{B}}-\eta_{M-1}^{\mathrm{A}}}\right)\right).$$

Both the element-wise definition from (2.25), and the matrix definition from (2.26) make sure that A follows a 'cyclic' structure of $\mathcal{I}^{\mathrm{c}}$ where the entries '1', are replaced by the diagonal entries of H^{A}. Likewise, we define matrix B, as follows,

$$B_{kl} := -\eta^{\mathrm{B}}_{\lfloor k/M \rfloor} \cdot \mathcal{I}^{\mathrm{b}}_{kl},$$

where $0 \leq k, l < M(N-M)$. Matrix B can also be defined alternatively by

$$B := \operatorname{Diag}(H_M^{\mathrm{B}}, ..., H_{N-1}^{B}),$$

where the auxiliary matrix H_j^{B} is defined as

$$H_j^{\mathrm{B}} = \begin{bmatrix} -\eta_j^{\mathrm{B}} & \cdots & -\eta_j^{\mathrm{B}} \\ \vdots & \ddots & \vdots \\ -\eta_j^{\mathrm{B}} & \cdots & -\eta_j^{\mathrm{B}} \end{bmatrix},$$

for $M \leq j < N$, similar to A, such that B follows the structure of $\mathcal{I}^{\mathrm{b}}$ where the entries '1', are replaced by the entries of H^{B}. To stress, the dimensions of A, B are $M(N-M) \times M(N-M)$ and not $MN \times MN$, because the variables x_{ij} for $i, j \in S$, denoting the trade between export regions, are not part of $\mathbf{x}$. Note, if $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}} > 0$, then A is strictly positive while B is strictly negative. The difference $B - A$ is thus also strictly negative. However, $B - A$ does not capture all of the revenues of $\beta_j - \alpha_i$, because we have seen from (2.18) that $\alpha_i(n_i(X)) \cdot x_{ij}$ also has a linear part. As such, we define the vector $\mathbf{c}$ which includes the linear part in (2.9). We use the re-indexing from (2.19) to define vector $\mathbf{c} = (c_k)_{0 \leq k < M(N-M)}$, in an element-wise fashion, as follows

$$c_{Mj+i} := \frac{H}{\theta_{i,j+M}}\left(b_{j+M} - a_i + \frac{\eta_i^{\mathrm{A}}(a_i - b_i)}{-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}}}\right) - \tau_{i,j+M}, \tag{2.27}$$

for $0 \leq i < M$ and $0 \leq j < N - M$, which leads to a vector $\mathbf{c}$ of length $M(N-M)$ containing the linear parts of the revenue, transportation costs and the passage cost. Using P as given by (2.24), and $\mathbf{c}$ as given by (2.27), we get that

$$\mathbf{x}^{\intercal} P \mathbf{x} + \mathbf{c}^{\intercal}\mathbf{x} = \sum_{\substack{i \in S \\ j \in R \setminus S}} \frac{H}{\theta_{ij}}\left[\beta_j(m_j(X)) - \alpha_i(n_i(X))\right] x_{ij} - \tau_{ij} x_{ij}.$$

2.2.2 Constraints as Linear Matrix Inequalities

In the following section, the inequality constraints from the optimisation problem in (2.15) will be given by the linear inequality

$$D\mathbf{x} \leq \mathbf{v}, \tag{2.28}$$

which is part of the quadratic optimisation problem given in (2.20). Remember that, the equality constraint in (2.10), does not need be included in the constraints of the quadratic optimisation, because of the implicit presence in the objective function, by (2.16). Before we rewrite the constraints into the matrix notation, we will reduce the number of effective constraints present in the optimisation. Substituting x_{ii}, given by (2.16), in the non-negative trade flows constraint of (2.15) results in a constraint, given by

$$E_i(\mathbf{x}) \leq \frac{b_i - a_i}{\eta_i^{\mathrm{A}}}, \text{ for } 0 \leq i < M. \tag{2.29}$$

Similarly, by substitution of x_{ii}, given by (2.16), together with the re-indexing as in (2.19), we find that the production limit given by (2.13) can be written as

$$\frac{b_i - a_i}{\eta_i^{\mathrm{A}}} \leq E_i(\mathbf{x}) \leq \frac{(-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}})L_i + b_i - a_i}{-\eta_i^{\mathrm{B}}}, \text{ for } 0 \leq i < M, \tag{2.30}$$

which, together with the remaining export constraint (2.29), shows that we have redundant bounds on the export $E_i(\mathbf{x})$. We can reduce the number of constraints by replacing the lower and upper bounds by a maximum and minimum, respectively, given by

$$\max\left\{0, \frac{a_i - b_i}{-\eta_i^{\mathrm{B}}}\right\} \leq E_i(\mathbf{x}) \leq \min\left\{C_i, \frac{(-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}})L_i + b_i - a_i}{-\eta_i^{\mathrm{B}}}, \frac{b_i - a_i}{\eta_i^{\mathrm{A}}}\right\}, \tag{2.31}$$

for $0 \leq i < M$. The overview of all constraints is given by

$$\begin{aligned}
&\mathrm{x}_{Mj+i} \geq 0, \qquad \text{for } 0 \leq i < M,\ 0 \leq j < N - M,\\
&\sum_{\substack{i \in S\\ j \in R\setminus S}}^{K} \frac{\theta_{ij}}{H} \mathrm{x}_{Mj+i} \leq \kappa,\\
&\max\left\{0, \frac{a_i - b_i}{-\eta_i^{\mathrm{B}}}\right\} \leq E_i(\mathbf{x}) \leq \min\left\{C_i, \frac{(-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}})L_i + b_i - a_i}{-\eta_i^{\mathrm{B}}}, \frac{b_i - a_i}{\eta_i^{\mathrm{A}}}\right\},\ \text{for } 0 \leq i < M,\\
&0 \leq I_j(\mathbf{x}) \leq C_j, \qquad \text{for } 0 \leq j < N - M.
\end{aligned} \tag{2.32}$$

To adhere to $D\mathbf{x} \leq \mathbf{v}$, the constraints (2.32) need to be rewritten such that we only have upper bound inequalities. This requires the following adaptation:

$$\mathrm{x}_{Mj+i} \leq 0, \qquad \text{for } 0 \leq i < M,\ 0 \leq j < N - M, \tag{2.33a}$$

$$\sum_{\substack{i \in S\\ j \in R\setminus S}}^{K} \frac{\theta_{ij}}{H} \mathrm{x}_{Mj+i} \leq \kappa, \tag{2.33b}$$

$$E_i(\mathbf{x}) \leq \min\left\{C_i, \frac{(-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}})L_i + b_i - a_i}{-\eta_i^{\mathrm{B}}}, \frac{b_i - a_i}{\eta_i^{\mathrm{A}}}\right\}, \quad \text{for } 0 \leq i < M, \tag{2.33c}$$

$$-E_i(\mathbf{x}) \leq \min\left\{0, \frac{a_i - b_i}{-\eta_i^{\mathrm{B}}}\right\}, \qquad \text{for } 0 \leq i < M, \tag{2.33d}$$

$$I_j(\mathbf{x}) \leq C_j, \qquad \text{for } 0 \leq j < N - M, \tag{2.33e}$$

$$-I_j(\mathbf{x}) \leq 0, \qquad \text{for } 0 \leq j < N - M. \tag{2.33f}$$

Each of the inequalities from (2.33) can be translated as $D^{(\omega)}\mathbf{x} \leq \mathbf{v}^{(\omega)}$ for $\omega = 1, ..., 6$, such that we have a collection of inequalities, which can be combined in a vertical concatenation to form the inequality $D\mathbf{x} \leq \mathbf{v}$ given by

$$\begin{bmatrix} D^{(1)} \\ \vdots \\ D^{(6)} \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} \mathbf{v}^{(1)} \\ \vdots \\ \mathbf{v}^{(6)} \end{bmatrix}.$$

To define some of the matrices $D^{(\omega)}$, we use $\mathcal{I}^{(b)}, \mathcal{I}^{(c)}$ from (2.22) such that the matrix multiplication encodes the required summations in the constraints. The constraint (2.33a)translates to defining

$$D^{(1)} := -\mathbb{I}, \qquad \mathbf{v}^{(1)} := \mathbf{0},$$

where $\mathbb{I} \in \mathbb{R}^{M(N-M)\times M(N-M)}$ is the identity matrix. For the global shipping capacity constraint in (2.14), we sum over all elements of the amounts traded x_{Mj+i}. Each element x_{Mj+i} needs to be multiplied by the time scale (2.6), for $i \in S, j \in R\setminus S$. We translate constraint (2.33b) by defining row vector

$$D^{(2)} := \boldsymbol{\mu}^{\intercal} = (\mu)_{0 \leq k < M(N-M)}, \qquad \text{where} \qquad \mu_{Mj+i} := \frac{\theta_{i,j+M}}{H},$$

for $0 \leq i < M,\ 0 \leq j < N - M$. In addition, we define $\mathbf{v}^{(2)} = \kappa$ as scalar value. For the upper and lower bounds on the export $E_i(\mathbf{x})$ given in (2.33c) and (2.33d), respectively, we use the cyclic structure from (2.21b) to define the matrix $D^{(3)}$ and $D^{(4)}$, as follows

$$D^{(3)} := \mathcal{I}^{\mathrm{c}}, \qquad D^{(4)} := -\mathcal{I}^{\mathrm{c}},$$

where $\mathcal{I}^{\mathrm{c}}$ is given by (2.22), and the accompanying vectors $\mathbf{v}^{(3)} = (\mathrm{v}_i^{(3)})_{0\leq i<M}$ and $\mathbf{v}^{(4)} = (\mathrm{v}_i^{(4)})_{0\leq i<M}$ of matrices $D^{(3)}$ and $D^{(4)}$ are, in an element-wise fashion, given by

$$\mathrm{v}_i^{(3)} := \min\left\{C_i, \frac{(-\eta_i^{\mathrm{B}} - \eta_i^{\mathrm{A}})L_i + b_i - a_i}{-\eta_i^{\mathrm{B}}}, \frac{b_i - a_i}{\eta_i^{\mathrm{A}}}\right\}, \tag{2.34}$$

$$\mathrm{v}_i^{(4)} := \min\left\{0, \frac{a_i - b_i}{-\eta_i^{\mathrm{B}}}\right\}, \tag{2.35}$$

for $0 \leq i < M$. We use the block-wise structure of $\mathcal{I}^{(b)}$ from (2.21b) to define $D^{(5)}, D^{(6)}$ which translates the lower and upper bounds on the imports given by (2.33e) and (2.33f), respectively, to the matrix notation. This is similar to the export constraints where $\mathcal{I}^{\mathrm{c}}$ is used to define $D^{(3)}, D^{(4)}$. Matrices $D^{(5)}, D^{(6)}$ are defined as follows

$$D^{(5)} := \mathcal{I}^{\mathrm{b}}, \qquad D^{(6)} := -\mathcal{I}^{\mathrm{b}},$$

where $\mathcal{I}^{\mathrm{b}}$ is given by (2.22). The bounds $\mathbf{v}^{(5)} = (v_j^{(5)})_{0\leq j<N-M}$ and $\mathbf{v}^{(6)} = (v_j^{(6)})_{0\leq j<N-M}$ are defined in an element-wise fashion by

$$\mathrm{v}_j^{(5)} := C_j, \qquad \mathrm{v}_j^{(6)} := 0,$$

for $0 \leq j < N$. The final reduced optimisation problem with minimized constraint bounds is given by

$$\begin{aligned} &\max_{\mathbf{x}} \ \mathbf{x}^{\intercal} P\mathbf{x} + \mathbf{c}^{\intercal}\mathbf{x}, \\ &\text{subject to} \\ &D\mathbf{x} \leq \mathbf{v}. \end{aligned} \tag{2.36}$$

2.2.3 Properties of the Optimisation Problem

To have a guaranteed global optimal solution, we need the optimisation problem (2.36) to be a convex optimisation problem [17]. This requires the matrix P to be negative semi-definite and the constraints $D\mathbf{x} \leq \mathbf{v}$ to be a convex set. Starting with the latter, the constraints given by linear inequalities $D\mathbf{x} \leq \mathbf{v}$ are convex, because from any linear combination of $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \{\mathbf{x} \in \mathbb{R}^{M(N-M)} \ : \ D\mathbf{x} \leq \mathbf{v}\}$ where $t \in (0,1)$, we see that the following holds

$$\begin{aligned} D(t\mathbf{x}^{(1)} + (1-t)\mathbf{x}^{(2)}) &= tD\mathbf{x}^{(1)} + (1-t)D\mathbf{x}^{(2)}, \\ &\leq t\mathbf{v} + (1-t)\mathbf{v}, \\ &= \mathbf{v}, \end{aligned}$$

and thus indeed

$$t\mathbf{x}^{(1)} + (1-t)\mathbf{x}^{(2)} \in \{\mathbf{x} \in \mathbb{R}^{M(N-M)} \ : \ D\mathbf{x} \leq \mathbf{v}\},$$

shows that the linear constraints are convex. A twice-differentiable objective function $\varphi(\mathbf{x})$ is concave when φ has a convex domain and the Hessian $\nabla^2\varphi$ is negative definite [17]. For our quadratic optimisation problem, with $\varphi(\mathbf{x}) = \mathbf{x}^{\intercal}P\mathbf{x} + \mathbf{c}^{\intercal}\mathbf{x}$, we get $\nabla^2\varphi(\mathbf{x}) = P$. This means that we require P to be negative semi-definite, i.e., $\mathbf{x}^{\intercal}P\mathbf{x} \leq 0$, for all $\mathbf{x} \in \mathbb{R}^{M(N-M)}$. Guaranteeing that P is negative semi-definite is difficult because the time-scale Θ can scale the components of $B - A$ such that the resulting element-wise multiplication $\Theta_{kl}(B_{kl} - A_{kl})$ can result in indefinite matrix P.

In Chapter 4, we solve the optimisation problem (2.36) to perform scenario analysis via perturbations compared to the baseline scenario which is based on realistic data. Note that P is a negative matrix, i.e., all components are negative. For each realistic problem (2.36), we also approximate the eigenvalues of P numerically. Next, we check if the matrix P has any eigenvalues equal to zero. If that is the case, then we conclude that the matrix P is indefinite, because the eigenvector $\hat{\mathbf{x}}$ corresponding with the zero-valued eigenvalue would mean that $\hat{\mathbf{x}}^{\intercal}P\hat{\mathbf{x}} = 0$. Throughout all our experimentation, matrix P has been negative semi-definite based on realistic data.

Chapter 3

Calibration of Market Parameters

The goal of this chapter is to formulate the methodology which is used to calibrate the market equilibrium model from Chapter 2, such that the modelled trade flows and prices are as close as possible to realistic trade flows and prices. In Section 3.1, we present a general calibration procedure defined as a bilevel optimisation problem and a way to translate the bilevel optimisation such that solving the model is feasible in reasonable computer time. In Section 3.2, we define the bilevel optimisation problem which calibrates the market equilibrium problem (2.36) from which we obtain a single level optimisation to solve the problem in reasonable computer time. In Section 3.3, we share the results from an accuracy analysis of the calibration process.

3.1 Defining a Calibration as Bilevel Optimisation

In this section, we will define a general bilevel optimisation problem which can be used to calibrate parameter $\mathbf{y} \in \mathbb{R}^m$ such that the solution $\mathbf{x} \in \mathbb{R}^n$ of a general optimisation model is as closely as possible to realistic data $\bar{\mathbf{x}}$. In Section 3.2, we apply this framework to our optimisation model (2.36) from Chapter 2 to calibrate for the parameters $\mathbf{y} := (\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b})$ given realistic data $\bar{\mathbf{x}}$. During this section, we reuse the notation from Chapter 2 to make the application to the market equilibrium model easier to understand. Calibration of an optimisation problem can be modelled as a hierarchical optimisation problem, in the same way as a hierarchical optimisation model is used to model Stackelberg competition. We define a single objective function which scores the calibration process. As such, we use an MPEC, given by (1.9), to model the calibration. The lower level optimisation problem is defined as the optimisation problem which needs to be calibrated. For example, the OP (2.36) will be used as lower level optimisation in the calibration in Section 3.2.2. The upper level optimisation is also an OP, for which we specify the details later on. A hierarchical optimisation of two OPs is called a bilevel optimisation. The upper level optimisation model aims to minimize the difference between the solution of the lower level optimisation, $\mathbf{x}$, and the observed data, $\bar{\mathbf{x}}$, to which the lower level optimisation model needs to be fitted as closely as possible. Consider a (lower level) optimisation problem given by

$$\begin{aligned} \max_{\mathbf{x}} \ & \varphi_{\mathbf{y}}(\mathbf{x}), \\ & \text{subject to} \\ & \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0}, \\ & \boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{3.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the optimised variable and $\mathbf{y} \in \mathbb{R}^m$ a vector with fixed parameters. The functions $\varphi_{\mathbf{y}}, \boldsymbol{\psi}_{\mathbf{y}}, \boldsymbol{\xi}_{\mathbf{y}}$ are the objective function, inequality constraints and equality constraints, respectively, similar to the functions in the OP given in (1.5), but their values depend on a fixed parameter $\mathbf{y}$.

The objective of the upper level optimisation problem is to minimize the distance between the model output $\mathbf{x}$ and the realistic data $\bar{\mathbf{x}}$ where model parameters $\mathbf{y}$ are free variables, in contrast to (3.1) where $\mathbf{y}$ is fixed. To quantify how well we fit the model output $\mathbf{x}$ to the realistic data $\bar{\mathbf{x}}$, we define an upper level objective function $f(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}})$ which represents a loss function. Notice that this objective function can include the realistic data $\bar{\mathbf{x}}$, which we write explicitly to show that f is

used for calibration of market equilibrium solution $\mathbf{x}$ to $\bar{\mathbf{x}}$. We define the vector function $\boldsymbol{g}(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}})$, which defines the constraints of the calibration process. The definitions of f, g are similar to the functions in the MPEC, given by (1.9). The bilevel optimisation is given by

$$\begin{aligned}
&\min_{\mathbf{x},\mathbf{y}} \ f(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}), && \text{(3.2a)}\\
&\quad \text{subject to} \\
&\quad \boldsymbol{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0}, && \text{(3.2b)}\\
&\quad \max_{\mathbf{x}} \ \varphi_{\mathbf{y}}(\mathbf{x}), \\
&\qquad \text{subject to} \\
&\qquad \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0}, && \text{(3.2c)}\\
&\qquad \boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0},
\end{aligned}$$

where (3.2a) and (3.2b) form the upper level optimisation and (3.2c) is the lower level optimisation, as given by (3.1). We can view the optimisation (3.2) from an economic perspective, where the upper level problem represents the optimisation of a leader's objective and the lower level problem is optimising a follower's objective. The leader, in this case the calibration process, anticipates the optimal solution of the follower, in this case the market equilibrium.

The upper level problem is not uniquely defined when the lower level problem has multiple global solutions. There are two ways to alter the bilevel optimisation such that the lower level problem outputs one global solution: optimistic and pessimistic bilevel optimisation. The descriptions (3.3) and (3.4) are based on [65]. Optimistic bilevel optimisation assumes that the follower's optimum will be the best for the leader's objective function among all possible optima. This can be expressed as:

$$\begin{aligned}
&\min_{\mathbf{x},\mathbf{y}} \ f(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}), && \text{(3.3a)}\\
&\text{subject to} && \text{(3.3b)}\\
&\boldsymbol{g}(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}) \leq \mathbf{0}, && \text{(3.3c)}\\
&\mathbf{x} = \operatorname*{argmin}_{\mathbf{x}} \big\{ f(\mathbf{x}, \mathbf{y}) \ : \ \mathbf{x} \in \{ \max_{\mathbf{x}} \varphi_{\mathbf{y}}(\mathbf{x}) \quad \text{subject to} \quad \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0}, \boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0} \} \big\}, && \text{(3.3d)}
\end{aligned}$$

where the lower level optimisation (3.2c) gets replaced by (3.3d). Whereas pessimistic bilevel optimisation assumes the opposite, meaning that the follower returns the optimum which is the worst for the leaders objective function, i.e., $\mathbf{x}$ is the argument for a maximum instead of a minimum, which is given by:

$$\begin{aligned}
&\min_{\mathbf{x},\mathbf{y}} \ f(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}), && \text{(3.4a)}\\
&\text{subject to} && \text{(3.4b)}\\
&\boldsymbol{g}(\mathbf{x}, \mathbf{y}; \bar{\mathbf{x}}) \leq \mathbf{0}, && \text{(3.4c)}\\
&\mathbf{x} = \operatorname*{argmax}_{\mathbf{x}} \big\{ f(\mathbf{x}, \mathbf{y}) \ : \ \mathbf{x} \in \{ \max_{\mathbf{x}} \varphi_{\mathbf{y}}(\mathbf{x}) \quad \text{subject to} \quad \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0}, \boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0} \} \big\}, && \text{(3.4d)}
\end{aligned}$$

where the lower level optimisation (3.2c) gets replaced by (3.4d). The assumption for pessimistic bilevel optimisation is stronger than that for optimistic bilevel optimisation. Thus, any algorithm which solves the pessimistic problem (3.4) will also solve for optimistic problem (3.3), the opposite does not hold in general.

It is important to note that an (optimistic) bilevel optimisation problem is a non-convex and non-differentiable optimisation problem, even if the functions f, φ defined in the upper and lower level optimisation are all convex and smooth [20]. A linear function is the simplest non-trivial objective function for f and φ in the bilevel optimisation, but even linear objective functions will result in a bilevel optimisation problem which is NP-hard [15]. There are a number of techniques which can be used to change the bilevel optimisation such that solving becomes feasible in reasonable computer time. Two techniques, we will cover here are, rewriting the bilevel optimisation to a single level optimisation and penalty functions. Other methods, which we do not use, are mentioned in Appendix B.2.

Bilevel optimisations are hierarchical in nature, but can be rewritten to a single level optimisation when the lower level optimisation is convex, analogously to rewriting an MPEC (1.9) into an MPCC (1.10) via the use of the KKT-conditions (1.6). We can rewrite the problem, given by (3.2), as a single optimisation problem if the lower level optimisation in (3.2c) is convex[1]. The convexity condition is needed to guarantee that the solution of the KKT-conditions corresponds to a global optimum. Under convexity assumptions, the lower level optimisation problem (3.1) can be written by the KKT-conditions, similar to (1.6), as

$$\begin{aligned} &\nabla_{\mathbf{x}}\mathcal{L}_{\mathbf{y}}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu}) = \mathbf{0}, \\ &\boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0}, \\ &\mathbf{0} \leq \boldsymbol{\lambda} \perp \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0}, \end{aligned} \tag{3.5}$$

where $\boldsymbol{\lambda}, \boldsymbol{\mu}$ are the Lagrange multipliers from the lower level optimisation and $\mathcal{L}_{\mathbf{y}}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu})$ is the Lagrangian function, given by

$$\mathcal{L}_{\mathbf{y}}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu}) := \varphi_{\mathbf{y}}(\mathbf{x}) + \boldsymbol{\lambda}^{\intercal}\boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) + \boldsymbol{\mu}^{\intercal}\boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}).$$

We replace the lower level problem (3.2c) in the bilevel optimisation problem (3.2a), such that a single level optimisation describes the calibration problem:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{y},\boldsymbol{\lambda}} \; & f(\mathbf{x},\mathbf{y};\bar{\mathbf{x}}), \\ & \text{subject to} \\ & g(\mathbf{x},\mathbf{y};\bar{\mathbf{x}}) \leq \mathbf{0}, \\ & \nabla_{\mathbf{x}}\mathcal{L}_{\mathbf{y}}(\mathbf{x},\boldsymbol{\lambda},\boldsymbol{\mu}) = \mathbf{0} \\ & \boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0} \\ & \mathbf{0} \leq \boldsymbol{\lambda} \perp \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) \leq \mathbf{0} \end{aligned} \tag{3.6}$$

Note that expressing the bilevel optimisation (3.2) as a single level optimisation (3.25) increases the number of variables for which we optimise, because we have the additional Lagrange multipliers $\boldsymbol{\lambda}, \boldsymbol{\mu}$ present in the optimisation. The literature on MPCCs is well-developed and several solver implementations are available in open source code, such as [73]. In general, rewriting a bilevel optimisation as a single level optimisation is not always possible because it relies on the convexity condition of the lower level problem.

