UNIVERSITY OF OSLO
Department of Informatics

The northern European gas market as a linear programming model

Master thesis

Lars Erik Jordet

Spring 2014

The northern European gas market as a linear programming model

Lars Erik Jordet

Spring 2014

Abstract

The European gas market is a huge network of producers, consumers, and various forms of gas processing and storage. To understand this market fully, all of these factors have to be explored and explained.

When the gas market is described as a network flow problem a multitude of factors - including data quality, poorly defined limitations, decisions only known to the individual actors in the market, and solution speed - put a limit on how accurate this description can be. This thesis shows that based on public and semi-public data the entire northern European gas network can be modelled as a general network flow problem and solved by a mixed-integer-problem solver.

This model is then shown to be easily converted into a fully linear program, which can be solved in significantly less time with almost no loss in precision.

Keywords: gas market, linear optimisation, linear programming, mixed-integer programming, network flow, natural gas flow

Acknowledgements

To start out, I would like to thank my colleagues at Nena for making this thesis possible - over the 5 years I've been working with this degree, I've been allowed the freedom necessary to submerge myself in the studies when needed, and hopefully I have at least repaid some of this freedom in the form of applied new knowledge over these years.

My three supervisors have been of great help; Carlo Mannino and Geir Dahl at the university, and my colleague Pål Svendsen at Nena have all pointed me in the right direction and helped me immensely in learning the particularities and peculiarities of optimization and gas markets, respectively.

I've also had good help from an old friend, Finn Olav Sveen, who provided suggestions from an outside perspective in the last stages of writing.

Thank you for all the help!

Contents

List of Figures

List of Tables

[1] Department of Energy and Climate Change 2013; European Commission 2014.

Nomenclature

mcm million (Standard) cubic meters, see Sm^3

Sm^3 Standard cubic meter; 1 cubic meter of gas at 1 atmosphere pressure and 15°C

bank holiday A public holiday in the UK; markets are not trading on bank holidays

base gas The volume of gas in a gas storage used to maintain the pressure needed to withdraw gas

BBL Balgzand-Bacton Link, a pipeline between NL and UK

calorific value The energy released by the combustion gas, varies with different qualities/mixtures of natural gas, measured in MJ/Sm^3 - see Sm^3

CBC COIN-OR Branch and Cut, a free, open source MILP solver

CCGT Combined cycle gas turbine

CHP Combined heat and power

COIN-OR COmputational INfrastructure for Operations Research, a set of OR packages

CPLEX by IBM, a commercial mathematical programming solver

dry gas also called lean gas, gas without heavy liquid hydrocarbons; mostly methane

EGV Expected Gas Value - value of gas left in a storage facility at the end of a modelling period

forward curve the prices for a product in for all traded future periods

full cycle running a storage from empty, to full, to empty again (or the other way around

LNG Liquefied natural gas

MIP/MILP Mixed-integer (linear) program, an optimization program with a mix of integer and continuous variables

National Grid UK gas grid operator http://www.natgrid.co.uk

NBP National Balancing Point, virtual trading location for the UK market

NCS Norwegian Continental Shelf

NGL Natural Gas Liquids, a group of hydrocarbons found in wet gas

nomination A request for transport of a quantity of gas

NTS National Transmission System, the UK-internal gas network

OCGT Open cycle gas turbine

PuLP An interface for LP solvers from Python

Python A programming language

send out Volume of gas sent out from (LNG) storage

SOS2 Special Ordered Set of type 2, a way of linearising constraints; see section 5.2.3 on page 38

therm 1 British therm is approximately equivalent to 29.3071kWh

TTF Title Transfer Facility, virtual trading location for the Dutch market

UKCS UK Continental Shelf

WDNW Working days next week, e.g. Mon-Fri unless there are bank holidays

wet gas also called rich gas, is dry gas with additional hydrocarbons

working gas The usable volume of gas in a full gas storage - total capacity minus base gas

Preface

Part I

Introduction

In this thesis I will explain the northern European gas market as seen from an outside observer, based on the public data from the actors in the system, and explore how it can be analysed fully.

Starting out, the first chapter contains a quick primer on the gas market, as a lot of jargon will be used to explain the behaviour of the model.

While mixed-integer models exist for parts of the northern European system, for instance the work by Tomasgard et al. 2007 which describes a model for continuous modelling of the flow based on the physical properties of the transmission of gas, and the similar work by De Wolf and Smeers 2000a looking at the details of the Belgian domestic gas network, this thesis explores the northern European gas network as a whole on a day-by-day basis over a 14 month period. A discussion of other work on the subject can be found in chapter 2.

Secondly, chapters 3 and 4, on *the problem* explain how this thesis came to be, and what the challenges in creating a model are.

For this market, we're looking at a huge amount of data, covering most - but not all - aspects of the system, in mostly - but not always - the form we want it.

After a quick run-down of the mathematical concepts used, there are presented two models for this gas market, the first model being a mixed-integer model covering all the elements of the system in chapter 6 - and the second model in chapter 7has all the integer variables removed or replaced, making a linear programming model which is easily solvable by even a low-powered desktop computer. As a result of running the models on real world data, some extensions to the model are proposed in chapter 8.

Chapter 9 presents the algorithm and the software involved, while chapter 10 looks into the impact of these models and their results, concluding this thesis.

Part II

Background theory

The aim of this part is to present readers with little to no prior experience with the European gas market an introduction to the terms and concepts used in the thesis.

Chapter 1

The gas markets and distribution systems

1.1 The European gas market

1.1.1 Futures markets in general

A futures contract is a contract on the delivery of a quantity of a commodity at a specified time future, at a specific future price - to be paid at the time of delivery. Utilities which depend on the physical assets of the commodity in question, like industry with high consumption of the commodity, will be trading future commodity to coincide with a high production period, to protect their position (hedge), being able to lock down their expected cost. At the same time producers of the commodity may want to hedge to maximize their guaranteed profits from the commodity.

Financial investment sectors speculate on making a profit on the development of the commodity price. These sectors may not have interest in owning any physical assets of the commodity in question, and needs to make sure they have no open contracts (position) at the time of expiry. Pindyck 2001 goes deeper into both the basics and more advanced concepts not covered by this thesis.

Natural gas is one of the commodities traded this way, and the topic of this thesis.

Figur 14.1 Eksisterande og planlagde rørleidningar *(Kjelde: Oljedirektoratet)*

Figure 1.1: Existing and projected pipelines Source: Figure 14.1 in Norwegian Ministry of Petroleum and Energy 2012

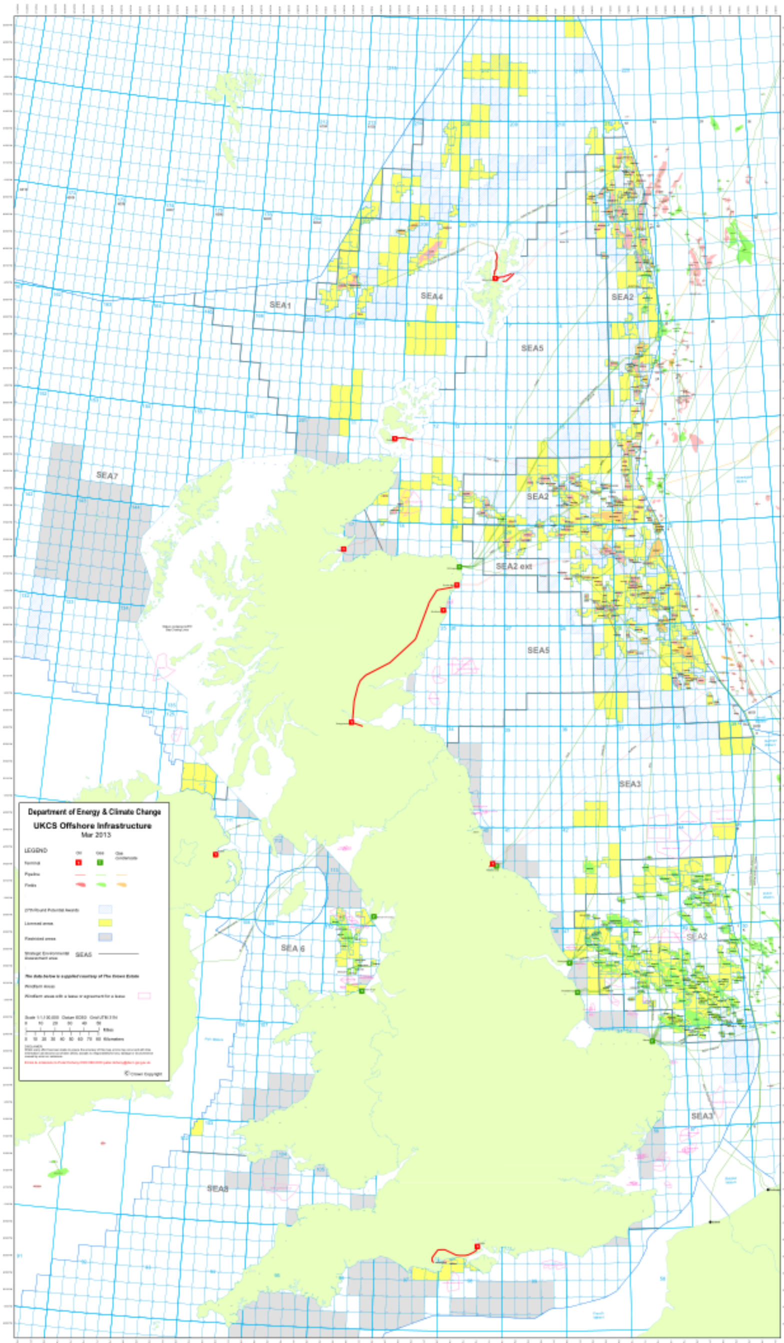


Figure 1.2: Source: UK Department of Energy & Climate Change, *08 March 2013*

1.1.2 Natural gas markets

The forward curve

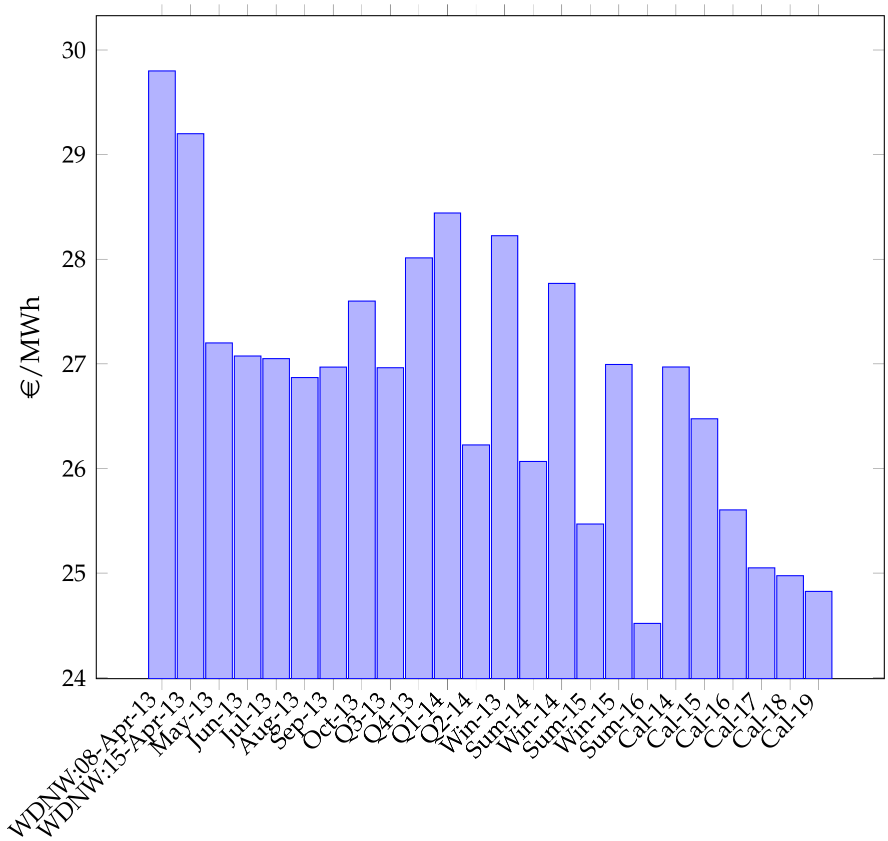


Figure 1.3: Endex TTF forward curve, fetched 2013-03-28

The forward curve is the curve of prices for the future periods traded on a given market; an example of such a curve is in figure 1.3. Typically it defines the price in a continuous set of periods with varying resolutions, and some overlap - in the example curve Q4-13 (fourth quarter, 2013) and Q1-14 combined comprises the period defined by Win(ter)-13.

At the time such a curve is published, it will in a perfect market reflect all information available on events, probabilities and other effects of the supply and demand on the price at the time, like planned maintenance affecting supply, seasonal/weather driven changes to demand, and - though realistically uncertainties in the production or underlying delivery system leads to the need to hedge against price changes.

Each market place has its own forward curve, where each period price is set based on the trades done for that period at the exchange.

1.1.3 The markets

Four traded gas markets in Europe are considered in this thesis; the National Balancing Point (NBP) in the UK, the Title Transfer Facility (TTF) in the Netherlands (as represented in the forward curve in figure 1.3)[1], PEG NORD in northern France and GASPOOL in Germany. Virtual trading places relieves the sellers and buyers of having to directly trade or otherwise set up delivery with the other party - in essence the gas enters the system at an entry point, and can be sold any number of times before leaving at an exit point (delivery).

1.2 The gas network

There are two sides to the network, the supply - the production of natural gas - and the demand - consumption in the domestic markets. In addition there's storage, which effectively can work as both supply and demand depending on whether the storage is in withdrawal or injection mode.

1.2.1 The supply side

Gas production in the North Sea continental shelf is done at several off-shore gas and gas/oil fields, owned by different companies. Gas fields consist of one or more gas and/or oil reservoirs, each with producing platforms where gas is pumped up from the reservoirs on the continental shelf. Generally the gas is grouped into associated gas, which is gas extracted in conjunction with the extraction of oil, and non-associated gas, which is gas extracted from purely gas fields.

These are further grouped into dry (lean) gas, which is mostly methane, and is "ready for consumers"; it does not contain any liquid/heavy hydrocarbons, and wet (rich) gas, which contains other hydrocarbons that need to be removed before the gas can be used (essentially turning it into dry gas).

Associated gas is wet gas, and has to pass through a processing plant to strip out the Natural Gas Liquids (NGLs) - these NGLs are turned into other petroleum products and sold - which turns it into dry gas.

Non-associated gas can be wet, and the above applies, or dry, where the methane can more or less be sent directly to the consumers. [2]

Gas fields are linked to hubs, on-shore natural gas processing plants, and terminals linked to a domestic market[3] - see figure 1.1 on page 10 for the Norwegian Continental Shelf (NCS) and figure 1.2 on page 11 for the UK Continental Shelf (UKCS). Not directly shown in these figures are the production facilities for the Netherlands, Denmark and Germany, whose production is linked directly to shore in the respective countries

Inside the EU, the Netherlands and the UK provide the majority of the natural gas production, with 44% and 26% respectively in 2012[4], Romania

[1] http://www.gasunietransportservices.nl/en/products-services/entry-exitcapacity/ttf

[2] Definitions of terms at http://www.eia.gov/tools/glossary/index.cfm

[3] Interactive map of the Norwegian Continental Shelf at http://www.gassco.no/wps/wcm/connect/Gassco-EN/Gassco/Home/norsk-gass/gas-transport-system/

[4] European Commission 2014.

and Germany provided 7% each, and the remaining 16% are produced in, amongst others, Italy, Denmark, Poland, Hungary.

Russia exports 27% of their gas production to the EU, which in turn is 33.0% of the total gas imports of the EU (2010)[5].Norway had 28% of the gas imports the same year. Recently, the African production was linked to Europe, through a pipeline from Algeria, which covers another 14% of the imports.

The gas is pumped from the platforms through large undersea pipelines. Compressors increase the gas pressure by injecting gas at one end of the pipeline, and gas flows out at the other end. Riser platforms and compressor platforms serve to maintain the network pressure and diverts the gas through various legs of the gas network.

Terminals and quality

Via the system of platforms, compressors and risers in the North Sea, gas eventually finds its way on shore, and hits the terminals and processing plants. Here the gas is controlled, compressed and mixed to the quality specification of the market it's intended for. Various qualities of gas have different calorific values, which effects the energy gained from each Sm^3 of gas. Operators like National Grid monitor this calorific value constantly, and report it on their web pages - they report the general range of calorific values as $37.5MJ/m^3$ to $43.0MJ/m^3$National Grid 2014a.

Wet gas has a higher calorific value[6] by about 10%, but the gas that is consumed by the end-users is, as mentioned, dry gas.

LNG

Additional gas - 81.63 billion Sm^3 in 2010 - is imported in the form of liquefied natural gas (LNG) which arrives by ship from, amongst others, Qatar, Nigeria, Trinidad, and Norway to the re-gasification terminals (Gas LNG Europe 2011; King & Spalding 2006). For this thesis LNG terminals in the UK, northern France, Belgium, Netherlands and Germany will be considered.

LNG is stored in its liquid form at the terminals, since the volume required 600 times lower than the original gas volume[7], and re-gasified as it is sold or used. For a cryogenic gas storage like this, boil-off - the venting of evaporated gases to keep the pressure in the container below a critical level - leads to a loss from the tank which is cooled and fed back into the tank (by expending some of the gas to produce the power to cool).

The gas withdrawn from an LNG storage is called the "send out", and is injected into the regular gas grid.

[5]European Commission 2014.

[6]Energuide 2012.

[7]Gas Infrastructure Europe 2014.

1.2.2 The demand side

The UK government and the operator National Grid directly publish a lot of details about their operations and statistics[8]. For the rest of the EU, the statistics are provided by the statistics office of the European Union, Eurostat[9].

The major gas consumers are power production, industry and other commercial users, as well as the largest group, domestic end-users for heating and cooking.

Domestic use

The dominant part of the demand side is linked to the gas use in heating; UK figures from 2013 indicate that 40% of the gas demand goes to domestic use. Outside of direct heating, gas is used for water heating and cooking. These demands are sensitive to temperature, day of week (less industrial demand in weekends, but more domestic demand) and holidays.

This domestic usage is largely insensitive to the price of the gas, as people need food and heating independent of the market movements - in contrast to the demand in power production.

Power production

The second largest consumer of gas is the power sector, which is linked to the demand in electrical power. This demand is price sensitive, as it is dependent on the prices of other forms of power production, notably the coal price. The competitive edge of gas power prices has been weakened the last few years in relation to the coal price.

European gas power plants use a few different technologies to generate power; for handling peak loads *open cycle gas turbines* (OCGT) are used. These use a straight forward system as shown in figure 1.4 on the following page, and vent the exhaust gases to the atmosphere. Current OCGT plants have a maximum efficiency of between 35% and 42%.

A more advanced gas turbine uses the heat from the exhaust gases to create steam and run a steam turbine, generating further power. This *combined cycle gas turbine* (CCGT) is illustrated in figure 1.5 on page 17 and can reach an efficiency of 55 - 59%(Energy Research Centre 2014).

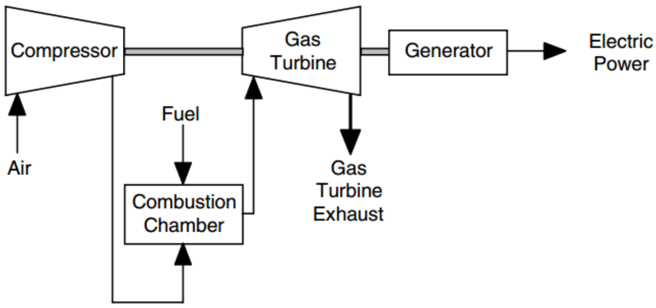
In addition, both of these technologies can be run in a *combined heat and power*(CHP) configuration, in which the excess heat from the power production is used to generate heating for nearby industry and housing. A CHP CCGT plant can reach a total energy efficiency in excess of 80%[10]. Figures 1.4 on the following page and 1.5 on page 17 both show these alternate configurations, and how they differ from a non-CHP plant.

See table 1.1 on the following page for statistics on how much of the gas demand goes to power production, and how much of the country's total

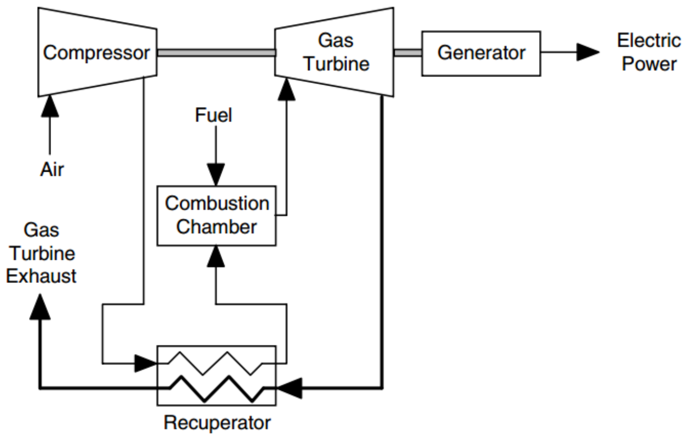
[8] National Grid 2014b.

[9] European Commission 2014.

[10] Lako 2010.



Open-Cycle Gas Turbine without Recuperator



Open-Cycle Gas Turbine with Recuperator

Figure 1.4: An open cycle gas turbine, with/without CHP; Source: Oland 2004

power production comes from natural gas, respectively. As shown in the table, some countries depend fully on gas for their power production (like the Netherlands), while others only rely on gas to handle the peak demand in regular or exceptional cases (like France).

Table 1.1: Power production from gas for some European countries[11]

Country	% gas to power	% power from gas
UK (2013)	25%	23%
Germany (2010)	24%	7%
France (2010)	20%	under 1%
Netherlands (2010)	41%	91%
Belgium (2010)	30%	0%

Industrial

In 2013, about 13% of the total gas demand in UK was industrial use. The largest users of gas in this group are the food/beverages. chemical, mineral product, and paper/printing industries.

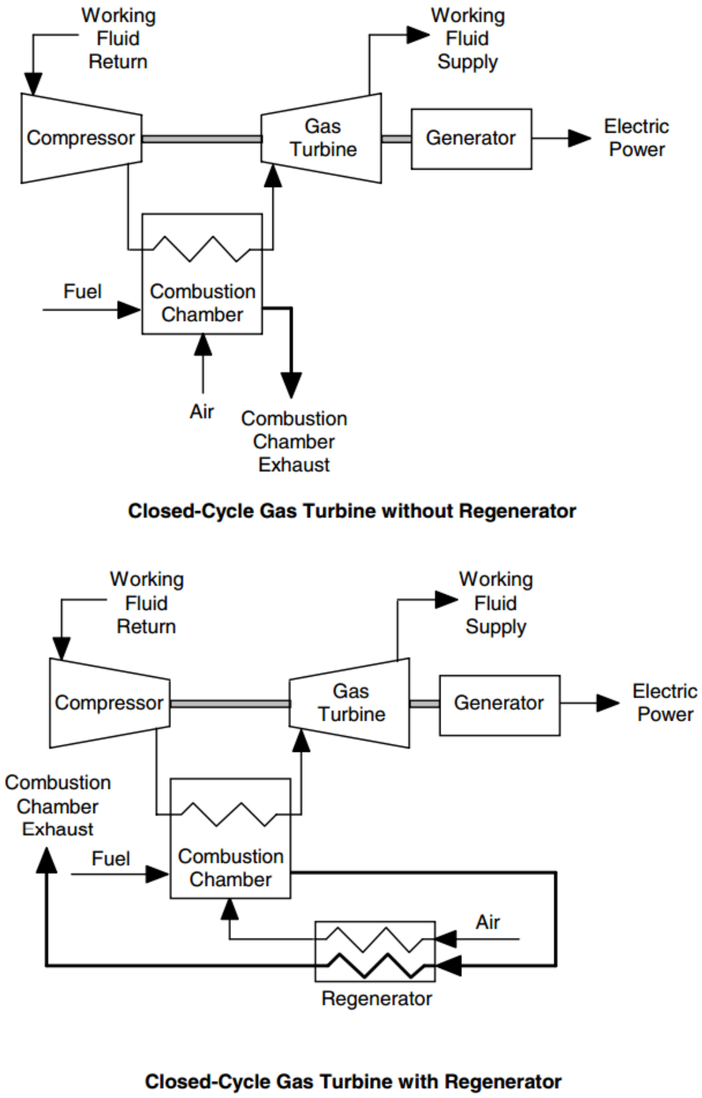


Figure 1.5: A combined cycle gas turbine, with/without CHP; Source: Oland 2004

1.2.3 Storage

Storage can be considered to be on both the supply- and demand-side.

On shore (and close to shore) there are storage facilities that help smooth out the supply to fit with the demand of gas. During low demand periods these storage facilities utilize low gas prices to inject gas for storage, to be used in periods of high demand (and hence, higher prices). These storage facilities range from storage tanks on land to caverns and previously depleted gas fields close to land.

In the European gas market there are 142 storage facilities of varying sizes and purposes (Eurogas 2013) - from LNG tanks above ground for handling short-term shortages (as described elsewhere), to underground salt caverns and aquifers able to store enough gas to minimize the impact of temporary changes in supply and/or demand, to depleted gas fields under the sea floor, used to handle the seasonal variation in demand - or, looking at it from the other side, being able to buy gas when the price is low, expecting to be able to sell it at a higher price.

A gas storage facility needs an initial amount of gas to establish the pressure needed to be able to withdraw gas, called the base gas. For some types of storage this is a major part of the cost of creating a new storage facility. When "depleted" gas fields are used, there's often a volume of gas left which helps in establishing the base gas, while aquifers and salt caverns require a huge investment of gas - up to 75% of the total volume - before they are viable for use.

The pressure inside the storage affects the possible injection and withdrawal rates, with a nearly full storage having a lower injection rate than an empty storage - and correspondingly a near empty storage has a lower withdrawal rate than a full storage.

Line pack

At all times, there is a volume og gas present in the gas distribution grid. This volume is referred to as the "line pack", and is necessary to maintain the pressure required to use the grid. In the UK (2013), this volume varies between 293 and 361 million Sm^3 depending on the season[12], with the most gas as line pack during winter where the temperatures are low - as Sm^3 is defined at a specific temperature, and pressure is temperature dependent.

As mentioned regarding the pipelines, the compressors increase the pressure at one end of the pipeline to enable extraction at the other end - in a way, each pipeline is a storage with forced withdrawal at one end when gas is injected at the other, though that distinction is not used in this thesis.

Like the base gas in a storage, the line pack needs to be injected before a grid is usable.

[12]National Grid 2014b.

Chapter 2

Existing literature

In the paper *Applied Mathematical Programming in Norwegian Petroleum Field* (2010) a lot of the work done in applied mathematical programming the last 30 years is summed up, and it was an excellent starting point for finding additional papers to work with. There are surprisingly few (public) papers on the gas transmission, as noted by Nygreen and Haugen 2010 "Still, research literature is relatively sparse on descriptions and reports of the above alleged model use, with some noteworthy exceptions."

