



NTNU – Trondheim
Norwegian University of
Science and Technology

Optimization of Oil Production - Applied to the Marlim Field

Lamija Dzubur
Andrea Sundby Langvik

Industrial Economics and Technology Management

Submission date: June 2012

Supervisor: Henrik Andersson, IØT

Co-supervisor: Vidar Gunnerud, Department of Engineering Cybernetics
Alex Teixeira, Petrobras

Norwegian University of Science and Technology

Department of Industrial Economics and Technology Management

Preface

This master thesis is the result of the final work completed to achieve a Master of Science degree with specialization in Applied Economics and Optimization at the Department of Industrial Economics and Technology Management at the Norwegian University of Science and Technology (NTNU).

The motivation for this master thesis is the Ph.D thesis by Gunnerud (2011), presenting petroleum production optimization models for the Troll C petroleum field owned by Statoil. An additional motivation factor was the summer internship Lamija had for the IO-Center during the summer 2011, where she was presented the problem first hand. During the process several meetings with Petrobras have been conducted. They have given us necessary input and important information about the system.

This work is a continuation of Dzubur & Langvik (2011) and some parts of the thesis are taken directly from that project work. The report is written in LATEX and the modelling was performed in Mosel using the Xpress-MP optimizer as commercial solver. We have enjoyed working with the thesis, and acquiring knowledge about the petroleum industry and production planning. A wide variety of theory on optimization has been studied and we have gained important new experience with solving large scale optimization problems.

We would like to thank our supervisor, Associate Professor Henrik Andersson, at the Department of Industrial Economics and Technology Management, for constructive discussions and guidance. We would also like to thank our cosupervisor, Post Doctor Vidar Gunnerud, at the Department of Engineering Cybernetics. He has provided insight into the problem and contributed with valuable input to this master thesis. Additionally, we want to thank Petrobras, represented by Alex Teixeira, for the interesting case study this thesis is based on.

Trondheim, June 25, 2012

Lamija Dzubur

Andrea Sundby Langvik

Abstract

This thesis considers short-term petroleum production planning at the Marlim field operated by the Brazilian energy company, Petrobras. The production asset to be optimized is an Floating Production and Storage Operating unit, including several oil producing wells and gas lift technology. Current practice for oil companies is to optimize the different parts of a production system separately. To avoid sub-optimal decisions, there is a need for integrated optimization models that consider all relevant decision variables simultaneously.

The objective of the models presented in this thesis is to maximize total oil production extracted from the reservoir. It is assumed that producing as much oil as possible is always economically preferable. The decisions to be made include production rate allocation of water, oil and gas from each well. In addition, allocation of lift gas to wells and pressure at different components of the system are important outcomes. Routing of flow adds further complexity to the model.

The optimization problem studied has integer requirements, in addition to several non-linear flow and pressure relationships. These have been approximated using piecewise linearization with special ordered sets of type 2, resulting in a mixed integer linear programming model. Different methods are implemented to solve the problem and evaluated to find the most appropriate optimization method. The optimization model was initially solved using Branch & Bound. Due to the block-angular structure of the problem, Dantzig-Wolfe decomposition was implemented to improve solution efficiency. Further, different heuristics were added to reduce the duality gap.

Generating realistic datasets was done with the PIPESIM simulator. The algorithms were implemented in Mosel, which is a high-level modelling and programming language, and solved with the Xpress-MP solver.

The results show that decomposing the model has a huge effect when the complexity of the system increases. Adding heuristics to the decomposed model is found to be an effective way of generating good integer programming solutions. When deciding what method to use, the trade-off between model realism and solution time must be weighted. A good, heuristically obtained solution could often be more useful than using exact solution methods that are very time consuming.

Sammendrag

Denne masteroppgaven tar for seg kortsiktig produksjonsplanlegging av petroleum på Marlim-feltet, som eies av det brasilianske energiselskapet, Petrobras. Produksjonsheten som skal optimeres er en flytende produksjon og lagrings driftenhet, inkludert flere oljeproduserende brønner og gassløft teknologi. Gjeldende praksis for oljeselskapene er å optimalisere de ulike delene av produksjonssystemet separat. For å unngå sub-optimale beslutninger, er det behov for integrerte optimeringsmodeller som vurderer alle relevante beslutningsvariable samtidig.

Målet med modellene som presenteres i denne masteroppgaven er å maksimere total oljeproduksjon, utvunnet fra reservoaret. Det antas at å produsere så mye olje som mulig alltid er økonomisk lønnsomt. De beslutninger som tas inkluderer produksjonsrater av vann, olje og gass fra hver brønn. I tillegg er allokering av løftegass til brønner og trykk på de ulike komponentene i systemet viktige utfall. Ruting av produksjonsrater tilfører ytterligere kompleksitet til modellen.