Part of the difficulty of the bilevel programs lies in the fact that both upper and lower problems are constrained. A way of relaxing this, is by formulating an unconstrained variant of the upper and/or lower problem. It is possible to rewrite the objective function to include penalty terms when the constraint inequalities are not adhered to. This idea was first applied to the lower level problem [11], but the hierarchy between the upper and lower level problem remains. Later on, both programs were relaxed via a penalty function [47]. The penalty function relaxation is not unique to bilevel optimisation, but can be used for any constrained optimisation, a combination of reducing the problem to a single optimisation and then applying the penalty function relaxation is also used, namely in [74]. The algorithm used by the solver IPOPT to solve the single level optimisation problem (3.25) uses a penalty term to solve the optimisation problem. We have used IPOPT to solve the calibration problem. We will discuss this algorithm and the relaxation applied by IPOPT in Section A.1.

3.2 Calibration of the Market Equilibrium Model

In this section, we present the formulation of the bilevel optimisation problem, of the form (3.2), which is used to calibrated the market equilibrium model (2.36). We divide this section into the definition of the objective function (3.2a) in Section 3.2.1 and the definition of the constraints (3.2b) in Section 3.2.2. We show that the bilevel optimisation can be written as a single level optimisation of the form (3.25).

[1]Note that with convex, we also include maximisation problems with a concave objective function.

3.2.1 Objective Function of the Calibration

The goal of the calibration is to find the values for $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$, such that the linear inverse supply and demand functions capture the demand and supply behaviour corresponding to realistic data. It should be noted that there is no accurate data available of the parameters $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$. Hence, we need to perform a calibration to approximate them. During the calibration, these values are bounded within in a realistic bandwidth. The calibration is challenging because of the relation between the trade flows and the prices. This relation is given by the linear inverse supply and demand functions

$$\underbrace{\alpha_i \left(\sum_{j=M}^{N-1} x_{ij} \right)}_{\substack{\text{model} \\ \text{output}}} = \underbrace{a_i}_{\substack{\text{model} \\ \text{parameter}}} + \underbrace{\eta_i^{\mathrm{A}}}_{\substack{\text{model} \\ \text{parameter}}} \sum_{j=M}^{N-1} \underbrace{x_{ij}}_{\substack{\text{model} \\ \text{output}}},$$

$$\underbrace{\beta_j \left(\sum_{i=0}^{M-1} x_{ij} \right)}_{\substack{\text{model} \\ \text{output}}} = \underbrace{b_j}_{\substack{\text{model} \\ \text{parameter}}} - \underbrace{\eta_j^{\mathrm{B}}}_{\substack{\text{model} \\ \text{parameter}}} \sum_{i=0}^{M-1} \underbrace{x_{ij}}_{\substack{\text{model} \\ \text{output}}},$$

which show that the market model parameters and output are linked. This linkage results in a calibration in which, improvement of one variable can deteriorate the score of the other variables.

We calibrate the model (2.36), such that the modelled trade flows and prices are as close as possible to realistic trade flows $\bar{x}_{ij}$, for $i \in S, j \in R \backslash S$, and $\bar{x}_{ii}$, for $i \in S$ and prices $\bar{\mathrm{p}}$. Recall from (2.5) that the parameters of the market model are the price slopes $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}$ and price levels $\boldsymbol{a}, \boldsymbol{b}$ which result into $2M + 2N$ parameters, which determine the inverse supply and demand. We will minimize the difference between model trade flows X and realistic trade flows $\bar{X}$. The same is done for model prices $\mathbf{p} := (\beta_j(m_j(x))_{j \in R}$ where β_j from (2.5b) and realistic prices $\bar{\mathbf{p}}$. Minimizing these differences results in a baseline scenario. Clearly, there is no guarantee that a perfect match between trade flows $(X; \bar{X})$ and prices $(\mathbf{p}; \bar{\mathbf{p}})$ exists, since it is unlikely that our model captures all dynamics of the market needed for perfect calibration. Whether or not a particular calibration is sufficiently accurate, depends on the use case. Given the challenges, we set the following objectives which our calibration method needs to fulfil:

- The calibrated price $\mathbf{p}$ will need to as close as possible to the observed price $\bar{\mathbf{p}}$.
- The LNG trade flows $\mathbf{x}$ will need to as close as possible to the observed trade flows $\bar{\mathbf{x}}$.
- The consumption in the export region (local trade) x_{ii} for $i \in S$ needs to be as close as possible to the observed data $\bar{x}_{ii}$ for $i \in S$.

For the equilibrium model presented in Chapter 2, the trade flows $\mathbf{x}$ and prices $\beta(\mathbf{x})$ in (2.5) depend on an unknown market parameter

$$\mathbf{y} := (\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}), \tag{3.7}$$

which corresponds to the variable $\mathbf{y}$ in the general bilevel optimisation (3.2). Here, we define f for the calibration of the market equilibrium model (2.36). The calibration will be based on time-averaged trade flows and prices listed on Bloomberg. In Table 3.1, we give the units and typical order of magnitudes for the entries of these variables. Considering these orders of magnitudes, the objective function f will be based on the non-dimensionalized variables,

$$\frac{x_{ij}}{\bar{x}_{ij}}, \qquad \text{and} \qquad \frac{\mathrm{p}_j}{\bar{\mathrm{p}}_j},$$

for $i \in S, j \in R$, such that all trade flow and price magnitudes are dimensionalized equally by the objective function in the calibration. The non-dimensional scores are all one when we have a perfect calibration result. We define a loss function ℓ_ρ based on non-dimensionalized quantities given by

$$\ell_\rho(\mathbf{z}, \bar{\mathbf{z}}) := \| \mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1} \|_\rho, \tag{3.8}$$

Variable Names	Mathematical Notation	Unit	Order of Magnitude
LNG trade flows	$\bar{x}_{ij}$ for $i \in S, j \in R \backslash S$	MWh/day	$[10^4, 10^6]$
Local trade flows	$\bar{x}_{ii}$ for $i \in S$	MWh/day	$[10^6, 10^7]$
Regional prices	$\bar{p}_j$ for $j \in R$	€/MWh	$[10^0, 10^2]$

Table 3.1: Overview of the order of magnitude of the observed trade flows and prices.

where $\mathbf{z}, \bar{\mathbf{z}} \in \mathbb{R}^s$ for $s := M + M(N-M) + N$ are defined as concatenated vectors given by

$$\mathbf{z} := \begin{bmatrix} \mathbf{x} \\ \mathbf{d}(\mathbf{x}, \mathbf{y}) \\ \beta(\mathbf{x}, \mathbf{y}) \end{bmatrix}, \qquad \text{and} \qquad \bar{\mathbf{z}} := \begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\mathbf{d}} \\ \bar{\mathbf{p}} \end{bmatrix}, \tag{3.9}$$

where $\mathbf{d}(\mathbf{x}, \mathbf{y}) := (x_{00}, ..., x_{MM})$ represents the local trade x_{ii} in $i \in S$ given by (2.16), which depends on $\mathbf{x}, \mathbf{y}$. Vector $\bar{\mathbf{d}} := (\bar{x}_{00}, ..., \bar{x}_{MM})$ is the corresponding data. Vector function $\beta(\mathbf{x}, \mathbf{y})$ is a vector of $\beta_j(m_j(X))$ from (2.5b) for $j \in R$, which depend on $\mathbf{y}$, which is a free variable during the calibration, thus written explicitly in (3.9). In Figure 3.1, we present an illustration which shows how the respective parts from X corresponding to $\mathbf{x}$ and $\mathbf{d}(\mathbf{x}, \mathbf{y})$ form the concatenation of vectors $\mathbf{z}$. The orange and blue circles represent the export and import regions, respectively. The

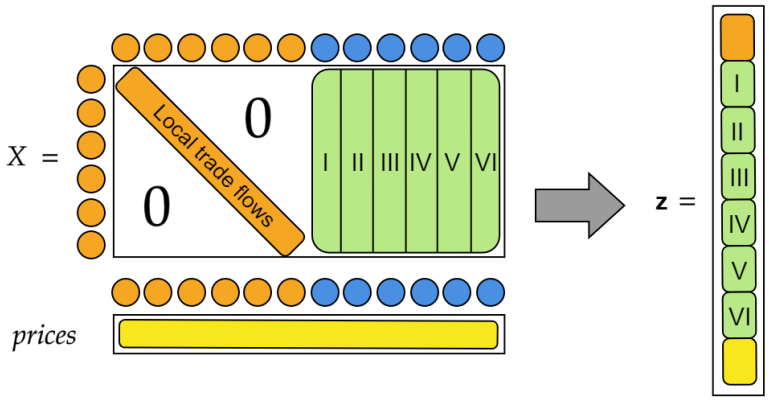


Figure 3.1: Illustrative example of the concatenated vector $\mathbf{z}$.

vector $\mathbb{1} := (1, ..., 1) \in \mathbb{R}^s$ represents the all ones vector and the operation $\mathbf{z} \circ \bar{\mathbf{z}} = (z_1 \cdot \bar{z}_1, ..., z_s \cdot \bar{z}_s)$ is a component-wise vector multiplication. A vector $\mathbf{z}^{-1}$ is defined as $\mathbf{z}^{-1} := \left(\frac{1}{z_1}, ..., \frac{1}{z_s}\right)$. The ρ-norm[2] is given by

$$\|\mathbf{z}\|_\rho := \left(\sum_{i=0}^{s} z_i^\rho \right)^{1/\rho}, \tag{3.10}$$

for $z \in \mathbb{R}^s$ and $\rho \in \mathbb{N}$. The loss function ℓ_ρ in (3.8) normalizes the difference $\mathbf{z} - \bar{\mathbf{z}}$ based on the data vector $\bar{\mathbf{z}}$ resulting in $\mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1}$. Then, we evaluate the ρ-norm of the scaled difference $\mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1}$, resulting in (3.8). Choosing $\rho = \infty$ corresponds to ℓ_ρ being the maximal non-dimensionalized difference.

Minimizing the maximum non-dimensionalized difference of $\mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1}$ will lead to a calibration solution where components z_i are all relatively close to their target $\bar{z}_i$. We define the upper level objective function f from (3.2a) as f_∞ given by

$$f_\infty(\mathbf{x}, \mathbf{y}) := \ell_\infty(\mathbf{z}, \bar{\mathbf{z}}) = \max \left\{ \left| \frac{x_{ii}}{\bar{x}_{ii}} - 1 \right|, \left| \frac{x_{ij}}{\bar{x}_{ij}} - 1 \right|, \left| \frac{\mathrm{p}_j}{\bar{\mathrm{p}}_j} - 1 \right| \ : \ i \in S, j \in R \backslash S \right\}, \tag{3.11}$$

which leads to an optimal solution for (3.2) for which the maximal scaled difference between the components of $\mathbf{z}$ and $\bar{\mathbf{z}}$ is minimized. However, we have to choose a numerical value for ρ

[2]This norm is commonly referred to as p-norm. However, as p is already used for price, we will use ρ instead.

because the gradient methods we use to solve the bilevel optimisation do not perform well with non-differentiable functions, like $\|.\|_\infty$. Choosing a relatively large value for ρ, such as $\rho = 30$, leads to an approximation of ℓ_∞. In Sections 4.1.1 and 4.2.1, we show that for the considered scenarios, this choice of ρ is sufficient, such that we approximate f_∞ by

$$f_{30}(\mathbf{x}, \mathbf{y}) \approx f_\infty(\mathbf{x}, \mathbf{y}). \tag{3.12}$$

3.2.2 Constraints of the Calibration

In this section, we define the constraints on the market parameters $\mathbf{y} = (\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b})$ given by (3.2b). The goal is to have a calibration result for realistic market behaviour. To illustrate undesirable market behaviour, if $\eta_0^{\mathrm{A}} := 0.0001$ and $\eta_1^{\mathrm{A}} := 1000$, then region 0 will raise its ask price by 0.0001 euro when they sell one MWh of LNG. By contrast, region 1 will raise its ask price by 1000 euro when they sell one MWh of LNG. This relatively large difference in η_0^{A} and η_1^{A} leads to selling behaviour of the regions which we deem to be unrealistic. In practice, export regions behave relatively similarly. We impose bounds on $\mathbf{y}$ to prevent an unrealistic solution from the calibration process. The constraints reduce the feasible region of the optimisation (3.2), and as a consequence the accuracy is likely worse than without these constraints. By achieving more realistic market behaviour, we likely give up accuracy on the calibration objective function f. Bounding $\mathbf{y}$ also reduces the search space and thus the solution techniques converge quicker when the upper and lower bounds are closer together. We will not define $\boldsymbol{g}$ from (3.2b) as a vector function, instead, we define several inequalities to model the constraints, which combined correspond to (3.2b).

First, we will define bounds for the price levels. For an accurate calibration, we ideally want the functions α_i, β_i from (2.5) to intersect at price $\bar{p}_i$ for $i \in S$. Thus, it is a natural choice to bound the price levels a_i and b_j around the observed price $\bar{p}_i$ where we add a lower bound on a_i and an upper bound on b_i with a scaling parameter $t \in \mathbb{R}$. We interpret factor t by how much the price levels (representing prices when nothing is bought or sold) can differ from the observed price. The constraints on components of $\boldsymbol{a}, \boldsymbol{b}$, are given by

$$(1-t) \cdot \bar{p} \le a_i \le \bar{p}_i \le b_i \le (1+t) \cdot \bar{p}_i, \qquad \text{for } i \in S, \tag{3.13a}$$

$$\bar{p}_j \le b_j \le (1+t) \cdot \bar{p}_j, \qquad \text{for } j \in R \backslash S. \tag{3.13b}$$

Combining the bounds (3.13) and the ask and bid functions from (2.5), we find bounds on the price slopes $\eta^{\mathrm{A}}, \eta^{\mathrm{B}}$. Starting with the ask functions α from (2.5a), our goal is to calibrate the model such that $\alpha_i \approx \bar{p}_i$. Let $\bar{n}_i$ be the observed quantity sold by region $i \in S$, then we find that using the lower bound from (3.13a) results in:

$$\bar{p}_i = \eta_i^{\mathrm{A}} \bar{n}_i + a_i \ge \eta_i^{\mathrm{A}} \bar{n}_i + (1-t) \cdot \bar{p}_i, \tag{3.14}$$

which can be rewritten as

$$\eta_i^{\mathrm{A}} \le \frac{t \bar{p}_i}{\bar{n}_i}.$$

The same can be done by applying $a_i \le \bar{p}_i$ such that

$$\bar{p}_i = \eta_i^{\mathrm{A}} \bar{n}_i + a_i \le \eta_i^{\mathrm{A}} \bar{n}_i + \bar{p}_i, \tag{3.15}$$

which can be rewritten as

$$\eta_i^{\mathrm{A}} \bar{n}_i \ge 0, \qquad \Rightarrow \qquad \eta_i^{\mathrm{A}} \ge 0, \quad \text{because} \quad \bar{n}_i \ge 0.$$

Thus, we find that

$$0 \le \eta_i^{\mathrm{A}} \le \frac{t \bar{p}_i}{\bar{n}_i}, \qquad \text{for } i \in S. \tag{3.16}$$

Similar to the ask function α_i, we can use (3.13) together with bid function β_j and m_j from (2.5b) and (2.4), respectively, such that we obtain the inequalities

$$\bar{p}_j = -\eta_j^{\mathrm{B}} \bar{m}_j + b_j \ge \eta_j^{\mathrm{B}} \bar{m}_j + (1+t) \cdot \bar{p}_j, \tag{3.17a}$$

$$\bar{p}_j = \eta_i^{\mathrm{A}} \bar{n}_j + a_j \le \eta_i^{\mathrm{A}} \bar{n}_j + \bar{p}_j, \tag{3.17b}$$

similar to (3.14) and (3.15). Rewriting the expressions in (3.17) and combining the results gives us

$$0 \leq \eta_j^{\mathrm{B}} \leq \frac{t\bar{p}_j}{\bar{m}_j}, \qquad \text{for } j \in R. \tag{3.18}$$

Combining the bounds from (3.13), (3.16) and (3.18) form the constraints g from (3.2b).

Next, we define the functions $\varphi_{\mathbf{y}}, \boldsymbol{\psi}_{\mathbf{y}}, \boldsymbol{\xi}_{\mathbf{y}}$ of the lower level optimisation from (3.2c), given by

$$\varphi_{\mathbf{y}}(\mathbf{x}) = \mathbf{x}^\intercal P_{\mathbf{y}} \mathbf{x} + \mathbf{c}_{\mathbf{y}}^\intercal \mathbf{x}, \tag{3.19}$$

$$\boldsymbol{\psi}_{\mathbf{y}}(\mathbf{x}) = D\mathbf{x} - \mathbf{v}_{\mathbf{y}}, \tag{3.20}$$

$$\boldsymbol{\xi}_{\mathbf{y}}(\mathbf{x}) = \mathbf{0}, \tag{3.21}$$

such that the lower level optimisation (3.2c) is equivalent to market equilibrium problem (2.36), where the coefficients $P_{\mathbf{y}}, \mathbf{c}_{\mathbf{y}}, \mathbf{v}_{\mathbf{y}}$ are the same as $P, \mathbf{c}, \mathbf{v}$ in (2.36), to make clear that their values depend on $\mathbf{y}$. The resulting bilevel optimisation is given by

$$\begin{aligned}
&\min_{\mathbf{x},\mathbf{y},\boldsymbol{\lambda}} \ \ell_\infty \left(\begin{bmatrix} \mathbf{x} \\ \mathbf{d}(\mathbf{x},\mathbf{y}) \\ \beta(\mathbf{x},\mathbf{y}) \end{bmatrix}, \begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\mathbf{d}} \\ \bar{\mathbf{p}} \end{bmatrix} \right), \\
&\text{subject to} \\
&(1-t)\cdot\bar{p} \leq a_i \leq \bar{p}_i \leq b_i \leq (1+t)\cdot\bar{p}_i, \quad \text{for } i \in S, \\
&\bar{p}_j \leq b_j \leq (1+t)\cdot\bar{p}_j, \quad \text{for } j \in R\backslash S, \\
&0 \leq \eta_i^{\mathrm{A}} \leq \frac{t\bar{p}_i}{\bar{n}_i}, \quad \text{for } i \in S, \\
&0 \leq \eta_j^{\mathrm{B}} \leq \frac{t\bar{p}_j}{\bar{m}_j}, \quad \text{for } j \in R, \\
&\max_{\mathbf{x}} \varphi_{\mathbf{y}}(\mathbf{x}), \\
&\quad \text{subject to} \\
&\quad D\mathbf{x} \leq \mathbf{v}_{\mathbf{y}}.
\end{aligned} \tag{3.22}$$

As we mentioned in Section 1.2.2, if P in (2.36) is negative semi-definite, then the objective function φ is concave. As a result, the KKT-conditions from (3.5), given by

$$\begin{aligned}
&\nabla_{\mathbf{x}} \mathcal{L}_{\mathbf{y}}(\mathbf{x},\mathbf{y},\boldsymbol{\lambda}) = \mathbf{0}, \\
&\mathbf{0} \leq \boldsymbol{\lambda} \perp D\mathbf{x} - \mathbf{v}_{\mathbf{y}} \leq \mathbf{0},
\end{aligned} \tag{3.23}$$

where

$$\mathcal{L}(\mathbf{x},\mathbf{y},\boldsymbol{\lambda}) = \mathbf{x}^\intercal P_{\mathbf{y}} \mathbf{x} + \mathbf{c}_{\mathbf{y}}^\intercal \mathbf{x} + \lambda^\intercal (D\mathbf{x} - \mathbf{v}_{\mathbf{y}}), \tag{3.24}$$

are used to reduce the bilevel optimisation problem (3.2) to a single level optimisation problem (3.25) by introducing the Lagrange multiplier $\boldsymbol{\lambda}$. In other words, we define the MPEC as an MPCC, under convexity assumption of the lower level optimisation problem. The single level optimisation problem, which we use for the calibration, is then given by

$$\begin{aligned}
&\min_{\mathbf{x},\mathbf{y},\boldsymbol{\lambda}} \ \ell_\infty \left(\begin{bmatrix} \mathbf{x} \\ \mathbf{d}(\mathbf{x},\mathbf{y}) \\ \beta(\mathbf{x},\mathbf{y}) \end{bmatrix}, \begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\mathbf{d}} \\ \bar{\mathbf{p}} \end{bmatrix} \right), \\
&\text{subject to} \\
&(1-t)\cdot\bar{p} \leq a_i \leq \bar{p}_i \leq b_i \leq (1+t)\cdot\bar{p}_i, \quad \text{for } i \in S, \\
&\bar{p}_j \leq b_j \leq (1+t)\cdot\bar{p}_j, \quad \text{for } j \in R\backslash S, \\
&0 \leq \eta_i^{\mathrm{A}} \leq \frac{t\bar{p}_i}{\bar{n}_i}, \quad \text{for } i \in S, \\
&0 \leq \eta_j^{\mathrm{B}} \leq \frac{t\bar{p}_j}{\bar{m}_j}, \quad \text{for } j \in R, \\
&\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x},\mathbf{y},\boldsymbol{\lambda}) = \mathbf{0}, \\
&\mathbf{0} \leq \boldsymbol{\lambda} \perp D\mathbf{x} - \mathbf{v}_{\mathbf{y}} \leq \mathbf{0},
\end{aligned} \tag{3.25}$$

where Lagrangian $\mathcal{L}$ is given by (3.24).

3.3 Accuracy Analysis of Calibration

In this section, we present an accuracy analysis of the calibration w.r.t. the number of regions. Bilevel optimisation problems, even the simplest versions, are NP-hard [15]. In Appendix A.1, we showcase the methodology of IPOPT, which finds a local solution to the optimisation problem (3.25) based on the interior point method. This method has a number of weaknesses, including the dependence on good initial solutions [58]. IPOPT has implemented measures to improve the convergence to a global optimum, but it cannot guarantee a global optimum. In Appendix A.2, we introduce several meta-heuristics based on evolutionary computing (EC) to improve the convergence to a globally optimal solution. Our description of ECs is based on [34].

We have performed an accuracy analysis for two meta-heuristics: multi-start local search (MSLS) and iterative local search (ILS), which are calibrated on a realistic scenario from summer 2022, which we describe in detail in Section 4.1.1. We tested the MSLS and ILS heuristics for 2×2, 3×3, ..., 6×6 regions, i.e., same number of export and import regions. Each meta-heuristic solves the problem (3.25) for a population of solutions (in our case with a population size of 600 solutions). Each solution has an associated objective value. In Figures 3.2 and 3.3, we present percentiles of the objective value of the MSLS heuristic and ILS heuristic, respectively. Note that the vertical axis is in logarithmic scale. The black line represents the 0th percentile of each problem size, i.e., the solution(s) in the population with the lowest objective value found by the respective meta-heuristic of each problem size. By connecting the 0th percentiles of each problem, we show how the objective value of the best solution(s) increases exponentially w.r.t. the number of regions. The orange line is the 100th percentile, i.e., the solution(s) with the highest objective value found by the meta-heuristic and shows the worst performing solution(s). We also plot the 25th, 50th and 75th percentiles, to show the general distribution of objective values in the population.

We observe that the 0th percentile for the ILS algorithm performs better than the MSLS algorithm. Thus, we will use the ILS algorithm to perform the calibration in our scenario analysis in Section 4. For the other percentiles (25th–100th) ILS performs similar to the MSLS. The large spread between the percentiles shows that both meta-heuristics have high variability of the upper level objective value within the population of solutions. This means that both will need a sufficiently large population size and long computing time to sufficiently cover the search space.