Several papers exist on optimization of gas flow, and a fair guess would be that all the big gas companies have internal, proprietary models as industrial secrets. For this thesis I have based my work on a selection of papers which define the problem at various detail levels, for various time intervals and lengths, and with various *flavours* - in that they have different criteria for what is being optimized, and different focus areas as to how the network is modelled.

Based on the detail level and the and reliance on information that's usually not public, it seems likely that most of the papers are written by or for the actors involved in the market - and in many cases this is stated explicitly.

2.1 Model scale

Natural gas is a heterogeneous product, where different wells give different compositions of hydrocarbons. As the markets require a specific quality of gas, the processing plants have to separate the heavier hydrocarbons from the lighter so the product is as contractually expected. The model in Ulstein, Nygreen, and Sagli (2007) considers this and has constraints for each of the components of the product. As an example, they have the following constraint[1]:

> High levels of carbon dioxide may cause erosion problems and some countries impose environmental taxes on emissions of the gas. In Eq. (12) this quality restriction $q \in Q$ has $\delta_q^k = 1$ for $k = carbondioxide$ and $\delta_q^k = 0$ for all other components. The

[1] Ulstein, Nygreen, and Sagli 2007, page 557.

maximum relative carbon dioxide content in the flow is limited by the upper bound u_{ijq}. There is no lower bound.

Tomasgard et al. 2007 present a software package, GASSOPT, developed in co-operation with the operators GASSCO and STATOIL, that can be used on a problem of the same scale - the example model they show[2] covers the Norwegian Continental Shelf. An illustration by Ulstein, Nygreen, and Sagli (2007, fig. 7, page 559) shows a comparable network, effectively covering the GASSCO operated gas network.

The main difference between the two models, is that Tomasgard et al. (2007) model the compression of gas in the pipelines and more of the physical properties of the system; the pressure in the pipelines is not linear, and the paper describes an equation for calculating the flow in a pipeline as a function of the pressure difference of the input and output end of the pipeline. This is called the Weymouth equation, given as follows by Tomasgard et al. (2007, page 12):

> In the Weymouth equation $W_{ij}(p_{ij}^{in}, p_{ij}^{out})$ is the flow through a pipeline going from node i to node j as a consequence of the pressure difference between p_{ij}^{in} and p_{ij}^{out}:
>
> $$W_{ij}(p_{ij}^{in}, p_{ij}^{out}) = K_{ij}^{W} \sqrt{{p_{ij}^{in}}^2 - {p_{ij}^{out}}^2}, j \in \mathcal{N}, i \in \mathcal{I}(n)$$
>
> Here K_{ij}^{W} is the Weymouth constant for the pipeline going from i to j. This constant depends among others on the pipelines length and its diameter and is used to relate the correct theoretical flow to the characteristics of the specific pipeline.

(here, $\mathcal{N}$ is given as the set of all nodes in the network, and $\mathcal{I}(n)$ is the set of nodes with pipelines going into node n)

At a smaller scale, De Wolf and Smeers (2000a) describe a model of the internal gas flow in Belgium - which is behaves in a similar way: In the Belgian model, the terminal/import nodes were the producers, while the domestic consumers were the actual end points of the flow. De Wolf and Smeers (2000a, page 1456) shows basic model presented attempts to minimize the total cost of gas; the point of view is that of a company "(...) where the gas merchant and transmission functions are integrated in a single company"[3]. In a way, this is close to the approach taken in this thesis - there are no clear delineations between the companies, like a single big integrated company handling the total end-to-end operations.

Möller (2004) uses a small network as an example[4], containing two sources, three sinks, three compressors, and a few valves. Even so, the principles are generally the same as in a larger network. Like the other models, the model is also working with pressures, but on a smaller scale.

[2]Tomasgard et al. 2007, fig. 2, page 9.

[3]De Wolf and Smeers 2000a, page 1455.

[4]Möller 2004, fig. 4.12, page 52.

2.2 Time intervals and lengths

A major difference between the models described by papers is, as mentioned, how the time dimension is factored into the model - with the main distinction being whether time is factored in at all.

De Wolf and Smeers 2000a describes a linear program using an extension to the simplex method. The paper builds on earlier work by the same authors (De Wolf and Smeers 2000b). It is not clear from neither this paper nor the earlier works what kind of time scale the model operates on, in relation to resolution and length of modelling.

Ulstein, Nygreen, and Sagli 2007 describes their model as an *operational* model, for day-to-day planning from the point-of-view of the gas producer, and from the illustration shown by Ulstein, Nygreen, and Sagli (2007, fig. 7, page 559) plus the constraints, time is not shown as a factor. It is assumed that the model only considers the balance of the system for a single unit of time - a single day seems most likely, as typically operators in the market have to submit their *nominations* - requests for the transport of quantities of gas - for the next gas day[5] at a fixed time the day before.

Möller (2004) notes that at the detail level presented, that[6] "our implementation and first tests of the transient model showed that from the beginning there was no prospect of solving the full problem in an acceptable time range." The *transient model* would have the dimension of time, like the model presented in this thesis - but in the context of this paper, it would be far too detailed to feasibly run (or even set all the parameters), and what's solved is time independent (the *stationary case*) due to this complexity.

Closer to the same concept of time intervals and -lengths as the model in this thesis is, as stated by Tomasgard et al. (2007, page 7):

> To be able to handle the complexity needed for our models, we leave the concept of modelling the transient behaviour of natural gas and approximate the time dimension by discrete time periods of such length that steady-state descriptions of the flow will be adequate. When the time resolution of the model are months, weeks, and maybe days, rather than minutes and hours, we can assume that the system is in a steady-state in each time period.

2.3 Model quality

In all of the papers mentioned here, there's a lot of focus on the efficiency and general time taken by the models, but what is missing is a discussion of the quality of the results. Ulstein, Nygreen, and Sagli (2007, figure 7, page 559) shows a single data point with a good hit, and a few more figures discussing the result, but it's difficult to know the overall quality of the

[5]Gasunie Transport Services 2014.

[6]Möller 2004, page 114.

model based on this. The other papers have no qualitative measures usable for comparison.

Part III

The problem

Chapter 3

The problem

3.1 Analysis

Market analysts want to be able to forecast the behaviour of the market in both the short- and long-term perspective. The first step in being able to forecast, is to understand the current state of the market. However, when you're not a direct actor in the market, a lot of information about the market is hidden from view, as they are industrial secrets.

State-owned operators and EU data transparency laws makes the task of understanding the system slightly easier, though, and provide views of slices of the system - but much is still hidden.

As such, a model of the obscured parts of the system would be a great analysis tool, as it would let the system be analysed as a whole.

3.2 The question

This thesis is borne out of the first step mentioned above, starting with the question: *Is it feasible to create a functional model of the northern European gas market based on public and semi-public data?*

To clarify, public data here means freely available to anyone on a regular basis, with no monetary cost associated, and the distinction semi-public means that the data is available to anyone on a regular basis, but is delivered through a commercial/subscription based service. In other words, is it possible for someone who's not an actor in the market to gain relatively complete understanding of the system? Even actors in the market do not have perfect information, but with the extra information they have, it would be possible to predict the behaviour of competitors.

With *functional* it is meant that the model should be able to complete a scenario in short enough time to be usable for analysis. A model which takes 24 hours to run is already out-of-date when it completes, and not a usable model in practise.

A key assumption is that a model of the history can be turned into a good forecasting tool.

3.3 The assumptions

For implementing the model, unless proven otherwise, everything is based on the assumptions that the market is *fair* and attempting *optimal profit*. The former implies that the market is not driven by any monopolies, and that all competition is above-board and for the general good of the market place[1], and the latter implies that nobody, as the saying goes, "leaves money on the table". Barring the existence of long-term contracts unbalancing the market, all actors should behave in the way that lets them end up with the highest possible profit over time - *legally*.

3.4 Free variables

Two sets of variables are considered the free variables in this analysis: the gas flows per pipeline, and the storage levels (which, in essence, are implied by the flows in and out of the storage). The distribution of the gas given a certain price pattern (e.g. the forward curve for forecasting) and demand pattern will be a great tool for running scenario forecasts.

This is not an arbitrary choice, but rather rooted in the availability of data for forecasting; models for demand, production and prices already exist (and/or will be assisted by this model), while the flows and storages are significantly harder to set in advance.

3.5 Practicalities

A more detailed discussion of the data sources involved can be found in the next chapter. On a general basis, the arbitrary choice of modelling period was chosen as July 2012 to August 2013 inclusive. This 14 month period represents enough time for the largest gas storages to run a full cycle - that is, going from empty, to full, and back to empty again. The aforementioned forecast model would need to have a horizon of at least 14 months to replace existing models. This also goes into the definition of *functional* above - while it's hard to quantify "how short is short time", as a rule of thumb a regular desktop computer should be able to run the model for these 14 months in less than 30 minutes. Of course, hardware improves steadily, and computational power can be rented by the hour these days, so this requirement is not set in stone. Looking at it from the other side, given better algorithms and/or better hardware, some of the simplifying assumptions could be dropped to provide a better analysis in the time allotted. The simplifications described in this thesis are largely modular - that is, they can be dropped individually, without changing other simplifications.

The entirety of the European gas system involves - in addition to the various domestic production and networks, and aforementioned

[1] Proving that previously undetected market manipulation occurs is outside the scope of this thesis.

Norwegian-based pipelines - a significant import by pipeline from Russia and Africa, and further imports by LNG carrier ships from the Middle East, Africa and the US. For the purpose of this thesis, a subset, here defined as the *northern European gas market* - UK, Belgium, the Netherlands, northern France and the north-western half of Germany[2], is modelled.

It is not practical to start out developing the model on the forecast, as any uncertainties in the forecasts of the input parameters - price, demands, production - would add to, or even magnify, the uncertainty inherent in the model. Coming from the other end, it's not practical to model all the physical properties of the model, since most of the details are not available from the operators of the system - and it seems prudent to choose a baseline in information used across the model, as discussed in the next chapter.

[2] see section 4.2 on page 31 for more details

Chapter 4

Input data

As mentioned, the model is based on public and semi-public data, though some concessions will have to be made to get the data on a level base for use in the model.

4.1 Data quality

Generally I am assuming that data from the respective countries' equivalent of a national department of energy, the data directly from the operators, and the data delivered through European Commission 2014 are correct - and that no better data is accessible outside of the operators themselves.

A general challenge in the making of this thesis is that the data is not in the resolution required, of which the largest problem being the production data from the Norwegian continental shelf (as provided by *Olje og Energidepartementet*). Production data only exists as monthly sums, and as the basic case, I have chosen to distribute the production evenly over the days of the month, but this may lead to problems, especially around the transition from one month to another, which is stair-stepped as illustrated in figure 4.1 on the following page. The chapter on results, specifically section 10.3.3 on page 80 discusses another variant of converting the monthly production aggregates to a daily production figure.

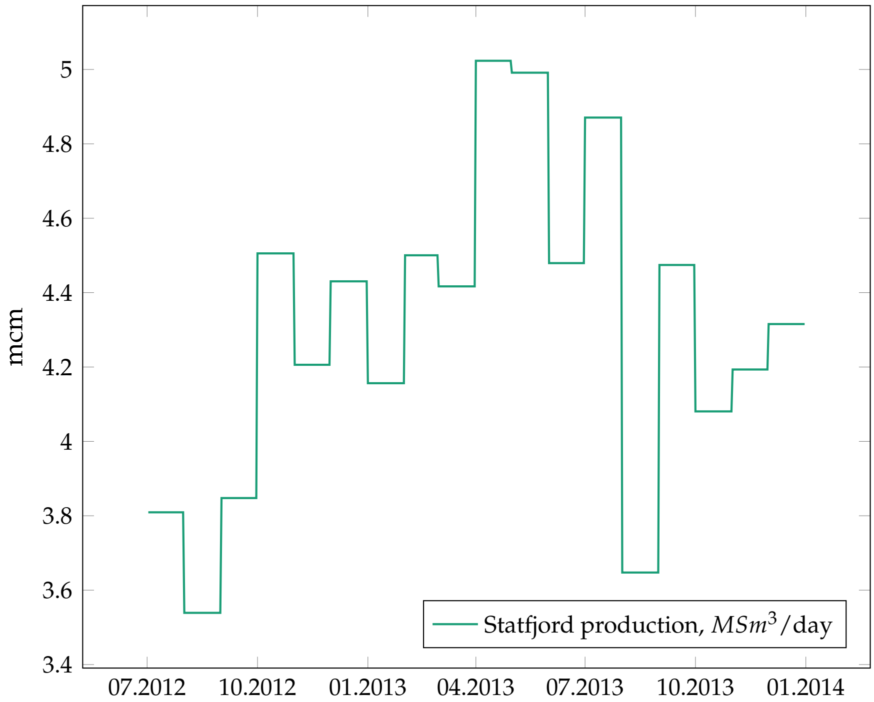


Figure 4.1: Production at the Statfjord platforms, daily average showing stair-stepping of values

The other big challenge is that there are no (public) data for most of the nodes of my network; the producers share little of what's going on in the hubs and risers in the North Sea. As this model is made by observing the network from the outside, I think the choices made here make sense. This also affected the choice to not go to the detail level modelled in for instance Tomasgard et al. 2007, as I don't have access to any information that would make it possible to model the pressure in the pipelines over the time frame the model is run over. I have also elected not to contact any of the operators asking for these details; it would be a huge undertaking to get details on a significant amount of the compressors in the system, and to verify that a model partially using compressors and partially plain linear pipelines actually is a valid simplification of the system.

For parts of the network where flow effectively is funnelled through a single node, like the gas fields feeding into Kårstø, I have put the aggregated production on the Kårstø node, to simplify the model[1]. As the North Sea part of the network is a standard linear network flow system, no precision would be gained from modelling the individual nodes, as there is no choice involved in where the gas has to flow.

As I am not modelling the intricacies of the different countries' internal gas network, I have replaced the entire grid for each country with a single node, with infinite capacity going in and out where appropriate, and linked the countries with a single pipeline offering the total capacity of the cross-border flows.

[1]See figure 6.2 on page 47

For some demand figures I have used modelled demands rather than reported, as the reported data either is too low resolution or missing/not updated.

While the gas flow is a combination of gas of various qualities - both wet and dry gas - all the gas that hits the market is processed to dry gas, so qualities are not taken into consideration. No public data on qualities, except some countries reporting the calorific value, are published.

4.2 Model limits

In addition to quality, the other factor here is the amount of data needed for the model to approach a realistic view of the network.

The model has a virtual southern border going across France - at the natural border between the northern and southern market areas, where the only pipeline connection is of limited capacity. The south-eastern border is loosely defined by a diagonal across Germany, from the French/Swiss/German triple border to the German/Polish/Baltic Sea triple border. For brevity I call this the *northern European* gas market, and this area is implied when I talk about the model in this thesis.

Numbers used have interactions with Switzerland, Austria, Czech Republic, and Poland subtracted - and, by extension, Russian and Algerian imports too.

With how the model is set up - as described in the next part - it can be extended to cover the entire European gas market at a later stage, given the figures for extra-EU imports, Mediterranean production, and the details on markets and pipelines.

Part IV

The model

Chapter 5

Mathematics, graph theory and optimization

For this chapter I will follow Wolsey 1998 for the most part, though the syntax will be substituted for the syntax used in this thesis where appropriate.

This chapter describes the concepts from graph theory and optimization used in the model for this thesis, and how they apply - the details of exactly how they apply can be found in chapter 6 on page 41, while the general theory can be found here.

5.1 Graphs

5.1.1 Directed graphs

The gas delivery system is, in this thesis, modelled as a directed graph $G = (V, E)$.

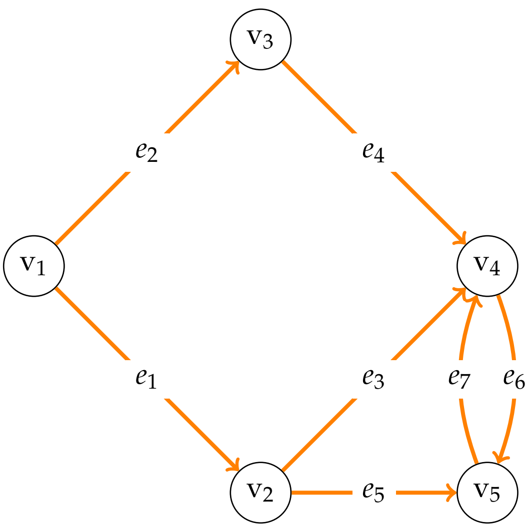


Figure 5.1: A simple directed graph

Here V is the set of nodes representing both physical entities like platforms and storage facilities, and representing logical entities like *the UK gas market* and *the sum of gas entering the EU system from Russia/Eastern Europe*.

The edges E represent the pipelines transporting the gas, both as an analogue to a physical pipeline, and theoretical pipelines to indicate how the logical entities in V are connected.

This system is a directed graph since each edge $e = (v_1, v_2) \in E$ where $v_1, v_2 \in V$ can only transport gas in one direction - in this notation, only from v_1 to v_2 where the real system does have some bidirectional pipelines, these have been represented by two edges (v_1, v_2) and (v_2, v_1). In figure 5.1 on the preceding page there's an example of this with edges e_6 and e_7.

5.1.2 Network flow

In a basic form of the network flow problem most relevant to this thesis, there is a directed graph, where each edge has a maximum flow capacity, and each vertex is either a source node (a producer), a neutral node (a hub) or a target node (a consumer).

A solution to the network flow problem consists of the flows that transfer the production to the consumers, where all the individual flows conform to the constrains placed by the capacities on the edges, and the flow out of each node is equal to the flow into the node - except for the producing and consuming nodes.

Simplified, the model described here could be represented as a separate network flow problem for each day of the modelling period - where some of the values from the model of a given day are used as inputs for the following day.

5.2 Optimization

5.2.1 Mixed-integer program

A mixed integer program (MIP) is a linear program with an additional restriction on some variables, requiring integer values.

A use for this in models is for a system where there is production of units of a product, where having a fractional product doesn't make sense, but there are other continuous factors in the system - for example start-up costs, fixed costs, transportation costs based on distance rather than number of units.

The main integer part of the model in this thesis is the modelling of the injection/withdrawal from the storages, as this relies on the current level of gas in the storage. This is modelled via a SOS2 system described in section 5.2.3 on page 38. As gas production and flows physically depend on pressure, all gas volumes are continuous variables - even the discrete LNG tankers are modelled as continuous values, as there are no significant limitations that suggest that integer variables would be a good fit.

A linear program is easier to solve than an integer program - before adding integer constraints, an early version of my model ran 50-100 times faster than the immediate next version with integer variables (though, of course, this version had more inequalities). Because of this, one often tries to solve the *linear relaxation* of the (mixed) integer program - where all the integer constraints are replaced by linear constraints. The optimal solution to this linear relaxation is not guaranteed to be anywhere near an optimal solution for the original MIP, but it can be used to find further inequalities that can be added to the problem to get closer to an optimal solution. Sufficiently big integer problems may not be solvable in a reasonable amount of time, and one has to settle for the goal of finding a feasible solution to the integer program.

Binary integer program

A binary (or $0-1$) integer program restricts all variables to either 0 or 1. This kind of program is suited - among other things - to model problems that consist of binary choices, like picking items from a set to cover some overarching limitation. This thesis uses binary variables in SOS2 sets (discussed in more detail in section 5.2.3 on the following page) and to model LNG storages, with a variable to prevent the daily losses to be subtracted if the storage is empty.

5.2.2 Relaxations

Given an integer program

$$z = max\{c(x) : x \in X\} \tag{5.1a}$$
$$X = \{Ax \leq b, x \in \mathbb{Z}^n_+, A \in \mathbb{R}^{n,m}, b \in \mathbb{R}^m\} \tag{5.1b}$$

then the program

$$z_R = max\{c_R(x) : x \in X_R\} \tag{5.2}$$

where

$$X \subseteq X_R \tag{5.3a}$$
$$c(x) \leq c_R(x) \forall x \in X \tag{5.3b}$$

is a relaxation of the integer program in equation 5.1a.

Since we have the condition in equation 5.3b, this implies that $z \leq z_R$, that z_R is an upper bound on the value of z. Furthermore, if $x*$ is an optimal solution of the relaxation, and if $x* \in X$ and $c_R(x) = c(x)$, then $x*$ is an optimal solution of the original problem.

Relaxations are used implicitly and explicitly in this thesis to solve the model; for efficiency, LP software solve advanced optimisation problems by breaking them down, and solving relaxations to plan out the solution of the main problem P_0. The model described in chapter 7 on page 55 is a relaxation of the model described in chapter 6 on page 41 - not just a linear relaxation, where integer constraints are converted to linear constraints, but also a relaxation where groups of constraints are dropped.

To make the model give a feasible result in a reasonable time, a natural way of doing so is to solve a relaxation of the technically proper model, as the difference may not lead to a statistically significant added error in the solution - and at the same time, for some kinds of models, like the one described in this thesis, it's worth more to be able to run several scenarios with different inputs in short time, rather than running one perfect model to completion. In the model in this thesis, the value z does not represent any real-world figure, but is rather used as a means to a result; this result being the incumbent vector x representing the actual flows in the system.

5.2.3 Special Ordered Set of type 2

A *Special Ordered Set of type 2* is a way to approach a constraint that is not linear, but that can be reduced to or approximated by a piecewise linear function.

To define a SOS2 set[1], you have the SOS2 variables $\lambda_1, \lambda_2, \lambda_3 ... \lambda_n, 0 \leq \lambda_i \leq 1 \forall i = 1, 2, 3...n$ and the description of the curve in the constant vectors $\hat{x}$ and $\hat{y}$; $\hat{X} = \hat{x}_1\lambda_1 + \hat{x}_2\lambda_2 + ... + \hat{x}_n\lambda_n$ and $\hat{Y} = \hat{y}_1\lambda_1 + \hat{y}_2\lambda_2 + ... + \hat{y}_n\lambda_n$.

The inequalities for a SOS2 variables are:

$$\lambda_i \geq 0, i = 1, 2, 3...n \tag{5.4a}$$

$$\sum_{i=1}^{n} \lambda_i = 1 \tag{5.4b}$$

In addition, either one or two adjacent λ may be non-zero (e.g. λ_i and λ_{i+1}), which requires additional inequalities and new 0-1 variables $\delta_1, \delta_2, \delta_3 ... \delta_{n-1} \in 0, 1$. The inequalities are on the form:

$$\begin{aligned}
\lambda_1 &\leq \delta_1, \\
\lambda_2 &\leq \delta_1 + \delta_2, \\
\lambda_3 &\leq \delta_2 + \delta_3, \\
&\vdots \\
\lambda_{n-1} &\leq \delta_{n-2} + \delta_{n-1}, \\
\lambda_n &\leq \delta_{n-1}
\end{aligned}$$

Many LP solver software packages have simple functions to define SOS2 variables, where these inequalities are implied.

Non-linear functions like the example in figure 5.2 on the next page can result in a local maximum rather than the global maximum if solved by the simplex method.

[1]I will use the tautology "SOS2 set" to refer to the full system of variables, constants and inequalities for a single constraint in the mode - e.g. *Injection into storage* s *on day* i

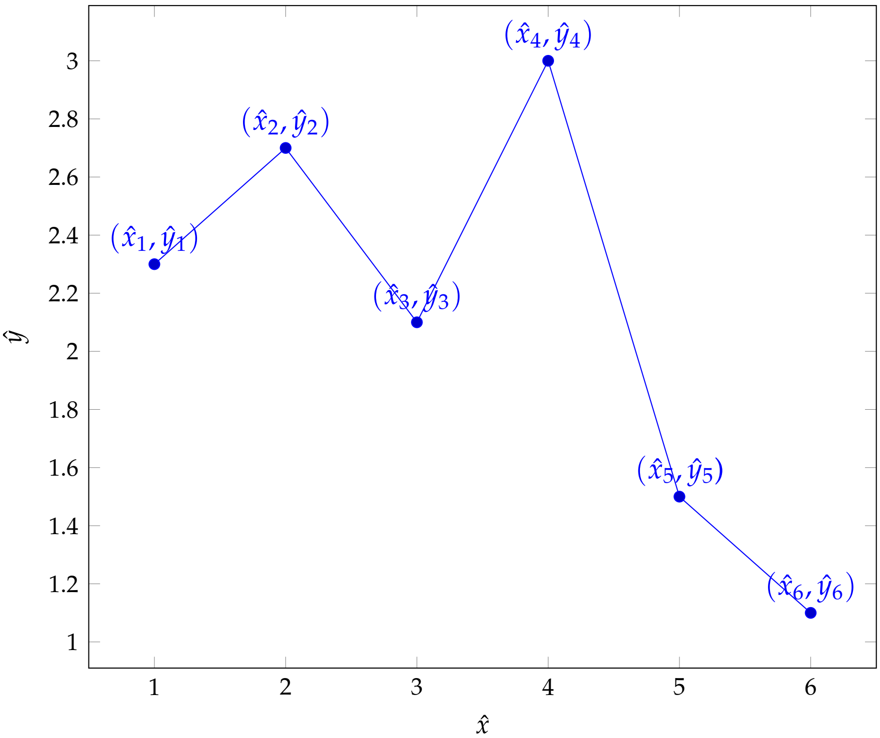


Figure 5.2: SOS2 coefficients for a non-convex/-concave function

As found in this thesis, adding SOS2 conditions significantly increases the complexity and running time of the model; this is due to the high number of integer variables and related inequalities added. Keeping the number of SOS2 groups down, and as few λ as possible, is key to keeping the model fast.

Special Ordered Sets of type 1

The main difference from SOS1 to SOS2 is that in SOS1 only a single λ can be non-zero, and thus $\lambda_i \in 0, 1 \forall i$ since $\sum_{i=1}^{n} \lambda_i = 1$.

These sets are used to solve other decision problems than SOS2 - where SOS2 does linear interpolation, SOS1 does discrete steps. An example would be finding the optimal size of packaging box for a product, where there are a number of discrete sizes to choose from; the product can only be in a single box, and one box will be the best size for the product. Another example would be choosing between production facilities, in cases where a production run needs to be done in its entirety in a single facility - choosing the smallest available production facility capable of producing enough for each run would be a SOS1 set, while distributing the run between at most two facilities would be a SOS2 set.

A SOS1 set would be faster than a SOS2 set since the adjacency inequalities and variables are no longer needed. SOS1 sets are not used in this thesis, but included here for completeness.