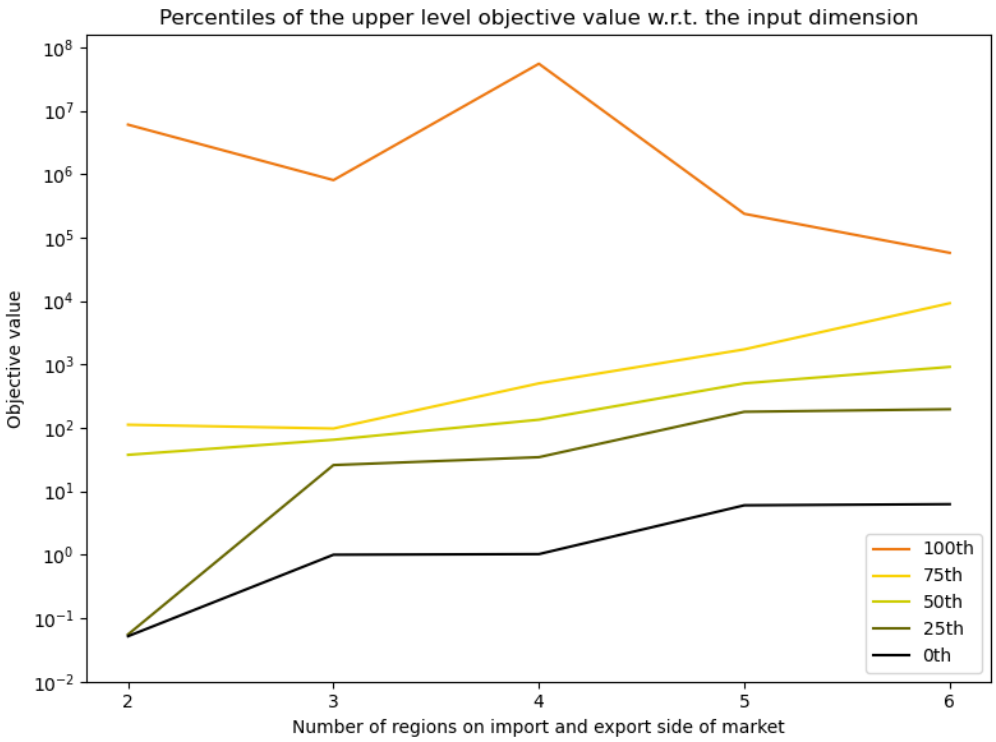


Figure 3.2: Performance of the multi-start local search.

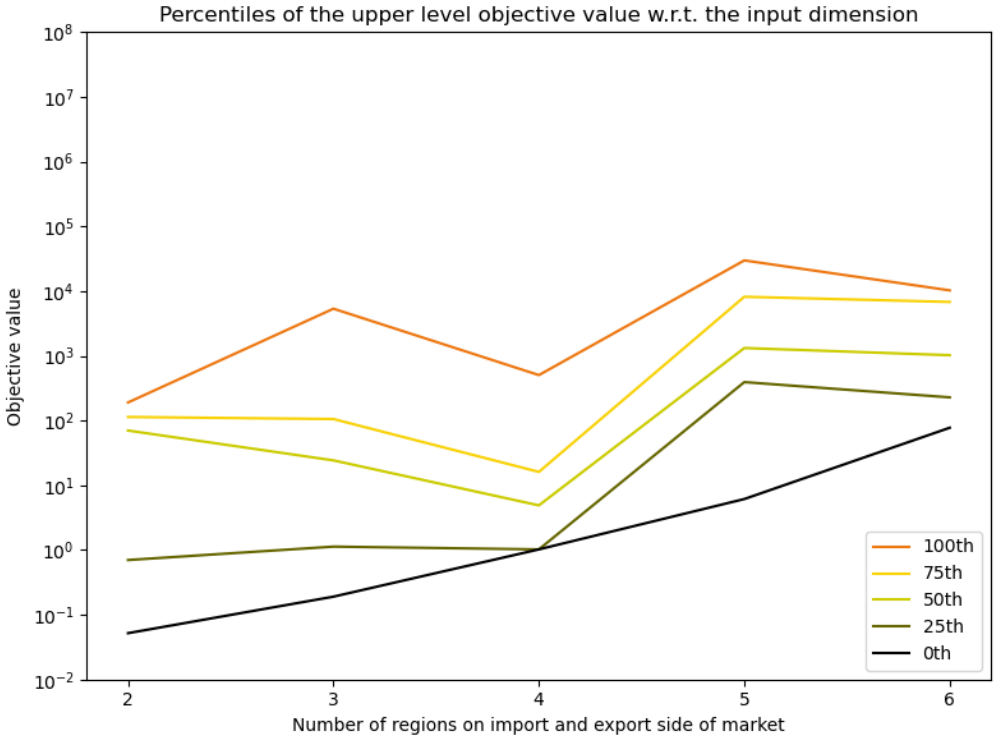


Figure 3.3: Performance of the iterative local search.

Chapter 4

Scenario Analysis

In this chapter, we will cover several realistic LNG transport scenarios based on events in 2022 and 2023 to illustrate how the model can be used as a scenario analysis tool, in line with the approach shown in Figure 1.1. For each study, we calibrate our baseline with market data applicable for the period of interest. Next, we perturb a parameter to simulate a hypothetical, but realistic scenario. The parameters which we perturb are: the inverse supply function α (by lowering η^{A}), the maximal global shipping capacity κ, the passage time $t^{(p)}$ through the Panama Canal, the distance matrix T and the port capacity C.

In Section 4.1, we study the Asian LNG market of summer 2022 to answer the question: What would the LNG price in Europe have been if China had not been in lockdown during the LNG price peak in Summer 2022? In Section 4.2, we study the effects of a drought in the Panama Canal during August 2023 [23]. In Section 4.3, we study the effects of reducing the Australian port capacity to limit export because there were speculations about labour strikes by Australian terminal workers [28].

4.1 Study of LNG market during Summer 2022

During the summer of 2022, LNG prices in Europe peaked [49] due to the reduction of pipeline NG from Russia to Europe [9]. As an alternative to pipeline NG, Europe imported LNG from abroad, which increased the demand on LNG ships significantly. At the same time, the Chinese LNG market imported around 25% less LNG from the spot market, compared to 2021. This decrease is illustrated by the green lines in Figure 4.1, where we see that the 70 days running average LNG imports drops from around 200,000 metric tonnes per day to around 150,000 metric tonne per day. The reduced LNG imports were partially caused by the strict Covid-19 lockdown in China, which led to reduced economic activities and hence less demand for LNG [5]. Our model indicates that if China imported similar quantities of LNG in 2022, compared to 2021, and the LNG shipping companies operate under peak capacity, then the European prices would have been approximately 5% higher.

In Section 4.1.1, we provide the calibrated baseline scenario of the LNG market of Summer 2022. During the calibration, we assume there was no global capacity on LNG trade. The baseline scenario is calibrated based on trade flow and price data from the period 1 June 2022 until 1 October 2022. In Section 4.1.2, we increase the North East Asian LNG consumption by decreasing the parameter $\eta^{\mathrm{B}}_{\mathrm{NEA}}$ such that the North East Asian import increases by around 33% [1]. This increase is not fully compensated by European or South American import, thus causing the utilized shipping capacity in this hypothetical scenario to increase by 26% w.r.t. the baseline. This increased utilized shipping capacity is unrealistic, because in Summer 2022, all shipping capacity was in use. Therefore, in Section 4.1.3, we put the maximal global shipping capacity κ at the utilized shipping capacity from the baseline.

[1] Note that the 33% increase precisely reverts the 25% drop in imported LNG.

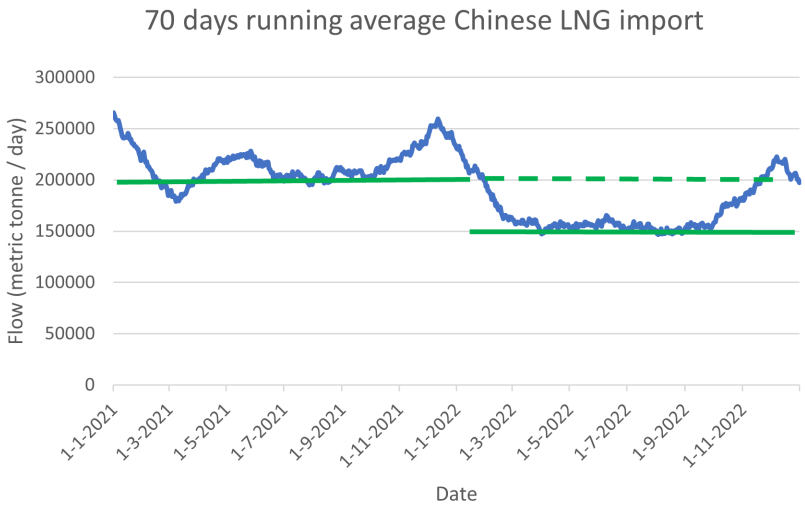


Figure 4.1: 70 days running average of Chinese LNG import from 2021 and 2022.

4.1.1 Baseline Scenario from Calibration

For the baseline scenario, we include North America (NA), the Middle East (ME), Australia (AU), Europe (EU), North East Asia (NEA) and South America (SA). The regions with a mature gas markets have a price benchmark. The price benchmark for Europe is based on the Title Transfer Facility (TTF), which is a virtual trading facility. In North East Asia, the price benchmark is based on the Japan/Korea Marker (JKM), which is a spot price index for the countries listed, but it is also relevant for Chinese and Taiwanese markets. The export prices of North America are based on the Henry Hub (HH) benchmark. We assume that other export regions such as Australia and the Middle East have a similar export price to HH, and that South American has a similar import price as TTF. In Tables 4.1 and 4.2, we provide the port capacity C_j, production limit L_i and global properties. The calibrated trade flows and prices, their relative differences from the market data, as well as the inverse demand and supply functions parameters are given in Tables 4.5, 4.4, 4.6 and 4.3, respectively. Worthwhile to mention for the baseline scenario is the objective value

$$f_{30}(\mathbf{x}, \mathbf{y}) \approx 0.1905,$$

where $f_{30}(\mathbf{x}, \mathbf{y})$ is given by (3.12). This means that the maximum absolute differences between calibrated prices and flows is approximately 19.05%. Recall that this was the maximum difference only if we had used $\rho = \infty$, which is in practice inconvenient to implement. We find that the maximum difference is 22.36%, which verifies that $\rho = 30$ is a sufficiently high power to approximate the infinity norm in (3.11) in this experiment. The relative differences in Tables 4.6, are given by

$$d(z, \bar{z}) := \frac{z - \bar{z}}{\bar{z}}, \tag{4.1}$$

where $z \in \mathbb{R}$ corresponds to the calibrated value point, and $\bar{z} \in \mathbb{R}$ corresponds to the observed data point. To measure the relative difference between the perturbation and the calibrated trade flows and prices in Tables 4.4, 4.8 and 4.10, we will also use (4.1), where z corresponds to the perturbed value point, and $\bar{z}$ corresponds to the calibrated value point.

	Type of Region -	Port Capacity MWh/day	Production Limit MWh/day
Australia (AU)	Export	31,826,800	2,531,507
North America (NA)	Export	35,978,120	31,917,808
Middle East (ME)	Export	27,952,230	1,852,055
Europe (EU)	Import	27,952,230	-
South America (SA)	Import	27,952,230	-
North East Asia (NEA)	Import	27,952,230	-

Table 4.1: Summer 2022 scenario: Regional Properties.

	Symbol	Value	Unit
Average speed ship	v_{avg}	19.5	knots
Transportation cost	c	0.00001	MWh/day/NM
Global trading capacity	κ	∞	MWh/day
Suez costs	$\pi^{(s)}$	0.1	euro/MWh/NM
Suez time	$t^{(s)}$	0.75	day(s)
Panama costs	$\pi^{(p)}$	0.05	euro/MWh/NM
Panama time	$t^{(p)}$	0.5	day(s)

Table 4.2: Summer 2022 scenario: Global Properties.

	$\boldsymbol{\eta}^{\text{A}}$ (€/MWh)/(MWh/day)	$\boldsymbol{a}$ €/MWh	$\boldsymbol{\eta}^{\text{B}}$ (€/MWh)/(MWh/day)	$\boldsymbol{b}$ €/MWh
AU	0.0000645	-58.44	0.0062235	1692.71
NA	0.0001039	-1642.79	0.0000788	1212.05
ME	0.0014996	-1686.54	0.0008264	731.27
EU	-	-	0.0002035	287.08
SA	-	-	0.0003590	154.81
NEA	-	-	0.0000900	290.05

Table 4.3: Summer 2022 scenario: Ask and bid parameters.

	Baseline $\mathbf{p}$ €/MWh	Relative diff. $d(\mathbf{p}, \bar{\mathbf{p}})$ -	Scenario 1 $\widetilde{\mathbf{p}}_1$ €/MWh	Relative diff. $d(\widetilde{\mathbf{p}}_1, \mathbf{p})$ -	Scenario 2 $\widetilde{\mathbf{p}}_2$ €/MWh	Relative diff. $d(\widetilde{\mathbf{p}}_2, \mathbf{p})$ -
AU	22.66	17.75%	36.04	59.02%	37.11	63.75%
NA	22.06	19.98%	30.68	39.08%	17.93	-18.72%
ME	30.46	-14.75%	38.61	26.77%	29.82	-2.10%
EU	153.01	15.96%	156.76	2.46%	161.06	5.26%
SA	92.88	22.36%	96.41	3.80%	99.47	7.10%
NEA	167.08	-15.15%	177.76	6.39%	193.54	15.83%

Table 4.4: Summer 2022 scenario: Baseline prices $\mathbf{p}$ with the relative difference compared to market data $\bar{\mathbf{p}}$ and perturbed prices $\widetilde{\mathbf{p}}_1, \widetilde{\mathbf{p}}_2$ with relative differences compared to the baseline $\mathbf{p}$.

4.1.2 Scenario 1: Increase North East Asian Demand

For the first scenario, we increase the NEA demand by decreasing $\eta^{\mathrm{B}}_{\mathrm{NEA}}$ by approximately 30% which achieves a North East Asian import which is approximately 33% higher compared to the baseline. Note that decreasing $\eta^{\mathrm{B}}_{\mathrm{NEA}}$ makes the slope of the inverse demand function, β_{NEA} in (2.5), flatter, which keeps prices higher, thus simulating aggressive bidding from NAE. This results in higher LNG imports by NEA. In Figure 4.2, we present an overview of the relative differences between the perturbation and the baseline. The orange and blue blocks represent the export and import regions, respectively. Variables n, m, p refer to the quantity sold in (2.3), quantity bought in (2.4) and the (bid) prices in (2.5b), respectively. The arrows represent the trade flows between regions. If a relative difference is larger than 2%, the arrow is coloured green or red depending if it is a relative increase or decrease. The full result of this perturbation on the trade flows, prices and the relative difference compared to the baseline are given in Table 4.7, 4.4, 4.8, respectively.

The 33% increased NEA imports is made up mostly from the trade flows AU-NEA and NA-NEA, where their market share increased by 21% and 103%, respectively. Note that NA is a large consumer, while AU is not. As a result, we observe that the overall sales have more significantly increased for the AU market than the NA market. Prices of the import regions remain relatively stable, with the largest percentage increase being 6% of the NEA price. Prices in export regions increase significantly, with the AU price showing the largest percentage increase of 59%. The utilized shipping capacity, which can be computed by the left-hand side of the global shipping capacity constraint (2.14), has increased by 26% compared to the baseline, which is an unrealistic increase for the market in 2022.

4.1.3 Scenario 2: Apply Global Constraint of 2022

During the summer of 2022, the market traded LNG close to the maximal global shipping capacity. We simulate this situation by defining the maximal global shipment capacity κ at the utilized shipping capacity of the baseline scenario, where we define κ as follows

$$\kappa := 47{,}343{,}902 \text{ MWh/day}.$$

This approach is chosen because estimating the shipping capacity from actual ship capacity (investigating each individual ship) is near-impossible, especially because it is not precisely clear which ship can be attributed to the spot market, because individual ships are not labelled as spot or contract delivery. The global shipping capacity constraint, from (2.14), will bound the perturbed market from Section 4.1.2, which simulated the increased NEA consumption. In Figure 4.3, we present an overview of the relative differences between the perturbation and the baseline. The detailed trade flows, prices and relative differences compared to the baseline are given in Tables 4.9, 4.4, 4.10, respectively.

We find that the NEA import increased only by 14% compared to 33% in the scenario without global shipping capacity constraint. Interestingly, the trade flow NA–NEA decreased by 29% compared to the baseline. This is counter-intuitive and opposite to the increase seen in the perturbation without global shipping capacity constraint (in Table 4.8). The trade route AU–NEA increased by 23%, with overall AU sales increasing by 18%, compared to 17% and 16% from the previous perturbation. The trade flow ME–EU significantly decreased with 19%, as the AU–NEA trade route becomes more profitable for LNG shipping companies.

EU prices increased by 5% which is the lowest price increase in the scenario, as NEA experiences a price increase of 16%. The NA price drops with 19%, while undergoing a minor decrease in their sales, and a minor increase in their local trade. The AU price increased with 64%, which is the largest price increase, with a slightly larger impact on the local trade flow. We see that despite the relatively large price changes, the consumption in all export regions remain stable.

Conclusion If China had not been in lockdown during 2022, and North East Asia imported LNG in similar quantities in 2021, then we would have seen price increase in Europe of 5% according to the presented model. North East Asia would have had the highest price increase (16%) from all the import regions, with the European price deviating the least (5%). The second scenario has

shown that the global shipping capacity constraint is a contributor to the increasing prices. If the 2022 LNG market was not trading close to the global shipping capacity, then an increase of 33% in NEA demand would have resulted only in 2% to 6% price increases in import regions, with Europe remaining the most stable.

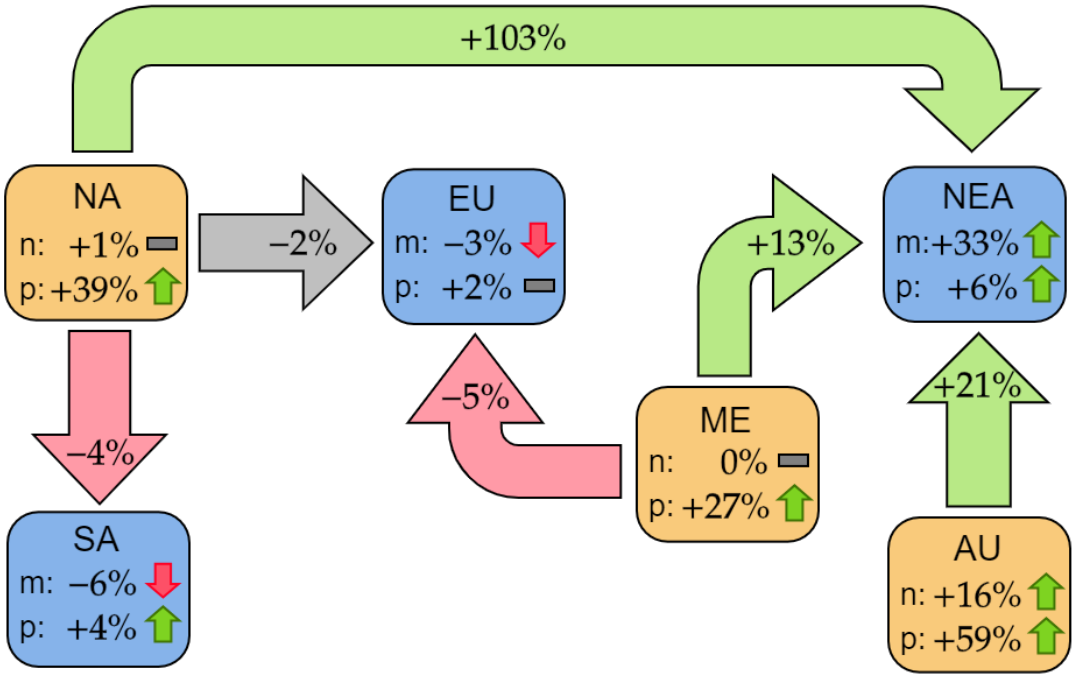


Figure 4.2: Summer 2022 scenario 1: Relative difference in prices, quantities bought and quantities sold compared to the baseline.

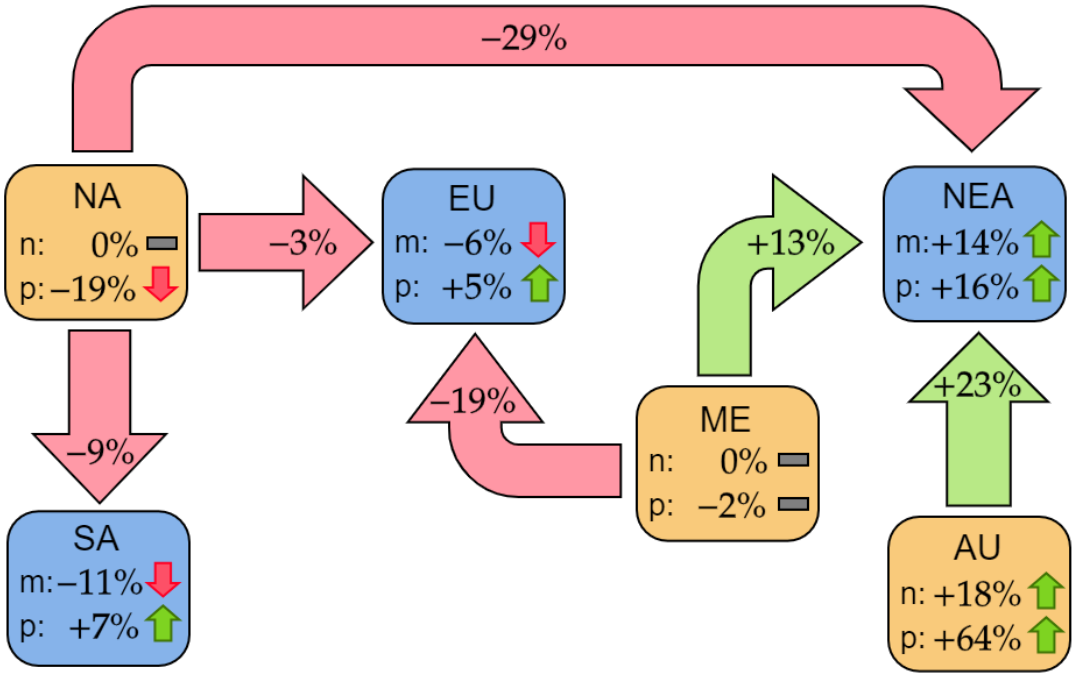


Figure 4.3: Summer 2022 scenario 2: Relative difference in prices, quantities bought and quantities sold compared to the baseline.

MWh/day	AU	NA	ME	EU	SA	NEA	Total Sold
AU	268,343	-	-	851	1,146	987,706	1,258,046
NA	-	15,107,068	-	535,103	170,193	204,737	16,017,101
ME	-	-	847,981	122,744	1,155	173,107	1,144,988
Total Bought	268,343	15,107,068	847,981	658,698	172,494	1,365,551	

Table 4.5: Summer 2022 baseline: Calibrated trade flows.

	AU	NA	ME	EU	SA	NEA	Total Sold
AU	11.30%	-	-	-	-	17.22%	16.09%
NA	-	10.84%	-	-16.71%	-15.99%	-15.66%	8.83%
ME	-	-	10.94%	19.06%	-	-17.79%	6.21%
Total Bought	11.30%	10.84%	10.94%	-11.65%	-15.99%	-16.23%	

Table 4.6: Summer 2022 baseline: Relative differences of the calibrated trade flows w.r.t. the data.

MWh/day	AU	NA	ME	EU	SA	NEA	Total Sold
AU	266,194	-	-	0	0	1,199,325	1,465,519
NA	-	14,997,641	-	523,998	162,659	415,730	16,100,028
ME	-	-	838,117	116,240	0	196,067	1,150,424
Total Bought	266,194	14,997,641	838,117	640,239	162,659	1,811,122	

Table 4.7: Summer 2022 scenario 1: Perturbed trade flows.

	AU	NA	ME	EU	SA	NEA	Total Sold
AU	-0.80%	-	-	-100.00%	-100.00%	21.43%	16.49%
NA	-	-0.72%	-	-2.08%	-4.43%	103.06%	0.52%
ME	-	-	-1.16%	-5.03%	-100.00%	13.26%	0.47%
Total Bought	-0.80%	-0.72%	-1.16%	-2.80%	-5.70%	32.63%	

Table 4.8: Summer 2022 scenario 1: Relative differences w.r.t. baseline trade flows.

MWh/day	AU	NA	ME	EU	SA	NEA	Total Sold
AU	266,022	-	-	0	0	1,216,137	1,482,159
NA	-	15,159,495	-	519,111	154,133	144,632	15,977,370
ME	-	-	848,754	100,030	0	195,777	1,144,561
Total Bought	266,022	15,159,495	848,754	619,141	154,133	1,556,546	

Table 4.9: Summer 2022 scenario 2: Perturbed trade flows.

	AU	NA	ME	EU	SA	NEA	Total Sold
AU	-0.87%	-	-	-100.00%	-100.00%	23.13%	17.81%
NA	-	0.35%	-	-2.99%	-9.44%	-29.36%	0.44%
ME	-	-	0.09%	-18.51%	-100.00%	13.10%	0.41%
Total Bought	-0.87%	0.35%	0.09%	-6.01%	-10.64%	13.99%	

Table 4.10: Summer 2022 scenario 2: Relative differences w.r.t. baseline trade flows.

4.2 Study of Impact Panama Canal Drought

During August of 2023, the Panama Canal faced problems related to drought of in-land lakes. These lakes are used to replenish the canal with water which is lost during the opening of water-gates [23]. As a consequence, the authorities decided to limit the number of passages a day from 36 to 32 passages [14]. During this time, the JKM (North East Asian) price benchmark increased by approximately 12%. Many ships were already on their way to the Panama Canal, thus a queue formed at the entrances. For these ships, the travel time from their departure to their destination suddenly increased, because of the waiting times outside of the canal.

In Section 4.2.1, we provide the baseline scenario, which corresponds to the market before the decision of the Panamanian authorities to reduce the throughput of the Panama Canal. We calibrate the model based on the trade flows and prices from 1 August 2023 until 31 August 2023. In Section 4.2.2, we perturb the baseline scenario by increasing the passage time of the Panama Canal. This scenario simulates the market situation after the reduction from 36 ships to 32 ships by the Panama authorities. In Section 4.2.3, we model the hypothetical closure of the Panama Canal.

4.2.1 Baseline Scenario from Calibration

For this study, we use the same regions (excluding South America), port capacities, production limits and global properties as in Section 4.1. The calibrated trade flows and prices, their relative differences from the market data, as well as the inverse demand and supply function parameters are given in Tables 4.13, 4.12, 4.14, and 4.11. The objective value of approximately 28.01% and the accuracy in Table 4.13 shows again that choosing $\rho = 30$ is sufficiently large to approximate the infinity norm. Similar as in Section 4.1, all relative differences are given by (4.1).

4.2.2 Scenario 1: Increase the Waiting Time

In August 2023, the number of ships which were allowed to move through the Panama Canal was reduced from 36 to 32 vessels per day. This reduction led to an average waiting time of 5-8 days. To mimic this waiting time, we define the passage time of the Panama canal to

$$t^{(p)} := 5 \text{ days}.$$

All other parameters remain equivalent to the baseline scenario. In Figure 4.4, we present an overview of the relative differences between the perturbation and the baseline. The detailed trade flows, prices and relative differences compared to the first perturbation are given in Tables 4.15, 4.12, 4.16.

In line with our expectation, increasing the passage time of the Panama Canal reduced the trade flow NA–NEA. However, a reduction of 95% is very significant. The LNG shipping companies increase their trade on trade routes AU–NEA and ME–NEA by 5% and 8%, respectively, according to our model. Still, North East Asian imports were reduced by 13% compared to the baseline scenario. All other trade flows remain relatively stable. The NEA price increased the most, which makes sense given the largest decline in NEA imports. Although the trade flows for AU and ME did not change much, their prices have increased by 9% and 8%, respectively. For EU and NA the prices remain stable.

4.2.3 Scenario 2: Close the Panama Canal

In this section, we model the hypothetical scenario that the Panama Canal has to close entirely in August 2023. For the regions we considered, this impacts the trade route NA–NEA, because we modelled the port of North America in the Gulf of Mexico, where Henry Hub is located. Thus, these ships now need to go around South America. We implemented this scenario by setting the Panama Canal usage parameter $\delta^{(p)}_{\text{NA,NEA}}$ to zero and we increased the distance of the trade route between North East Asia and America from 9,745 nautical miles to 14,211 nautical miles. All other distances and parameters will stay the same as the baseline scenario. In Figure 4.5, we present an overview of the relative differences between the perturbation and the baseline. The detailed trade

flows, prices and relative differences compared to the first perturbation are given in Tables 4.17, 4.12, 4.18.

In line with our expectations, by closing the Panama Canal, we find more impactful changes in the trade flows and prices compared to increasing the passage time of the Panama Canal to 5 days, as in Section 4.2.2. The trade flow NA–NEA (which was already quite low with the baseline and 5 days passage time scenarios) has changed to zero for the closed Panama Canal scenario. Again, LNG shipping companies move to the trade routes AU–NEA and ME–NEA with increases of 5% and 8%, respectively. Prices change similar to those found in Section 4.2.2, where the prices of AU, ME, NEA increase by 9%, 8% and 11%, respectively. Again, prices of EU and NA remain stable.

Conclusion Both perturbations show a decrease in NA–NEA trade flows, which is expected when the travel time on this route increases. The substitution of trade flows AU–NEA and ME–NEA is less obvious, but can be explained, by substitution of alternative trade flows. The increase of these flows led to price increases in Australia, the Middle East and North East Asia. The North American and European prices remained stable. It's noteworthy that both perturbations generated such similar trade flows and prices, which indicates that the passage time increasing to 5 days, has a similar effect to closing the Panama Canal.

It should be noted that the EU price relies heavily on the stability of the NA price. The ask parameter $\eta^{\mathrm{A}}_{\mathrm{NA}} = 0$, i.e., the constraint for $\eta^{\mathrm{A}}_{\mathrm{NA}}$ is at the boundary. As a consequence, the NA price is fixed. his means that, as long as it is profitable for the LNG shipping companies, the NA–EU trade flow can increase until the operation costs become too high. Thus, the NA–EU trade flow is able to meet EU demands, which keeps the EU price stable. This results is unrealistic.

	$\boldsymbol{\eta}^{\mathrm{A}}$	$\boldsymbol{a}$	$\boldsymbol{\eta}^{\mathrm{B}}$	$\boldsymbol{b}$
	(€/MWh)/(MWh/day)	€/MWh	(€/MWh)/(MWh/day)	€/MWh
AU	0.0000096	-7.16	0.0017640	507.33
NA	0.0000000	6.09	0.0000287	421.08
ME	0.0000808	-91.86	0.0004911	451.17
EU	-	-	0.0000630	90.27
NEA	-	-	0.0000146	52.57

Table 4.11: Panama drought baseline: Ask and bid parameters.

	Baseline $\mathbf{p}$	Relative diff. $d(\mathbf{p}, \bar{\mathbf{p}})$	Scenario 1 $\widetilde{\mathbf{p}}_1$	Relative diff. $d(\widetilde{\mathbf{p}}_1, \mathbf{p})$	Scenario 2 $\widetilde{\mathbf{p}}_2$	Relative diff. $d(\widetilde{\mathbf{p}}_2, \mathbf{p})$
	€/MWh	-	€/MWh	-	€/MWh	-
AU	5.88	-24.82%	6.38	8.58%	6.40	8.98%
NA	6.09	-24.82%	6.09	0.00%	6.09	0.00%
ME	9.41	23.85%	10.13	7.62%	10.16	7.96%
EU	42.56	26.55%	42.76	0.45%	42.76	0.47%
NEA	29.82	-21.14%	32.88	10.25%	33.03	10.76%

Table 4.12: Panama drought baseline: Baseline prices $\mathbf{p}$ with the relative difference compared to market data $\bar{\mathbf{p}}$ and perturbed prices $\widetilde{\mathbf{p}}_1, \widetilde{\mathbf{p}}_2$ with relative differences compared to the baseline $\mathbf{p}$.

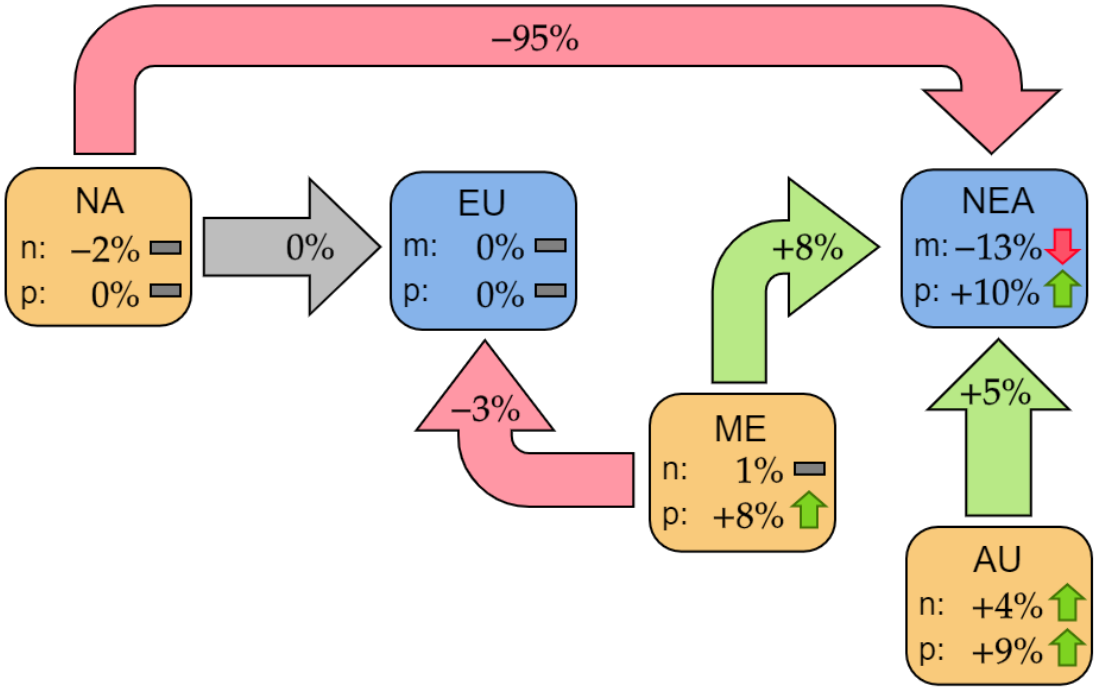


Figure 4.4: Panama drought scenario 1: Relative difference in prices, quantities bought and quantities sold compared to the baseline.

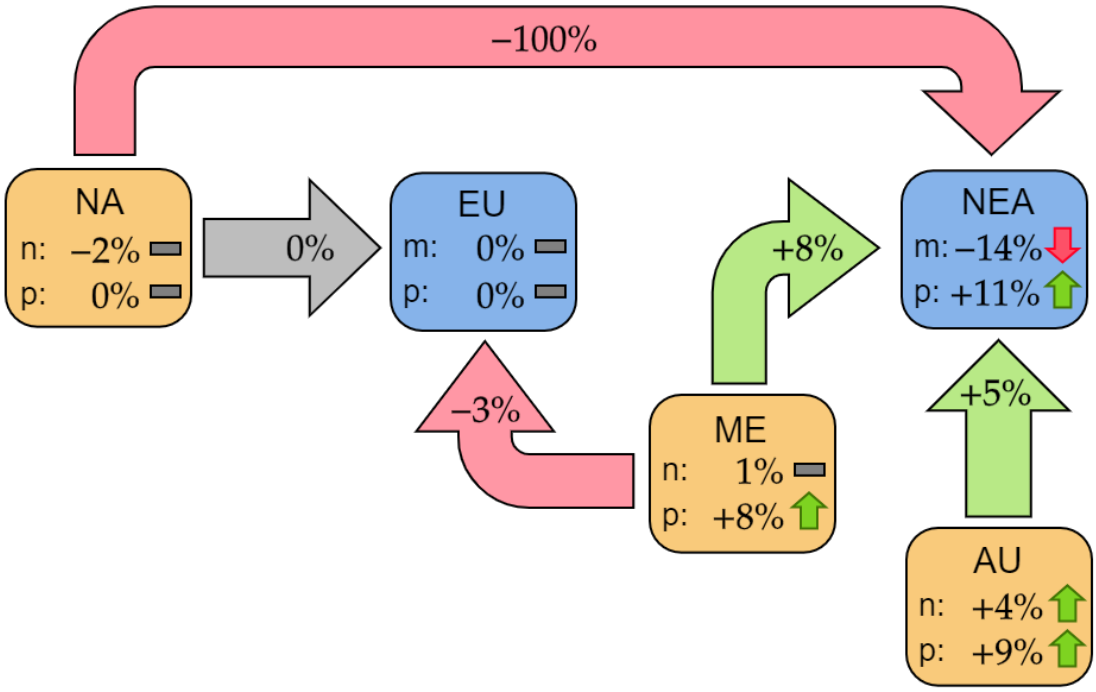


Figure 4.5: Panama drought scenario 2: Relative difference in prices, quantities bought and quantities sold compared to the baseline.

MWh/day	AU	NA	ME	EU	NEA	Total Sold
AU	284,271	-	-	772	1,069,534	1,354,577
NA	-	14,475,992	-	597,624	290,680	15,364,296
ME	-	-	899,453	159,449	193,954	1,252,856
Total Bought	284,271	14,475,992	899,453	757,845	1,554,168	

Table 4.13: Panama drought baseline: Calibrated trade flows.

	AU	NA	ME	EU	NEA	Total Sold
AU	17.91%	-	-	-	26.17%	24.41%
NA	-	6.21%	-	-25.50%	-25.80%	3.64%
ME	-	-	17.67%	27.36%	-26.61%	8.58%
Total Bought	17.91%	6.21%	17.67%	-18.28%	-26.23%	

Table 4.14: Panama drought baseline: Relative differences of the calibrated trade flows w.r.t. the data

MWh/day	AU	NA	ME	EU	NEA	Total Sold
AU	283,986	-	-	0	1,122,961	1,406,946
NA	-	14,475,993	-	599,956	13,497	15,089,446
ME	-	-	897,992	154,842	208,898	1,261,733
Total Bought	283,986	14,475,993	897,992	754,798	1,345,356	

Table 4.15: Panama drought scenario 1: Perturbed trade flows.

	AU	NA	ME	EU	NEA	Total Sold
AU	-0.10%	-	-	-100.00%	5.00%	3.87%
NA	-	0.00%	-	0.39%	-95.36%	-1.79%
ME	-	-	-0.16%	-2.89%	7.71%	0.71%
Total Bought	-0.10%	0.00%	-0.16%	-0.40%	-13.44%	

Table 4.16: Panama drought scenario 1: Relative differences w.r.t. baseline trade flows.

MWh/day	AU	NA	ME	EU	NEA	Total Sold
AU	283,972	-	-	0	1,125,438	1,409,410
NA	-	14,475,993	-	600,049	0	15,076,042
ME	-	-	897,927	154,631	209,571	1,262,129
Total Bought	283,972	14,475,993	897,927	754,680	1,335,009	

Table 4.17: Panama drought scenario 2: Perturbed trade flows.

	AU	NA	ME	EU	NEA	Total Sold
AU	-0.11%	-	-	-100.00%	5.23%	4.05%
NA	-	0.00%	-	0.41%	-100.00%	-1.88%
ME	-	-	-0.17%	-3.02%	8.05%	0.74%
Total Bought	-0.11%	0.00%	-0.17%	-0.42%	-14.10%	

Table 4.18: Panama drought scenario 2: Relative differences w.r.t. baseline trade flows.

4.3 Study of Impact Australian Labour Strike

During August 2023, the same time frame as the Panama drought study in Section 4.2, labour unions threatened to organise labour strikes at the AU LNG terminals. Intuitively, such a labour strike woud have impacted the trade route AU–NEA the most, but the reduced global supply could have had impact beyond this specific route. In August 2023, the NEA price increased approximately by 12% and the EU price increased by 22% on 9 August 2023 [7]. Later on, such an event has occurred again on 3 September 2023, this time the EU price increased insignificantly [62]. We will investigate the impact of the labour strike, and consider if the increased prices were an overreaction, or that they followed the new market equilibrium. This study assumes the standard Panama passage time of 0.5 days.

We use the calibrated model from Section 4.2.1, which was based on trade flows and price data from the period 1 August 2023 until 31 August 2023. In Section 4.3.1, we perturb the AU port capacity to mimic a labour strike occurring in AU LNG terminals. We show that although the make up of the NEA import changes significantly, the EU and NEA prices will remain stable.

4.3.1 Scenario: Reduction of Port Capacity

Two major Australian companies have been in talks with Australian Labour unions, who threatened with labour strikes if work conditions would not improve. These companies are responsible for half of Australia's export [28]. During this time, Australia traded most of the LNG to Asia. However, the instability caused by the possibility of cutting Australian exports in half, resulted in a 12% price increase in NEA and a 22% price increase in EU. The price increase in EU is noteworthy, because EU does not directly depend on AU exports. At the time, it was unclear when the strike would happen and if there would be global supply issues. To model the labour strike, we reduce the port capacity of Australia by half of the exported value of the baseline, corresponding to the fact that the Australian companies which are dealing with this strike, are responsible for half of AU export. The quantity exported by Australia $E_{\mathrm{AU}} = 1{,}070{,}306$ MWh/day is computed from the baseline scenario in Section 4.2.1. We set the Australian port capacity C_{AU} to 50% of the export capacity E_{AU} from the baseline, such that C_{AU} is defined as

$$C_{\mathrm{AU}} := \frac{1}{2} \cdot E_{\mathrm{AU}}.$$

In Figure 4.6, we present an overview of the relative differences between the perturbation and the baseline. The resulting trade flows, prices and relative differences compared to the baseline are given in Tables 4.20, 4.19, 4.21.

The Asian import remains stable, but distribution of trade flows change significantly. Firstly, we see that overall Australian sales reduced by 40%, mostly caused by the export on trade flow AU–NEA which reduced by 50% as expected. Secondly, the NA–NEA and ME–NEA trade flows increased by 168% and 17%, respectively, to compensate for the loss on the trade route AU–NEA. As a consequence, the trade flow ME–EU decreased by 6% .

The Australian price has decreased significantly by 87%. The Middle Eastern price has increased by 17%, while the North American price remains stable. The prices from the import regions remain stable as well, which is especially interesting for North East Asia, because the distribution of their import has changed significantly.

Conclusion This scenario analysis indicates that in August the market overreacted with the threat of on-going global supply shortage. The market response in September is much more in line with our scenario analysis which shows stable prices for Europe. Again, It should be noted that the EU price relies heavily on the stability of the NA price, which is caused by the unrealistic calibration of $\eta^{\mathrm{A}}_{\mathrm{NA}} = 0$.

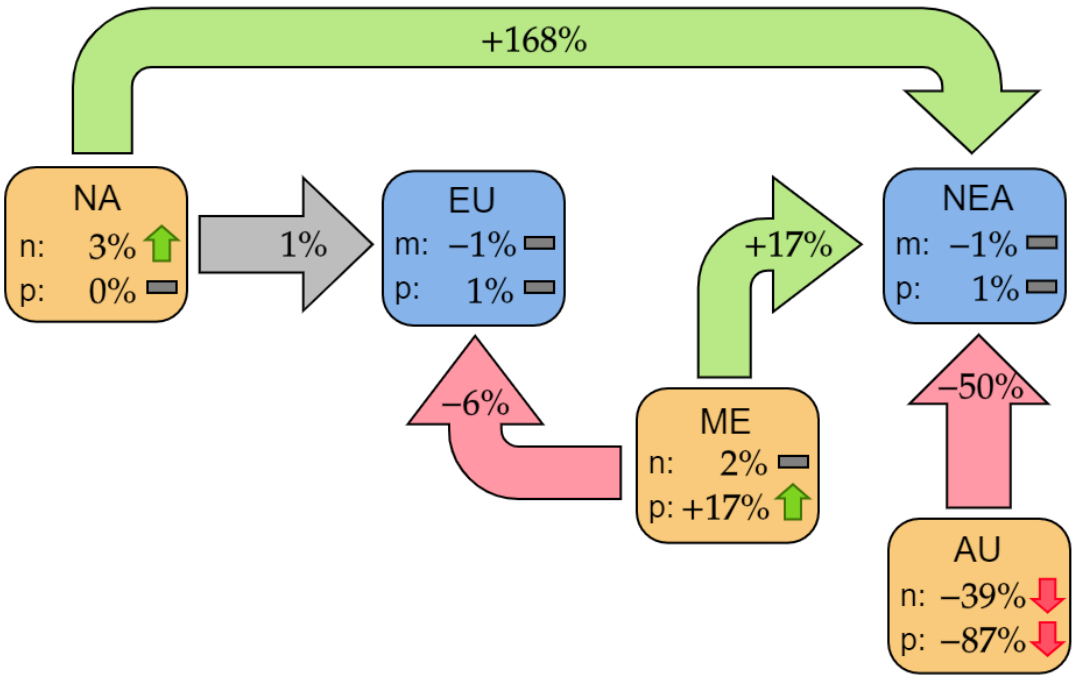


Figure 4.6: Labour strike scenario: Relative difference in prices, quantities bought and quantities sold compared to the baseline.

	Baseline $\mathbf{p}$ €/MWh	Scenario 1 $\widetilde{\mathbf{p}}$ €/MWh	Relative diff. $d(\widetilde{\mathbf{p}}, \mathbf{p})$ -
AU	5.88	0.75	-87.18%
NA	6.09	6.09	0.00%
ME	9.41	10.98	16.60%
EU	42.56	42.95	0.91%
NEA	29.82	30.02	0.68%

Table 4.19: Labour strike scenario: Baseline prices $\mathbf{p}$ and perturbed prices $\widetilde{\mathbf{p}}$ with relative differences compared to the baseline $\mathbf{p}$.

MWh/day	AU	NA	ME	EU	NEA	Total Sold
AU	287,175	-	-	-	535,153	822,328
NA	-	14,475,991	-	602,420	778,498	15,856,909
ME	-	-	896,272	149,272	226,645	1,272,189
Total Bought	287,175	14,475,991	896,272	751,692	1,540,296	

Table 4.20: Labour strike scenario: Perturbed trade flows.

	AU	NA	ME	EU	NEA	Total Sold
AU	1.02%	-	-	-100.00%	-49.96%	-39.29%
NA	-	0.00%	-	0.80%	167.82%	3.21%
ME	-	-	-0.35%	-6.38%	16.85%	1.54%
Total Bought	1.02%	0.00%	-0.35%	-0.81%	-0.89%	

Table 4.21: Labour strike scenario: Relative differences w.r.t. baseline trade flows.

Chapter 5

Conclusion and Discussion

In this final chapter, we discuss our findings, provide a conclusion and give an outlook for further research. In Section 5.1, we summarize our conclusions related to the market equilibrium model, the calibration process and scenario analysis. In Section 5.2, we present the discussion, in which we discuss the results from the model design, the algorithmic choices, scenario analysis. In Section 5.3, we present possible extensions to the model and to the optimisation algorithm used to calibrate.

5.1 Conclusion

We have developed a unique monopolistic market equilibrium model of the LNG spot market to model interaction between supply and demand, which assumes a strong market position of the LNG shipping companies in times of high demand. The market equilibrium is found by solving the quadratic optimisation problem (2.36). If matrix P is negative semi-definite, then the optimisation problem is convex and `quadprog` can efficiently return the optimal solution. However, we do not have such a convexity guarantee, but we did not discover any indefinite matrix P based on realistic data during our scenario analysis. Hence, the optimization problem appears to be convex in practice. To prove that P is always negative semi-definite is made challenging because of time-scale matrix Θ which can theoretically scale elements of P to any size, resulting in P becoming negative semi-definite.