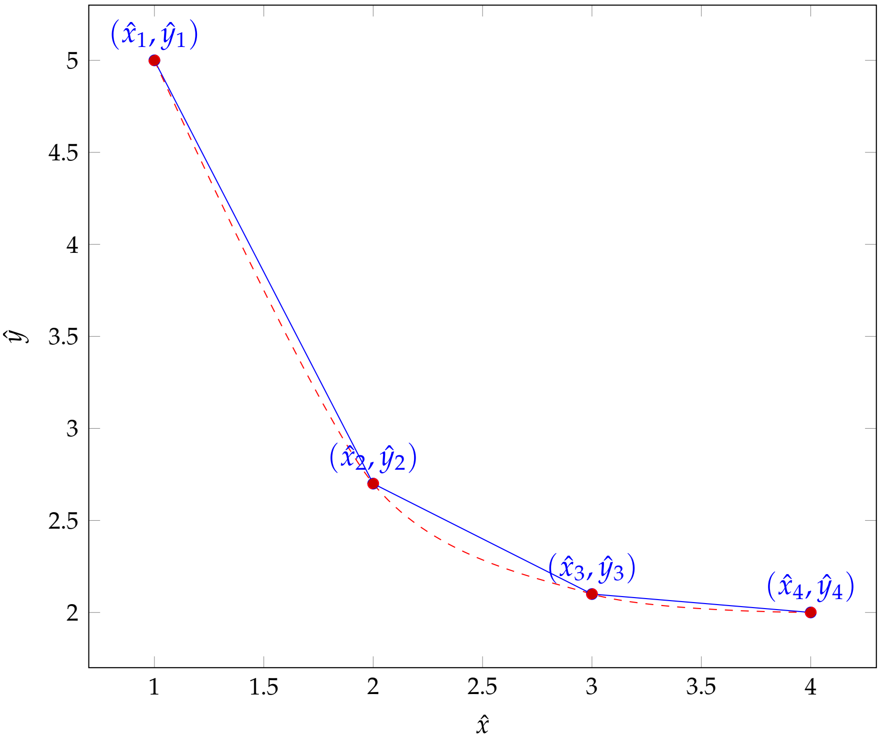


Figure 5.3: Estimating a nonlinear function by piecewise linear SOS2

Chapter 6

A MIP model for the gas network

6.1 The gas system as a model

Reducing the gas transportation system to a manageable model within the constraints of the use case for the model requires some simplifications and assumptions.

The real gas network is a complex graph where the flows are determined by the compressors on the transmitting end of pipeline, with pressure and gas quality determining the result received at the other end. In this model the facilities and pipelines are represented as a directed graph - the few bidirectional pipelines are set up as two unidirectional pipelines going opposite directions - so all flows x are positive.

As this model reduces the gas flow to a homogeneous product, no changes to the flow is assumed in any of the nodes outside of the producing nodes in F and the consuming nodes in M across all days I. However, on a given day $i \in I$ the storage nodes S may behave as a producer or consumer depending on whether the storage is in injection or withdrawal mode.

There is also no implied delays in the system; as the shortest unit of time in the model is a *day*, gas produced on that day can reach the market the same day. While this potentially leads to some inaccuracies for a given day of the model run, the overall aggregate is fairly consistent.

For practical purposes, this thesis will treat the level of the storage when the base gas is present as *empty* or 0, so the storage variables will only count the usable amount of gas (called the *working gas*). Initialization of a new storage is not covered by the model, but could feasibly be covered by defining a new storage with initial stock level < 0, or a new node type to simulate this initial demand of gas.

LNG storages are modelled as having a constant daily loss of stock due to boil-off, and ships come with fairly regular intervals, so we need to have room for the gas from each new ship as they come in. The official numbers show a discrepancy between stock level changes and the amount of gas sent out, which for the purpose of this model is taken as the internal losses in the LNG storage. Looking at the values for the year 02.07.2012 to

01.01.2013 (which is the period used for the implementation of the model), I have looked at the reported negative send-out values minus the daily changes in the reported storage level, which gives a positive number when the storage gets an LNG delivery, and negative when there is a send-out. This gives some unexplained negative values, which I take to be the daily loss. This is especially clear with the figures from Isle of Grain - see figure 6.1 - where 62% of the daily losses fall in the range $[-0.5, -0.7]$ GWh.

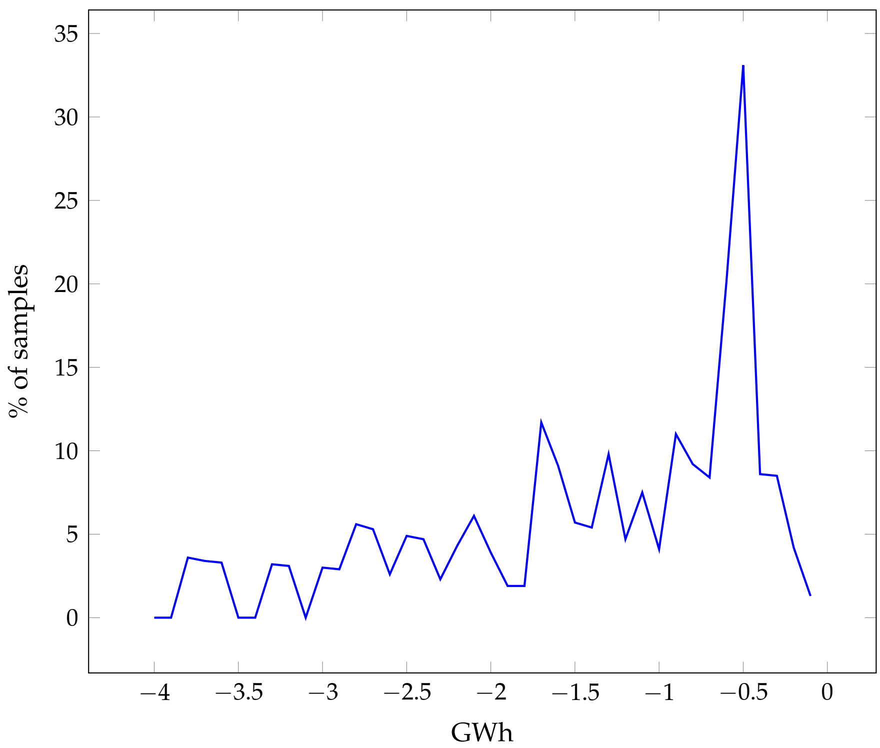


Figure 6.1: Distribution of assumed LNG losses at Isle of Grain, 02.07.2012-01.07.2013

A key part of the model is the behaviour of the storages at the end the model period; Tomasgard et al. 2007, page 30-33 argue for having an Expected Gas Value (EGV) to avoid the storages attempting to get the level as low as possible at the end. This is especially important for the large storages, where emptying the entire stock takes months, and leads to a ripple effect of wrong decisions early in the modelling period. Tomasgard et al. 2007 refers to a few ways to estimate the value at the horizon, and suggests using a stochastic model. I have chosen to go with using the price on the last day of the model period as a value for the entire stock, to simplify things. This choice is not perfect, it could be better to look at the forward curve at the end, and combine that with the horizon of the storage (i.e. how long does it take to fill it and then empty out completely, or the other way around) to set a potential value for the storage).

Without an EGV, the gas in the storage is valued at zero, and leads to a ripple of extra injection followed by complete withdrawal of the storage.

Setting the EGV too high will lead to extra filling at the end, while a too low EGV will lead to extra withdrawals prior to the end. In theory a perfect EGV would make the model possible to run in segments of days, e.g. running for the days $I_1 = 1...50$, then for the days $I_2 = 51...100$ and so on, using the results on the last day the previous run as the initial value for the next run. Such a model could be a valuable analytical tool, as parameters could be adjusted more or less on-the-fly, and several scenarios could be spun off from a short initial model run.

To mitigate the issues caused by imprecisions in the EGV, I am running the model for a few days extra - but solving a simplified version of the model on these days. The results from these days are discarded, so errors in the stock level due to simplified injection/withdrawal can be ignored. Ideally this extra period should be enough for a full cycle of the largest storage, but in practice the EGV should keep values in line enough that a shorter period can be used.

Line pack is not part of the model, but rather wrapped into the market node as a part of the demand side. As no new grids need to be line packed within the run-time of the model, variations in the line pack follow the seasonal and temperature dependent demand, day-to-day changes are minimal.

6.2 Definitions

The gas network is described as a directed graph $G = (V, E)$ with nodes in one of several categories described below

6.2.1 Generic properties and variables

I the set of days to run the model for, $i \in I$ being a given day, typically $I = 1...365$

$init$ as a subscript refers to the day preceding I

$last$ as a subscript refers to the last day in I

x_i^e the flows on the pipelines, $x_i^e \geq 0 \forall e \in E, i \in I$

y_i^v the level of the gas storage facilities, $y_i^v \geq 0 \forall v \in S, i \in I$

y_{init}^v the initial level of the gas storage facilities, $y_{init}^v \geq 0 \forall v \in S$

6.2.2 The physical elements of the graph

$F \subseteq V$ the set of facilities producing or distributing gas (platforms, processing plants, receiving terminals, hubs).

$S \subseteq V$ the set of gas storage facilities.

$H \subseteq V$ the set of ports receiving LNG by ship.

$L \subseteq S$ the subset of gas storage facilities used for LNG - have some extra constraints to handle boil-off.

$M \subseteq V$ the set of markets where gas is consumed.

$P \subseteq E$ the set of pipelines not connected to storages.

$\bar{P} \subseteq E$ the set of pipelines connected to storages.

6.2.3 The properties of the graph

c^v the maximum flow for pipeline $v \in P \cup \bar{P}$.

c^v the maximum capacities for storage $v \in S$.

g^v the daily boil-off/other losses at storage $v \in S$.

t_i^v the production/extraction of gas at facility $v \in F$ on day $i \in I$.

d_i^v the demand for gas in market $v \in M$ on day $i \in I$.

b_i^v the price for gas in market or storage $v \in M \cup S$ on day $i \in I$. For convenience, each storage is assigned a market price.

b_{EV}^v the expected value for a unit of gas left in a storage $v \in S$ at the end of I

q_i^v LNG delivered or exported by ship to/from port $v \in H$ on day $i \in I$.

6.2.4 Subsets of the above

δ_+^v the set of edges $v \in E$ going out of node $v \in V$

δ_-^v the set of edges $v \in E$ going into node $v \in V$

6.2.5 Extra variables

For the SOS2 sets, there are a number of points defining the piecewise linear curve that is approximated by the set. For this model, each SOS2 system uses the same number of points for all the sets contained within. For the regular storages, this number is referred to by the symbol A, and for LNG A_{LNG} is used.

The variable types in each set are described in detail in section 5.2.3 on page 38.

A *SOS2* number of constants.

$y_{SOS2inj}^a, a = 1...A$ *SOS2* constants for storage levels, implemented as a fraction $\frac{y^s}{c^s}$ - this set is for injection to the storage.

$x_{SOS2inj}^a, a = 1...A$ *SOS2* constants for the max flow at the storage level at the corresponding y_{SOS2}^a, implemented as a fraction $\frac{maxinjectionatstoragelevel}{maxinjection}$.

$y^a_{SOS2wdr}, a = 1...A$ *SOS2* constants for storage levels, implemented as a fraction $\frac{y^s}{c^s}$ - this set is for withdrawal from the storage.

$x^a_{SOS2wdr}, a = 1...A$ *SOS2* constants for the max flow at the storage level at the corresponding y^a_{SOS2}, implemented as a fraction $\frac{maxwithdrawalatstoragelevel}{maxwithdrawal}$ - withdrawal.

$\lambda^{(s,inj)}_{(a,i)} \in [0,1]^{|S||I|A}$ The set of A *SOS2* variables for each storage/day, injection.

$\alpha^{(s,inj)}_{(a,i)} \in \{0,1\}^{|S||I|(A-1)}$ The set of $A-1$ *SOS2* adjacency variables for each storage/day, injection.

$\lambda^{(s,wdr)}_{(a,i)} \in [0,1]^{|S||I|A}$ The set of A *SOS2* variables for each storage/day, withdrawal.

$\alpha^{(s,wdr)}_{(a,i)} \in \{0,1\}^{|S||I|(A-1)}$ The set of $A-1$ *SOS2* adjacency variables for each storage/day, withdrawal.

A_{LNG} *SOS2* number of constants.

$y^a_{LNG}, a = 1...A_{LNG}$ *SOS2* constants for storage levels, implemented as a number $[0, c^s]$.

$x^a_{LNG}, a = 1...A_{LNG}$ *SOS2* constants for the loss at the storage level at the corresponding y^a_{LNG}, will be in the range $[0, g^s]$.

$\gamma^s_{(a,i)} \in [0,1]^{|G||I|A_{LNG}}$ The set of A_{LNG} *SOS2* variables for each LNG storage/day.

$\beta^s_{(a,i)} \in [0,1]^{|G||I|(A_{LNG}-1)}$ The set of $A_{LNG}-1$ *SOS2* adjacency variables for each LNG storage/day.

6.3 The network

An example of what the network looks like is shown in figure 6.2 on page 47 - for the sake of readability, some nodes are omitted (most notably storage), and the position of nodes is not indicative of absolute geographic positioning - compare with figure 1.1 on page 10.

A detail of this network is shown in figure 6.3 on page 48, to explain how the network is laid out. Each country has it's own hub node representing the internal network in that country. *NTS* is the actual name for the *UK-hub*, so that name is used directly, otherwise I use the general terminology *YY-hub*, where *YY* is the two-letter ISO 3166-1 country code. Likewise, cross-border flows are represented by pipelines from hub to hub, except where a more detailed picture is needed - like how *BBL* and *Interconnector* are represented. Generally the official and/or commonly recognized names are used for pipelines and nodes where appropriate.

Zeebrugge LNG is modelled separate from Zeebrugge since it needed the extra node for LNG storage. As the Zeebrugge-to-BE-hub pipeline is modelled as having infinite capacity, attaching *Zeebrugge LNG storage withdrawal* directly to *BE-hub* is functionally equivalent to attaching it to *Zeebrugge*.

Production where there is no forking/decision as to where the gas is transported is placed on a node close to the hub of the country in question - as an example, the domestic production of the Netherlands, is modelled as being produced in the Balgzand node.

6.4 Equalities and inequalities

6.4.1 Basic network flow

Conservation of mass in production and distribution nodes

Every day i, the difference between flow out of the node and flow into the node is the production in the node.

$$\sum_{a\in\delta^f_+} x^a_i - \sum_{b\in\delta^f_-} x^b_i = t^f_i, f \in F, i \in I \tag{6.1}$$

Here we are assuming no loss in the node, and that production is counted as the exportable volume of gas from the production, with any production related losses already subtracted.

A potential improvement here would be to add in a loss element in each node, so that the implicit cost of sending gas around is covered. On the scale of the system as modelled here, retrieving realistic loss coefficients may be difficult or impossible, as no such figures - as far as I can tell - are published. Tomasgard et al. 2007, page 19 consider the various petrochemical components of the gas, and how the processing plants extract these components, which could be considered a kind of loss, but the node equation[1] is equivalent to the one specified here.

De Wolf and Smeers 2000a, page 1456 have no modelled loss in the nodes, but rather, like Tomasgard et al. 2007, have a focus on the compressors and the gas pressure they create in the pipelines.

Ulstein, Nygreen, and Sagli 2007 refer to the losses as fuel gas consumption in the compressors, and that it can be modelled as a linear approximation based on the gas flow, and that[2] "As Fig. 5 shows, linearisation of the fuel gas consumption introduces errors, but this is considered acceptable as fuel gas consumption is less than 2% of the total gas production."

The model discussed by Möller 2004 also models a loss of gas as fuel in the compressors - and indeed has as the objective function to minimize this fuel loss[3] - but even so the equations for the other nodes in his model are

[1] Tomasgard et al. 2007, page 11.
[2] Ulstein, Nygreen, and Sagli 2007, figure 5, page 558.
[3] Möller 2004, page 19.

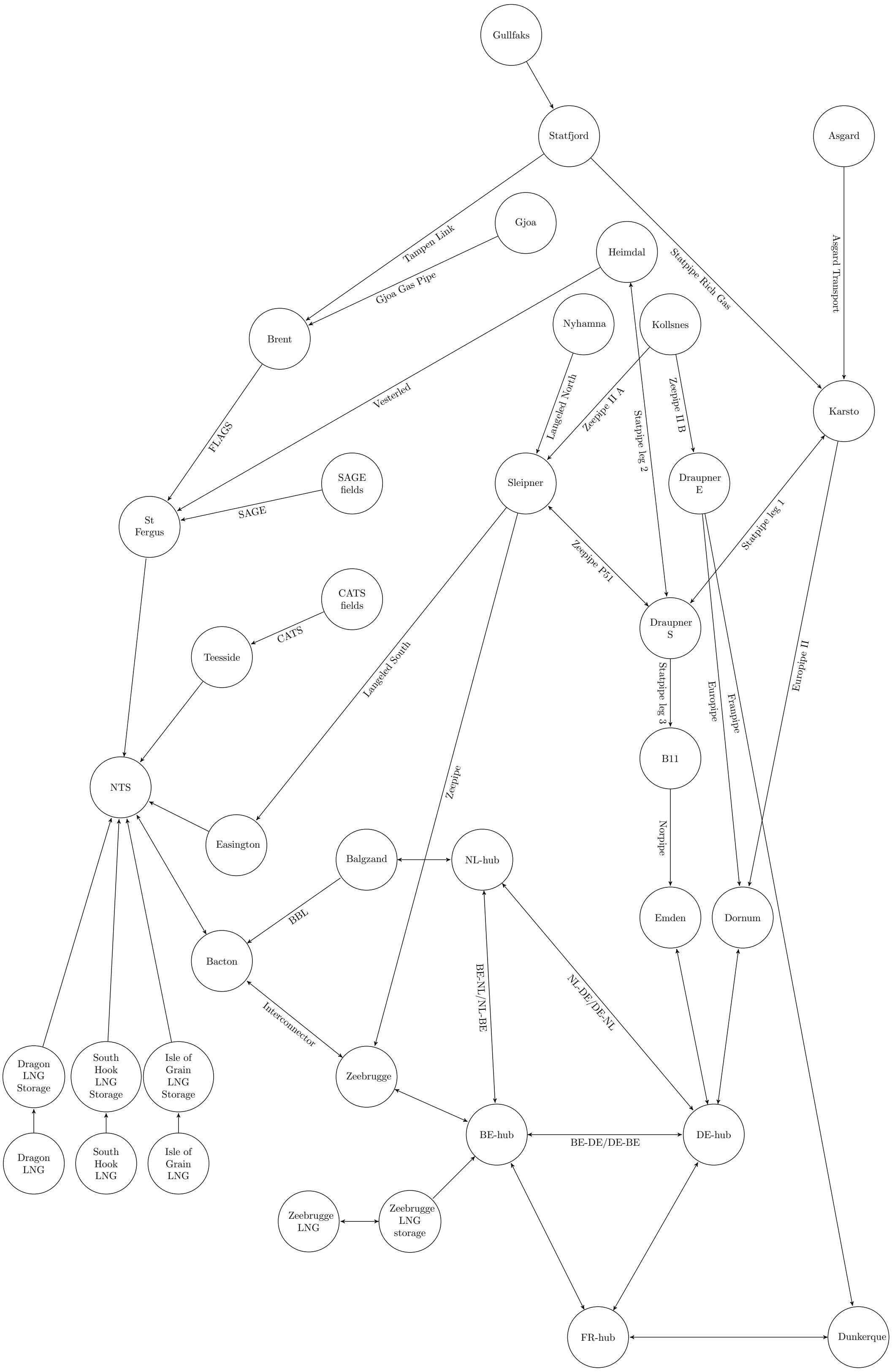


Figure 6.2: Network without storage nodes - rough geographic positioning

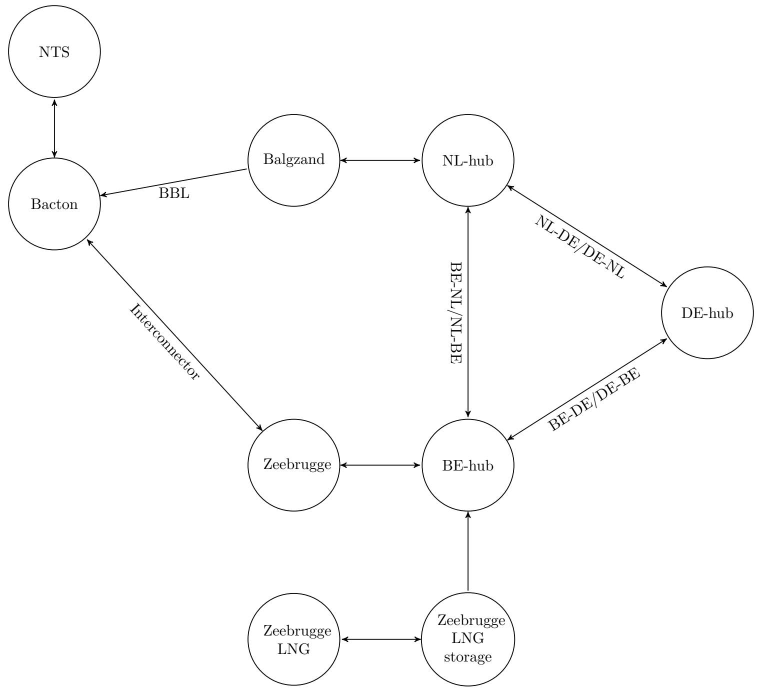


Figure 6.3: Detail of network model, BE/NL/UK

without loss. The model goes into details of the mechanics of the network to such a degree that using a similar model for this size of network is not practically feasible, but adding a linear loss would be possible - as shown in section 8.2.3 on page 64. It was, however skipped in the initial version of the model, as no clear figures other than the above quote from Ulstein, Nygreen, and Sagli 2007 have been found.

Conservation of mass in the market nodes

Each day i, flow to the sink nodes should cover the demand of that particular market.

$$\sum_{a \in \delta_{-}^{m}} x_i^a \geq d_i^m, m \in M, i \in I \tag{6.2}$$

In this model, we're assuming the demand and price are constants for a given day, and the model's job is to handle the supply side, which allows users of the model to run several scenarios to test the volatility in the system for various values of these input variables. We're assuming, for the purposes of this model, that changes in supply do not affect neither price nor demand.

Conservation of mass in storages

This inequality makes sure that today's difference in flow plus yesterday's stock should equal today's stock.

$$\sum_{a \in \delta_+^s} x_i^a - \sum_{b \in \delta_-^s} x_i^b + y_{i-1}^s = y_i^s, s \in S, i \in I \tag{6.3}$$

Non-LNG storages do not have significant reported losses, so we can for this model assume that the stock doesn't disappear.

6.4.2 Limits

Maintain limits on storages

Making sure the storages don't exceed their capacity.

$$0 \leq y_i^s \leq c^s, s \in S, i \in I \tag{6.4}$$

Here, the 0 level represents the minimum amount we can reach when withdrawing. The storage will still contain the base gas required for the storage to work, and since this model won't cover opening of new storages, this base gas is never touched.

Maintain limits on non-storage pipelines

Pipelines not connected to storages have simple upper/lower bounds:

$$x_i^e \leq c^e, e \in P, i \in I \tag{6.5}$$

As discussed in regards to the conservation of mass in nodes, we're not modelling the compressors, nor the pressure differentials in the pipelines. At the 1-day scale of I it is assumed that the operators maintain the necessary pressure for the grid to function, and since all volumes are reported in Sm^3 (which is linked to a standard condition and pressure) the volumes are pressure independent for our purposes.

Maintain limits on pipelines for storage injection/withdrawal

The inequality for a simple linear storage is the following - which is used for the LNG storages:

$$x_i^e \leq c^e, e \in \bar{P}, i \in I \tag{6.6}$$

However for a more realistic injection curve the SOS2 inequalities are used - one SOS2 set for each storage, each day, and for injection and withdrawal. The inequalities follow a standard SOS2 set as described in section 5.2.3 on page 38. Starting out, we have the constraints on the regular SOS2 variables:

$$\sum_{a=1}^{A} \lambda_{(a,i)}^{(s,inj)} = 1, a \in 1...A, s \in S, i \in I \tag{6.7}$$

And the constraints using the adjacency variables, making sure that no more than two λ are non-zero, and that they are adjacent:

$$\begin{aligned}
\lambda_{(1,i)}^{(s,inj)} &\leq \alpha_{(1,i)}^{(s,inj)} \\
\lambda_{(2,i)}^{(s,inj)} &\leq \alpha_{(1,i)}^{(s,inj)} + \alpha_{(2,i)}^{(s,inj)} \\
\lambda_{(3,i)}^{(s,inj)} &\leq \alpha_{(2,i)}^{(s,inj)} + \alpha_{(3,i)}^{(s,inj)} \\
&\vdots \\
\lambda_{(A,i)}^{(s,inj)} &\leq \alpha_{(A-1,i)}^{(s,inj)} \\
&s \in S, i \in I
\end{aligned} \tag{6.8}$$

The constraints that do the actual model work start with the constraint to "select" the λs according to storage level:

$$\sum_{a=1}^{A} \lambda_{(a,i)}^{(s,inj)} y_{SOS2inj}^{a} = \frac{y_i^s}{c^s}, s \in S, i \in I \tag{6.9}$$

And the constraint to limit the flow according to the selected λs in the previous equation:

$$x_i^e \leq \sum_{a=1}^{A} c^e \lambda_{(a,i)}^{(s,inj)} x_{SOS2inj}^{a}, s \in S, e \in P, i \in I \tag{6.10}$$

And of course an equal set for withdrawal (included for completeness):

$$\sum_{a=1}^{A} \lambda_{(a,i)}^{(s,wdr)} = 1 \tag{6.11}$$

$$\begin{aligned}
\lambda_{(1,i)}^{(s,wdr)} &\leq \alpha_{(1,i)}^{(s,wdr)} \\
\lambda_{(2,i)}^{(s,wdr)} &\leq \alpha_{(1,i)}^{(s,wdr)} + \alpha_{(2,i)}^{(s,wdr)} \\
\lambda_{(3,i)}^{(s,wdr)} &\leq \alpha_{(2,i)}^{(s,wdr)} + \alpha_{(3,i)}^{(s,wdr)} \\
&\vdots \\
\lambda_{(A,i)}^{(s,wdr)} &\leq \alpha_{(A-1,i)}^{(s,wdr)} \\
&s \in S, i \in I
\end{aligned} \tag{6.12}$$

$$\sum_{a=1}^{A} \lambda_{(a,i)}^{(s,wdr)} y_{SOS2wdr}^{a} = \frac{y_i^s}{c^s}, s \in S, i \in I \tag{6.13}$$

$$x_i^e \leq \sum_{a=1}^{A} c^e \lambda_{(a,i)}^{(s,wdr)} x_{SOS2wdr}^{a}, s \in S, e \in P, i \in I \tag{6.14}$$

All in all, this lets us have varying injection and withdrawal rates depending on the storage level.

6.4.3 LNG

LNG uses the linear storage constraint, as the reported send-outs of the LNG storages seem to follow this linear usage, and injection typically fills a ship in a single day.

Handle incoming ships

Going with the reported gas flows and storage levels[4], the incoming ships unload their cargo of LNG within a single day, and likewise outgoing ships load in a single day.

$$\sum_{a \in \delta_+^v} x_i^a - \sum_{a \in \delta_-^v} x_i^a = q_i^v, v \in H, i \in I \tag{6.15}$$

This is solved by having infinite capacity on the pipelines going to and from the harbour, and having the real limitation on the pipelines from the LNG storages to the rest of the network.