The solver `quadprog` can efficiently solve optimisation problems with more regions than we calibrated for during the scenario analysis. The main limitations of our model is the calibration process which is inaccurate for larger scenarios as shown in Figures 3.2 and 3.3. As a result, we can only perturb with a small number of regions during the scenario analysis for which we have calibrated $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$.

The calibration challenge stems from the non-convexity of MPCCs. Additionally, the introduction of Lagrange multipliers and slack variables in (A.3) significantly increases the computing time of IPOPT and increases the number of solutions for which we find a unique calibration score, i.e., more distinct local optima. The ILS has improved finding a stationary point with sufficient accuracy for scenarios analysis with low number of regions. The runtime of the ILS depends of the number of local optima found by IPOPT. Since, more distinct solutions require a longer search to ensure we have explored the whole search space to converge to a sufficiently accurate solution. Increasing the scenario by one import and one export region results in the number of variables increasing quadratically, while the degrees of freedom increase linearly. When the number of variables outnumbers the degrees of freedom, we observe that the calibration accuracy decreases.

We have shown that our model can be used to study (hypothetical) scenario outcomes via parameter perturbations. The solutions for the summer 2022 scenarios are more realistic than the August 2023 scenarios because for the latter scenarios the parameter $\eta^{\mathrm{A}}_{\mathrm{NEA}}$ was at the boundary, which kept the prices unrealistically stable. This means that the calibration is artificially limited. By model design, the calibration works best if the LNG shipping companies have a lot of market power (close to monopolistic) due to high demand of LNG transport, which leads to high profits for LNG shipping companies. As such, the calibration based on the dataset from 2022 had better

accuracy with a maximum relative difference of 19%. The dataset from August 2023 has lower prices and the price differentials are smaller, which is crucial for our objective function because the price differentials will determine the potential profits. Subsequently, we observe a lower accuracy with a maximum relative difference of 26% for a baseline scenario with five regions.

Lastly, our model only outputs steady state equilibriums. This is by design, but it does limit the usability of the model for scenario analysis because the impact of speculation effects, such as the European price spike in August 2023 [7], which cannot be modelled realistically, as they can have a lot of short term impact.

5.2 Discussion

In this section, we discuss the results and shortcomings of the model design, the algorithmic choices. We will cover the following topics regarding the model design, firstly, we discuss the decrease in calibration accuracy when the number of region increases. Secondly. we discuss the high variability of the ask and bid parameters for different market scenarios. Lastly, we discuss the difference between the calibration accuracy of the summer 2022 and August 2023 scenarios. Regarding the algorithmic choices and implementation, firstly, we discuss the long runtime of the ILS algorithm to calibrate the model to a desired accuracy. Secondly, we present the possibility for a speed-up of the calibration process by excluding market equilibrium constraints $D\mathbf{x} \leq \mathbf{v}$.

Discussion of model design We have been unable to accurately calibrate the model for more regions during the scenario analysis. The solution found is accurate ($f < 25\%$) for low dimension sizes, e.g., up to three export and three import regions. When we increase the number of regions it becomes exponentially more difficult to calibrate correctly, as shown by the increasing 0th percentile in Figure 3.3. Note that the number of trade flows and prices $x_{ii}, x_{ij}, \mathrm{p}_j$ for $i \in S, j \in R \backslash S$ which need to be matched as closely as possible to realistic data are given by $M(N-M)+M+N$ number of variables. When M and N are increased simultaneously, the increase in the number of variables in the model output is quadratic. On the contrary, the number of parameters $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$ are given by $2M+2N$ variables, which is linear when M and N are increased simultaneously. This means that increasing the model by import and export regions increases the data that needs to be fitted more quickly, than the increase in degrees of freedom.

We also observe that the calibrated parameters $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$ are unstable for increasing number of regions. For example, assume we have calibrated a baseline scenario including North America (NA) and the Middle East (ME) as export regions and Europe (EU) and North East Asia (NEA) as import regions. We have experienced that the addition of Australia (AU) as a export region significantly changes the value of all parameters $\boldsymbol{\eta}^{\mathrm{A}}, \boldsymbol{\eta}^{\mathrm{B}}, \boldsymbol{a}, \boldsymbol{b}$ compared to the calibration with only four regions. It is logical that a LNG market without AU as export region would indeed behave different in reality. However, it does prevent us to scale down the problem to obtain ask and bid parameters which are still usable for larger scenarios.

During the scenario analysis, we observed that the maximum difference for summer 2022 with six regions in Section 4.1.1 was 19%, which is considerably better than 26% maximum difference for August 2023 with five regions in Section 4.2.1. We attribute this to the 2022 data fitting our model assumptions better, in particular the large price differential between import and export leads to high profit margins for the LNG shipping companies in our model. In summer 2022, the LNG shipping companies did have more market power compared to August 2023, thereby fitting our monopolistic assumption of the LNG shipping companies better.

Discussion of algorithmic choises To improve the performance of IPOPT, we have implemented three meta-heuristics to improve the convergence to the global optimum. All methods help to find a accurate stationary point. However, these meta-heuristics are relatively slow with a computing time being in the order of hours. Naturally, this depends on the settings of the user, such as the termination criterion. Depending on the market scenario, we observe that the number of unique local optima can vary significantly. The higher the diversity in the populations, the longer the meta-heuristic should run. Next, we discuss our experience with a possible speed-up.

Under assumption of an accurate calibration process, the calibrated flows will be relatively close to the data such that when the data trade flows $\bar{\mathbf{x}}$ are in the interior of the feasible region, i.e., $D\bar{\mathbf{x}} < \mathbf{v}$, then the calibrated flows $\mathbf{x}$ are likely also in the interior of the feasible region. In other words, when the data does not violate any constraints, then the calibrated flows will (likely) not violate any constraints either. This is the case during our scenario analysis calibration in Section 4. Note that we do not deny the existence of such boundary, we only observe that the calibrated flows $\mathbf{x}$ do not violate the constraints when they are close to the data. Under the assumption of accurate calibration, it is beneficial to remove the constraints from the lower level optimisation problem, such that the optimisation (3.25) is simplified by the removal of unviolated constraint $D\mathbf{x} \leq \mathbf{v}$ and the removal of the corresponding multipliers $\boldsymbol{\lambda}$. This reduction in variables leads to a speed-up in computing time, from order of hours to order of minutes, and generally finds more accurate stationary points.

It should be noted that reducing $D\mathbf{x} \leq \mathbf{v}$, also removes part of the slack variables given by $\mathbf{s}$, which are needed to transform the inequality constraint to an equality constraint in (A.4). With the inclusion of constraints $D\mathbf{x} \leq \mathbf{v}$, a significant portion of the initial solution did not converge to a stationary point. On the contrary, without $D\mathbf{x} \leq \mathbf{v}$, the majority of initial solutions did converge. This means that we can reduce the termination criterion of the meta-heuristic which reduces the runtime of the iterative local search algorithm.

Unfortunately, this still does not allow for accurate calibration of market scenarios which use larger number of regions. We believe the inaccuracy of the model is not caused by the solver missing the global optimum, but rather that our model does not allow for enough degrees of freedom to fit to realistic data. This idea is based on the calibration process mimicking a system of equations when the variables $\mathbf{z}$ match the data $\bar{\mathbf{z}}$ perfectly. A detailed explanation is given in Appendix B.3.

Efforts have been made to use a GLS algorithm which aims to mix solutions via a crossover operator to find better 'offspring' initial points. These initial points are used to find new solutions. This approach is however unsuccessful in reducing the computation time needed to find a stationary point which is accurate enough. We believe this is caused by the fact that the solutions $\mathbf{z}$ do not have a 'block' structure, where certain components of the "parent '' solutions can be combined to form "offspring" solutions which are an improvement compared to the parent solutions. Without such structure, using a GLS algorithm will likely not lead to an improvement compared to an ILS algorithm. Although we have done our best efforts to tune the GLS algorithm, a more sophisticated tuning, or even alternative GLS algorithm, could find structures in the solutions which we were unsuccessful to exploit.

5.3 Further Research

In this section, we give an overview of possible extensions, with the focus on suggestions based on our experiences with the usage of this model.

Least Squares with regularization We explored the use of the least-squares objective function with regularization $\|\mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1}\|_2^2 + \|\mathbf{y}\|_2^2$ to perform the calibration process. This method is mathematically simple and efficient to solve because least-squares problems are convex [17]. This objective function with regularization and the exclusion of a majority of the constraints $D\mathbf{x} \leq \mathbf{v}$, as mentioned before, creates an optimisation problem which is in line with the standard optimisation problem designed to be solved by IPOPT, i.e., only simple upper and lower bounds on variables $\mathbf{x}, \mathbf{y}$ and no inequality constraints. In Appendix B.3, we present this calibration on the dataset from summer 2022. We believe that this method is promising, because the objective function is simple, and the inclusion of a regularisation could have positive effects on the convergence. However, this method led to a handful of outliers in the calibrated trade flows which were significantly off. Understanding under which conditions the least squares approach is accurate (without outliers) would make this simplification useful to apply in practice.

Outlook on implementation We have seen that reducing the constraints $D\mathbf{x} \leq \mathbf{v}$ during the calibration process leads to less variability in the calibration scores associated to each solution in

the population. This means that we can use a smaller population for the meta-heuristic. Resulting in reduced computer time necessary to find a sufficiently low accuracy. We propose the following strategy. Firstly, calibrate the single level optimisation (3.25) without constraints $D\mathbf{x} \leq \mathbf{v}$ via MSLS to find $\mathbf{x}$ where the constraints are reduced to take advantage of the reduced computer time. Then check if any of the constraints by $D\mathbf{x} \leq \mathbf{v}$ are violated. If not, then no action is necessary and the solution can be used for scenario analysis. If any of the constraints are violated, then solve the single level optimisation problem (3.25) (where the constraints $D\mathbf{x} \leq \mathbf{x}$ are included) via ILS.

Alternative flows between two regions In our model, we have modelled a single trade route between regions, where we assume that all transportation moves through the same canal, or avoids a canal. For example, we have the trade route ME–EU where we assume that all transport uses the Suez Canal. However, in reality there is an alternative to go around Africa instead. Additionally, we can include an average waiting time at the LNG ports before the LNG is offloaded. Our current model indicates when trade flows change to a different destination, but not if an alternative route for the same destination is used. This would be especially relevant if the ships have to wait before they can offload. Implementing multiple routes between a pair of regions together with waiting times at the ports could answer questions, such as: 'How much LNG is transported around Africa instead of using the Suez Canal between ME and EU when the EU port waiting time has increased by 20% in Europe?'. Ships will have to wait at the destination port, so they would rather avoid the Suez Canal, go around Africa to make up the time and save on the costs from Suez Canal. Due to the rising tensions in the Middle East from the Israel-Hamas conflict, another question which is relevant is: 'How much LNG is transported around Africa instead of using the Suez Canal between ME and EU when the Suez Canal passage time has doubled?'.

Improving Calibration of the Market Equilibrium Model We shared our interpretation of the calibration as a system of equations in Section 5.2. To extend the model to be able to calibrate accurately for more regions, we suggest to extend the degrees of freedom of the model. Our accuracy analysis in Section 3.3, shows that our current model cannot calibrate for scenarios where the number of regions is larger than 3 import and 3 export regions.

Additionally, some fixed parameters can be replaced by free variables during the calibration process, such as the transportation cost c. We experimented with this technique because as far as we know, no data on parameter c is not published. This additional (free) variable in the optimisation will allow c to be calibrated based on the trade flow and price data. Bounds on c are introduced to keep the resulting multiplication cT_{ij}, for $i \in S, j \in R \backslash S$, realistic. However, from experience, the parameter c is at the boundary and thus is artificially limited. Note that relaxing the bound would make the parameter unrealistic. Hence, this method does not work well in practice. Investigating which fixed parameters can be transformed to free variables would be the next step.

CO2 Certificate In 2023, the European Union has introduced the CO_2 certificates which will be used in the future for maritime transportation. A natural question with regards to our model is: How does this new legislation impact the LNG market? Can it already be incorporated in our model? We would incorporate this as follows. Currently, from the perspective of the ship companies, little would change because they already operate on boil-off LNG (evaporated LNG from the load the ship transports). Note that LNG has less CO_2 emissions than crude oils used by other large vessels. When there is a noticeable difference in CO_2 emission between vessels used on specific trade routes, then is it relevant to incorporate an inefficiency cost to trade routes which have more CO_2 emissions on average. Such costs can be incorporated into the transportation costs τ_{ij} for $i \in S, j \in R \backslash S$.

Bibliography

[1] The role of gas in today's energy transitions. *U.S. Energy Information Administration*, July 2019. Visited on 5 November 2023.

[2] Explainer: Should europe use more long term lng contracts? *Reuters*, 7 February 2022. Visited on 5 November 2023.

[3] Natural gas explained. *U.S. Energy Information Administration*, 7 November 2022. Visited on 20 October 2023.

[4] Run op lng schepen: 'we komen straks echt te kort'. *BNR Webredactie*, 6 March 2022. Visited on 21 April 2023.

[5] China's natural gas consumption and lng imports declined in 2022, amid zero-covid policies. *U.S. Energy Information Administration*, 1 June 2023. Visited on 20 October 2023.

[6] Gams, version 43.4.0. https://www.gams.com/, May 2023.

[7] Gigantische stijging gasprijs door dreigend lng-tekort. *BNR*, 9 August 2023. Visited on 8 September 2023.

[8] Reducing carbon emissions: Eu targets and policies. *European Parliament News*, 27 April 2023. Visited on 20 October 2023.

[9] Waar halen we komend jaar ons gas vandaag en tegen welke prijs? *NOS nieuws*, 8 January 2023. Visited on 20 April 2023.

[10] Ibrahim Abada and Olivier Massol. Security of supply and retail competition in the european gas market.: Some model-based insights. *Energy Policy*, 39(7):4077–4088, 2011.

[11] Eitaro Aiyoshi and Kiyotaka Shimizu. Hierarchical decentralized systems and its new solution by a barrier method. *IEEE Transactions on Systems, Man and Cybernetics*, (6):444–449, 1981.

[12] Martin S. Andersen, Joachim Dahl, and Lieven Vandenberghe. Cvxopt, version 1.3.0. https://cvxopt.org/index.html, 2023.

[13] Kenneth J Arrow and Gerard Debreu. Existence of an equilibrium for a competitive economy. *Econometrica: Journal of the Econometric Society*, pages 265–290, 1954.

[14] Lisa Baertlein and Parraga Marianna. Focus: Historic drought, hot seas slow panama canal shipping. *Reuters*, 21 August 2023. Visited on 6 October 2023.

[15] Jonathan F Bard. Some properties of the bilevel programming problem. *Journal of optimization theory and applications*, 68(2):371–378, 1991.

[16] Maroeska G. Boots, Fieke A.M. Rijkers, and Benjamin F. Hobbs. Trading in the downstream european gas market: A successive oligopoly approach. *The Energy Journal*, 25(3):73–102, 2004.

[17] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[18] Stéphane Caron. qpsolvers, version 3.3.1. https://github.com/qpsolvers/qpsolvers, April 2023.

[19] Chi Kong Chyong and Benjamin F. Hobbs. Strategic eurasian natural gas market model for energy security and policy analysis: Formulation and application to south stream. *Energy Economics*, 44:198–211, 2014.

[20] Stephan Dempe. Bilevel optimization: theory, algorithms, applications and a bibliography. *Bilevel Optimization: Advances and Next Challenges*, pages 581–672, 2020.

[21] Caroline Dieckhoner. Simulating security of supply effects of the nabucco and south stream projects for the european natural gas market. *The Energy Journal*, 33(3), 2012.

[22] Caroline Dieckhöner, Stefan Lochner, and Dietmar Lindenberger. European natural gas infrastructure: the impact of market developments on gas flows and physical market integration. *Applied energy*, 102:994–1003, 2013.

[23] Ruben Eg. Files bij panamakanaal door droogte in regenseizoen. *NOS nieuws*, 23 August 2023. Visited on 4 September 2023.

[24] Ruud Egging. Benders decomposition for multi-stage stochastic mixed complementarity problems–applied to a global natural gas market model. *European Journal of Operational Research*, 226(2):341–353, 2013.

[25] Ruud Egging, Franziska Holz, and Steven A Gabriel. The world gas model: A multi-period mixed complementarity model for the global natural gas market. *Energy*, 35(10):4016–4029, 2010.

[26] Ruud Egging-Bratseth, Tobias Baltensperger, and Asgeir Tomasgard. Solving oligopolistic equilibrium problems with convex optimization. *European Journal of Operational Research*, 284(1):44–52, 2020.

[27] Ruud Egging-Bratseth, Franziska Holz, and Victoria Czempinski. Freedom gas to europe: Scenarios analyzed using the global gas model. *Research in International Business and Finance*, 58:101460, 2021.

[28] Nic Fildes and Hudson Lockett. Australian lng producers in talks to avert strike as energy prices climb. *Financial Times*, 10 August 2023. Visited on 9 October 2023.

[29] Robert Fourer, David M Gay, and Brian W Kernighan. Ampl: A mathematical programming language. *Management Science*, 36(5):519–554, 1990.

[30] Steven A Gabriel, Antonio J Conejo, J David Fuller, Benjamin F Hobbs, and Carlos Ruiz. *Complementarity modeling in energy markets*, volume 180. Springer Science & Business Media, 2012.

[31] Steven A Gabriel, Supat Kiet, and Jifang Zhuang. A mixed complementarity-based equilibrium model of natural gas markets. *Operations research*, 53(5):799–818, 2005.

[32] Steven A Gabriel and Yves Smeers. Complementarity problems in restructured natural gas markets. In *Recent Advances in Optimization*, pages 343–373. Springer, 2006.

[33] Steven A Gabriel, Shree Vikas, and David M Ribar. Measuring the influence of canadian carbon stabilization programs on natural gas exports to the united states via a 'bottom-up'intertemporal spatial price equilibrium model. *Energy Economics*, 22(5):497–525, 2000.

[34] Carlos García-Martínez and Manuel Lozano. Local search based on genetic algorithms. In *Advances in metaheuristics for hard optimization*, pages 199–221. Springer, 2008.

[35] Github. Quadprog, version 0.1.11. https://github.com/quadprog/quadprog#readme, 2021.

[36] Donald Goldfarb and Ashok Idnani. A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical programming*, 27(1):1–33, 1983.

[37] Rolf Golombek, Eystein Gjelsvik, and Knut Einar Rosendahl. Effects of liberalizing the natural gas markets in western europe. *The Energy Journal*, 16(1), 1995.

[38] P. Goulart. Clarabel version 0.5.1. https://github.com/oxfordcontrol/Clarabel.rs, 2023. GitHub repository.

[39] Julian Hall, Ivet Galabova, Michael Feldmeier, and Filippo Zanetti. Highs, version 1.5.1. https://github.com/ERGO-Code/HiGHS, April 2023. GitHub repository.

[40] William E Hart, Carl D Laird, Jean-Paul Watson, David L Woodruff, Gabriel A Hackebeil, Bethany L Nicholson, John D Siirola, et al. *Pyomo-optimization modeling in python*, volume 67. Springer, 2017.

[41] William E Hart, Carl D Laird, Jean-Paul Watson, David L Woodruff, Gabriel A Hackebeil, Bethany L Nicholson, John D Siirola, et al. Pyomo, version 6.6.1. https://github.com/Pyomo/pyomo, May 2023.

[42] Peter R Hartley and Kenneth B Medlock. *The Baker Institute world gas trade model*. James A Baker III Institute, 2005.

[43] Peter R. Hartley and Kenneth B. Medlock. Potential futures for russian natural gas exports. *The Energy Journal*, 30:73–95, 2009.

[44] Franziska Holz, Christian Von Hirschhausen, and Claudia Kemfert. A strategic model of european gas supply (gasmod). *Energy Economics*, 30(3):766–788, 2008.

[45] Qi Huangfu and JA Julian Hall. Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, 10(1):119–142, 2018.

[46] Berend Huisman. Scenarios for the natural gas market: An outlook to 2050. Master's thesis, 2018.

[47] Yo Ishizuka and Eitaro Aiyoshi. Double penalty method for bilevel optimization problems. *Annals of Operations Research*, 34(1):73–88, 1992.

[48] Charles D Kolstad and Leon S Lasdon. Derivative evaluation and computational experience with large bilevel mathematical programs. *Journal of optimization theory and applications*, 65:485–499, 1990.

[49] Rob Koster. Rusland aan verliezende hand in energie-oorlog. *NOS nieuws*, 23 February 2023. Visited on 20 April 2023.

[50] Noureddine Krichene. World crude oil and natural gas: a demand and supply model. *Energy economics*, 24(6):557–576, 2002.

[51] Guoshan Liu, Jiye Han, and Shouyang Wang. A trust region algorithm for bilevel programing problems. *Chinese science bulletin*, 43:820–824, 1998.

[52] Stefan Lochner. Identification of congestion and valuation of transport infrastructures in the european natural gas market. *Energy*, 36(5):2483–2492, 2011.

[53] Stefan Lochner and David Bothe. From russia with gas: an analysis of the nord stream pipeline's impact on the european gas transmission system with the tiger-model. Technical report, EWI Working Paper, 2007.

[54] Stefan Lochner and David Bothe. The development of natural gas supply costs to europe, the united states and japan in a globalizing gas market model-based analysis until 2030. *Energy policy*, 37(4):1518–1528, 2009.

[55] Alan S Manne, Kjell Roland, and Gunter Stephan. Security of supply in the western european market for natural gas. *Energy policy*, 14(1):52–64, 1986.

[56] Alfred Marshall. *Principles of Economics*. Macmillan, London, 1890.

[57] Michael D McKay, Richard J Beckman, and William J Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61, 2000.

[58] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999. Second edition.

[59] Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, June 2016.

[60] Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Source code SCS: Splitting conic solver, version 3.2.3. https://github.com/cvxgrp/scs, November 2022.

[61] Arvind U Raghunathan and Lorenz T Biegler. An interior point method for mathematical programs with complementarity constraints (mpccs). *SIAM Journal on Optimization*, 15(3):720–750, 2005.

[62] Financiële redactie. Gasprijs slechts fractie hoger ondanks stakingsdreiging australië. *De Telegraaf*, 5 September 2023. Visited on 8 September 2023.

[63] Andreas Seeliger. *Entwicklung des weltweiten Erdgasangebots bis 2030: eine modellgestützte Prognose der globalen Produktion, des Transports und des internationalen Handels, sowie eine Analyse der Bezugskostensituation ausgewählter Importnationen*, volume 61. Oldenbourg Industrieverlag, 2006.

[64] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. Efficient evolutionary algorithm for single-objective bilevel optimization. *arXiv preprint arXiv:1303.3901*, 2013.

[65] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2017.

[66] Statista. Number of liquefied natural gas storage vessels worldwide from 2010 to 2021, May 2022.

[67] Bartolomeo Stellato, Goran Banjac, Paul Goulart, Alberto Bemporad, and Stephen Boyd. Osqp: An operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4):637–672, 2020.

[68] AS Strekalovsky, Andrei Vasil'evich Orlov, and Anton Valentinovich Malyshev. Local search in a quadratic-linear bilevel programming problem. *Numerical Analysis and applications*, 3:59–70, 2010.

[69] Sunduck Suh and Tschangho John Kim. Solving nonlinear bilevel programming models of the equilibrium network design problem: a comparative review. *Annals of operations research*, 34(1):203–218, 1992.

[70] Ewan Thomson. These charts show europe's reliance on gas before the war in ukraine. *World Economic Forum*, 10 November 2022. Visited on 28 April 2023.

[71] Lieven Vandenberghe. The cvxopt linear and quadratic cone program solvers. *Online: http://cvxopt.org/documentation/coneprog.pdf*, 2010.

[72] H. von Stackelberg. Marktform und Gleichgewicht. *The Economic Journal*, 45(178):334–336, 06 1935.