Handle boil-off and losses

The losses in LNG storages can all be summarised as losses to operation and maintenance. To cool and maintain the pressure on the LNG, and to handle boil-off, gas is used. The operators report this as changes in the stock level without a matching send-out of gas from the facility, and from historical data we can see that these daily losses are relatively constant per day, and not related to the stock level. This constant loss is modelled as a single value for the storage across all the days I, with the SOS2 set making sure the losses don't exceed available gas, using the curve in figure 6.4. That is, for each storage s we have $A_{LNG} = 3$, and the coordinates are $(0,0)$, (g^s, g^s), and (c^s, g^s).

$$y_{i-1}^s - \sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s y_{LNG}^a g^s + \sum_{a \in \delta_-^s} x_i^a - \sum_{b \in \delta_+^s} x_i^b = y_i^s, s \in L, g \in G \tag{6.16}$$

$$\sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s = 1 \tag{6.17}$$

$$\sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s y_{LNG}^a = y_{i-1}^s \tag{6.18}$$

[4]*National Grid, supplementary reports* 2014.

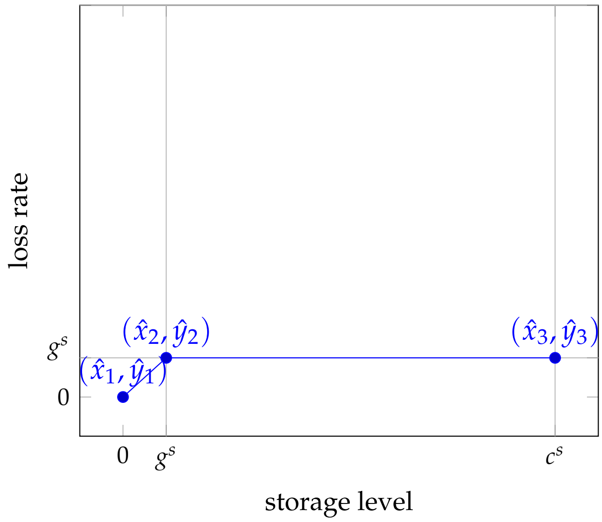


Figure 6.4: SOS2 set for LNG losses

$$\begin{aligned} \gamma^{(s,wdr)}_{(1,i)} &\leq \beta^s_{(1,i)} \\ \gamma^{(s,wdr)}_{(2,i)} &\leq \beta^s_{(1,i)} + \beta^s_{(2,i)} \\ \gamma^{(s,wdr)}_{(3,i)} &\leq \beta^s_{(2,i)} + \beta^s_{(3,i)} \\ &\vdots \\ \gamma^{(s,wdr)}_{(A,i)} &\leq \beta^s_{(A-1,i)} \end{aligned} \tag{6.19}$$

$$\sum_{a=1}^{A} \lambda^{(s,wdr)}_{(a,i)} y^a_{SOS2wdr} = \frac{y^s_i}{c^s} \tag{6.20}$$

6.5 Objective function

We want to maximize the *value of gas flowing into the different markets*, plus *maximize the revenues of the storage facilities* with some compensation for *gas left in the storage* at the end of the modelling period.

$$\begin{aligned} max \quad & \underbrace{\sum_{i \in I} \sum_{\substack{m \in M \\ e \in \delta^m_-}} b^m_i x^e_i}_{\text{Maximize sold gas value}} + \underbrace{\sum_{i \in I} \sum_{s \in S} \Big(\sum_{v \in \delta^s_+} b^s_i x^v_i - \sum_{w \in \delta^s_-} b^s_i x^w_i \Big)}_{\text{Maximize storage revenue}} \\ + & \underbrace{\sum_{s \in S} y^s_{last} b^s_{EV}}_{\text{Value of gas at end of modelling period}} \end{aligned} \tag{6.21}$$

This goes back to the basic assumptions in section 3.3 on page 26 - with these assumptions in mind this relatively simple objective function should be sufficient, and is in line with the objective functions used by Tomasgard et al. 2007, page 11 and Ulstein, Nygreen, and Sagli 2007, page 557, while De Wolf and Smeers 2000a, page 1456 has the objective to minimize operator cost. For my model I set b^s_{EV} to be equal to b^s_{last}. An argument can be made for this to be set to some future price the gas could be sold at given the stock level and the length of a full inject/withdraw cycle for the individual storage - Tomasgard et al. 2007, page 27 use a stochastic model to derive this future price - but since this model is running for a few months past the core analysis period, the effect of a wrong b^s_{EV} is diminished.

Chapter 7

A linear programming relaxation for the gas network

7.1 SOS2 for withdrawal and injection

The model in the previous chapter is a mixed-integer model, due to the SOS2 constraints and LNG loss modelling constraints. Looking at the SOS2 constraints, the shape of the chart of the SOS2 coordinates - as based on the reported withdrawal rates - is rather irregular, as shown for the storage at Hatfield Moor in figure 7.1 on the next page and Humbly Grove in figure 7.2 on the following page.

The line in the chart shows the initial SOS2 set I assigned for each storage to approximate the maximum. There's some uncertainty as to whether the few outliers for Humbly Grove (figure 7.2 on the next page) captured by the $(\hat{x}_3, \hat{y}_3), (\hat{x}_4, \hat{y}_4), (\hat{x}_5, \hat{y}_5)$ points should be included. I haven't had the opportunity to analyse the conditions leading up to these values, so I am going with the default assumption that since it's reported, it's possible to achieve this flow. It's also a bit unsure whether the maximum rate stays at the maximum possible all the way up to full storage; there's few samples going above the base line at the highest level, so I have kept it at maximum right now. Adding more points to reduce the rate at the top does not change the premise for this model, though.

The charts for injection look similar, though mirrored horizontally as the injection rate sinks when the storage is getting full. The drop-off is also a bit more abrupt, as exemplified by Hatfield Moor in figure 7.4 on page 57.

What was notable about these SOS2 constants, was that with this arrangement of the coordinates, it's close to a concave graph - the SOS2 coordinates almost describe a convex polygon. By adjusting the SOS2 constants to form an admittedly less accurate, but concave graph, the adjacency constraints in SOS2 can be dropped: As all the SOS2 sets are implicitly part of the (maximization) objective function, and the SOS2 coordinates are set up to describe a convex polygon, which means the convex combination of SOS2 λs won't exceed the original graph, but the optimal value will lie on the convex hull without the need for adjacency constraints (see figure 7.5 on page 58). For purposes of comparing with

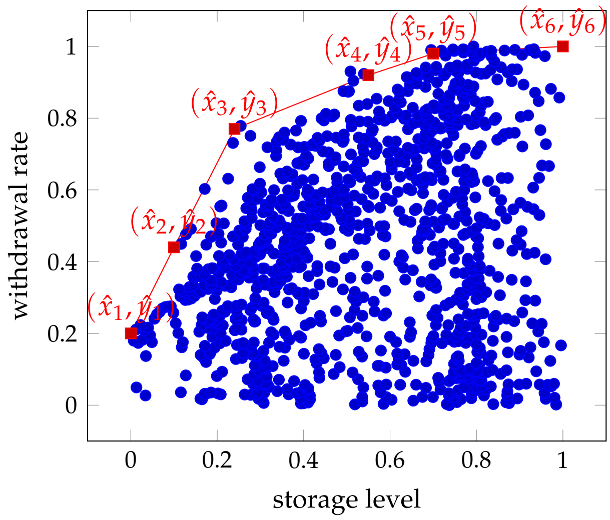


Figure 7.1: Base data for SOS2 constants, withdrawal, Hatfield Moor, 01.10.2007 to 31.03.2014

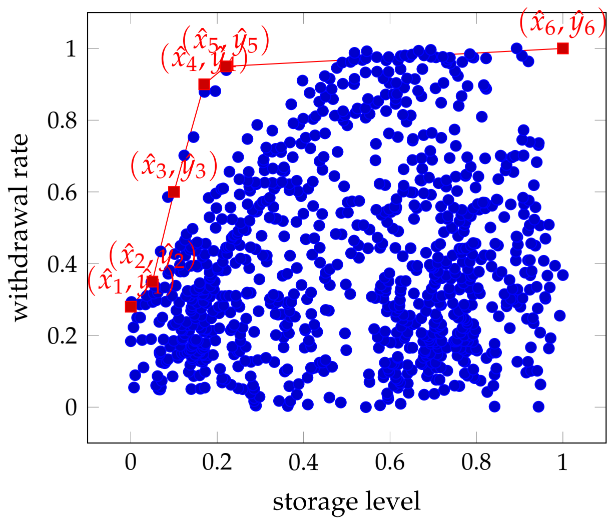


Figure 7.2: Base data for SOS2 constants, withdrawal, Humbly Grove, 01.10.2007 to 31.03.2014

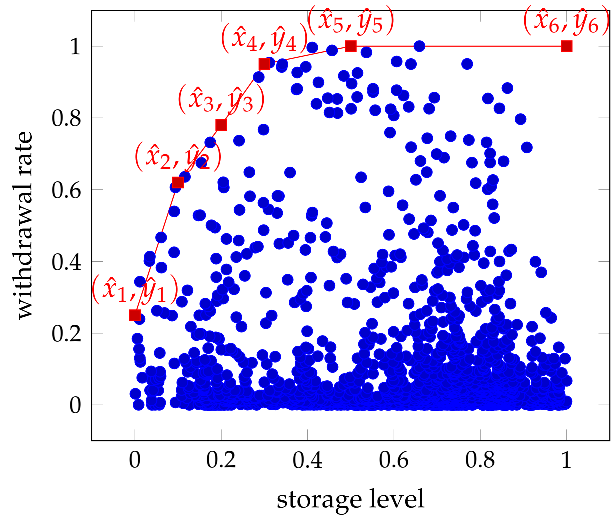


Figure 7.3: Base data for SOS2 constants, withdrawal, Hornsea, 01.10.2007 to 31.03.2014

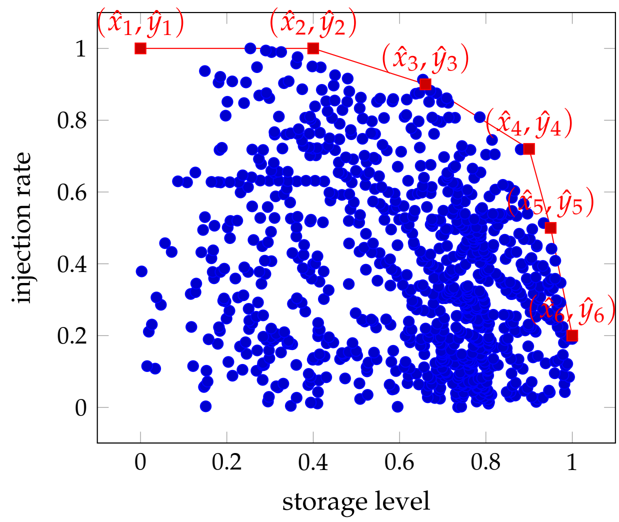


Figure 7.4: Base data for SOS2 constants, injection, Hatfield Moor, 01.10.2007 to 31.03.2014

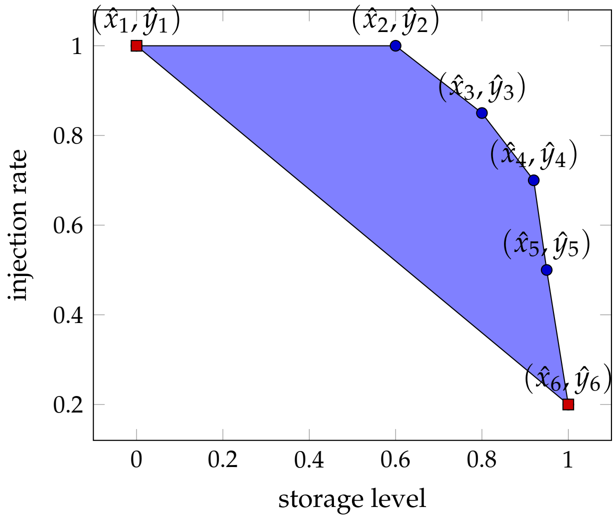


Figure 7.5: Convex polygon described by a concave SOS2 graph

the version including the adjacency constraints, I will refer to the version where they are removed as the *relaxed SOS2 system*.

This simplifies the model immensely, as there are $|I| \times |S| \times 2 \times (A - 1)$ binary adjacency variables in the SOS2 system of the original model. As pointed out by Tomasgard et al. 2007, page 17 "solution time will increase exponentially with the numbers of SOS2 sets needed."

This eliminates one set of integer variables from the model, plus the set of constraints 6.8 on page 50 and 6.12 on page 50.

7.2 SOS2 for LNG losses

The other SOS2 system are the γs and βs on the LNG storages. To eliminate the βs, we need to introduce a new constant; relaxed SOS2 only works when the variables follow the objective function - in a way, that the direction of the objective function pushes the variables to the right edge of the convex polygon. We solve this here by adding an element to maximize the LNG losses.

W_{LNG} The weight to give LNG losses in the objective function, to make sure they're not optimized away.

I use $W_{LNG} = 1000$ to ensure that the LNG losses are significant enough in the objective function.

This eliminates the last set of integer variables, and the constraints 6.19 on page 52.

7.3 New objective function

$$\begin{aligned} max \quad & \underbrace{\sum_{i \in I} \sum_{\substack{m \in M \\ e \in \delta_-^m}} b_i^m x_i^e}_{\text{Maximize sold gas value}} + \underbrace{\sum_{i \in I} \sum_{s \in S} \Big(\sum_{v \in \delta_+^s} b_i^s x_i^v - \sum_{w \in \delta_-^s} b_i^s x_i^w \Big)}_{\text{Maximize storage revenue}} \\ + & \underbrace{\sum_{s \in S} y_{last}^s b_{EV}^s}_{\text{Value of gas at end of modelling period}} + \underbrace{W_{LNG} \sum_{s \in G} \sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s}_{\text{Maximize LNG loss}} \end{aligned} \tag{7.1}$$

The objective function is largely unchanged, except for the new element to handle LNG loss.

In the next chapter the run time and precision of these two models are compared.

Chapter 8

Potential model extensions

The models above exhibited some specific behaviours when applied to real data, as the chapters on results will show. As a reaction to this, I propose a few extensions to the models to adjust the results. These extensions are presented here in the order they were tried, and possible combinations are specified. For discussion of the efficacy of each extension (and their combinations), see section 10.3 on page 77.

8.1 Weighting the storage element of the objective function

This came up in the comparison between the full and relaxed models: The element of the objective function that sums up gas revenue to market also counts the gas originating from storages, and the storages have to buy their gas at market price, while the model does not factor in the production costs at the gas fields (so the produced gas is essentially free), this means the element in the objective function that sums profit across storages is small compared to revenue element.

To mitigate this, I attempted to add a new constant, $W_{storage}$, which is functionally similar to W_{LNG}, weighting the storage element in the objective function:

$$\begin{aligned}
max \quad & \underbrace{\sum_{i \in I} \sum_{\substack{m \in M \\ e \in \delta_-^m}} b_i^m x_i^e}_{\text{Maximize sold gas value}} + \underbrace{\sum_{i \in I} \sum_{s \in S} \Big(\sum_{v \in \delta_+^s} b_i^s x_i^v - \sum_{w \in \delta_-^s} b_i^s x_i^w \Big)}_{\text{Maximize storage revenue}} \\
+ \quad & \underbrace{\sum_{s \in S} y_{last}^s b_{EV}^s}_{\text{Value of gas at end of modelling period}} + \underbrace{W_{storage} \sum_{s \in S} \sum_{a=1}^{A} \big(\lambda_{(a,i)}^{(s,inj)} + \lambda_{(a,i)}^{(s,wd)} \big)}_{\text{Maximize storage capacity}} \\
& \underbrace{+ W_{LNG} \sum_{s \in G} \sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s}_{\text{Maximize LNG loss}}
\end{aligned} \tag{8.1}$$

8.2 Damping variations

Most likely due to the data quality and resolution, the model behaves strangely when run with the reported monthly production figures divided evenly over the days of the month - the flows turn out very *binary*; alternating movement between two levels of flow.

An attempt to fix this was to add some constraints to limit the maximum variation between two days:

ξ the maximum day-to-day variation of all pipelines, as a fraction of the capacity; $0 < \xi \leq 1$.

$x_0^p = x_{init}^p$ The flow value of the day preceding the first day in I.

$$|x_{i-1}^p - x_i^p| \leq \xi c^p, p \in P, i \in I \tag{8.2}$$

Basically, the maximum variation from one day to the next was some percentage of the capacity of the pipeline - with the same constant used for all pipelines.

8.2.1 Dynamic production adjustments

As mentioned in chapter 4 on page 29, the production data is only presented as monthly sums. The initial model fixed this by using the average daily production for each day of the month, which lead to a jump in value on each month change.

To fix the values, some more variables and constraints were added:

$F_{mth} \subseteq F$ The subset of facilities where production is specified by month.

z_i^v The production/extraction at facility $v \in F_{mth}$ on day i.

$r \in R$ The months covered by the model.

$ri : I \rightarrow R$ A function to get the month containing the day i.

$i_r \subseteq I$ The days in month r.

t_r^v The production/extraction of gas at facility $v \in F_{mth}$ for month $r \in R$ stated as a daily average; that is: $t_{ri(i)}^v = t_i^v \forall i \in I, v \in F_{mth}$. item[$z_{free} \in [0,1]$] The fraction of the production considered flexible, as in not as associated gas or other forced production constraints.

$$\begin{gathered}(1 - z_{free}) t_{ri(i)}^v \leq z_i^v \leq \frac{1}{1 - z_{free}} t_{ri(i)}^v, i \in I, v \in F_{mth} \\ \sum_{i \in i_r} z_i^v = |i_r| t_r^v, v \in F_{mth}\end{gathered} \tag{8.3}$$

The first equation allows each day's production to vary the fraction z_{free} up/down from the monthly average - the simplified assumption is that about this much of the production is flexible on a general basis, while the remaining is forced production, for instance as associated gas. The second equation makes sure the month sum is still the same as before. The rationale for using the daily average production rather than the sum here, is that sometimes the model does not end cleanly on a month boundary; using $|i_r|t_r^v$ ensures the right sum even if only a single day of the month r is included in the model.

This extension can be used at the same time as the damping constraints.

8.2.2 Smoothness as an objective

Combining the idea and implementation of the dynamic production adjustment, with the damping constraints lead to the next extension: While huge day-to-day variations are technically possible in the gas system, the model variations are not near the actual observed values even if the average is. To tune this, I improved equation 8.2 on the preceding page by adding some new variables and constraints, plus a new element in the objective function.

New variables, replacing the ξ used above:

$P_{dmp} \subseteq P$ The subset of pipelines to reduce variations on.

$\xi_i^p \in [0,1]^{|P_{dmp}||I|}$ Parameters for the variation reduction of pipeline $p \in P_{dmp}$ for day $i \in I$.

W_{dmp} The weight used for these variables in the objective function.

Equation 8.2 on the facing page is replaced with an equation using these new ξs:

$$|x_{i-1}^p - x_i^p| \leq (1 - \xi_i^p)c^p, p \in P_{dmp}, i \in I \tag{8.4}$$

Here, the variation is reduced as ξ_i^p increases, so we maximize the sum of all ξs, with the new objective function:

$$\begin{aligned} max \quad & \underbrace{\sum_{i \in I} \sum_{\substack{m \in M \\ e \in \delta_-^m}} b_i^m x_i^e}_{\text{Maximize sold gas value}} + \underbrace{\sum_{i \in I} \sum_{s \in S} \Big(\sum_{v \in \delta_+^s} b_i^s x_i^v - \sum_{w \in \delta_-^s} b_i^s x_i^w \Big)}_{\text{Maximize storage revenue}} \\ + & \underbrace{\sum_{s \in S} y_{last}^s b_{EV}^s}_{\text{Value of gas at end of modelling period}} + \underbrace{W_{LNG} \sum_{s \in G} \sum_{a=1}^{A_{LNG}} \gamma_{(a,i)}^s}_{\text{Maximize LNG loss}} \\ + & \underbrace{W_{dmp} \sum_{\substack{p \in P_{dmp} \\ i \in I}} \xi}_{\text{Minimize curve variation}} \end{aligned} \tag{8.5}$$

For each pipeline, each day can have a different damping constraint. This should lead to overall tighter damping constraints, while allowing for some variation on days where there are no other feasible options.

This extension is of course mutually exclusive with the plain damping constraints, but can be combined with the dynamic production adjustments.

8.2.3 Node loss

As stated by Ulstein, Nygreen, and Sagli (2007, page 553) "Compressors increase the gas pressure to transport gas through the pipelines. The gas itself is used as fuel for the compressors."

A test implementation of node loss as a percentage of flow was implemented with a new constant:

K The fraction of gas lost in each transportation node $v \in F$.

The following changed mass conservation constraint replaces equation 6.1 on page 46:

$$(1 - K) \sum_{a \in \delta_+^f} x_i^a - \sum_{b \in \delta_-^f} x_i^b = t_i^f, f \in F, i \in I \tag{8.6}$$

Part V

Results and analysis

Chapter 9

Algorithms

9.1 Choice of software

The software choice is influenced by compatibility with existing systems, and general familiarity with the programming languages involved.

9.1.1 PYTHON

The PYTHON programming language[1] is a dynamic programming language developed since 1989(van Rossum 2014). It has libraries for interfacing with many optimization packages and programs, while letting you use the full extent of PYTHON syntax. In addition, due to the wide range of libraries for databases and web downloads, a model implemented in PYTHON can load and pre-process its own data before solving.

PYTHON is also extensively used in the system where the operational model will run, which means the implementation work necessary to put it in production is reduced.

9.1.2 COIN-OR

COmputational INfrastructure for Operations Research is a set of various software projects[2] pertaining to OR, and PULP comes bundled with their COIN-OR *Branch and Cut* MILP solver[3] (CBC)as a default solver. As such, it has been used as the default solver in this thesis for comparing the speed of various implementations, and results presented will be from this solver unless otherwise noted.

A problem occurred when the model reached a certain size, where CBC was unable to run it in a reasonable time. It turns out that the WINDOWS version of CBC does not support neither 64-bit nor multi threading, leading to a rather small limit on model size, and poor utilisation of the available resources. Furthermore, CBC is not as efficient as commercial solvers - at one point the single-threaded CBC used 56 minutes to find a solution,

[1] http://www.python.org/

[2] http://www.coin-or.org/projects/

[3] http://www.coin-or.org/projects/Cbc.xml

while the multi threaded CPLEX used 3 minutes (though the former ran on a regular desktop computer, and the latter ran on the university's server, getting free reign of the 24 cores available). As much of the model development was done without access to this CPLEX server (and the final result will most likely run on CBC or similar), my first solution was to dedicate a computer with the 64-bit LINUX version of CBC to running the model.

CBC 64-bit

However, since CBC is open source, I was able to download the source code and modify it so I could compile a multi-threaded 64-bit Windows version after some work. This version solves problems significantly faster than the regular 32-bit single-threaded version, and I elected to use it as a base-line. In the rest of this thesis I refer to this custom version as CBC64[4], while the more "official" released version is referred to as CBC32[5]. Though some trial and error, I found that on the 4 core computer I ran the model on, 6 threads were optimal - no significant gains in going above this, but it was slightly faster than 4 threads (as the CPU in question, an Intel i7 3770K, has HYPERTHREADING, which splits each physical core into two virtual cores). Unless anything else is specified, CBC64 is run on 6 threads.

Of note is that not all parts of the solution process are able to run in parallel, but for projects of a significant size - and that have a large number of integer variables to branch on like the first model presented in this thesis, it's a clear improvement, as shown in the next chapter.

For an LP model, CBC is always single-threaded, and the benefit of using CBC64 is minimal for models that fit in 32-bit address space (depending on OS, this typically means 2-4GB). 64-bit seems to have some speed benefits even single threaded - maybe due to how the CPU architecture works (access to more CPU registers); I did an informal test of my compiled CBC v2.8.9 64-bit versus the latest official binary version from the COIN-OR website, which is CBC v2.8.8 32-bit; for a given problem (an arbitrary MPS file for a run of my MILP model), that CBC64 solved in an average of 34 seconds over 3 runs, this CBC 32-bit used 44 seconds on average over 3 runs - and nothing in the change log[6] explains the difference.

Based on this, I generally use CBC64 even if the problem in question doesn't need multi-threading or large amounts of memory.

9.1.3 PULP

The choice to use PULP[7] is due to its ability to interface with multiple LP/OR software packages, both free/open source (Coin-OR and GLPK) and commercial (CPLEX and GUROBI). It's also able to produce standard MPS files for other LP solvers.

[4]specifically *CBC v2.8.9 64bit*, which was the most recent source available at the time
[5]specifically I am using *CBC v2.7.1 32bit*, which is packaged with PULP
[6]https://projects.coin-or.org/Cbc/log/
[7]https://code.google.com/p/pulp-or/

One issue with PuLP is that it has no direct support for SOS2 variables, so for my model I have implemented the SOS2 inequalities explicitly; regular PYTHON constructs let me generate all the inequalities with few and simple lines of code[8].

PuLP is also a COIN-OR project, and is packaged with a version of the CBC32 solver as mentioned above.

CPLEX

On the university Linux server, I could run my model through the commercial solver IBM CPLEX[9]. As this was not always available to me, and required a few extra steps, I have mostly run the model on CPLEX as a comparison for timing tests.

Unless otherwise specified, CPLEX runs with 24 threads as default. `/proc/cpuinfo` on the server reports 64 CPUs with 8 cores each, but I found no significant speed improvement in a 200-day test model going from 24 to 256 threads[10].

The university server runs an old version of PYTHON, and required a lot of workarounds before PuLP will run, so the timing tests were performed with the MPS files generated by PuLP on my local computer.

Due to these limitations, I consider CBC64 my primary solver for this project.

9.2 Implementation choices

9.2.1 Data and code

While the model is capable of loading data from the sources in real-time, for this implementation I made the choice to pre-process the data by hand to make sure consecutive runs are deterministic in nature.

All data coming from external sources is fetched on a daily resolution for the period 02.07.2012 to 05.08.2013 (400 days, inclusive) where available. Data with lower temporal resolution is converted to daily resolution, typically by dividing evenly day-by-day. Price data is converted to the same currency/unit - in this case UK pence/therm, as an arbitrary choice, the model would work equivalently using EUR/MWh as this is a linear conversion.

Capacities, demands and production numbers are converted to *million* Sm^3 (mcm) where needed. If the calorific value is not specified, the average of $11.2kWh/Sm^3$[11] is generally assumed.

Variable names and similar use more expressive descriptions of their purpose rather than the mathematical notation described in chapter 6 on page 41, an example is from the generation of the SOS2 system:

[8] See listing 9.1 on the following page

[9] http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/

[10] The server may have limits for each user - I ran these outside of "prime time", but several users were logged on at the same time as me

[11] National Grid 2014a.

Listing 9.1: PYTHON code snippet - SOS2 system generation

```python
noadjacency = True
N = len(curve)
for n in src.storage:
    for d in days[1:num_core_days]: # days after core days use basic flow controls
        # withdraw
        s2w = LpVariable.dicts("sos2_lambda_%s_wd_%03d" % (n, d), range(N), 0.0, 1.0, cat=LpContinuous)
        # SOS2 conditions
        prob += lpSum(s2w) == 1.0 , "sum of SOS2 variables %s wd day %d" % (n, d)
        if not noadjacency:
            s2gw = LpVariable.dicts("sos2_delta_%s_wd_%03d" % (n, d), range(N-1), cat=LpBinary) # 1 shorter than s2
            prob += s2w[0] <= s2gw[0] , "adjacency start SOS2 variables %s wd day %d" % (n, d)
            for i in range(1, N-2):
                prob += s2w[i] <= s2gw[i-1] + s2gw[i], "adjacency %d SOS2 variables %s wd day %d" % (i, n, d)
            prob += s2w[N-1] <= s2gw[N-2], "adjacency end SOS2 variables %s wd day %d" % (n, d) # lpvariable.dicts does not support negative indexing
            prob += lpSum(s2gw) == 1.0 , "sum of SOS2 adjacency variables %s wd day %d" % (n, d)

        # flow conditions
        prob += lpSum([s2w[i]*curvewd[i][0] for i in range(N)]) == stock[(d-1, n)]/src.storage[n][1], "main SOS2 condition (select) %s wd day %d" % (n, d)
        prob += lpSum([routes[(d, (i, j))] for (i,j) in list(src.pipelines.values()) if i == n]) <= lpSum([src.storage[n][3]*s2w[i]*curvewd[i][1] for i in range(N)]), "main SOS2 condition (constrain)%s wd day %d" % (n, d)


        # inject
        s2i = LpVariable.dicts("sos2_lambda_%s_in_%03d" % (n, d), range(N), 0.0, 1.0, cat=
```

```
        LpContinuous)
    # SOS2 conditions
    prob += lpSum(s2i) == 1.0, "sum of SOS2 variables %s in day %d" % (n, d)
    if not noadjacency:
        s2gi = LpVariable.dicts("sos2_delta_%s_in_%03d" % (n, d), range(N-1), cat=LpBinary) # 1 shorter than s2
        prob += s2i[0] <= s2gi[0] , "adjacency start SOS2 variables %s in day %d" % (n, d)
        for i in range(1, N-2):
            prob += s2i[i] <= s2gi[i-1] + s2gi[i], "adjacency %d SOS2 variables %s in day %d" % (i, n, d)
        prob += s2i[N-1] <= s2gi[N-2], "adjacency end SOS2 variables %s in day %d" % (n, d) # lpvariable.dicts does not support negative indexing
        prob += lpSum(s2gi) == 1.0 , "sum of SOS2 adjacency variables %s in day %d" % (n, d)

    # flow conditions
    prob += lpSum([s2i[i]*curve[i][0] for i in range(N)]) == stock[(d-1, n)]/src.storage[n][1], "main SOS2 condition (select) %s in day %d" % (n, d)
    prob += lpSum([routes[(d, (i, j))] for (i,j) in src.pipelines.values() if j == n]) <= lpSum([src.storage[n][2]*s2i[i]*curve[i][1] for i in range(N)]), "main SOS2 condition (constrain)%s in day %d" % (n, d)

    if not noadjacency:
        sos2sets.append((s2w, s2gw, s2i, s2gi))
    else:
        sos2sets.append((s2w, None, s2i, None))
```

In the notation used in this thesis, `N` is A, `src.storage` is S, `s2w` is λ, `curve` is the set of pairs (y_{SOS2}, x_{SOS2}), and so on. In PuLP, constraints are defined on the form

```
prob += lpSum(X) == 1.0, "unique constraint description"
```

(i.e. "adding the named constraint to the problem `prob`") which is roughly equivalent to the formula

$$\sum_{x \in X} x = 1$$

The MPS files PuLP generates for CBC are compatible with CPLEX, which I have used for comparison of run time in the next chapter.

9.2.2 Solver parameters

Both solvers used lets the user tune a multitude of parameters pertaining to the strategy for solving MILP problems; the choice of which variables to branch on, how to do this branching, generating new constraints (”cuts”), heuristics, and other solver-specific strategies may affect the solution time for a given problem.

For the most part while solving the MILP version of my model, I used the default strategies. While in the process of comparing strategies and seeing if the MILP solution time could be improved, I found the LP solution to the problem, and did not pursue the parameters further.

Santos (2011) lists some of these parameters for CBC; I attempted to both reduce and increase limits around the default to see if any significant changes to the processing time occurred. In short, none of the parameters I tried lead to an improvement of solution time over the default, but I did not test every single parameter extensively. There might still be improvements over the defaults in tuning parameters, but in general it seems like the defaults are well chosen.

Chapter 10

Results

10.1 Model complexity with SOS2 adjacency constraints

10.1.1 Problem size and solution time

CBC reports that a particular iteration of the model, when run for $I = 1...100$, has 62846 rows and 60777 columns. The first passes reduce this to < 5000 rows and < 5000 columns. Most of the size comes from the MIP parts of the program, specifically the SOS2 system. The number of SOS2 steps A increases the size and time taken by the program in a fairly linear fashion when using CPLEX, but has a huge impact in CBC, as can be seen in table 10.1. For the minimal curve to give a realistic injection/withdrawal time, a minimum of 3 SOS2 steps are required, as shown in figure 10.1 on the next page, as two points are required to define the full flow capacity area, and the third to define the lowest flow capacity. It's possible to refine this by having the curve be more detailed in the first parts of I, and gradually get more coarse in the later parts - even falling back to the default non-SOS2 constraint. This would affect model precision, but also increase speed considerably.

The time it takes to get an optimal solution increases significantly with the size of I when running CBC, as is apparent in table 10.2. Note that, for some reason, CBC seems to get stuck not able to find a solution some times. This is especially obvious when comparing the solution times for CBC32 on *75 days* and *100 days*, which seem counter-intuitive. For some of the larger sets - note that this is with the full SOS2 system for all days - it didn't seem

Table 10.1: Model at various # of SOS2 steps (A), 50 days, run time (mm:ss)

A	CBC32	CPLEX
3	00:16	00:08
4	00:24	00:10
5	02:58	00:15
6	02:29	00:18
10	41:03	00:28

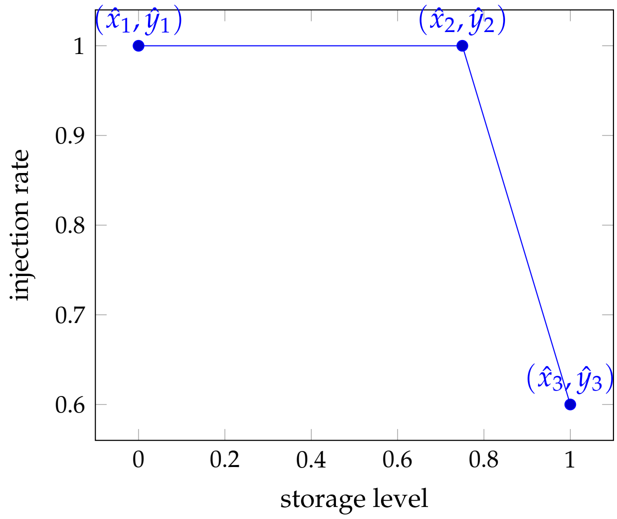


Figure 10.1: Minimal SOS2 constants to linearise storage injection rate

Table 10.2: Model complexity with $A = 3$, run time (mm:ss)

days	CBC32	CBC64	CPLEX
10	00:01	00:01	00:01
25	00:03	00:03	00:04
50	00:15	00:09	00:08
75	45:27	07:22	00:13
100	07:42	17:45	00:19
150	(n/a)[1]	(n/a)[2]	00:30
200	(n/a)	(n/a)	00:44
250	(n/a)	(n/a)	00:53

to get a solution at all, even when left running for more than 12 hours.

10.2 Model complexity with linear SOS2 constraints

After the initial model was developed, I spent some time trying to work out the general strategy employed by the solvers, hoping to figure out the extreme difference between CPLEX and CBC64.

This lead me to investigate the elimination of integer variables, which again lead to the linear programming model described in chapter 7 on page 55.

With this relaxed SOS2 system, I ran the same test as in table 10.1 on the previous page, varying the number of steps A in the SOS2 set - the results are in table 10.3.

This allowed me to use a more realistic curve than the one shown in figure 10.1, as the number of steps no longer was an issue - and I chose to use a 6 step curve like the one in figure 10.2 based on the average of UK storages, but there's also a possibility to use a separate curve for each storage. In the minimal example the withdrawal curve was a mirrored

Table 10.3: Model at various # of SOS2 steps (A), 50 days, run time (mm:ss), relaxed SOS2

A	CBC32
3	00:01
4	00:02
5	00:02
6	00:03
10	00:03
20	00:05
40	00:09
80	00:14

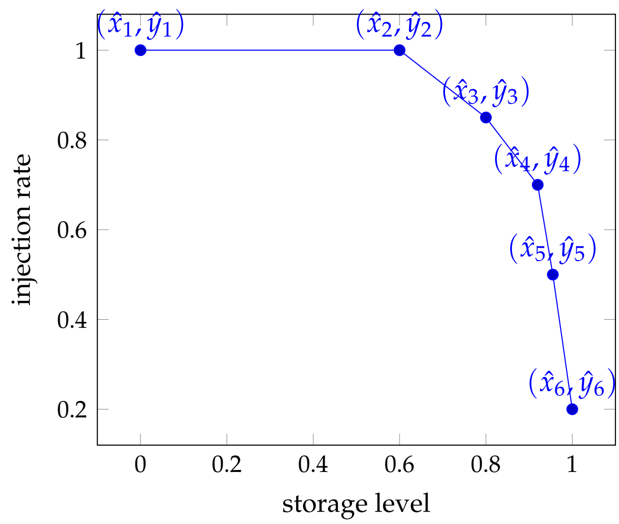


Figure 10.2: SOS2 constants to linearise storage injection rate, based on UK average

version of the injection curve, but with the relaxed SOS2 system's speed, it can be of any size and shape (as long as it's convex, of course), and I constructed one from the UK storages in the same way - as shown in figure 10.3. The reason for using UK storage data is due to it having the most detailed reporting, enabling the plot of storage level to flow rate as seen in figure 7.1 on page 56.

With the relaxed SOS2 system, the run time of a full 400 day model is below 15 minutes even with CBC32 (as the linear solver is single-threaded).

10.2.1 Differences between the full and relaxed SOS2 systems

Running a short test on both models, just to look at how much the change does for the results.

Figure 10.4 on the following page shows a 200-day run of the flow for the NCS part of the Easington terminal (i.e *Langeled South* flow) with $A = 4$; the values are mostly the same, except for some differences in the

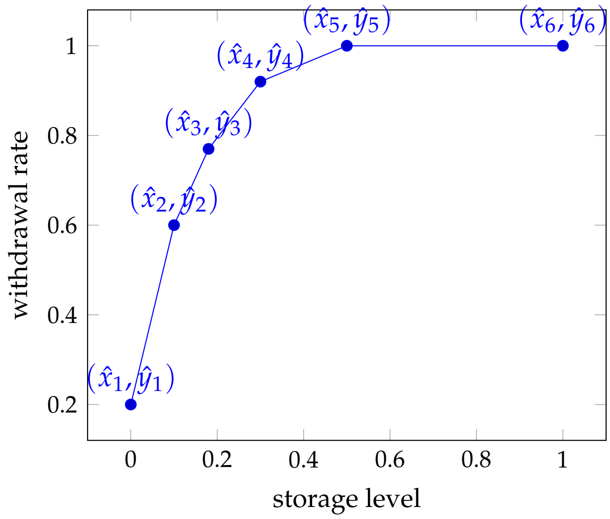


Figure 10.3: SOS2 constants to linearise storage withdrawal rate, based on UK average

August-September 2012 figures. There is larger differences in the numbers for Dornum (*Europipe II* specifically), where the relaxed SOS2 has smaller spikes, but a higher number of spikes overall, as shown in figure 10.5 on the next page.

The run time for these two tests was 71 minutes for the full SOS2 model, and 45 seconds for the relaxed SOS2 model.

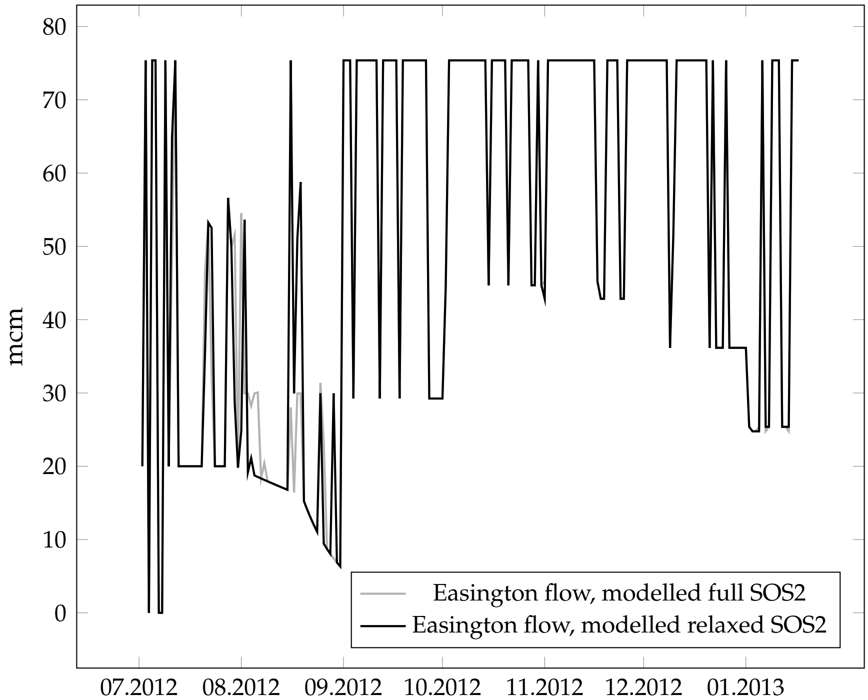


Figure 10.4: Flow at Easington, full SOS2 vs relaxed SOS2

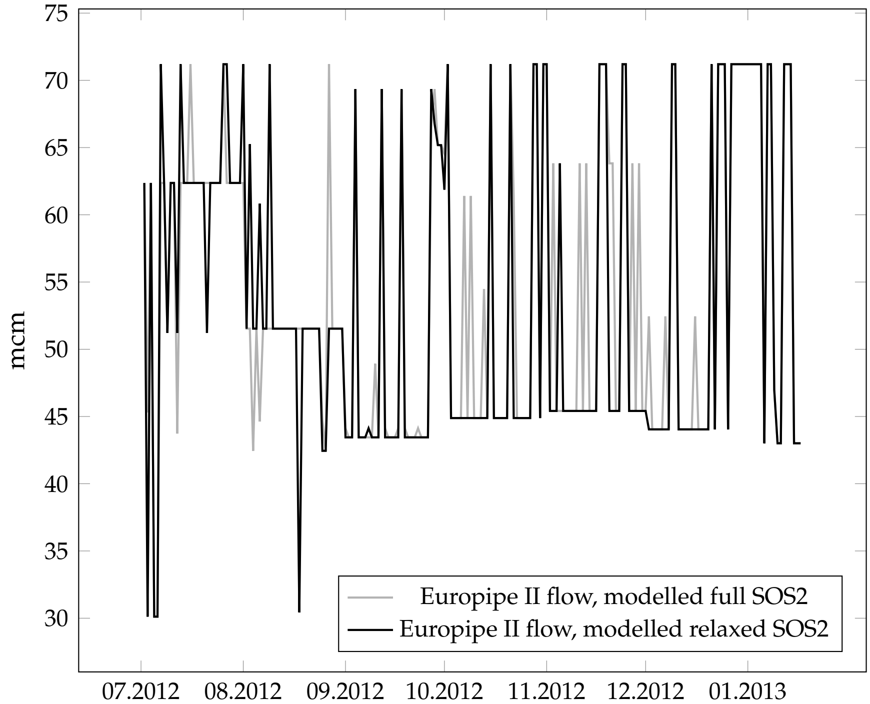


Figure 10.5: Flow at Dornum, full SOS2 vs relaxed SOS2

10.3 Extensions and efficacy

10.3.1 Weighting the storage element of the objective function

I found that by adding a weighting element to the objective function, I got less differences. I set $W_{storage} = W_{LNG} = 1000$, and got a seemingly more equal result with than before, as shown in figure 10.6 on the following page.

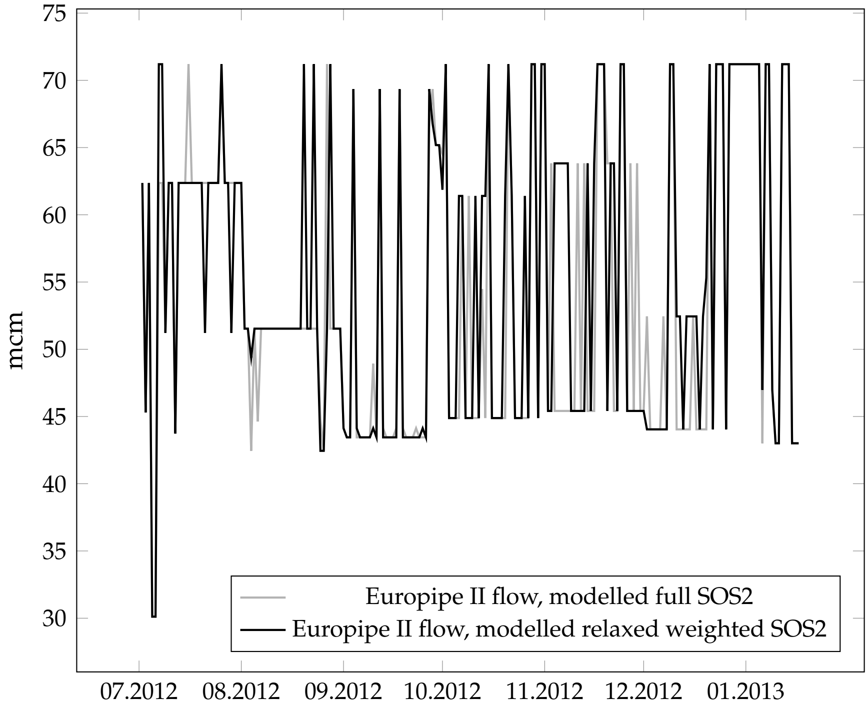


Figure 10.6: Flow at Dornum, full SOS2 vs relaxed weighted SOS2

At first glance, it seems to have more overlapping with the full model, but I found that the non-weighted version was closer overall, when looking at the sum of flows over the 200 day period, as shown in table 10.4. The values for every single country is closer to the full model (though individual pipelines were better in weighted form, but not enough to be significant); for this reason, the idea of weighting of the SOS2 variables was discarded. As all the differences for the non-weighted model are $< \pm 2\%$, it's reasonable to use it for the further testing of the accuracy of the model.

Table 10.4: Difference between full model and two relaxed variants

Pipeline	Full model (mcm)	Relaxed model (% difference)	Weighted model (% difference)
Germany	18 760	1.24%	-6.40%
France	10 361	0.53%	2.11%
Belgium	8 967	-1.55%	-2.06%
UK	19 432	1.77%	1.87%

10.3.2 Damping variations

GASSCO report the flows for the terminals for gas from the Norwegian continental shelf[3] at 5 minute intervals. I am using daily aggregates of these numbers (aligned to the gas day, 06:00 to 06:00) compared to the model flows for these same terminals as a measure of the precision and validity of

[3] See http://flow.gassco.no

the model, as these are the numbers most directly comparable to the model results.

As shown in figure 10.7, the day-to-day variations are extreme; the actual sum of flows over the period was ±5% to the modelled for the flows checked, which is a good sign, but the flow for an individual day is less valuable this way.

I found 15%, $\xi = 0.15$, to be a minimum - anything less than 0.13, and the solver was unable to find a feasible solution, and less than 0.15 took 10 times longer to compute. This made the figures look better, as shown in figure 10.8 on the next page, and the sum is still within ±5% of the actual sum.

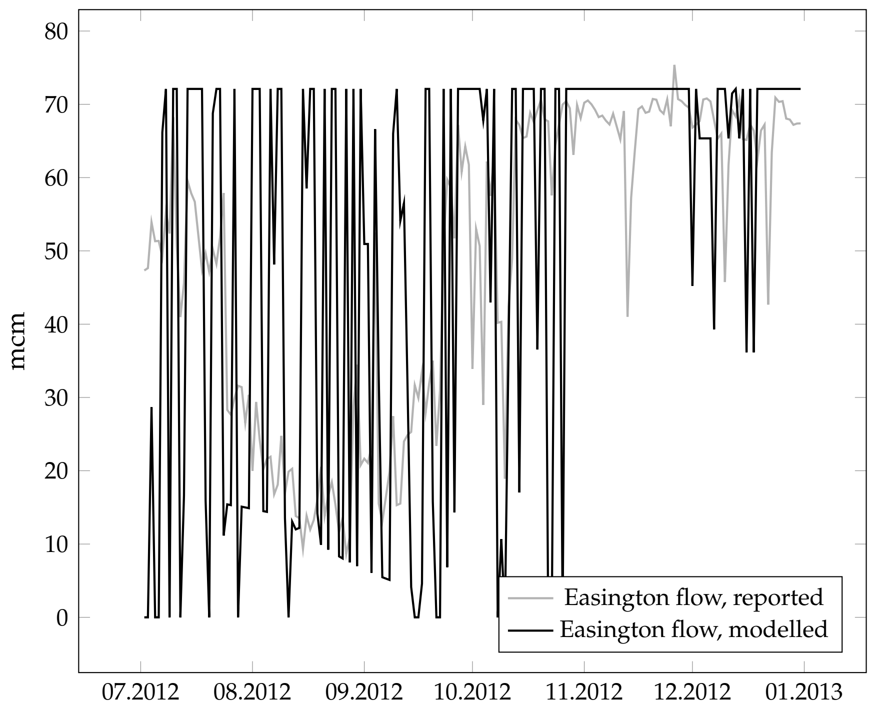


Figure 10.7: Flow at Easington, showing binary nature of initial model

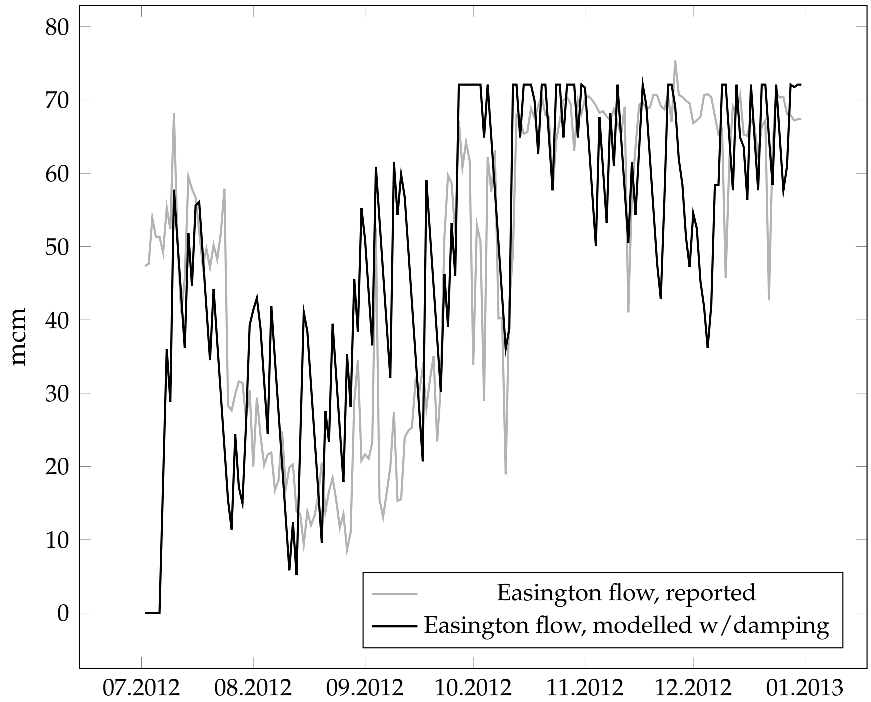


Figure 10.8: Flow at Easington, with damping constraints added

Adding these constraints did affect solution time, on average 20% higher than the unmodified model.

10.3.3 Dynamic production adjustments

I ran the model with a dynamic production adjustment with $z_{free} = 0.33$ - that is, 33% flexibility in the production.

This did, however, not make any significant difference to the result, day-to-day variations were on the same level as 10.8; this adjustment was still kept, and combined with the next extension to the model:

10.3.4 Smoothness as an objective

For this I kept the dynamic production adjustments, and tested several values of W_{dmp}.