[73] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106:25–57, 2005.

[74] Douglas J White and G Anandalingam. A penalty function approach for solving bi-level linear programs. *Journal of Global Optimization*, 3:397–419, 1993.

[75] GTJ Zwart and M Mulder. A welfare-economic analysis of the dutch gas depletion policy. *CPB Memorandum*, 2006.

Appendix A

Details of the Optimisation Algorithms

In this chapter, we discuss the optimisation algorithms to find a solution to the optimisation problem (2.36) from the market equilibrium model and the optimisation problem (3.25) from the calibration process. In Section A.1, we discuss the IPOPT solver with a simplified description from the original paper [73]. IPOPT is used to find a solution to (3.25). However, IPOPT relies on a good choice of initial solution to find the global optimum. In Section A.2, we discuss a global search algorithm based on the ILS meta-heuristic. This heuristic uses the stationary points found by the IPOPT algorithm to create new initial values to which IPOPT is applied again. This procedure helps to improve the calibration procedure. In Section A.3, we discuss the `quadprog` solver, which we use to solve the quadratic optimisation problem (2.36). Our description is a simplified version of the original paper [36]. Algorithm 1 combines the calibration and perturbation to perform a scenario analysis. The implementation is programmed in Python 3.10, run on a licence of Microsoft Windows 10 64-bit operating system on an Intel i5-10310U @ 1.70 GHz (2.21 GHz turbo boosted) processor with 16 GB @ 2667 MHz RAM.

Algorithm 1 Scenario Analysis Framework.

input: Data $\bar{X}, \bar{\mathbf{p}}$.
parameters: $\kappa, L, C, T, t^{(s)}, t^{(p)}, c$.
output: baseline: $X, \mathbf{p}$, perturbation: $\widetilde{X}, \widetilde{\mathbf{p}}$.

1: $\bar{X}, \bar{\mathbf{p}} \leftarrow$ `convert_units`$(\bar{X}, \bar{\mathbf{p}})$. ▷ in Appendix A.6
2: $X, \mathbf{y}, \mathbf{p} \leftarrow$ `calibrate`$(\bar{X}, \bar{\mathbf{p}}\ ;\ \kappa, L, C, T, t^{(s)}, t^{(p)}, c)$. ▷ in Section A.1 and A.2
3: perturb parameter(s): $\mathbf{y}, L, C, T, t^{(s)}, t^{(p)}, c$.
4: $\widetilde{X}, \widetilde{\mathbf{p}} \leftarrow$ `market_equilibrium`$(\mathbf{y}\ ;\ \kappa, L, C, T, t^{(s)}, t^{(p)}, c)$. ▷ in Section A.1
5: **Return:** baseline: $X, \mathbf{p}$, perturbed: $\widetilde{X}, \widetilde{\mathbf{p}}$ and differences: $\widetilde{X} - X, \widetilde{\mathbf{p}} - \mathbf{p}$.

A.1 Bilevel Optimisation for the Calibration

To find a solution to an optimisation problem, solvers typically interact with algebraic modelling languages, which allow for relative easy coding of mathematical programs. The algebraic modelling language called General Algebraic Modeling Language (GAMS) [6] is standard for many optimisation applications. An alternative is the open-source algebraic modeling language package Pyomo [41] in the programming language Python. This package is well-developed, based on many contributions, and is well-documented [40]. It uses both open-source solvers (e.g. BARNLP, MOSEK & IPOPT) and licensed solvers (e.g. KNITRO & Gurobi). However, one open-source non-linear solver is able to solve the optimisation problem in (3.25): Interior Point OPTimiser (IPOPT). IPOPT version 3.x is written in C++ (with older versions based on the original paper [73] implemented in Fortran and C) and is used as a pre-compiled solver application. The extension to solve complementarity problems via IPOPT–C (with the C standing for complementarity) is published in [61]. For more information on IPOPT(–C), we refer to details in the original papers [73, 61]. We use IPOPT–C to find a local solution of MPCC (3.25).

IPOPT-C is, as the full name (partially) implies, based on the primal-dual interior point with filter line search algorithm which is specifically well-suited to solve non-linear constrained optimisation problems (NLPs) and has an extension to solve complementarity problems. IPOPT is extensive and contains a lot of intricacies, which are designed to provide for a robust solver. Here, we will describe some of the most notable parts of IPOPT based on the original paper [73] and we cover the fundamental idea of the extension IPOPT–C. The IPOPT-C extension is designed to solve the MPCC, given by

$$\min_{\mathbf{u},\mathbf{w},\boldsymbol{\lambda}} \; f(\mathbf{u},\mathbf{w},\boldsymbol{\lambda}), \tag{A.1a}$$

$$\text{subject to} \tag{A.1b}$$

$$\boldsymbol{h}(\mathbf{u},\mathbf{w},\boldsymbol{\lambda}) = \mathbf{0}, \tag{A.1c}$$

$$\mathbf{u} \geq \mathbf{0}, \tag{A.1d}$$

$$\mathbf{0} \leq \boldsymbol{\lambda} \perp \mathbf{w} \geq \mathbf{0}, \tag{A.1e}$$

where $\mathbf{u} \in \mathbb{R}^n$, $\mathbf{w}, \boldsymbol{\lambda} \in \mathbb{R}^m$, $f : \mathbb{R}^{n+2m} \to \mathbb{R}$ and $h : \mathbb{R}^{n+2m} \to \mathbb{R}^q$. We present an equality on h in (A.1c), because inequality constraints can be incorporated via slack variables. It is possible to rewrite the complementarity (A.1e) to a constraint in a non-linear optimisation problem (NLP) given by

$$\min_{\mathbf{u},\mathbf{w},\boldsymbol{\lambda}} \; f(\mathbf{u},\mathbf{w},\boldsymbol{\lambda}), \tag{A.2a}$$

$$\text{subject to} \tag{A.2b}$$

$$\boldsymbol{h}(\mathbf{u},\mathbf{w},\boldsymbol{\lambda}) = \mathbf{0}, \tag{A.2c}$$

$$\text{Diag}(\boldsymbol{\lambda})\mathbf{w} \leq \mathbf{0}, \tag{A.2d}$$

$$\mathbf{u},\boldsymbol{\lambda},\mathbf{w} \geq \mathbf{0}, \tag{A.2e}$$

where $\text{Diag}(\boldsymbol{\lambda})$ is the diagonal matrix with $\boldsymbol{\lambda}$ on the diagonal. In [61], it is proven that the solution found by solving (A.2) is the same solution to solving (A.1). The original solver IPOPT, in [73], is designed to solve an NLP of the form:

$$\begin{aligned} &\min_{\mathbf{z}\in\mathbb{R}^{n+2m+q}} \; f(\mathbf{z}), \\ &\text{subject to} \\ &\boldsymbol{g}(\mathbf{z}) = \mathbf{0}, \\ &\mathbf{z} \geq \mathbf{0}. \end{aligned} \tag{A.3}$$

Through the use of slack variable $\mathbf{s} \in \mathbb{R}^q$ and defining $\mathbf{z} := (\mathbf{u},\mathbf{w},\boldsymbol{\lambda},\mathbf{s})$, we can formulate (A.2d) as

$$\text{Diag}(\boldsymbol{\lambda})\mathbf{w} + \mathbf{s} = \mathbf{0}, \tag{A.4}$$

such that (A.2) can be expressed as (A.3). Inequality constraints, such as in the calibration problem in (3.25), can be solved by IPOPT with the introduction of slack variables to incorporate the inequality (A.2d) in $\boldsymbol{g}(\mathbf{z})$. Creation of slack variables is handled by Pyomo. The optimisation problem given by (A.3) is relaxed via a penalty term, as we have mentioned in Section 3.1, to a *barrier method*, which is given by

$$\begin{aligned} &\min_{\mathbf{z}\in\mathbb{R}^{n+2m+q}} \; F_\mu(\mathbf{z}) := f(\mathbf{z}) - \mu \sum_{i=1}^{n} \log\left(\mathbf{z}^{(i)}\right) \\ &\text{subject to} \\ &\boldsymbol{g}(\mathbf{z}) = \mathbf{0} \end{aligned} \tag{A.5}$$

where $\mu \in \mathbb{R}$ is the barrier parameter scaling term, which scales the penalty term $\sum_{i=1}^{n} \log\left(\mathbf{z}^{(i)}\right)$, which is a relaxation of the constraint $\mathbf{z} \geq \mathbf{0}$. This approach is common in NLP techniques which use the interior point method, as seen in [58]. The original NLP (A.3) can be rewritten via the KKT-conditions (1.6) to primal-dual equations

$$\begin{aligned} \mathcal{L}(\mathbf{z},\boldsymbol{\nu},\boldsymbol{\omega}) := \nabla f(\mathbf{z}) + \nabla \boldsymbol{g}(\mathbf{z})\boldsymbol{\nu} - \boldsymbol{\omega} &= 0, \\ \boldsymbol{g}(\mathbf{z}) &= \mathbf{0}, \\ \text{Diag}(\mathbf{z})\text{Diag}(\boldsymbol{\omega})\mathbb{1} - \mu\mathbb{1} &= \mathbf{0}, \end{aligned} \tag{A.6}$$

where $\boldsymbol{\nu}, \boldsymbol{\omega}$ represent the Lagrangian multipliers of the equality constraints $g(\mathbf{z})$ and the bound constraint $\mathbf{z} \geq 0$, respectively. Note that ∇f is a gradient vector, and $\nabla \boldsymbol{g}$ is a Jacobian matrix, because $f, \boldsymbol{g}$ have images $\mathbb{R}$ and $\mathbb{R}^m$, respectively. The logarithmic function $\log(.)$ is convex and if $f, \boldsymbol{g}$ are also convex, then (A.6) is convex as well, because of the additivity rule of convex functions [17]. To solve (A.5) for fixed values of μ, a damped Newton's method is used to the primal-dual equations (A.6) given by:

$$\begin{bmatrix} \nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\omega}) & \nabla \boldsymbol{g}(\mathbf{z}) & -\mathbb{I} \\ \nabla \boldsymbol{g}(\mathbf{z})^\intercal & 0 & 0 \\ \text{diag}(\boldsymbol{\omega}) & 0 & \text{diag}(\mathbf{z}) \end{bmatrix} \begin{pmatrix} \boldsymbol{\delta}^{\mathbf{z}}_k \\ \boldsymbol{\delta}^{\boldsymbol{\nu}}_k \\ \boldsymbol{\delta}^{\boldsymbol{\omega}}_k \end{pmatrix} = - \begin{pmatrix} \nabla f(\mathbf{z}) + \nabla \boldsymbol{g}(\mathbf{z}) \boldsymbol{\nu} - \boldsymbol{\omega} \\ \boldsymbol{g}(\mathbf{z}) \\ \text{Diag}(\mathbf{z}) \text{Diag}(\boldsymbol{\omega}) \mathbb{1} - \mu \mathbb{1} \end{pmatrix}, \tag{A.7}$$

which can be solved in two steps. Firstly, by applying row elimination, we rewrite (A.7) to

$$\begin{bmatrix} \nabla^2_{\mathbf{zz}} \mathcal{L}(\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\omega}) + \text{Diag}(\boldsymbol{\omega} \circ \mathbf{z}^{-1}) & \nabla \boldsymbol{g}(\mathbf{z}) \\ \nabla \boldsymbol{g}(\mathbf{z})^\intercal & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{\delta}^{\mathbf{z}}_k \\ \boldsymbol{\delta}^{\boldsymbol{\nu}}_k \end{pmatrix} = - \begin{pmatrix} \nabla F(\mathbf{z}) + \nabla \boldsymbol{g}(\mathbf{z}) \boldsymbol{\nu} - \boldsymbol{\omega} \\ \boldsymbol{g}(\mathbf{z}) \end{pmatrix}, \tag{A.8}$$

where $\circ$ and $\mathbf{z}^{-1}$ are the same operators as in (3.8). Note that in the right hand side of (A.8) is the derivative of F from the barrier method. By solving (A.8), $\boldsymbol{\delta}^{\mathbf{z}}_k$ can be used to compute search direction $\boldsymbol{\delta}^{\omega}_k$ as

$$\boldsymbol{\delta}^{\boldsymbol{\omega}}_k = \mu \mathbf{z}^{-1} - \mathbf{z} - \text{Diag}(\boldsymbol{\omega} \circ \mathbf{z}^{-1}) \boldsymbol{\delta}^{\mathbf{z}}_k. \tag{A.9}$$

Combining all the search directions gives rise to the iterative solution approach, where we compute each iterate $(\mathbf{z}_k, \boldsymbol{\nu}_k, \boldsymbol{\omega}_k)$ as follows:

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{z}_k + \alpha \boldsymbol{\delta}^{\mathbf{z}}_k, \\ \boldsymbol{\nu}_{k+1} &= \boldsymbol{\nu}_k + \alpha \boldsymbol{\delta}^{\boldsymbol{\nu}}_k, \\ \boldsymbol{\omega}_{k+1} &= \boldsymbol{\omega}_k + \alpha_{\boldsymbol{\omega}} \boldsymbol{\delta}^{\boldsymbol{\omega}}_k, \end{aligned} \tag{A.10}$$

where $\alpha, \alpha_{\boldsymbol{\omega}} \in \mathbb{R}$ are step parameters. IPOPT uses the line search interior point method, which can be interpreted as a bi-objective optimisation method, simultaneously minimising the objective function $F_\mu(\mathbf{z})$ and the constraint violations $\theta(\mathbf{z}) := \|\boldsymbol{g}(\mathbf{z})\|_1$, with emphasis on the latter quantity. Algorithm 2, based on a formulation in Nocedal & Wright [58], describes the line-search interior point algorithm in broad steps. For a detailed description, we refer to the original paper [73], where the algorithm in IPOPT is presented.

Algorithm 2 IPOPT: Line search interior point algorithm.

input: Initial solution $(\mathbf{z}_0, \boldsymbol{\nu}_0, \boldsymbol{\omega}_0)$ and barrier parameter μ.
parameters: Set scaling factors $\sigma, \tau \in (0, 1)$ and termination tolerances $\epsilon_{\text{tol}}, \epsilon_\mu > 0$.
output: Stationary Point $(\mathbf{z}_k, \boldsymbol{\nu}_k, \boldsymbol{\omega}_k)$.

Set $k \leftarrow 0$.
while $\mathcal{E}(\mathbf{z}_k, \boldsymbol{\nu}_k, \boldsymbol{\omega}_k; 0) > \epsilon_{\text{tol}}$ **do**
  while $\mathcal{E}(\mathbf{z}_k, \boldsymbol{\nu}_k, \boldsymbol{\omega}_k; \mu) > \epsilon_\mu$ **do**
    Compute the search direction $(\boldsymbol{\delta}^{\mathbf{z}}_k, \boldsymbol{\delta}^{\boldsymbol{\nu}}_k, \boldsymbol{\delta}^{\boldsymbol{\omega}}_k)$.
    Compute maximal step size $\alpha^{\max} \leftarrow \max\{\alpha \in (0, 1] \ : \ \mathbf{z} + \alpha \boldsymbol{\delta}^{\mathbf{z}} \geq (1 - \tau)\mathbf{z}\}$.
    Compute maximal step size $\alpha^{\max}_{\omega} \leftarrow \max\{\alpha \in (0, 1] \ : \ \boldsymbol{\omega} + \alpha \boldsymbol{\delta}^{\boldsymbol{\omega}} \geq (1 - \tau)\boldsymbol{\omega}\}$.
    Compute step sizes $\alpha \in (0, \alpha^{\max}]$ and $\alpha_{\boldsymbol{\omega}} \in (0, \alpha^{\max}_{\boldsymbol{\omega}}]$.
    Compute $(\mathbf{z}_{k+1}, \boldsymbol{\nu}_{k+1}, \boldsymbol{\omega}_{k+1})$ according to (A.10).
    Set $k \leftarrow k + 1$.
  Set $\mu \leftarrow \sigma\mu$ and update ϵ_μ.
return $(\mathbf{z}_k, \boldsymbol{\nu}_k, \boldsymbol{\omega}_k)$.

The termination criterion is given by

$$\mathcal{E}(\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\omega}; \mu) := \max \left\{ \|\nabla f(\mathbf{z}) - \nabla g(\mathbf{z}) \boldsymbol{\nu}\|_\infty, \ \|g(\mathbf{z})\|_\infty, \ \|\text{Diag}(\mathbf{z}) \text{Diag}(\boldsymbol{\omega}) \mathbb{1} - \mu \mathbb{1}\|_\infty \right\},$$

where $\|.\|_\infty$ is the infimum norm as given in (3.10) with $\rho = \infty$. The goal of IPOPT is to solve the barrier method (A.5), where $\mu \to 0$. This is achieved by the outer while-loop, which checks

if the termination criterion holds true for the user defined tolerance $\epsilon_{\text{tol}} \in \mathbb{R}$, which is typically small. This outer-loop represents the barrier method for $\mu = 0$. At the start of the algorithm, $\mu > 0$, where each iteration of the outer loop, the parameter μ is reduced by a user-defined factor $\sigma \in (0, 1)$. The inner while-loop, computes the solution of the barrier method for $\mu > 0$, where each iteration computes a new search direction (by solving (A.8) and computing (A.9)) to update the solution (A.10), iteratively. The step sizes $\alpha, \alpha_{\boldsymbol{\omega}}$ need to be determined based on the variable $\boldsymbol{z}$ and Lagrange multiplier $\boldsymbol{\omega}$, respectively. The user-defined parameter $\tau \in (0, 1)$ makes sure that the optimisation does not converge to the lower bound too quickly [73].

Next, we give a brief overview of the extension applied by IPOPT to create a more robust solver to find a (global) solution of NLPs. The extensions we cover are as follows:

- Filter method, which prevents cycling between solutions.
- Second-order correction, which corrects infeasible solutions based on information in the Jacobian.
- Inertia correction of the KKT matrix, which approximates the system of equations (A.8) that may have become singular throughout the iterative process.
- Feasibility restoration phase, which is the final mechanism to find a (non-optimal) solution which is, at least, feasible.

Filter Method For each iteration k, IPOPT maintains a 'filter' $\mathcal{F}_k \subset \{(\theta, f) \in \mathbb{R}^2 \ : \ \theta \geq 0\}$ which contains the combinations of objective values f and constraint violations θ that are prohibited. If $(\theta(\mathbf{z}_k), f(\mathbf{z}_k)) \in \mathcal{F}_k$ then the candidate solution $\mathbf{z}_k$ is rejected. The filter is initialized with

$$\mathcal{F}_0 := \{(\theta, f) \in \mathbb{R}^2 \ : \ \theta \geq \theta^{\max}\},$$

where $\theta^{\max} \in \mathbb{R}$ is user defined parameter. During the algorithm, the filter is also updated every iteration $k + 1$ by:

$$\mathcal{F}_{k+1} := \mathcal{F}_k \cup \left\{(\theta, f) \in \mathbb{R}^2 \ : \ \theta \geq (1 - \gamma_\theta)\theta(\mathbf{z}_k) \quad \text{and} \quad f \geq f_\mu(\mathbf{z}) - \gamma_f \theta(\mathbf{z}_k)\right\}, \tag{A.11}$$

where $\gamma_\theta, \gamma_f \in \mathbb{R}$ are user-defined parameters. This filter is designed to prevent the cycling between two solutions, where one improves the objective value f and the other the constraint violation θ. If a candidate is found, which does not improve either the objective value f or the constraint violation θ, then the candidate is discarded.

Second-order correction When a candidate solution is rejected, a secord order correction can be used to improve the step size. Such a second-order correction (an additional search direction step based on the Jacobian $\nabla f(\mathbf{z})$) applies addition Newton-like steps to the algorithm, with the goal of reducing the infeasibility. This correction step replaces the candidate solution $\mathbf{z}_k$. The additional correction step $d_k^{\mathbf{z},\text{cor}}$ is found by solving

$$\nabla f(\mathbf{z})^\intercal \boldsymbol{\delta}_k^{\mathbf{z},\text{soc}} + \boldsymbol{g}(\mathbf{z}_k + \alpha_k \boldsymbol{\delta}_k^{\mathbf{z}}) = \mathbf{0},$$

where $\boldsymbol{\delta}_k^{\mathbf{z},\text{soc}}$ is the second-order step, and f, g are the objective function and constraints from (A.3). Then, the correction step $\boldsymbol{\delta}_k^{\mathbf{z},\text{cor}}$ is defined as

$$\boldsymbol{\delta}_k^{\mathbf{z},\text{cor}} := \alpha_k \boldsymbol{\delta}_k^{\mathbf{z}} + \boldsymbol{\delta}_k^{\mathbf{z},\text{soc}}.$$

There are more correction techniques applied, such as updating the step size α_k and updating constraint g_k based on the corrected point. This second-order correction technique is applied consecutively until the corrected point is feasible. For more details on the second-order correction and the intricacies of updating α_k, g_k, we refer to Section 2.4 in [73].

Inertia correction of the KKT matrix To be able to evaluate the search directions (A.10), we need non-singular matrices for (A.7) to solve the system. In addition, the matrix in (A.8) needs to be positive definite to have global convergence, i.e., the optimisation returns a globally optimal solution. The inertia of the iteration matrix (found in (A.8)) is defined by a tuple[1] signifying n positive eigenvalues, m negative eigenvalues and 0 zero-valued eigenvalues. To try to keep the inertia $(n, m, 0)$, a heuristic is used by IPOPT to improve the robustness of the algorithm, and prevent zero-valued eigenvalues. In short, no action is taken whenever the inertia is $(n, m, 0)$, but when zero-valued eigenvalues are present, a modified version of (A.8) is solved, where

$$\begin{bmatrix} \nabla^2_{\mathbf{zz}}\mathcal{L}(\mathbf{z}, \boldsymbol{\nu}, \boldsymbol{\omega}) + \text{Diag}(\boldsymbol{\omega} \circ \mathbf{z}^{-1}) + \varepsilon_1 \mathbb{I} & \nabla \boldsymbol{g}(\mathbf{z}) \\ \nabla \boldsymbol{g}(\mathbf{z})^\intercal & -\varepsilon_2 \mathbb{I} \end{bmatrix} \begin{pmatrix} \boldsymbol{\delta}^{\mathbf{z}}_k \\ \boldsymbol{\delta}^{\boldsymbol{\nu}}_k \end{pmatrix} = - \begin{pmatrix} \nabla F(\mathbf{z}) + \nabla \boldsymbol{g}(\mathbf{z})\boldsymbol{\nu} - \boldsymbol{\omega} \\ \boldsymbol{g}(\mathbf{z}) \end{pmatrix},$$

for $\varepsilon_1, \varepsilon_2 > 0$ are relatively small values. This allows for an acceptable approximation of the singular system of equations, such that IPOPT can still find a solution for problem descriptions with zero-valued eigenvalues during the optimisation process. For more information, we refer to Section 3.1 in [73].

Feasibility Restoration Phase Whenever the line-search procedure cannot correct an infeasible solution, and the step size has become too small to 'escape' the convergence, a feasibility restoration is applied to return a non-optimal solution, which is at least feasible. If the solution is infeasible under the aforementioned conditions, then the restoration phase computes a new acceptable iterate to the updated filter $\mathcal{F}_{k+1}$ by decreasing the infeasibility. IPOPT adjusts the objective function to

$$\begin{aligned} &\min_{\mathbf{z} \in \mathbb{R}^n} \|\boldsymbol{g}(\mathbf{z})\|_1 + \zeta \|D_R(\mathbf{z} - \hat{\mathbf{z}})\|_2^2, \\ &\text{subject to} \\ &\mathbf{z} \geq \mathbf{0}, \end{aligned} \tag{A.12}$$

where $\hat{\mathbf{z}}$ is the infeasible reference solution at which IPOPT converged and needs to be restored. The matrix D_R is a scaling matrix, the parameter $\zeta > 0$ is used as a weight. The minimization of $\|\boldsymbol{g}(\mathbf{z})\|_1$ is the key objective, with $\|D_R(\mathbf{z} - \hat{\mathbf{z}})\|_2^2$ being a regularization such that the solution is close to the reference point $\hat{\mathbf{z}}$ at which the restoration phase is initiated. The parameter ζ needs to be chosen carefully to guarantee improvement of the constraint violation, while staying in the neighbourhood of $\hat{\mathbf{z}}$. There are alternative optimisations possible during the restoration phase compared to (A.12). We only present the basic idea here. For more information, we refer to Section 3.3 in [73].

A.2 Searching for the Global Solution

EC can be used to improve the solution. The basic idea in EC mimics biological evolution. The goal in EC is to solve a problem with use of a population of solutions which are iteratively altered. Candidate solutions are typically referred to as chromosomes, where each chromosome has a fitness determining if it will be used to form the new solution. In optimisation problems, the fitness is usually the objective function of the optimisation problem. A set of chromosomes is called a population, which evolves iteratively with the use of crossover (mixing chromosomes) and mutation (perturb a single chromosome). Using an evolutionary algorithm can be relatively slow, but can result in better solutions when the objective function is non-smooth, non-differentiable, or has many local optima. In these cases, using a gradient-based algorithm would converge locally. However, gradient-based algorithms have relatively quick convergence and low computing time depending on the problem size. Combining the properties of the gradient-based algorithms with an evolutionary framework can result in a relatively efficient and accurate algorithm. We will refer to this as a meta-heuristic, because it is a heuristic which is not problem specific. There are many types of meta-heuristics, as seen in [34]. The types we consider are, in order of increased complexity, given by:

[1] The variables m, n have nothing to do with the amounts bought and sold in Chapter 2 $(n, m, 0)$, and is used for notational ease.

Multi-start local search As the name implies, MSLS uses an efficient local search algorithm for a number of different initial starting values. It solves the problem with a local solver for each initial value and considers all the found optima and uses the local optimum with the lowest objective value. This is assumed to be the global optimum, but there is no guarantee that it is. There is no interaction or mixing between solutions and solutions are not put back into the local search algorithm, and thus MSLS local search is the simplest form of EC.

Iterative local search Similar to the MSLS, multiple problems with different initial values are solved. Based on the objective values, the best ranking solutions are chosen. These can be changed slightly, referred to as a mutation, to obtain new initial values, which are optimised again. This iterative process is more complex because of the mutation of the local optima found by the algorithm. However, there is still no interaction or mixing between solutions.

Genetic local search A crossover operation between multiple (usually two) solutions is used at each iteration to combine good performing optima to find even better optima. This can be combined with a mutation operation which only changes one solution. The inclusion of a crossover operation in between iterations is similar to the genetic algorithms traditionally found in EC.

In our project, we have implemented all three methods, but in the end the ILS algorithm was sufficient in finding a solution. Applying the global search algorithm did not improve the solution. In that case, it is better to use the ILS algorithm, because it has reduced computing time, compared to GLS. The MSLS and GLS algorithms are given in Appendix A.5, respectively. The pseudocode of the ILS algorithm is presented here, in Algorithm 3, for which all notation corresponds to the variables and functions defined for the calibration (3.25) of the market equilibrium model.

Algorithm 3 Iterative local search with IPOPT.

input: Observed market data $(\bar{\mathbf{x}}, \bar{\mathbf{d}}, \bar{\mathbf{p}})$ from (3.9).
parameters: Set scaling parameter $\sigma \in (0,1)$, termination tolerance $\epsilon_{\text{improv}} > 0$, $c_{\max} > 0$.
output: Stationary solution $(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda})^{(k)}$

$k \leftarrow 0, \quad \zeta \leftarrow 1, \quad c \leftarrow 0.$
Choose initial $\breve{\mathbf{y}}$ whose components are in the middle of the constraints (3.25).
Initialize population $\mathcal{P}_0 \leftarrow$ `initialize_population`$(\bar{\mathbf{x}}, \breve{\mathbf{y}}, \zeta)$
while $\mathcal{E}((\mathbf{x}, \mathbf{y}, \lambda)^{(k)}, c)$ **do**
    Locally optimal solutions $\mathcal{S}_k \leftarrow$ `parallel_IPOPT`$(\mathcal{P}_k)$
    $(\breve{\mathbf{x}}, \breve{\mathbf{y}}, \breve{\boldsymbol{\lambda}})^{(k)} \leftarrow \operatorname{argmin}(\mathcal{S}_k)$
    if $f_k\left((\breve{\mathbf{x}}, \breve{\mathbf{y}}, \breve{\boldsymbol{\lambda}})^{(k)}\right) - f_{k-1}\left((\breve{\mathbf{x}}, \breve{\mathbf{y}}, \breve{\boldsymbol{\lambda}})^{(k-1)}\right) < \epsilon_{\text{improve}}$ **then**
        $c \leftarrow c + 1$
    else
        $c \leftarrow 0$
    $\mathcal{P}_{k+1} \leftarrow$ `initialize_population`$(\bar{\mathbf{x}}, \breve{\mathbf{y}}, \zeta)$
    set $k \leftarrow k + 1, \quad \zeta \leftarrow \sigma\zeta$
return $(\mathbf{x}, \mathbf{y}, \lambda)^{(k)}$

The termination criterion $\mathcal{E}((\mathbf{x}, \mathbf{y}, \lambda)^{(k)}, c)$ is given by

$$\mathcal{E}((\mathbf{x}, \mathbf{y}, \lambda)^{(k)}, c) := \begin{cases} \texttt{true}, & k < k_{\max} \text{ and } c < c_{\max}, \\ \texttt{false}, & \text{otherwise.} \end{cases}$$

Lastly the minimal improvement criterion $c < c_{\max}$ counts how many times the objective value has not improved sufficiently. If there is no noticeable improvement after $c_{\max}$ iterations, we terminate the algorithm. The function `parallel_IPOPT`$(\mathcal{P}_k)$ is the parallelized application of the IPOPT algorithm, i.e., we distribute the computations of IPOPT over multiple processors, because each instance can be solved independently. For example, when the population $\mathcal{P}_k$ has 22 initial values with 4 processors, then 2 processors solve IPOPT for 6 initial solutions, and the other 2 processors solve for 5 initial solutions.

The initialization of the population by `initialize_population`$(\bar{\mathbf{x}}, \check{\mathbf{y}}, \zeta)$ uses a Latin hypercube sampling (LHS) technique, which is presented in Algorithm 4. Let $\mathbf{y} \in \mathbb{R}^n$ be a vector which is sampled via LHS. Divide each dimension $i = 1, ..., n$ of $\mathbf{y}$ that needs to be sampled in equal intervals. Then, for each interval one value is uniformly sampled. Mix the samples from the different intervals such that the samples have a mix of uniformly sampled values from varying intervals. The intervals help to spread the samples evenly throughout the sample space. For more information on the Latin Hypercube sampling in comparison to other sampling techniques, we refer to [57].

Algorithm 4 Description of function: `initialize_population`.

input: Observed market data $\bar{\mathbf{x}}$, found optimal $\check{\mathbf{y}}$, population size $p \in \mathbb{N}$ and parameter $\zeta \in (0, 1)$.
output: Population $\mathcal{P}$

Step 1: Determine the partitions and perform sampling.
Compute the lower bounds $\mathbf{y}^l \leftarrow (1 - \zeta)\mathbf{y}$, and upper bounds $\mathbf{y}^u \leftarrow (1 + \zeta)\mathbf{y}$
for $i = 1, ..., |\mathbf{y}|$ **do**
    Partition interval $[\mathbf{y}_i^l, \mathbf{y}_i^u]$ into p subintervals, denoted by $[\mathbf{y}_{i,j}^l, \mathbf{y}_{i,(j+1)}^u]$ for $j = 1, ..., p - 1$.
    for $j = 1, ..., p - 1$ **do**
        Sample $\tilde{\mathbf{y}}_i^{(j)} \sim U([\mathbf{y}_{i,j}^l, \mathbf{y}_{i,(j+1)}^u])$

Step 2: Distribute the samples of each component to form initial population.
for $j = 1, ..., p$ **do**
    randomly choose $\mathbf{y}_i^{(k,j)} \leftarrow \tilde{\mathbf{y}}(k, j)_i$ without repeat.
$\boldsymbol{\lambda} \leftarrow$ `determine_lambda`$(\bar{\mathbf{x}})$
Initialize population $\mathcal{P} \leftarrow \left\{(\bar{\mathbf{x}}, \mathbf{y}^{(k,1)}, \boldsymbol{\lambda}), ..., (\bar{\mathbf{x}}, \mathbf{y}^{(k,p)}, \boldsymbol{\lambda})\right\}$
return $\mathcal{P}$

The function `determine_lambda`$(\bar{\mathbf{x}})$ sets entries of $\boldsymbol{\lambda}$ to zero when we know, based on $\bar{\mathbf{x}}$, that constraints are not at the boundary, and gives a random non-zero value when we know that a constraint is at the boundary. For example, for the non-negativity constraint of $\mathbf{x}$ in (2.15), when we know $\bar{\mathrm{x}}_0 = 0$, then $\lambda_0 \sim \mathcal{U}([0, 1])$ uniformly sampled, otherwise when $\bar{\mathrm{x}}_0 > 0$, then $\lambda_0 = 0$.

A.3 Details of the Quadratic Optimisation Algorithm

In this Section, we discuss the `quadprog` solver, which we use to solve the quadratic optimisation problem (2.36). Our description is a simplified version of the original paper [36]. The quadratic optimisation problem from (2.36) (with linear constraints) is a relatively simple mathematical description and there are several solvers available which can be used to solve this problem. Under (strict) convexity conditions, the quadratic optimisation algorithms find the global solution. Some solvers, such as `quadprog`, require a symmetric matrix P. In that case, the matrix P can be replaced by $\frac{1}{2}(P + P^\intercal)$, because

$$\frac{1}{2}\mathbf{x}^\intercal(P + P^\intercal)\mathbf{x} = \frac{1}{2}\mathbf{x}^\intercal P\mathbf{x} + \frac{1}{2}\mathbf{x}^\intercal P^\intercal\mathbf{x} = \mathbf{x}^\intercal P\mathbf{x}.$$

The solvers which we used are from the `qpsolver` [18] Python package running the open-source solvers: `Highs` [45, 39], `quadprog` [36], `Clarabel` [38], `OSQP` [67], `SCS` [59, 60] and `CVXOPT` [71, 12] which are mixtures of active-set, interior point and first order methods [17, 58, 67]. Through testing, we have found that `quadprog` based on [36] is the most robust solver. When the other solvers find a solution, the result is used to perform a sanity check on the final solution. The solver `quadprog` algorithm solves the quadratic minimization problems of the form

$$\begin{aligned} \min_{\mathbf{x}} \ & \hat{\varphi}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\intercal Q\mathbf{x} + \mathbf{q}^\intercal\mathbf{x}, \\ & \text{subject to} \\ & D\mathbf{x} \leq \mathbf{v}, \end{aligned} \tag{A.13}$$

where defining $Q := -(P + P^\intercal)$ and $\mathbf{q} = -\mathbf{c}$ means that minimization problem (A.13) is equivalent to the market equilibrium problem (2.36) for $\mathbf{x}, D, \mathbf{v}$ representing the same trade flows, constraint matrix and constraint bound, respectively. If the matrix Q is indefinite, and all solvers find the same solution, we have more confidence in the stationary point being a global optimum. In this section, we present a description of the `quadprog` solver, applied to the quadratic optimisation problem (A.13), based on the original paper. For further details, we refer to the paper [36] and the GitHub source code [35].

The `quadprog` solver is based on the Goldfarb/Idnani algorithm which is a projection type algorithm. This algorithm uses projections on the active sets of constraints. An active set $\mathcal{A} \subset \{1, 2, ..., K\}$ is a subset of K constraints, based on the K constraints given by D in (A.13). for which the inequalities are equalities, e.g., if $\mathcal{A} = \{1\}$ then $(D\mathbf{x})_1 \leq \mathrm{v}_1$ will be replaced by $(D\mathbf{x})_1 = \mathrm{v}_1$. Using the active sets, the constrained quadratic programming can be solved in two phases: First, a feasible point is found (usually non-optimal). Secondly, the problem is iteratively optimised, increasing the subproblem with one constraint per iteration, while the solution remains feasible. A natural initial point is the unconstrained optimum

$$\mathbf{x} = -Q^{-1}\mathbf{q},$$

which is used when this point is also feasible. In case it is infeasible, a feasible point routine is used to acquire a feasible point based on $\breve{\mathbf{x}}$.

The optimisation problem (A.13) is solved iteratively, where at each iteration a subproblem is used. Let $\mathcal{C} \subseteq \{1, ..., K\}$. A subproblem $\mathcal{P}(\mathcal{C})$ of the optimisation problem (A.13) is an optimisation problem with the same objective function, but with the constraints $(D\mathbf{x})_j \leq \mathrm{v}_j$ indexed by $j \in \mathcal{C}$. The unconstrained version of the optimisation problem would be given by $\mathcal{P}(\emptyset)$. The optimisation problem in (A.13) is given by $\mathcal{P}(\{1, 2, ..., K\})$. Note that for any subproblem the active set $\mathcal{A} \subset \mathcal{C}$ is a set of indices which indicate the constraints which are equalities from the subset of constraints. For example, let $K = 3$ and $\mathcal{P}(\{1, 2, 3\})$ be an optimisation problem given by (A.14). Let $\mathcal{P}(\{2, 3\})$ be an optimisation problem given by (A.15), which is a subproblem of $\mathcal{P}(\{1, 2, 3\})$ because it has a subset of the constraints, namely constraints $(D\mathbf{x})_2 \leq \mathbf{v}_2$ and $(D\mathbf{x})_3 \leq \mathbf{v}_3$. Let $\mathcal{A} := \{2, 3\}$ be an active set. Then the subproblem $\mathcal{P}(\mathcal{A})$ is given by (A.16). Note that the subproblem $\mathcal{P}(\mathcal{A})$ has equality constraints, while $\mathcal{P}(\{2, 3\})$ has inequality constraints. Additionally, a solution to $\mathcal{P}(\mathcal{A})$ is also a solution for $\mathcal{P}(\mathcal{A})$.

$\mathcal{P}(\{1,2,3\})$:

$$\begin{aligned} \min_{\mathbf{x}}\ & \mathbf{x}^\intercal Q\mathbf{x} + \mathbf{q}^\intercal\mathbf{x}, \\ \text{subject to}\ & \\ & (D\mathbf{x})_1 \leq \mathbf{v}_1, \\ & (D\mathbf{x})_2 \leq \mathbf{v}_2, \\ & (D\mathbf{x})_3 \leq \mathbf{v}_3, \end{aligned} \tag{A.14}$$

$\mathcal{P}(\{2,3\})$:

$$\begin{aligned} \min_{\mathbf{x}}\ & \mathbf{x}^\intercal Q\mathbf{x} + \mathbf{q}^\intercal\mathbf{x}, \\ \text{subject to}\ & \\ & (D\mathbf{x})_2 \leq \mathbf{v}_2, \\ & (D\mathbf{x})_3 \leq \mathbf{v}_3, \end{aligned} \tag{A.15}$$

$\mathcal{P}(\mathcal{A})$:

$$\begin{aligned} \min_{\mathbf{x}}\ & \mathbf{x}^\intercal Q\mathbf{x} + \mathbf{q}^\intercal\mathbf{x}, \\ \text{subject to}\ & \\ & (D\mathbf{x})_2 = \mathbf{v}_2, \\ & (D\mathbf{x})_3 = \mathbf{v}_3. \end{aligned} \tag{A.16}$$

The solution $\mathbf{x}$ found by solving $\mathcal{P}(\mathcal{C})$ with some linearly independent active set $\mathcal{A} \subseteq \mathcal{C}$, is given by the solution-pair $(\mathbf{x}, \mathcal{A})$. Note that a solution which solves (A.16), also solves (A.15), because the inequality is a relaxation compared to the equality constraint. In addition, note that when (A.15) is infeasible, so is (A.14). The linear independence of the active set $\mathcal{A}$ means that the normal vectors $\mathbf{n}_i$ which correspond with the constraint $(D\mathbf{x})_i \leq \mathrm{v}_i$ for $i \in \mathcal{A}$ are linearly independent. Based on the definitions of subproblems and solution-pairs, the algorithm used by `quadprog` iteratively expands the subproblem at each iteration. To start, we have $\mathcal{P}(\mathcal{A})$ with $\mathcal{A} = \emptyset$. Each iteration, we consider a violated constraint $k \in \{1, ..., K\} \backslash \mathcal{A}$. If adding the violated constraint as an active constraint, means that the subproblem $\mathcal{P}(\mathcal{A} \cup \{k\})$ becomes infeasible, then stop. Else, obtain a new solution-pair of $\mathcal{P}(A \cup \{k\})$. Solve to find the solution-pair $(\widetilde{\mathbf{x}}, \widetilde{\mathcal{A}} \cup \{k\})$, where $\widetilde{\mathcal{A}} \subseteq \mathcal{A}$ and

$$\widetilde{\mathbf{x}}^\intercal Q\widetilde{\mathbf{x}} + \mathbf{q}^\intercal\widetilde{\mathbf{x}} > \mathbf{x}^\intercal Q\mathbf{x} + \mathbf{q}^\intercal\mathbf{x}, \tag{A.17}$$

which replaces the old solution-pair $(\mathbf{x}, \mathcal{A})$. Note that the objective value has increased in (A.17) because we start with the unconstrained minimum (the lowest value possible without any constraint)

for the first solution-pair and add constraints consecutively, the objective value must monotonically increase as the feasible region decreases. Repeat this process, until none of the constraints is violated, then the optimal solution is found. This approach is outlined in Algorithm 5. Algorithm 6 is the detailed algorithmic description from the original paper [36] is given in Appendix A.4. The following notation will be used in the algorithms: The matrix N describes the normal vectors $\mathbf{n}_i$ of constraints in the active set $i \in \mathcal{A}$. The cardinality of $\mathcal{A}$ is given by $|\mathcal{A}|$. Matrix $\mathcal{N}^+$ is the matrix of normal vectors of $\mathcal{A} \cup \{k\}$ for $k \in \{1, ..., K\} \backslash \mathcal{A}$. Vector $\mathbf{n}^+$ indicates the normal (column) vector $\mathbf{n}_k$ added to N to obtain N^+. Pseudo-inverse N^* is defined by

$$\mathcal{N}^* := (\mathcal{N}^\intercal Q^{-1} \mathcal{N})^{-1} \mathcal{N}^\intercal Q^{-1}, \tag{A.18}$$

and a reduced inverse Hessian operator H, which is subject to the active set of constraints N, is defined by

$$\mathcal{H} := Q^{-1}(\mathbb{I} - \mathcal{N}\mathcal{N}^*),$$

such that a point $\bar{\mathbf{x}} = \hat{\mathbf{x}} - \mathcal{H}(D\hat{\mathbf{x}} - \mathbf{v})$ is an optimum of $\nabla\hat{\varphi}(\hat{\mathbf{x}})$ where $\hat{\mathbf{x}} \in \{\mathbf{x} \in \mathbb{R}^n \ : \ \mathbf{n}_i^\intercal \hat{\mathbf{x}} = \mathrm{v}_i, i \in \mathcal{A}\}$. During the optimisation, we also update the Lagrange multipliers, denoted by $\boldsymbol{\lambda}$. The Lagrange multiplier which corresponds to normal vector $\mathbf{n}^+$ is $\boldsymbol{\lambda}^+$.

Algorithm 5 Quadprog: Basic overview of the algorithm.

input: Coefficients of (A.13): $Q, \boldsymbol{q}$ and constraint $D\mathbf{x} \leq \mathbf{v}$.
output: Return optimal and feasible $\mathbf{x}$ or return an error.

1: Initialize $\mathbf{x}, \hat{\varphi}, \mathcal{H}, \mathcal{A}$.
2: **if** there is no violated constraint **then**
3: **return** solution $\mathbf{x}$, which is feasible and optimal.
4: **else**
5: select a violated constraint i.
6: Determine step directions $\mathbf{r}$ (for $\mathbf{x}$) and $\mathbf{z}$ (for $\boldsymbol{\lambda}$).
7: Compute step length t_1 for direction $\mathbf{x}$ and t_2 for direction $\boldsymbol{\lambda}$.
8: Compute minimum step size $t = \min\{t_1, t_2\}$.
9: **if** step size t (or t_2) is infinite **then**
10: **return** ERROR or go to step 2a.
11: Compute new iteration $\mathbf{x}$, $\boldsymbol{\lambda}^+$ and $\hat{\varphi}$.
12: **if** $t = t_1$ **then**
13: Set $\mathcal{A} \leftarrow \mathcal{A} \backslash \{k\}$, i.e., add constraint k to the active set.
14: Update matrices $\mathcal{H}$ and $\mathcal{N}$, based on new active set $\mathcal{A}$. Go to line (6).
15: **if** $t = t_2$ **then**
16: Set $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^+$ and $\mathcal{A} \leftarrow \mathcal{A} \cup \{i\}$, i.e., remove constraint i from the active set.
17: Update matrices $\mathcal{H}$ and $\mathcal{N}$, based on new active set $\mathcal{A}$. Go to line (2).

A.4 Details of Active Set Algorithm used by Quadprog

In Algorithm 6, we give a detailed description of the algorithms used by `quadprog`, based on the original paper [36].

Algorithm 6 Detailed description of `quadprog`.

input: Coefficients of (A.13): $Q, \mathbf{q}$ and constraint $D\mathbf{x} \leq \mathbf{v}$.
output: return optimal and feasible $\mathbf{x}$ or return an error.

Step 0: Initialize
1: $\mathbf{x} \leftarrow -Q^{-1}\mathbf{q}, \quad \hat{\varphi} \leftarrow \frac{1}{2}\mathbf{q}^\intercal\mathbf{x}, \quad \mathcal{H} \leftarrow Q^{-1}, \quad \mathcal{A} \leftarrow \emptyset.$

Step 1: Choose a violated constraint, if any:
2: Compute $(D\mathbf{x})_j - \mathbf{v}_j$ for all $j \in \{1, ..., K\}\backslash\mathcal{A}$.
3: **if** $V = \{j \in \{1, ..., K\}\backslash\mathcal{A} \ : \ (D\mathbf{x})_j - \mathbf{v}_j > 0\} = \emptyset$. **then**
4: **return** optimal and feasible $\mathbf{x}$.
5: **else**
6: Choose $i \in V$ and set $\mathbf{n}^+ \leftarrow \mathbf{n}_i$ and $\boldsymbol{\lambda}^+ \leftarrow \binom{\boldsymbol{\lambda}}{0}$.
7: **if** $|\mathcal{A}| = 0$. **then**
8: $\boldsymbol{\lambda} \leftarrow 0$.

Step 2: Check for feasibility and determine new S-pair.
Step 2a: Determine step directions r and z.
9: Compute $\mathbf{z} = \mathcal{H}\mathbf{n}^+$ (direction for $\mathbf{x}$).
10: **if** $|\mathcal{A}| > 0$. **then**
11: $\mathbf{r} = N^*\mathbf{n}^+$ (direction for $\boldsymbol{\lambda}$).

Step 2b: Compute step length t.
12: **if** $\mathbf{r} \leq 0$ or $|\mathcal{A}| = 0$. **then**
13: $t_1 \leftarrow \infty$.
14: **else**
15: $t_1 \leftarrow \min_{j \in \mathcal{A}} \left\{ \frac{\lambda_j^+(\mathbf{x})}{\mathrm{r}_j} \right\} = \frac{\lambda_k^+(\mathbf{x})}{\mathrm{r}_k}$.
16: **if** $|\mathbf{z}| = 0$. **then**
17: $t_2 \leftarrow \infty$.
18: **else**
19: $t_2 \leftarrow \frac{(D\mathbf{x})_i - \mathbf{v}_i}{\mathbf{z}^\intercal\mathbf{n}^+}$ where i is from line (6).
20: $t \leftarrow \min\{t_1, t_2\}$.