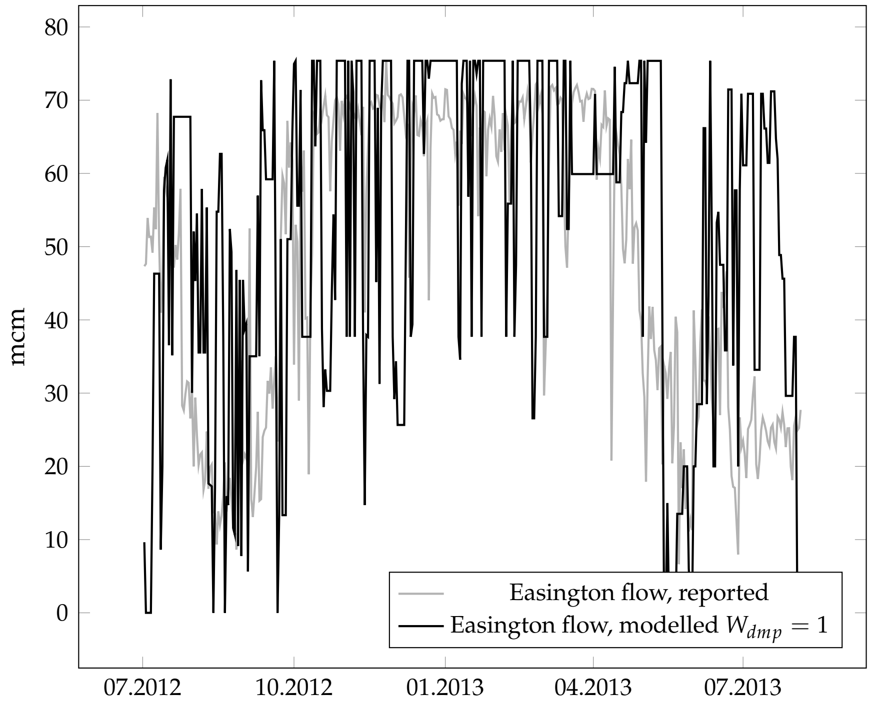


Figure 10.9: Flow at Easington, $W_{dmp} = 1$

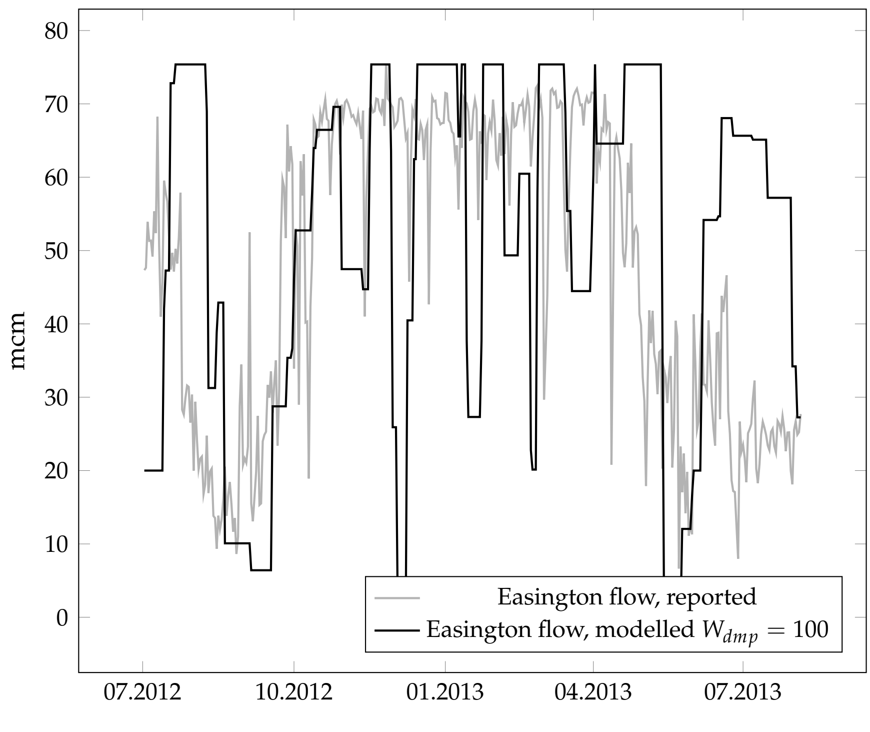


Figure 10.10: Flow at Easington, $W_{dmp} = 100$

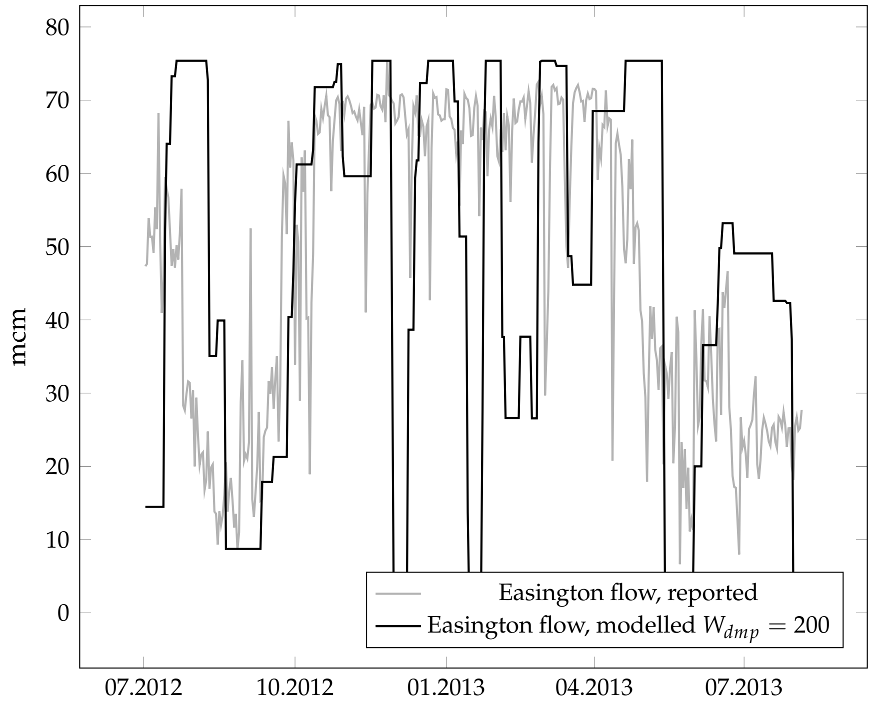


Figure 10.11: Flow at Easington, $W_{dmp} = 200$

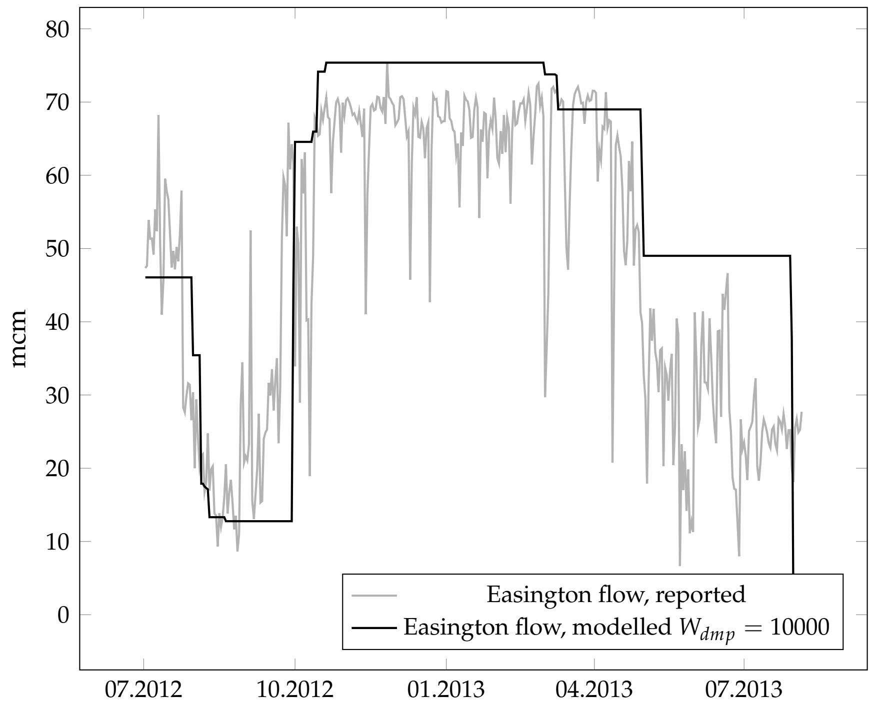


Figure 10.12: Flow at Easington, $W_{dmp} = 10000$

As can be seen in figure 10.9 on the preceding page, leaving the weight at 1 results in a chart fairly close to the one without weighting (compare figure 10.8 on page 80), while increasing to 100 (figure 10.10) smooths out the worst spikes. Both have significant spikes to 0 in December 2012, but they don't oscillate between 0 and full flow when $W_{dmp} = 100$. $W_{dmp} = 200$ is also included, as I found this value to give the best results; see section 10.3.5 on page 86.

Increasing further to $W_{dmp} = 10000$ gives the curve shown in figure 10.12 on the preceding page, which is significantly less spiky, but does not capture the details of the flow. Even with the initial flow set to the observation the day before the initial day, $W_{dmp} = 1$ lets the first day value deviate severely from the actual reported value.

Other flows

Easington was a fairly arbitrary choice for comparison; it had a decent amount of variation over the period, and was linked to a single pipeline, making it easy to show the differences.

The other pipelines show more of the same, here shown at $W_{dmp} = 200$. For other values of W_{dmp}, see appendix A on page 101.

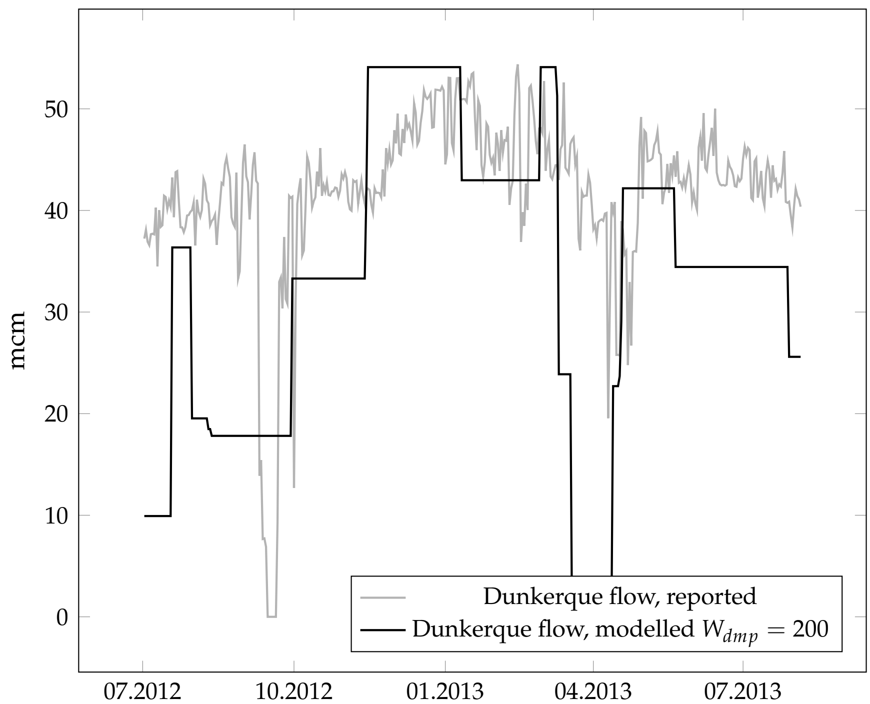


Figure 10.13: Flow at Dunkerque, $W_{dmp} = 200$

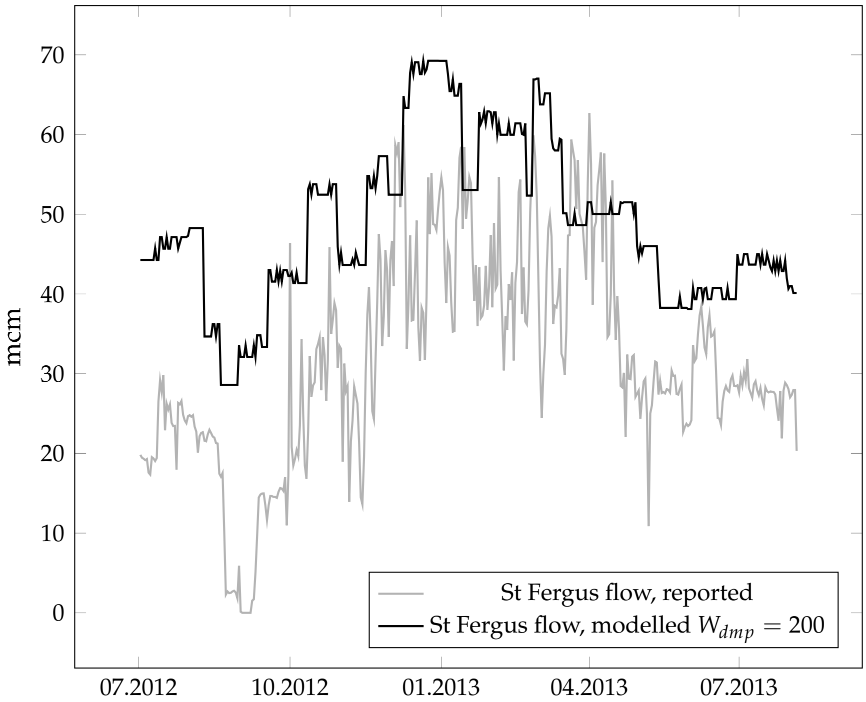


Figure 10.14: Flow at St Fergus, $W_{dmp} = 200$

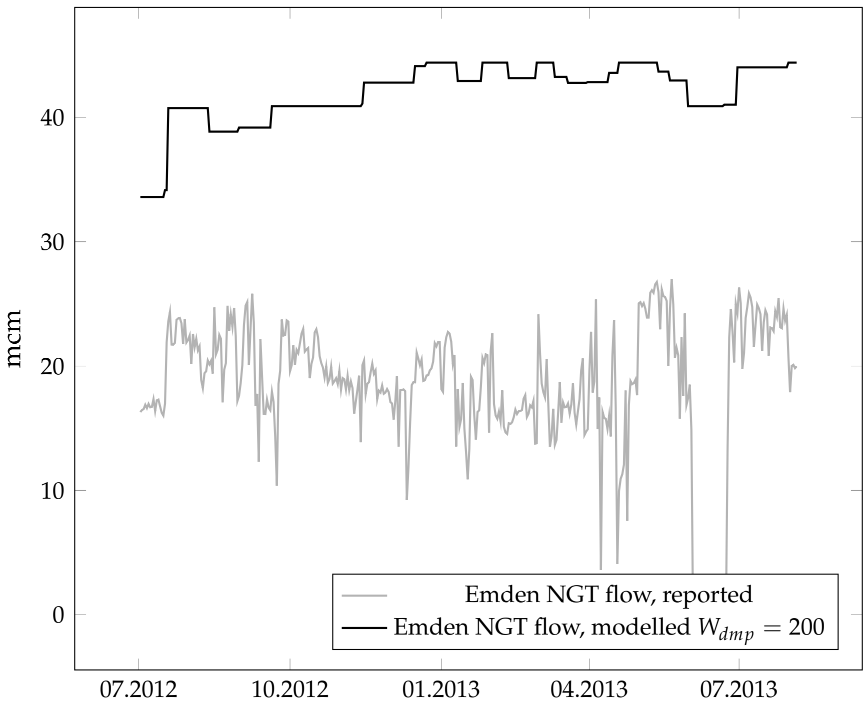


Figure 10.15: Flow at Emden, $W_{dmp} = 200$

The reason for the huge discrepancy in Emden is the way I have modelled the Emden/Dornum constellation; GASSCO reports three flows into Germany: Emden Norsea Gas Terminal (NGT) (shown in figure

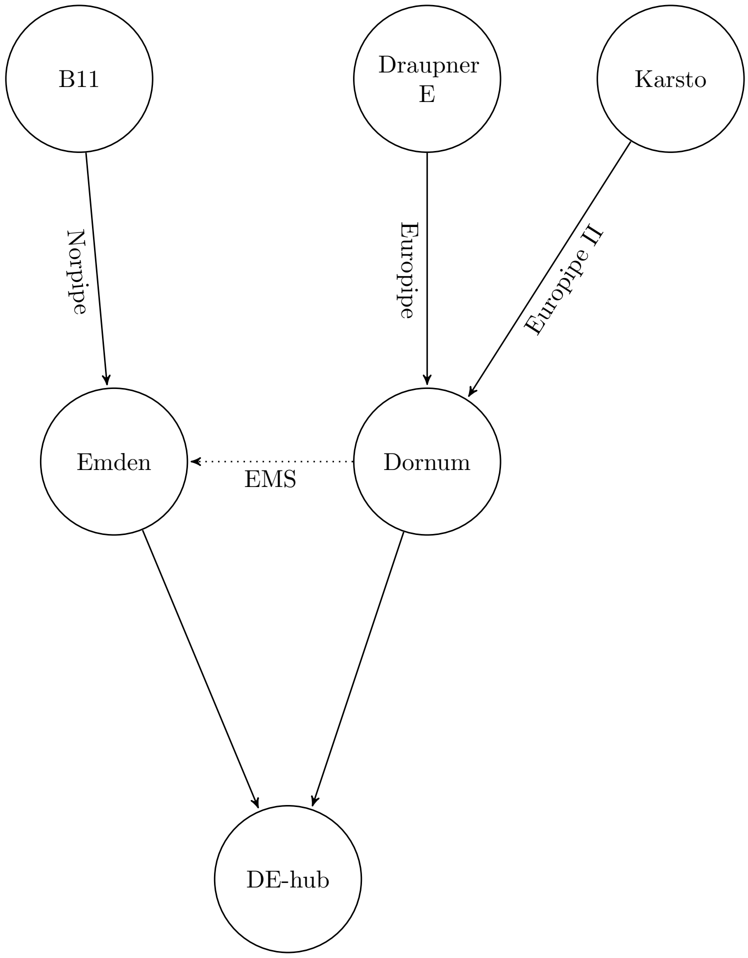


Figure 10.16: Detail of network model, Emden/Dornum

10.15), Emden Europipe Metering Station (EMS), and Dornum Europipe Receiving Facilities (ERF). As shown in figure 10.16 I have not modelled the pipeline labelled *EMS*, which forwards the gas from Europipe to the facility at Emden EMS, where the gas quality and volume is controlled. As both Emden and Dornum facilities lead to *DE-hub* this link was redundant in my system. The sum for the three pipelines leading into Germany add up to the correct flow as shown in figure 10.17.

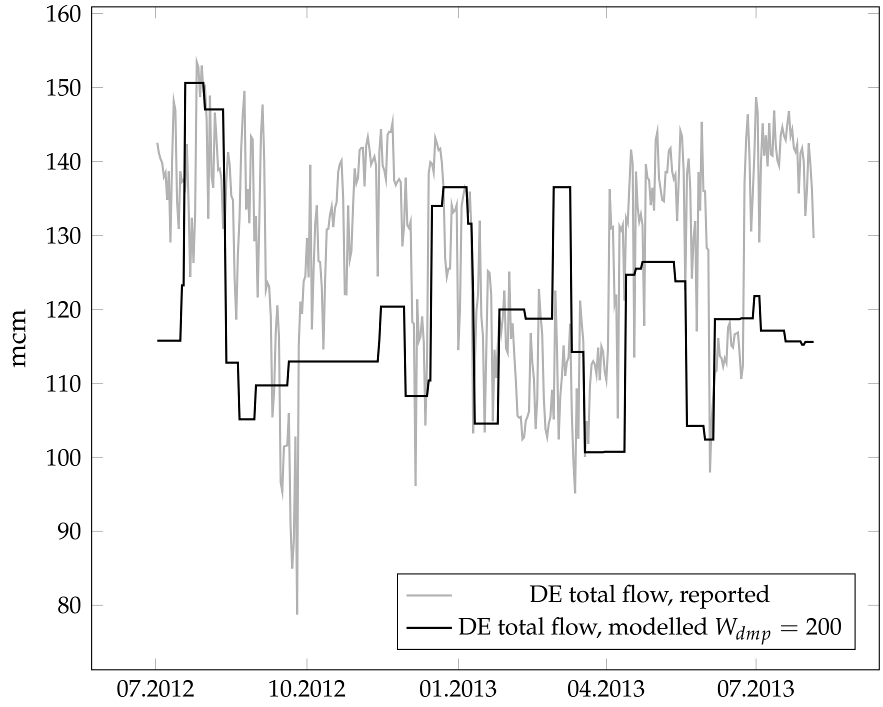


Figure 10.17: Flow in to Germany, $W_{dmp} = 200$

The reported flow for Emden EMS exceeds the reported capacity for Europipe by 20 mcm in the period considered here, which accounts for the extra 20 mcm shown at Emden NGT in my model. Corollary of this, the reported flow for Emden NGT is consistently at 20 mcm less than the official capacity in the period considered.

Adding these constraints does increase the model run time significantly. A run for 400 days without extensions takes less than 2 minutes, while damping constraints increases this to 15 minutes with $W_{dmp} = 200$.

10.3.5 Picking a value for W_{dmp}

As an attempt to quantify the quality of the model by changing W_dmp, a few symbols were introduced:

$E_{GC} \subset E$ the set of edges where GASSCO reports flows.

x^e the vector of modelled flows for a single edge $e \in E$ across all days I.

$\bar{x}^e$ the vector of GASSCO reported flows for a single edge $e \in E_{GC}$ across all days I.

$r_{x^e,\bar{x}^e}$ The Pearson correlation coefficient (*Pearson's r*) for the flow on a single edge $e \in E_{GC}$.

The following formulas were used as quality measures:

$$CORREL_{W_{dmp}} = \prod_{e \in E_{CG}} r_{x^e,\bar{x}^e} \tag{10.1}$$

Table 10.5: Product of correlation for model at various values of W_{dmp}, $I = 1...400$

W_{dmp}	$CORREL_{W_{dmp}}$	$MCORREL_{W_{dmp}}$
0	0.535×10^{-3}	
1	7.779×10^{-3}	0.240
10	3.652×10^{-3}	0.076
25	3.923×10^{-3}	0.074
50	6.672×10^{-3}	0.135
75	1.896×10^{-3}	0.053
100	1.534×10^{-3}	0.048
180	13.917×10^{-3}	0.244
190	15.584×10^{-3}	0.236
200	16.935×10^{-3}	0.270
210	12.718×10^{-3}	0.233
220	11.537×10^{-3}	0.227
300	4.236×10^{-3}	0.144
400	11.046×10^{-3}	0.193
500	8.688×10^{-3}	0.200
1000	15.890×10^{-3}	0.205

$$MCORREL_{W_{dmp}} = min_{e \in E_{CG}} r_{x^e, \bar{x}^e} \tag{10.2}$$

The rationale for multiplying the coefficients together is that due to the coefficients being in the range $[-1.0, 1.0]$ poor correlation in any flow will pull the value down, so the product with the highest value should have the highest overall correlation (barring negative correlation, which didn't occur here). There could be made a case for weighting each correlation coefficient with the relative capacity of the pipelines involved, but as I only have data for a few pipelines, I elected to not weight anything so that none of them were effectively insignificant.

Using this I calculated the product of the correlation coefficients, and the minimum correlation coefficient for several values of W_dmp for a 400-day model run, which gave the results in table 10.5 (with $W_{dmp} = 0$ shown as a base-line). As $W_{dmp} = 200$ gave the highest result in the first pass, I did extra runs for the values surrounding, which accounts for the majority of the results. The tables shown in this chapter contains a selection of runs for brevity, and full tables can be found in appendix B on page 113. As can be seen from the minimum correlation coefficients, no negative correlations occurred.

Having a slight smoothing seems to help in correlation, but there's no clear pattern for this; a proper choice for W_{dmp} may be period-specific, model-specific and even infeasible to set in advance.

Table 10.6: Total flows, 02.07.2012-04.08.2013, various values of W_{dmp}

W_{dmp}	Germany (mcm)	France (mcm)	Belgium (mcm)	UK (mcm)
1	47 502	13 323	13 408	38 847
10	48 535	13 008	13 438	38 148
25	47 937	12 877	14 202	38 168
50	46 978	13 543	14 509	38 149
75	48 666	12 120	13 134	39 279
100	48 146	12 328	13 848	38 878
180	48 275	11 872	14 455	38 594
190	47 712	11 940	15 362	38 176
200	47 221	13 255	14 374	38 342
210	48 569	11 732	14 877	38 013
220	48 376	11 904	15 080	37 831
300	49 019	9 474	15 825	38 820
400	51 926	6 540	16 433	38 258
500	49 266	7 156	16 637	40 002
1000	50 222	6 157	16 214	40 361
Reported	50 919	16 709	14 814	31 815

The totals

While the correlations in table 10.5 on the preceding page would indicate that $W_{dmp} = 200$ or $W_{dmp} = 1000$ would be good choices, the sums of flows to each country vary significantly with the smoothness, as shown in table 10.6. At least for the period considered, $W_{dmp} = 200$ is a local maximum, and seems a good choice. $W_{dmp} = 1000$ is missing quite a lot of flow to France, and overshoots heavily to the UK, plus a slight overshoot in Belgium. $W_{dmp} = 200$ has the same high level for the UK, but with closer values for Belgium. While the level for Germany is better with the former, the latter has closer absolute differences.

Looking at 100-day segments of the 400-day run, some patterns emerge in the data:

Table 10.7: Total flows, day 1-100, reported vs. model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	12 986	12 093	44 %	39 %
France	3 529	2 043	12 %	7 %
Belgium	3 020	2 675	10 %	9 %
UK	5 129	7 203	17 %	23 %
- *Easington*	3 423	3 740	11 %	12 %
- *St Fergus*	1 705	3 463	6 %	11 %
TOTAL	**29 793**	**31 216**		

Table 10.8: Total flows, day 101-200, reported vs. model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	13 130	11 844	31 %	27 %
France	4 589	4 592	11 %	11 %
Belgium	4 007	4 275	9 %	10 %
UK	10 434	11 251	24 %	26 %
- Easington	6 599	5 830	15 %	13 %
- St Fergus	3 835	5 421	9 %	13 %
TOTAL	**42 594**	**43 213**		

Table 10.9: Total flows, day 201-300, reported vs. model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	11 618	11 588	28 %	29 %
France	4 268	3 068	10 %	8 %
Belgium	3 956	4 182	10 %	11 %
UK	10 687	10 484	26 %	26 %
- Easington	6 405	5 529	16 %	14 %
- St Fergus	4 282	4 954	10 %	12 %
TOTAL	**41 215**	**39 807**		

Table 10.10: Total flows, day 301-400, reported vs. model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	13 185	11 696	41 %	37 %
France	4 323	3 551	13 %	11 %
Belgium	3 831	3 242	12 %	10 %
UK	5 565	6 693	17 %	21 %
- Easington	2 752	3 799	8 %	12 %
- St Fergus	2 813	2 894	9 %	9 %
TOTAL	**32 470**	**31 875**		

Across all four segments, the sum of reported flows is 114257 mcm, while the sum of modelled flows is 110280 mcm; about 3.5 % more is reported. Due to this, all the tables also show percentage of total reported flow and percentage of total model flow. The reason for this discrepancy may be in the source data from the Norwegian Continental Shelf, or that some (for this model) unknown producer is counted into the reported data.

In all four segments, not enough gas is routed to Germany when looking at the absolute figures; on average 3.2 % more gas should be sent to Germany. Similarly, 3.0 % more gas should be sent to France. On the other hand, the UK gets 3.3 % too much gas, almost all of which is routed to St Fergus. Belgium is not that bad, missing only 0.4 %.

For three of the segments, too much gas is routed to the UK, with the overall closest period in total being the third, shown in 10.9. Here all of the totals are close, except France, and the amount missing is the same as the amount missing from the total - that is, in this period the total modelled flow was 1409 mcm lower than the actual reported flow, while France is missing 1200 mcm. Also, while the UK total is very close to right for this period, the individual flows are diverging.

As the model doesn't have any friction for the gas, but assumes that it can flow across the entire grid within a day, some of this is to be expected. As both the British terminals connect to the exact same market, it doesn't matter which path the gas takes - while in the real world, there are more considerations involved, like maintaining contractually required pressure[4] or quality[5].

Adding friction

The longest path in the graph visits 11 nodes (Gullfaks to FR-hub via NTS, NL-hub, and DE-hub) - see figure 6.2 on page 47. According to Ulstein, Nygreen, and Sagli (2007, page 557) "(...) fuel gas consumption is less than 2% of the total gas production." As a quick test, I used the constant node loss $K = 0.001$, which keeps the total loss over the longest path approximately at 1.1%.

Model results with $W_{dmp} = 200$ can be summed up as close to the results in the previous section[6]; France is slightly improved in the second and third segments (days 101-300), but UK overshoots significantly, especially in the last segment (by 87%), and the figures for Germany are generally worse, with 24% discrepancy in the last segment.