Step 2b: Determine new S-pair and take step.
21: **if** $t = \infty$ **then**
22: **return** ERROR: subproblem $\mathcal{P}(\mathcal{A} \cup \{i\})$ and by extension (A.13) is infeasible.
23: **if** $t_2 = \infty$. **then**
24: $\boldsymbol{\lambda}^+ \leftarrow \boldsymbol{\lambda}^+ + t \cdot \binom{-\mathbf{r}}{0}$ and $\mathcal{A} \leftarrow \mathcal{A}\backslash\{k\}$ where k from line 15. Update $\mathcal{H}, \mathcal{N}^*$. Go to step 2a.
25: Compute new iteration: $\mathbf{x} \leftarrow \mathbf{x} + t\mathbf{z}, \quad \hat{\varphi} \leftarrow \hat{\varphi} + t(\frac{1}{2}t + \mathrm{u}^+_{|\mathcal{A}+1|}) \cdot \mathbf{z}^\intercal\mathbf{n}^+, \quad \boldsymbol{\lambda}^+ \leftarrow \boldsymbol{\lambda}^+ + t\binom{-\mathbf{r}}{0}$.
26: **if** $t = t_2$ **then**
27: $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda}^+, \quad \mathcal{A} \leftarrow \mathcal{A} \cup \{i\}$ where i is from line (6). Update $\mathcal{H}$ and $\mathcal{N}^*$. Go to step 1.
28: **if** $t = t_1$ **then**
29: $\mathcal{A} \leftarrow \mathcal{A}\backslash\{k\}$ where k from line (15). Update $\mathcal{H}$ and $\mathcal{N}^*$. Go to step 2a.

A.5 Other Meta-heuristics

Multi-start Local Search The multi-start local search algorithm we have implemented is given by Algorithm 7.

Algorithm 7 Multi-Start local search with IPOPT.

input: Observed market data $(\bar{\mathbf{x}}, \bar{\mathbf{d}}, \bar{\mathbf{p}})$ from (3.9).
output: Stationary solution $(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda})^{(k)}$
.

1: Choose initial $\breve{\mathbf{y}}$ whose components are in the middle of the constraints (3.25).
2: Initialize population $\mathcal{P}_0 \leftarrow$ `initialize_population`$(\bar{\mathbf{x}}, \breve{\mathbf{y}}, \zeta)$
3: Locally optimal solutions $\mathcal{S}_k \leftarrow$ `parallel_IPOPT`$(\mathcal{P}_k)$
4: $(\breve{\mathbf{x}}, \breve{\mathbf{y}}, \breve{\boldsymbol{\lambda}})^{(k)} \leftarrow \text{argmin}(\mathcal{S}_k)$
5: **return** $(\breve{\mathbf{x}}, \breve{\mathbf{y}}, \breve{\boldsymbol{\lambda}})^{(k)}$

Differential Evolution We have tried a GLS algorithm using the framework of Differential Evolution given in Algorithm 8, but altered this algorithm to use the local search algorithm IPOPT. Algorithm 8 generates a population of local optima for which the solutions are mixed (by crossover) to find new initial solutions. These initial solutions are used by the IPOPT solver to find new local optima. If the mixed initial gives a better local optima than the previous iteration, the solution is replaced, otherwise the old solution is kept. Parameter $CR \in (0, 1)$ is the crossover rate and parameter $F \in [0, 2)$ is used to the adjust the crossover operation, which is based on the crossover operation typically used by the Differential Evolution algorithm. The crossover operation is given in Algorithm 9. To determine if the crossover has led to a better solution, the population of initial solutions $\hat{\mathcal{P}}_k$ is solved via a parallelised execution of the IPOPT algorithm. In Algorithm 10, we give the details of the selection procedure we perform, which is also based on the selection used in Differential Evolution. To improve the robustness of the global search algorithm a 'replenishment' of the population of initial solutions is performed to make sure that there are enough initial solutions to find solutions. This safety measure is necessary because IPOPT does not always converge to a solution which can be used. At that point, an error is given by IPOPT, thus we need to replace the failed solution with a new initial value.

Algorithm 8 Differential Evolution with IPOPT.

input: Observed market data $\bar{\mathbf{x}}$, $\bar{\mathbf{d}}$, $\bar{\mathbf{p}}$.
parameters: Set $CR \in (0, 1)$, $F \in [0, 2)$, $\epsilon_{\text{tol}} > 0$.
output: Stationary solution $(\mathbf{x}, \mathbf{y}, \lambda)_l^{(k)}$.

1: $k \leftarrow 0$
2: Initialize population $\mathcal{P}_0 := \left\{(\mathbf{x}, \mathbf{y}, \lambda)_1^{(0)}, ..., (\mathbf{x}, \mathbf{y}, \lambda)_l^{(0)}\right\}$
3: Local solutions $\mathcal{S}_0 \leftarrow \text{IPOPT}(\mathcal{P}_0)$
4: **while** $\mathcal{E}((\mathbf{x}, \mathbf{y}, \lambda)_k) \leq \epsilon_{\text{tol}}$ and $k < k_{\max}$ **do**
5: $\quad \hat{\mathcal{P}}_k \leftarrow$ `crossover`$(\mathcal{S}_0, CR, F)$
6: $\quad$ Local solutions $\hat{\mathcal{S}}_k \leftarrow \text{IPOPT}(\hat{\mathcal{P}}_k)$
7: $\quad \mathcal{P}_{k+1} \leftarrow$ `selection`$(\mathcal{S}_k, \hat{\mathcal{S}}_k)$
8: $\quad$ Select $(\mathbf{x}, \mathbf{y}, \lambda)_l^{(k)} \leftarrow \text{argmin}(\mathcal{S}_k)$
9: $\quad \mathcal{P}_{k+1} \leftarrow$ `replenish`$(\mathcal{P}_{k+1})$
10: $\quad$ set $k \leftarrow k + 1$
11: **return** $(\mathbf{x}, \mathbf{y}, \lambda)_l^{(k)}$

Algorithm 9 Crossover Operation of the Differential Evolution algorithm.

input: Population of parent solutions $\mathcal{S}_k$.
parameters: Set $CR \in (0,1)$, $F \in [0,2)$.
output: Population of offspring initial values $\hat{\mathcal{P}}_k$.

$\hat{\mathcal{P}}_k := \emptyset$
for $(\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)} \in \mathcal{S}_k$ **do**
  $u_1, u_2, u_3 \leftarrow$ `uniform_sample`$([0,1))$
  random index $\iota \leftarrow$ `uniform_sample`$(\{0, \dim((\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)})\})$
  for $i \in \{0, \dim((\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)})\}$ **do**
    $r_i \leftarrow$ `uniform_sample`$([0,1))$
    if $r_i < CR$ or $i = \iota$ **then**
      $z_i \leftarrow (\mathbf{x}, \mathbf{y}, \lambda)_{u_1}^{(k)} + F \cdot \left[(\mathbf{x}, \mathbf{y}, \lambda)_{u_2}^{(k)} - (\mathbf{x}, \mathbf{y}, \lambda)_{u_3}^{(k)}\right]$
    else
      $z_i \leftarrow (\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)}$
  append $\hat{\mathcal{P}}_k \leftarrow \hat{\mathcal{P}}_k \cup \{z\}$
return $\hat{\mathcal{P}}_k$

Algorithm 10 Selection Operation of the Differential Evolution algorithm.

input: Parent and offspring solutions $\mathcal{S}_k, \hat{\mathcal{S}}_k$.
output: Population of initial solutions for next iteration $\mathcal{P}_{k+1}$.

$\mathcal{P}_{k+1} := \emptyset$
for $u = 1, ..., |\mathcal{S}_k|$ **do**
  select $(\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)} \in \mathcal{S}_k$ and $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\lambda})_u^{(k)} \in \hat{\mathcal{S}}_k$
  if $f\big((\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\lambda})_u^{(k)}\big) < f\big((\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)}\big)$ **then**
    add offspring solution $\mathcal{P}_{k+1} \leftarrow \mathcal{P}_{k+1} \cup \{(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\lambda})_u^{(k)}\}$
  else
    add parent solution $\mathcal{P}_{k+1} \leftarrow \mathcal{P}_{k+1} \cup \{(\mathbf{x}, \mathbf{y}, \lambda)_u^{(k)}\}$
return $\mathcal{P}_{k+1}$

A.6 Preprocessing: Data ingestion

For the model to perform well it is essential that the data which in ingested in the model is of sufficient quality to perform a accurate calibration. There are a number of pre-processing tasks necessary before we can calibrate the market equilibrium model. The different price benchmarks for the European and Asian markets are based on different currencies and different energy units. In addition, the trade flows are commonly published in a different unit than the price benchmarks are published. It is essential for the market equilibrium model that these trade flows units and currencies are equivalent for every route and region. Some regions also do not have a price benchmark. These missing data points need an estimate. This estimate is based on a price benchmark and a price differential, which we base on expert opinion. In general, we assume that export regions act comparable to each other, and thus have similar prices. In Algorithm 11, we show the conversion algorithm. Here, MMbtu stands for 10^6 British thermal units, Therms stands for 10^5 British thermal units, MWh stands for megawatt hour, USD is the United States Dollar, GBp is the Great British penny and EUR is the Euro.

Algorithm 11 Data Conversion.

input: Data $\bar{X}, \bar{\mathbf{p}}$.
output: Converted data $\bar{X}, \bar{\mathbf{p}}$.

$\bar{\mathbf{p}} \leftarrow$ `USDperMMbtu_to_EURperMWh`$(\bar{\mathbf{p}})$
$\bar{\mathbf{p}} \leftarrow$ `GBpperTherm_to_EURperMWh`$(\bar{\mathbf{p}})$
$\bar{X} \leftarrow$ `metric_tonne_to_MWh`$(\bar{X})$
return: $\bar{X}$ and $\bar{\mathbf{p}}$.

In addition, we also need to acquire data on $L, C, \kappa, \pi^{(s)}, \pi^{(p)}$. Some of these values are published. However, for missing data we need to find a suitable substitute. In general, we assume that export regions have similar behaviour and thus similar parameters. We assume the same for import regions. Alternatively, The North American terminals have relatively a lot of publications describing the port capacity in the Gulf of Mexico. This can be used to compute an 'overhead' ratio $\frac{E_{\text{NA}}}{C_{\text{NA}}}$ which describes how much port capacity is available on top of what is already exported, i.e., overhead. This ratio can be used to compute the port capacities of the other regions as

$$C_i := E_i \cdot \frac{E_{\text{NA}}}{C_{\text{NA}}},$$

for the export regions $i \in S$.

Our model focusses on the spot LNG market. Finding data which is exclusively on spot LNG market is difficult. As a solution, we use the combined trade flows (spot and long contract) and multiply the data by a factor of the spot market contribution. For example, the trade route NA–EU will be multiplied by 0.5, because 50% of the trade flow is spot market. While ME–NEA is multiplied by 0.05, because only 5% of the trade flow is spot market.

Appendix B

Additional Background and Results

B.1 Algebraic Modeling Language

The goal of using an algebraic modeling language (AML) is to provide a programming syntax which translates the mathematical formulation seen in LaTeX documents to readable code. It should be noted that AML are high level programming languages, intended to provide a framework to solve high complexity problems for large scale problems. The types of problems that AMLs are able to solve are linear, integer, quadratic, non-linear, complementarity and stochastic problems. These can be constrained or unconstrained programs. Another benefit of AMLs is the relative ease at which the programmer can use different solvers. Each solver has its own interface, but uses standardised 'A Mathematical Programming Language' [29] (AMPL) read (.nl) and write solution (.sol) file extensions, which are interpreted by the AML. It is possible for the user to link the AML to non-supported solvers, as long as the solver is compliant with standard AMPL .nl and .sol file extensions.

Pyomo translates the optimisation problem (3.25) from Python code to input of the solver application based on standard .nl file-format, which is used by algebraic modelling languages to inferface IPOPT (or any other solving application). The solution is in a standard solution .sol file-format. The algebraic modeling language Pyomo handles the translation from Python code to the .nl file-format and after solving the resulting .sol file is interpreted by Pyomo, which creates a Python object with all the information from the .sol file.

B.2 Other Methods to Simplify a Bilevel Optimisation

Descent Method A descent method uses the gradient of the lower or upper level problem which provides a direction to decrease the upper level objective value while the point remains feasible. The upper level problem is non-differentiable, but this can be approximated via several approximation methods, see [48]. The lower level problem is a standard single level optimisation on its own, thus gradient-based methods are applicable. This method is explored for quadratic-linear bilevel programs in [68] and non-linear bilevel programs in [69].

Trust-region Method This method can be seen as the opposite of the line search method. Here a model function is used for an approximation of a region of the objective function. The size of the region is changed based on the quality of the approximation. If the approximation is accurate, the the region is expended. Otherwise the size of the region is reduced. This method is applied to non-linear bilevel programs with convex objective functions in [51].

Meta-modelling For some solving techniques, such as global search algorithms, it can be problematic whenever the objective function is computationally expensive to evaluate. Instead, a meta-model – a surrogate model of the actual model – is used. The meta-models can be solved with an evolutionary algorithm, because evaluating the objective function is designed to be cheap.

The meta-model should be close enough to the actual model such that the solution of the meta-model can be used as an approximation for the solution of the actual model. There are several ways to employ meta-modelling:

The first approach, reaction set mapping, is based on the fact that the set mapping

$$\Psi = \{\varphi(\mathbf{x}) \quad \text{subject to} \quad D\mathbf{x} \leq \mathbf{v}\},$$

is not known. The alternative is to approximate Ψ, but this is not straightforward. Attempts have been made, such as in [64].

The second approach tries to approximate the optimal lower level value function $\phi_{\mathbf{x}}(\mathbf{y})$ which represents the minimum lower level function value for given upper level variable $\mathbf{x}$. However, $\phi_{\mathbf{x}}(\mathbf{y})$ is rarely known. Whenever it is known, $\phi_{\mathbf{x}}(y)$ can be used to reduce the bilevel program to a single level program when the lower level problem is not convex.

The third approach is bypassing the lower level problem, which assumes that the optimal $\mathbf{x}$ is a function of $\mathbf{y}$. With this assumption, a single level optimisation can be constructed because the lower level optimisation can be removed. This is likely to lead to ill-posed problems because of non-convexity, discontinuities and non-differentiability. In practice, it is also difficult to determine such function $\mathbf{x}(\mathbf{y})$.

B.3 Accuracy Analysis of Calibration without Market Equilibrium Constraints

Assume that our model can calibrate perfectly such that $\mathbf{z} = \bar{\mathbf{z}}$ as given by (3.9). This means that we have $M(N-M)+M+N$ equations given by $\mathrm{z}_k = \bar{\mathrm{z}}_k$ for $k = 1, ..., M(N-M)+M+N$. The calibration has $2M+2N$ degrees of freedom. When $M(N-M)+M+N < 2M+2N$, we can interpret the optimisation as an underdetermined system of equations. And when $M(N-M)+M+N > 2M+2N$ as an overdetermined system of equations. We show that the optimisation without market equilibrium constraints of the summer 2022 baseline is able to find accurate solutions consistently when $M(N-M)+M+N = 2M+2N$. In Table B.1, we given an overview the number of equations given by $\mathrm{z}_k = \bar{\mathrm{z}}_k$ for $k = 1, ..., M(N-M)+M+N$ and the degree of freedom for increasing number of regions for this specific calibration. In Figures B.1 and B.2, we show the accuracy analysis of these regions solved by the MSLS and ILS algorithms, respectively.

Notice that the scenario with 3 export and 3 import regions (3×3) is well equated (number of equations and degrees of freedom are the same) and the accuracy analysis in Figures B.1 and B.2 are the most consistent. The scenario with 2×2 regions is still more accurate, as shown by the lowest 0th percentile, but has more unique solutions, as seen by the spread between the other percentiles. We believe this is caused by the calibration mimicking an underdetermined system of equations. This calibration would benefit from a better solver, which quickly finds the minimum. We think the scenarios 4×4,...,6×6 are inaccurate because the calibration mimicking an overdetermined system of equations. As seen by the high minimum objective value and high objective values for all percentiles higher than 25th. In contrast to the underdetermined system, we think the model would need more degrees of freedom to accurately calibrate. In Table B.1, we give an overview of the number of equations and degrees of freedom, when the calibration is interpreted as a system of equations.

Additionally, we have calibrated the model by replacing the objective function by

$$\|\mathbf{z} \circ \bar{\mathbf{z}}^{-1} - \mathbb{1}\|_2^2 + \|\mathbf{y}\|_2^2,$$

where $\gamma \in \mathbb{R}$ (we choose 10^{-8}) and $\|.\|_2$ is the Euclidean norm. In Tables B.2, B.3, B.4 and B.5, we present the corresponding calibrated flows, prices,ask and bid parameters and the relative difference w.r.t. the realistic data.

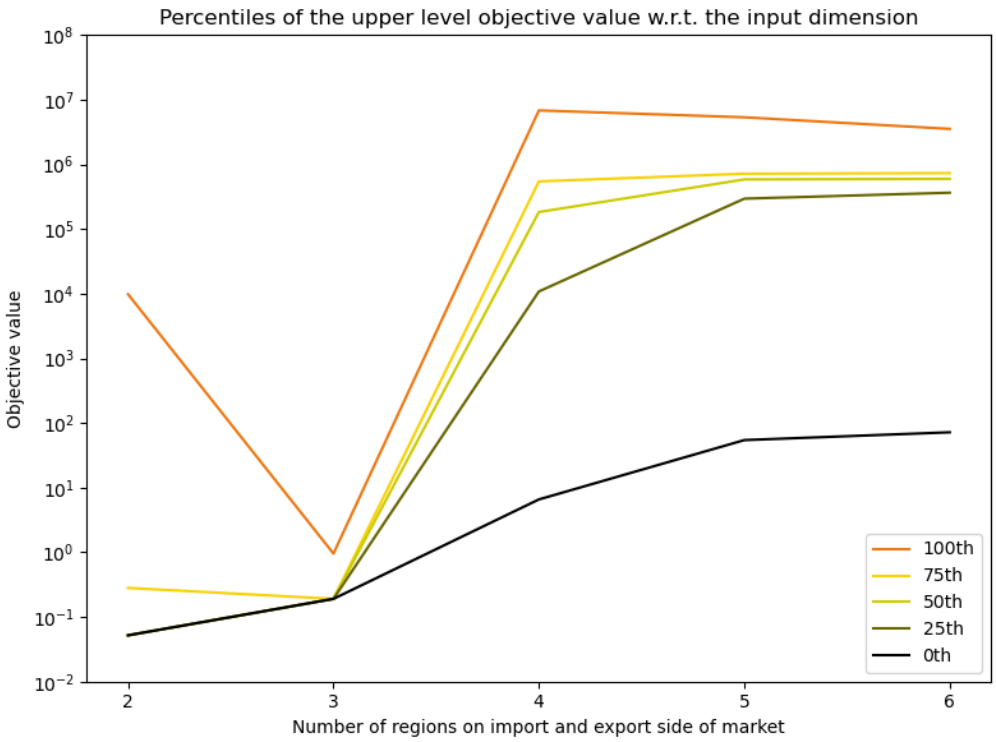


Figure B.1: Performance of the multi-start local search without market equilibrium constraints.

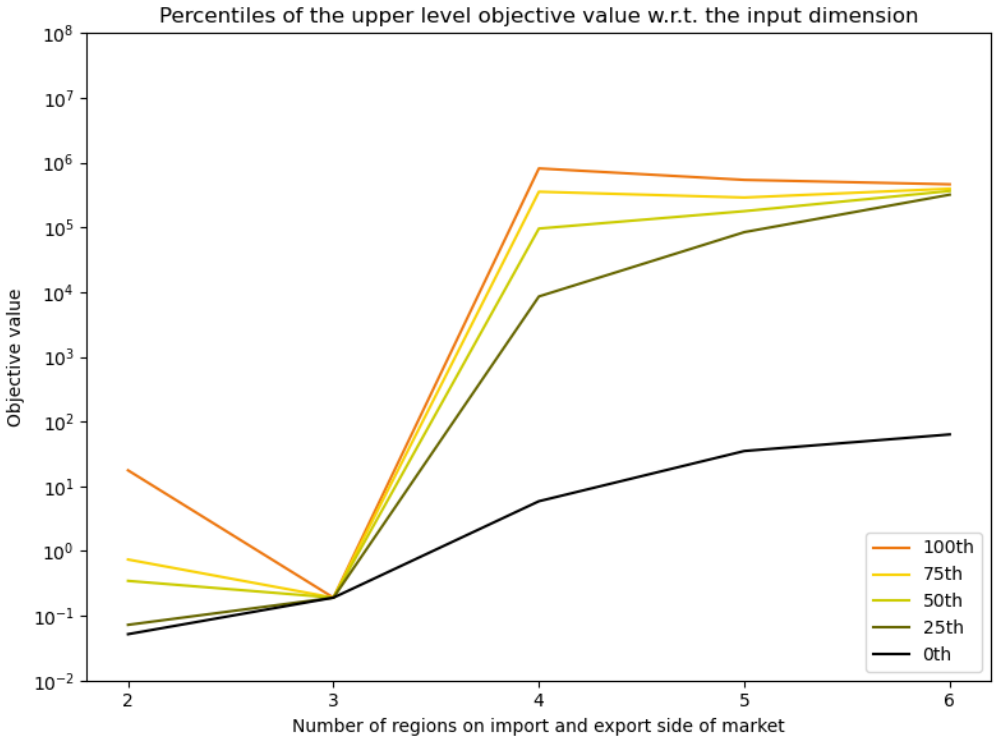


Figure B.2: Performance of the ILS without market equilibrium constraints.

Export × Import	Regions N	Number of equations $M(N-M)+M+N$	Degrees of freedom $2M+2N$	Type of system
2×2	4	10	12	underdetermined
3×3	6	18	18	well equated
4×4	8	28	24	overdetermined
5×5	10	40	30	overdetermined
6×6	12	54	36	overdetermined

Table B.1: Overview of number of equations and degrees of freedom.

	$\boldsymbol{\eta}^{\mathrm{A}}$ (€/MWh)/(MWh/day)	$\boldsymbol{a}$ €/MWh	$\boldsymbol{\eta}^{\mathrm{B}}$ (€/MWh)/(MWh/day)	$\boldsymbol{b}$ €/MWh
AU	0.0000881	-71.87	0.0001888	72.37
NA	0.0000025	-10.98	0.0000013	43.99
ME	0.0010659	-1071.84	0.0012911	999.67
EU	-	-	0.0002220	327.76
SA	-	-	0.0003002	153.63
NEA	-	-	0.0001061	297.43

Table B.2: Summer 2022 scenario calibration without market equilibrium constraint: Ask and bid parameters.

	Baseline $\mathbf{p}$ €/MWh	Relative diff. $d(\mathbf{p}, \bar{\mathbf{p}})$ -
AU	26.67	0.06%
NA	26.14	1.25%
ME	27.02	-3.90%
EU	163.93	8.23%
SA	91.97	23.57%
NEA	162.40	-12.70%

Table B.3: Summer 2022 scenario calibration without market equilibrium constraint: Baseline prices $\mathbf{p}$ with the relative difference compared to market data $\bar{\mathbf{p}}$.

MWh/day	AU	NA	ME	EU	SA	NEA	Total Sold
AU	242,086	-	-	999	1,000	874,953	1,119,039
NA	-	13,631,003	-	613,745	203,398	244,154	14,692,301
ME	-	-	753,326	123,314	1,002	153,293	1,030,934
Total Bought	242,086	13,631,003	753,326	738,058	205,400	1,272,400	

Table B.4: Summer 2022 baseline calibration without market equilibrium constraint: Calibrated trade flows.

	AU	NA	ME	EU	SA	NEA	Total Sold
AU	0.41%	-	-	-	-	3.84	3.26%
NA	-	0.01%	-	-4.47%	0.40%	0.58%	-0.17%
ME	-	-	-1.45%	19.61%	-	-27.20%	-4.37%
Total Bought	0.41%	0.01%	-1.45%	-1.01%	1.39%	-1.81%	

Table B.5: Summer 2022 baseline calibration without market equilibrium constraint: Relative differences of the calibrated trade flows w.r.t. the data.