10.4 Model quality

As the previous sections show, I have attempted in various ways to measure the quality of the model as presented here, and added to the model to improve the results after these measures. As mentioned in chapter 2, it is difficult to estimate how well this model performs as compared to the existing models - both because the focus and scales are different, and because the other models have very little in the way of qualitative results to work with.

It is highly likely that the parameter values used in this chapter are specific to the period modelled, and that given a different historical period or future period, that W_{dmp} (which is the parameter that has the highest impact) optimally should be a different value.

[4]Tomasgard et al. 2007, page 15.
[5]Ulstein, Nygreen, and Sagli 2007, page 557.
[6]Detailed results can be found in appendix C on page 117

Part VI

Conclusion

Chapter 11

Conclusion and future research

The linear programming model described here helps explain the system as a whole, even if the day-by-day precision could be improved. The assumptions as to how the gas is routed seems to hold for the most part, but there may be hidden constraints not available to this model that explain the discrepancies.

The model gives several parameters that can be tuned to trade precision for solution speed, and generally tune how the system is solved depending on the intended use-case. It's also possible to run several scenarios with different parameters in parallel to compare results. The base speed of the model is well within the limits suggested by the original problem on a regular desktop computer.

There's also the possibility of replacing parts of the network with more detailed models if any such are known, for instance modelling the compressors and pressure in the pipelines either by retrieving as much data as possible from the actors in the market, or attempting to infer these parameters by testing scenarios against real values.

Creating a working forecast model from this model is just a matter of input data; given a forecast for demand, prices and production, the model will give a forecast for the gas flows.

Part VII

References

Bibliography

Articles, books, reports

De Wolf, Daniel and Yves Smeers (2000a). "The gas transmission problem solved by an extension of the simplex algorithm". In: *Management Science*, pp. 1454–1465.

— (2000b). "The simplex algorithm extended to piecewise-linearly constrained problems". In:

Eurogas (2013). *Eurogas Statistical Report 2013: data for year 2012 and 2011*. Tech. rep. URL: http://www.eurogas.org/uploads/media/Eurogas_Statistical_Report_2013.pdf.

Gas LNG Europe (2011). "LNG Terminals in Europe". In: URL: http://www.gie.eu/index.php/events-diary/workshops/gle-workshops/cat_view/16-gle-workshops/33-2011-workshops/34-first-ceer-workshop-on-access-to-european-lng-terminals.

King & Spalding (2006). "LNG in Europe". In: URL: http://www.kslaw.com/library/pdf/LNg_in_Europe.pdf.

Lako, Paul (2010). "ETSAP Technology Brief E04 - Combined Heat And Power". In: URL: http://www.iea-etsap.org/web/E-TechDS/PDF/E04-CHP-GS-gct_ADfinal.pdf.

Möller, Markus (2004). "Mixed Integer Models for the Optimisation of Gas Networks in the Stationary Case". PhD thesis. TU Darmstadt. URL: http://tuprints.ulb.tu-darmstadt.de/438/.

Norwegian Ministry of Petroleum and Energy (2012). *Facts 2012 - The Norwegian Petroleum Sector*. Tech. rep. Norwegian Ministry of Petroleum, Energy, and Norwegian Petroleum Directorate. URL: http://www.regjeringen.no/upload/OED/pdf%20filer/Faktaheftet/OG_Facts_2012_web.pdf.

Nygreen, Bjørn and Kjetil Haugen (2010). "Applied Mathematical Programming in Norwegian Petroleum Field and Pipeline Development: Some Highlights from the Last 30 Years". In: *Energy, Natural Resources and Environmental Economics*. Springer, pp. 59–69.

Oland, CB (2004). *Guide to Combined Heat and Power Systems for Boiler Owners and Operators*. DOI: 10.2172/885547. URL: http://www.osti.gov/scitech/servlets/purl/885547.

Pindyck, Robert S (2001). "The Dynamics of Commodity Spot and Futures Markets: A Primer." In: *Energy Journal* 22.3.

Tomasgard, Asgeir et al. (2007). "Optimization Models for the Natural Gas Value Chain". In: ed. by Geir Hasle, Knut-Andreas Lie, and Ewald Quak, pp. 521–558.

Ulstein, Nina Linn, Bjørn Nygreen, and Jan Richard Sagli (2007). "Tactical planning of offshore petroleum production". In: *European Journal of Operational Research* 176.1, pp. 550–564. ISSN: 03772217. DOI: 10.1016/j.ejor.2005.06.060. URL: http://linkinghub.elsevier.com/retrieve/pii/S0377221705006363.

Wolsey, L.A. (1998). *Integer Programming*. Wiley Series in Discrete Mathematics and Optimization. Wiley. ISBN: 9780471283669. URL: http://books.google.no/books?id=x7RvQgAACAAJ.

Web pages

Department of Energy and Climate Change, UK (2013). *Digest of UK Energy Statistics (DUKES)*. URL: https://www.gov.uk/government/collections/digest-of-uk-energy-statistics-dukes (retrieved on 03/26/2014).

Energuide (2012). *"What is the difference between rich and lean gas?"* URL: http://www.energuide.be/en/questions-answers/what-is-the-difference-between-rich-and-lean-gas-does-this-have-an-impact-on-my-bill/85/ (retrieved on 03/30/2014).

Energy Research Centre, UK (2014). *Gas — UK Energy Research Centre*. URL: http://www.ukerc.ac.uk/support/Gas (retrieved on 03/26/2014).

European Commission (2014). *Eurostat*. URL: http://epp.eurostat.ec.europa.eu (retrieved on 03/26/2014).

Gas Infrastructure Europe (2014). *GIE - The Natural Gas Chain*. URL: http://www.gie.eu/KC/gasinfrastructure.html (retrieved on 03/26/2014).

Gasunie Transport Services (2014). *Nomination process*. URL: http://www.gasunietransportservices.nl/en/transportinformation/operational-handling/nomination-proces (retrieved on 04/26/2014).

National Grid, supplementary reports (2014). URL: http://www.nationalgrid.com/uk/Gas/Data/storage/ (retrieved on 04/18/2014).

National Grid, UK (2014a). *National Grid, Calorific Value Description*. URL: http://www2.nationalgrid.com/uk/Industry-information/Gas-transmission-operational-data/Calorific-Value-Description/ (retrieved on 03/26/2014).

— (2014b). *National Grid, Data Item Explorer*. URL: http://marketinformation.natgrid.co.uk/gas/DataItemExplorer.aspx (retrieved on 04/03/2014).

Santos, Prof. Haroldo Gambini (2011). *CBC - Short Guide to the Command Line Interface*. URL: http://www.decom.ufop.br/haroldo/files/cbcCommandLine.pdf (retrieved on 04/17/2014).

van Rossum, Guido (2014). *The History of Python: Personal History part 1, CWI*. URL: http://python-history.blogspot.no/2009/01/personal-history-part-1-cwi.html (retrieved on 03/26/2014).

Appendices

Appendix A

Flows at various values of W_{dmp}

A.1 $W_{dmp} = 1$

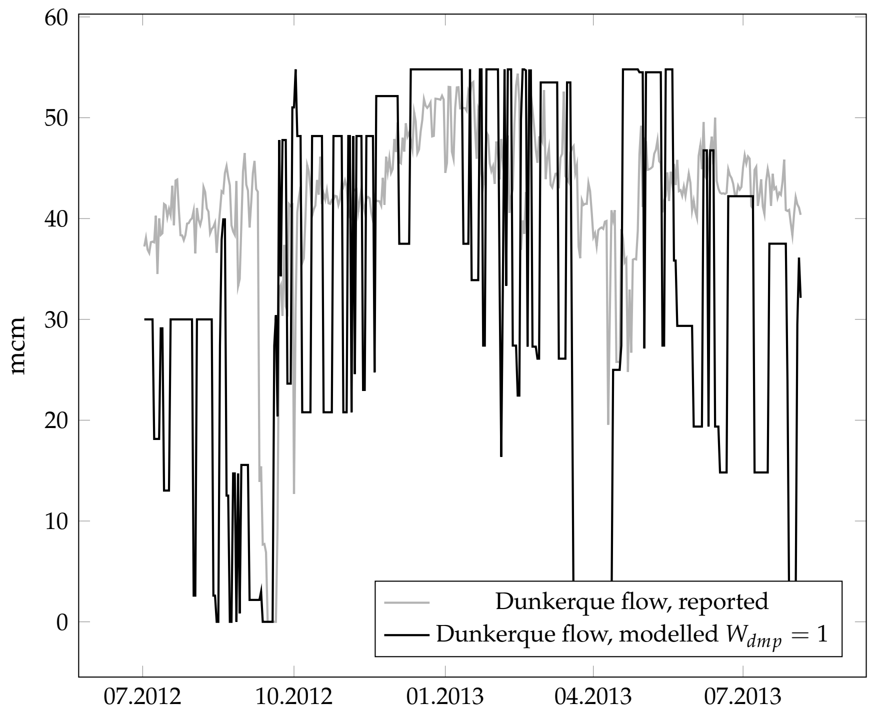


Figure A.1: Flow at Dunkerque, $W_{dmp} = 1$

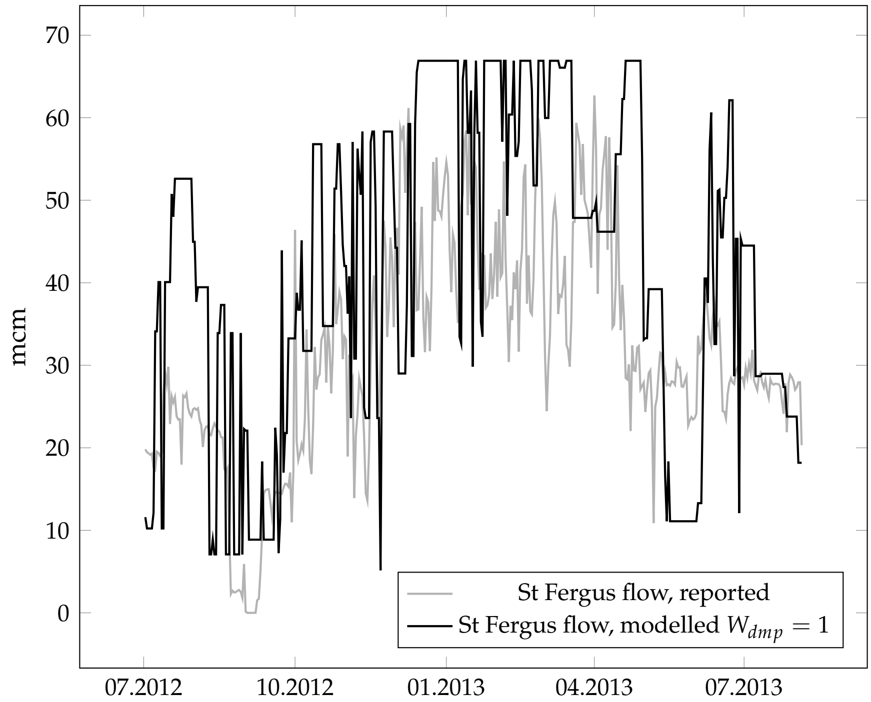


Figure A.2: Flow at St Fergus, $W_{dmp} = 1$

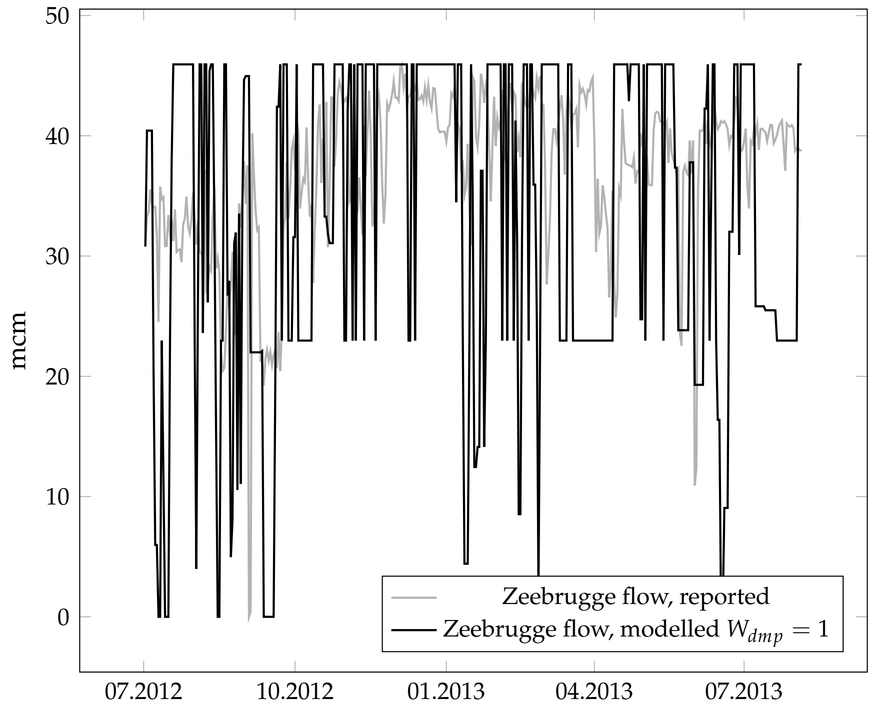


Figure A.3: Flow at Zeebrugge, $W_{dmp} = 1$

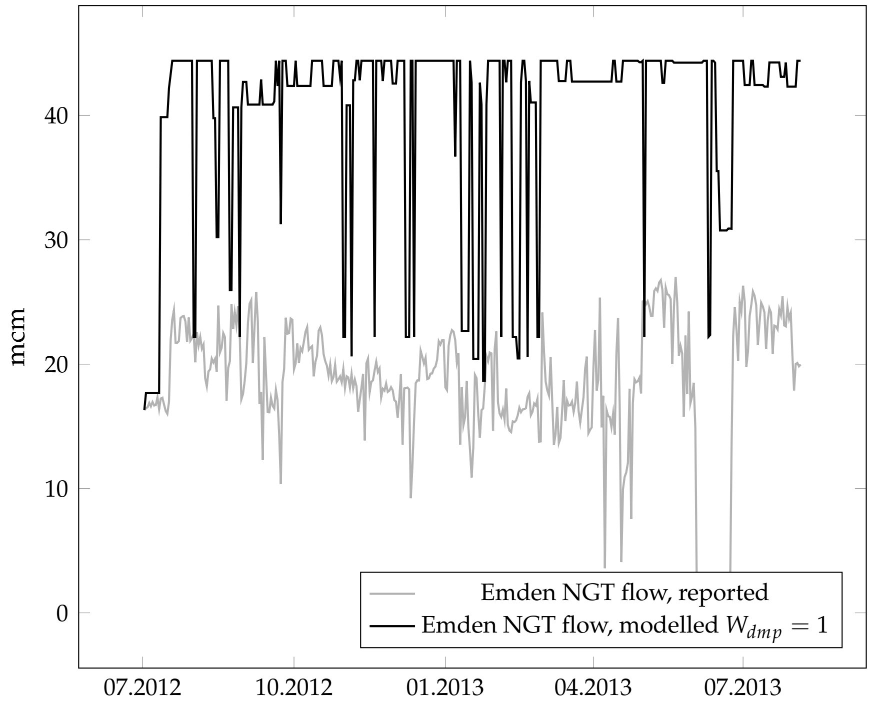


Figure A.4: Flow at Emden, $W_{dmp} = 1$

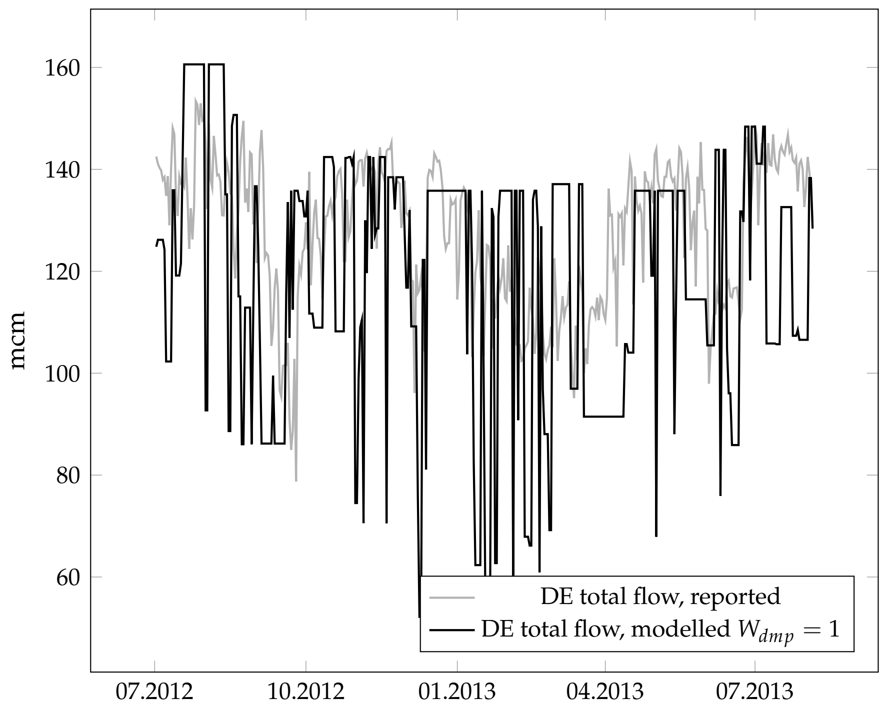


Figure A.5: Flow in to Germany, $W_{dmp} = 1$

A.2 $W_{dmp} = 100$

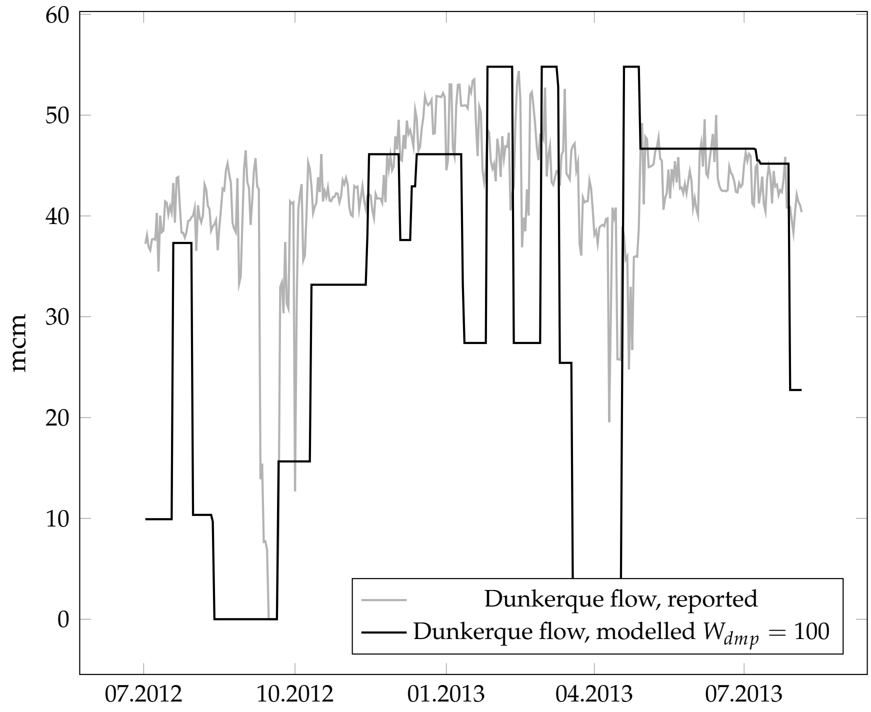


Figure A.6: Flow at Dunkerque, $W_{dmp} = 100$

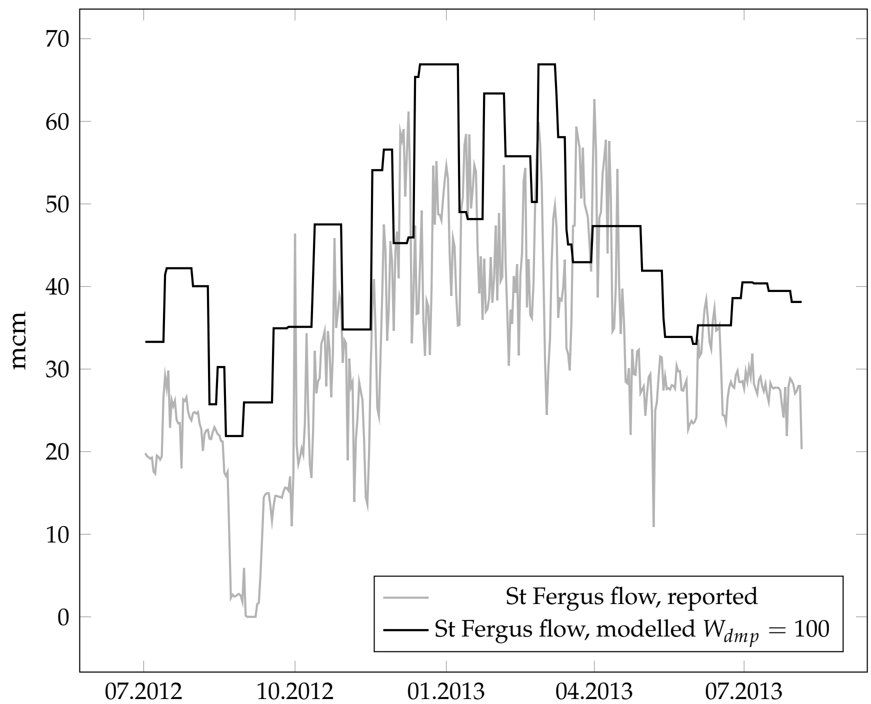


Figure A.7: Flow at St Fergus, $W_{dmp} = 100$

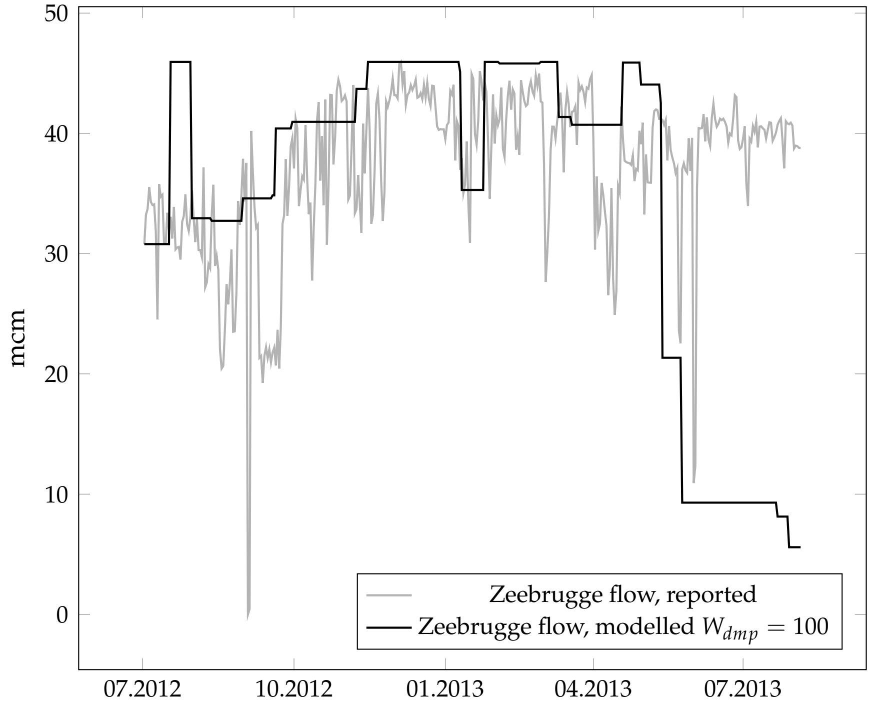


Figure A.8: Flow at Zeebrugge, $W_{dmp} = 100$

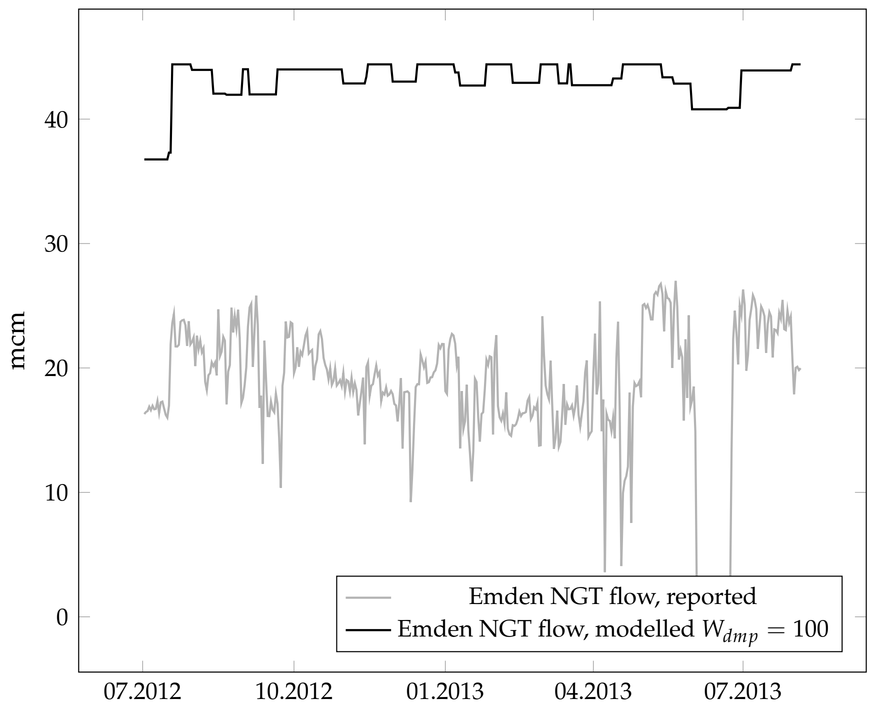


Figure A.9: Flow at Emden, $W_{dmp} = 100$

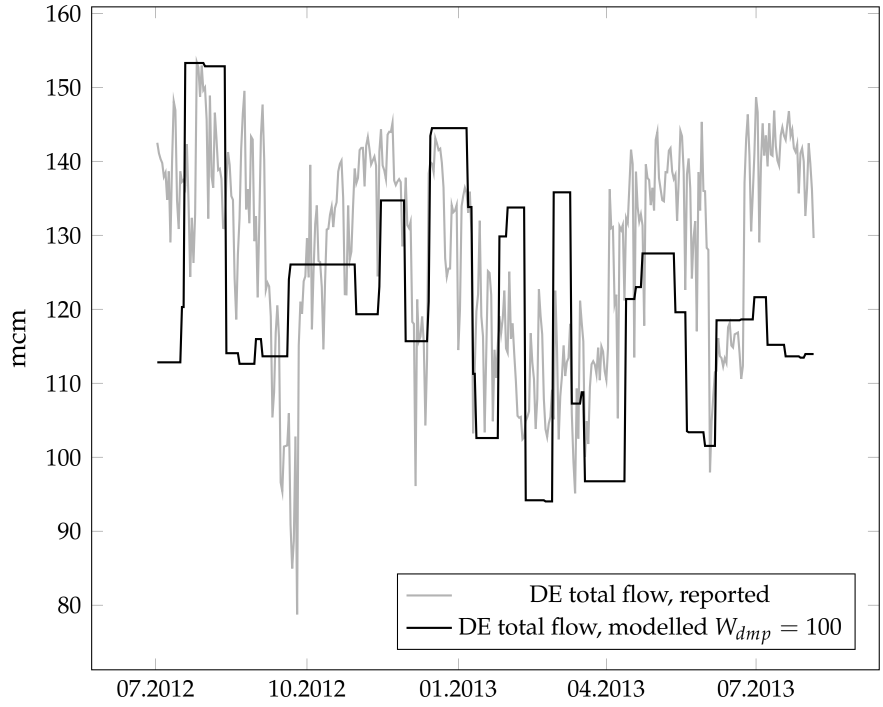


Figure A.10: Flow in to Germany, $W_{dmp} = 100$

A.3 $W_{dmp} = 1000$

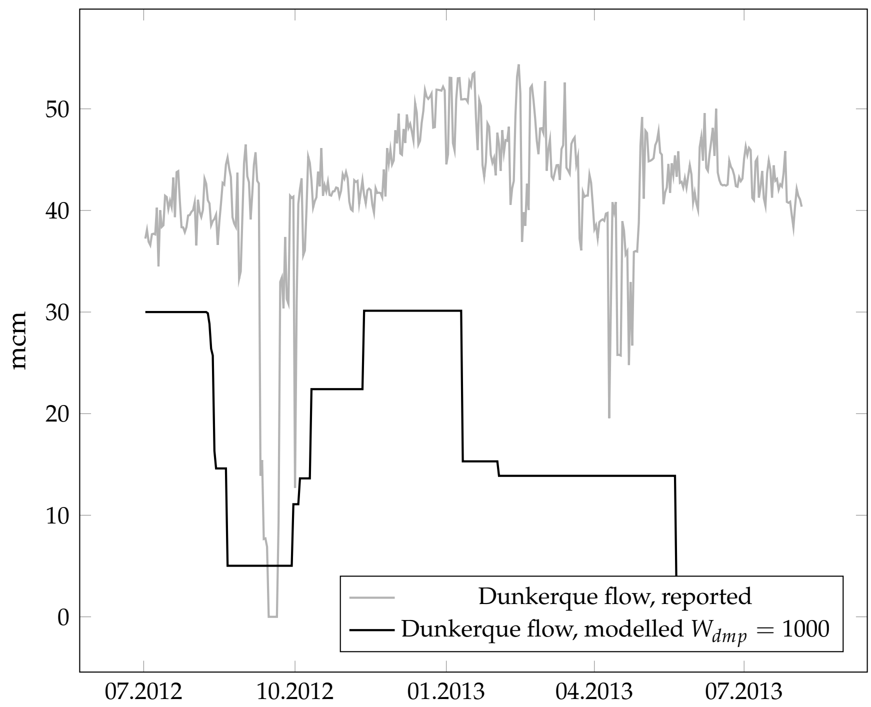


Figure A.11: Flow at Dunkerque, $W_{dmp} = 1000$

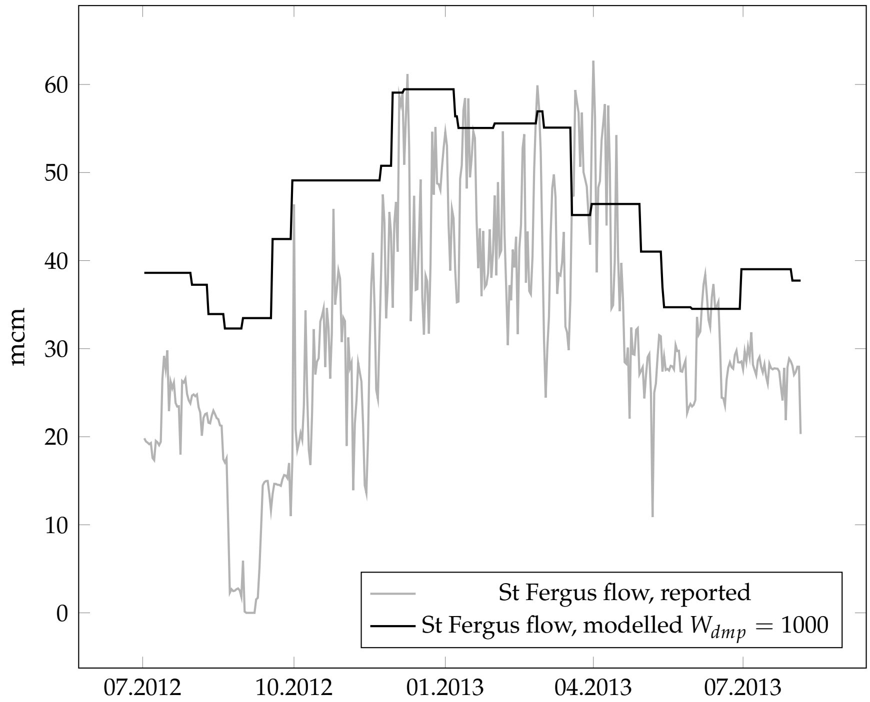


Figure A.12: Flow at St Fergus, $W_{dmp} = 1000$

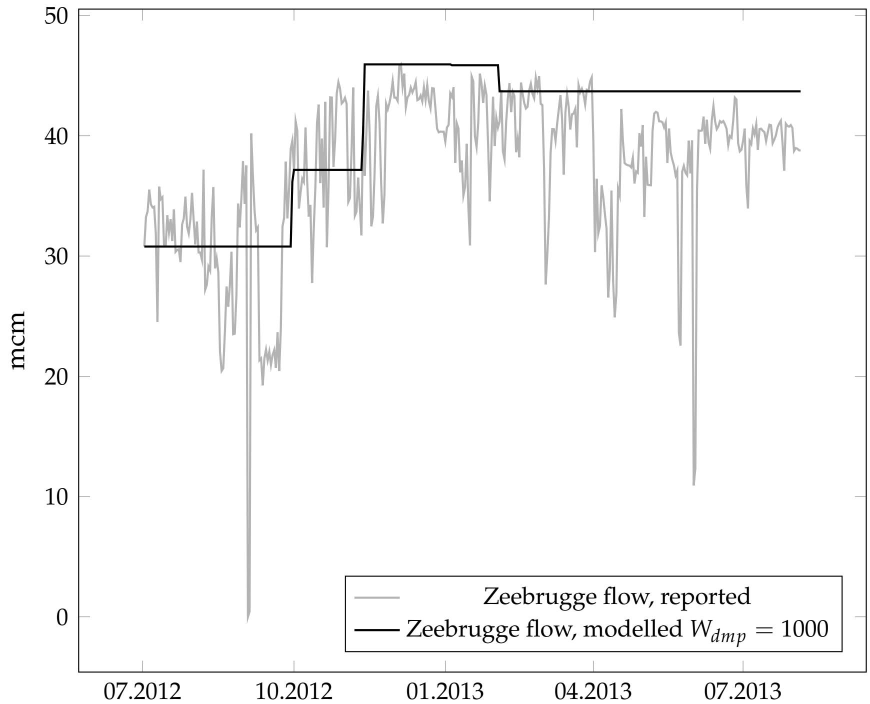


Figure A.13: Flow at Zeebrugge, $W_{dmp} = 1000$

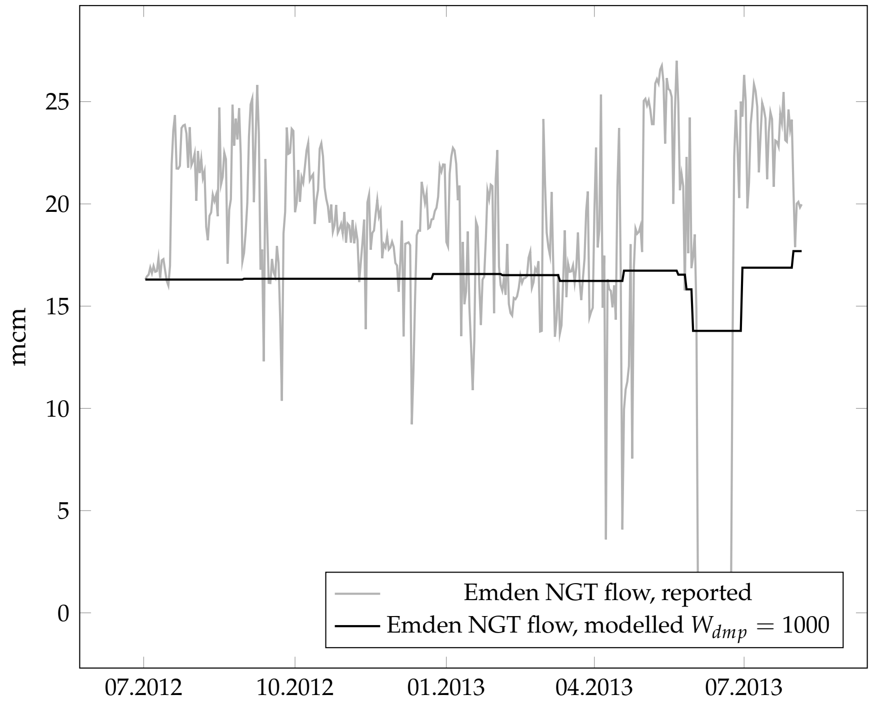


Figure A.14: Flow at Emden, $W_{dmp} = 1000$

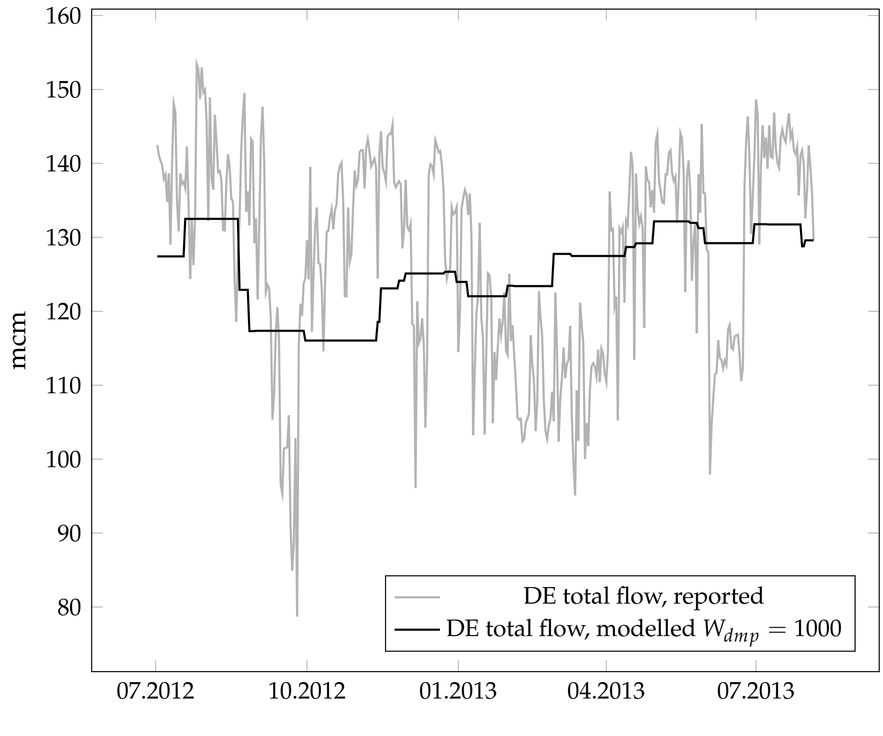


Figure A.15: Flow in to Germany, $W_{dmp} = 1000$

A.4 $W_{dmp} = 10000$

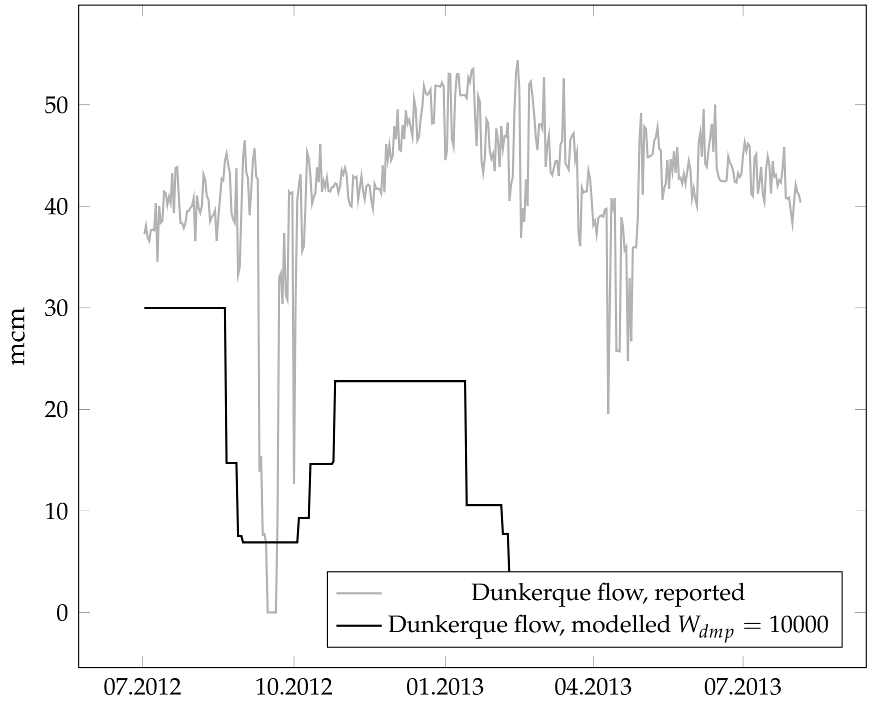


Figure A.16: Flow at Dunkerque, $W_{dmp} = 10000$

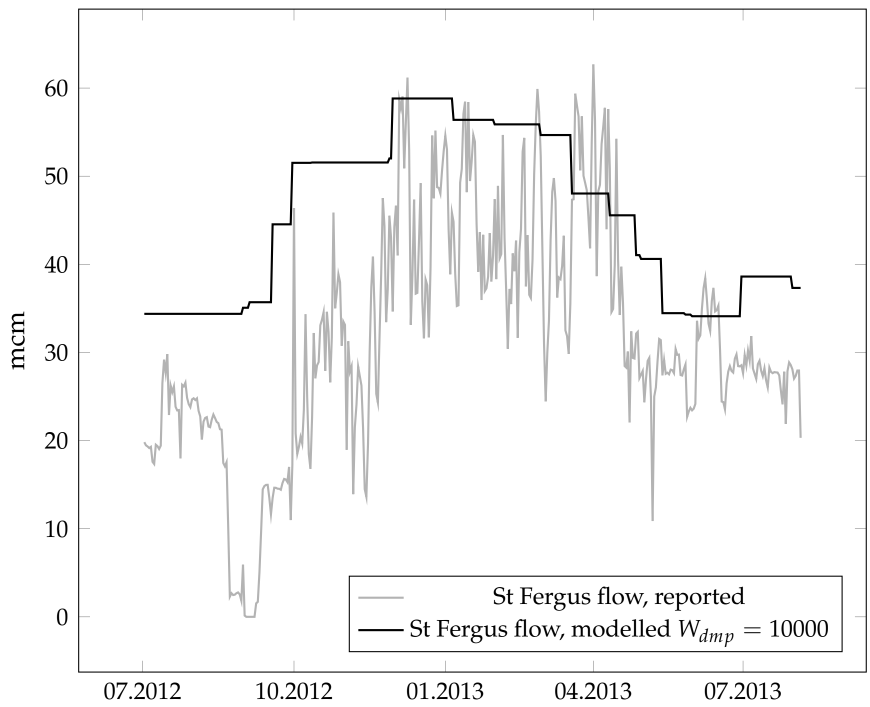


Figure A.17: Flow at St Fergus, $W_{dmp} = 10000$

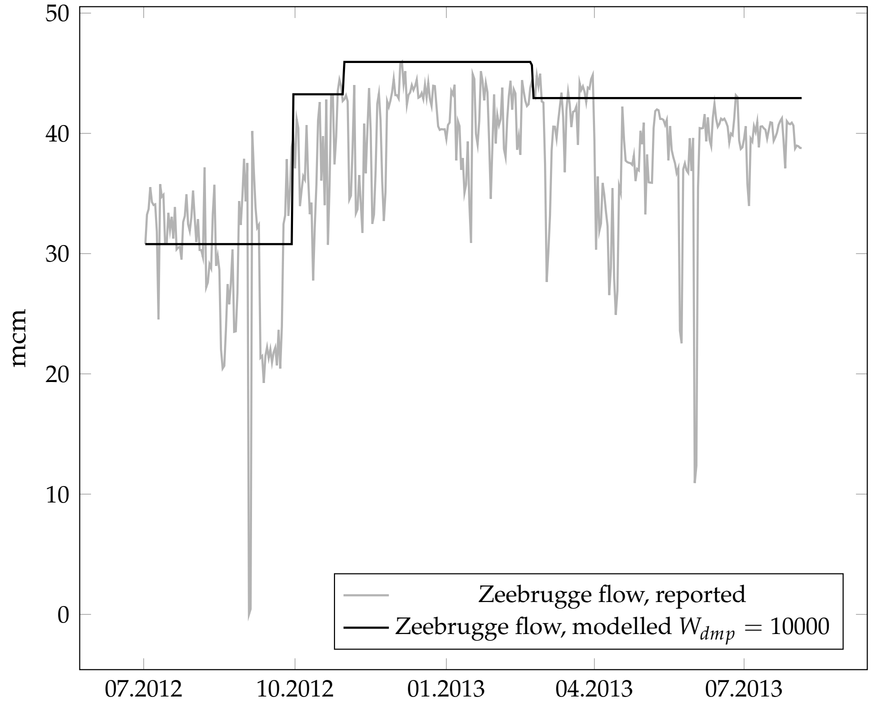


Figure A.18: Flow at Zeebrugge, $W_{dmp} = 10000$

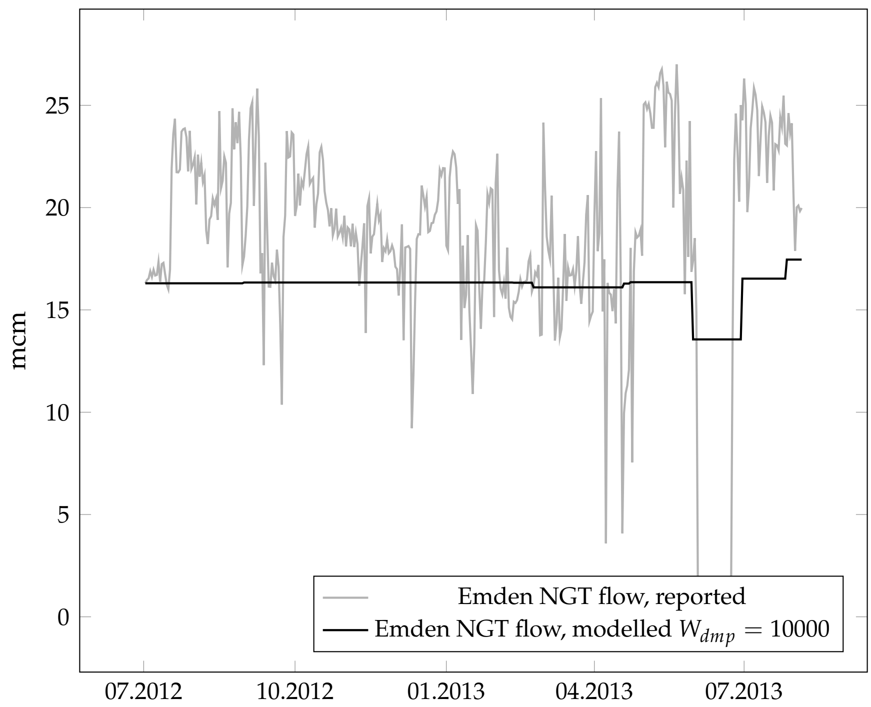


Figure A.19: Flow at Emden, $W_{dmp} = 10000$

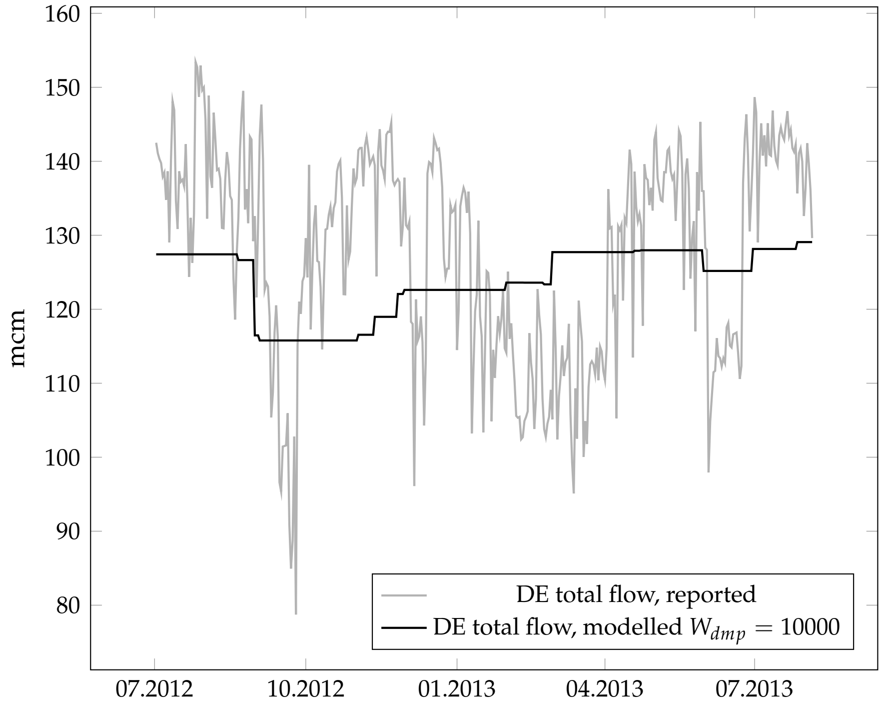


Figure A.20: Flow in to Germany, $W_{dmp} = 10000$

Appendix B

Correlations at various values of W_{dmp}

Multiple tests were run at intervals of 10 to find an approximate local maximum. The full table is shown below.

Table B.1: Product of correlation for model at various values of W_{dmp}, $I = 1...400$, full table

W_{dmp}	$CORREL_{W_{dmp}}$	$MCORREL_{W_{dmp}}$
0	0.535×10^{-3}	
1	7.779×10^{-3}	0.240
10	3.652×10^{-3}	0.076
25	3.923×10^{-3}	0.074
50	6.672×10^{-3}	0.135
75	1.896×10^{-3}	0.053
100	1.534×10^{-3}	0.048
150	5.390×10^{-3}	0.205
160	6.655×10^{-3}	0.234
170	12.142×10^{-3}	0.238
180	13.917×10^{-3}	0.244
190	15.584×10^{-3}	0.236
200	16.935×10^{-3}	0.270
210	12.718×10^{-3}	0.233
220	11.537×10^{-3}	0.227
230	9.295×10^{-3}	0.209
240	8.775×10^{-3}	0.190
250	7.335×10^{-3}	0.185
260	7.436×10^{-3}	0.178
270	6.260×10^{-3}	0.161
280	5.200×10^{-3}	0.138
290	3.827×10^{-3}	0.120
300	4.236×10^{-3}	0.144
400	11.046×10^{-3}	0.193
500	8.688×10^{-3}	0.200
1000	15.890×10^{-3}	0.205
10000	4.522×10^{-3}	0.078

Table B.2: Total flows, 02.07.2012-04.08.2013, various values of W_{dmp}, full table

W_{dmp}	Germany (mcm)	France (mcm)	Belgium (mcm)	UK (mcm)
1	47 502	13 323	13 408	38 847
10	48 535	13 008	13 438	38 148
25	47 937	12 877	14 202	38 168
50	46 978	13 543	14 509	38 149
75	48 666	12 120	13 134	39 279
100	48 146	12 328	13 848	38 878
150	48 564	12 006	14 540	38 086
160	48 122	12 160	14 496	38 418
170	48 149	11 691	14 810	38 546
180	48 275	11 872	14 455	38 594
190	47 712	11 940	15 362	38 176
200	47 221	13 255	14 374	38 342
210	48 569	11 732	14 877	38 013
220	48 376	11 904	15 080	37 831
230	48 146	11 865	15 341	37 841
240	49 027	10 148	16 064	37 963
250	49 320	9 486	16 023	38 380
260	49 330	9 383	16 130	38 364
270	49 163	9 516	15 954	38 546
280	48 995	9 741	15 953	38 450
290	48 689	9 776	16 030	38 643
300	49 019	9 474	15 825	38 820
400	51 926	6 540	16 433	38 258
500	49 266	7 156	16 637	40 002
1000	50 222	6 157	16 214	40 361
10000	49 580	4 564	16 436	41 337
Reported	50 919	16 709	14 814	31 815

Appendix C

Results from test with node loss

All tests with $f_{loss} = 0.001$.

Table C.1: Total flows, day 1-100, reported vs. node loss model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	12 986	11 969	44 %	39 %
France	3 529	2 251	12 %	7 %
Belgium	3 020	3 875	10 %	13 %
UK	5 129	6 214	17 %	20 %
- *Easington*	3 423	2 707	11 %	9 %
- *St Fergus*	1 705	3 506	6 %	11 %
TOTAL	**29 793**	**31 216**		

Table C.2: Total flows, day 101-200, reported vs. node loss model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	13 130	12 315	44 %	40 %
France	4 589	4 897	15 %	16 %
Belgium	4 007	2 938	13 %	10 %
UK	10 434	11 805	35 %	39 %
- *Easington*	6 599	6 445	22 %	21 %
- *St Fergus*	3 835	5 359	13 %	18 %
TOTAL	**42 594**	**43 758**		

Table C.3: Total flows, day 201-300, reported vs. node loss model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	11 618	11 323	39 %	37 %
France	4 268	4 399	14 %	14 %
Belgium	3 956	2 190	13 %	7 %
UK	10 687	12 808	36 %	42 %
- *Easington*	6 405	6 531	21 %	21 %
- *St Fergus*	4 282	6 278	14 %	21 %
TOTAL	**41 215**	**39 807**		

Table C.4: Total flows, day 301-400, reported vs. node loss model $W_{dmp} = 200$

Area	Reported (mcm)	Model (mcm)	Reported (%)	Model (%)
Germany	13 185	10 000	44 %	33 %
France	4 323	2 353	15 %	8 %
Belgium	3 831	3 565	13 %	12 %
UK	5 565	10 425	19 %	34 %
- *Easington*	2 752	5 701	9 %	19 %
- *St Fergus*	2 813	4 724	9 %	15 %
TOTAL	**32 470**	**31 875**		