Det studerte optimeringsproblemet har heltallskrav, i tillegg til flere ikke-lineære strømnings- og trykkrelasjoner. Disse er approksimert ved bruk av stykkevis linearisering med spesielt ordnede sett av type 2, noe som resulterer i en blandet heltallig og lineær optimeringsmodell. Ulike metoder blir tatt i bruk for å løse problemet. Disse evalueres for å finne den mest hensiktsmessige optimeringsmetoden. Optimeringsmodellen ble opprinnelig løst ved hjelp av Branch & Bound. På grunn av problemets blokk-struktur, ble Dantzig-Wolfe dekomponering implementert for å forbedre løsningseffektiviteten. Ulike heuristikker ble lagt til for forsøke å redusere dualitets gapet.

Realistiske datasett ble generert med simulatoren PIPESIM. Algoritmene ble implementert i Mosel, som er et avansert modellerings og programmerings språk, og løst med Xpress-MP.

Resultatene viser at dekomponering av modellen har stor effekt når kompleksiteten i systemet øker. Å kombinere den dekomponerte modellen med heuristikker, viste seg å være en effektiv måte å generere gode heltallsløsninger. Når man skal bestemme hvilken metode man skal benytte, må avveiningen mellom realisme og løsningsstid vektet. En god heuristisk løsning kan ofte være mer nyttig enn å bruke eksakte løsningsmetoder som er svært tidkrevende.

Contents

List of Figures	xii
List of Tables	xiv
List of Algorithms	xv
1 Introduction	3
1.1 Petroleum Industry	3
1.2 Planning within the Petroleum Industry	4
1.3 The Marlim Field	6
1.4 The Production Asset	7
1.4.1 FPSO	7
1.4.2 System description	8
1.4.3 Gas Lift	10
1.5 Purpose and scope	11
2 Literature Review	13
2.1 Petroleum production planning	13
2.2 Gas Lift	14
2.3 Optimization at Marlim	15
3 Model formulation	17
3.1 Identifying relevant decision variables	17
3.2 Assumptions	18
3.3 Mathematical Model	19
3.3.1 Declarations	19
3.3.2 Formulation	22
4 Data simulation	29
4.1 Units	29
4.2 Simulation	30
4.2.1 Software	30
4.2.2 Well flow	31
4.2.3 Wellhead pressure	32
4.2.4 Pressure drop across riser pipes	35
4.2.5 Gas lift pipes	37

5	Linear formulation	41
5.1	Mixed-Integer Linear Programming	41
5.1.1	Branch & Bound	42
5.1.2	The Big-M method	43
5.1.3	Special Ordered Sets of type 2	44
5.2	Linearization using Big-M	46
5.3	Linearization using SOS2	50
5.3.1	SOS2 formulation	51
5.3.2	Defining breakpoint intervals	55
6	Dantzig-Wolfe decomposition	59
6.1	Decomposition methods	59
6.2	Dantzig-Wolfe Decomposition	60
6.3	Different decomposition strategies	62
6.4	Decomposition using DWD	64
6.4.1	DWD approach with MILP	64
6.4.2	Decomposing on manifolds	64
6.4.3	Decomposing on subsea manifolds and topside wells	68
7	Introducing heuristics	71
7.1	DWD applied to MILP problems	71
7.1.1	Duality gap	71
7.1.2	Branch and Price	73
7.2	Heuristics	73
7.2.1	Motivation and Assumptions	74
7.2.2	Adding new columns with different routing	75
7.2.3	Depth first	77
7.2.4	Reducing the slack based on RMP-LP	79
7.2.5	Reducing the slack based on RMP-IP	83
7.2.6	Obtaining N solutions from the subproblems	84
8	Implementation	87
8.1	Software and hardware	87
8.2	Data pre-processing	88
8.3	Implementation of the original model	89
8.4	Implementation of DWD	90
8.4.1	Finding an initial solution	91
8.4.2	Column generation	91
8.4.3	Termination criteria	92
8.5	System data	93
8.5.1	Case 1	93
8.5.2	Case 2	94
8.5.3	Case 3	94
8.6	Removal of symmetry	95
9	Computational study	97

9.1	Original model	97
9.1.1	Technical analysis	98
9.1.2	Production analysis	98
9.2	The importance of Gas lift	104
9.3	Dantzig-Wolfe decomposition	105
9.3.1	DWD test cases	105
9.3.2	Comparison of decomposition methods	105
9.4	Heuristics for improving the IP solution	108
9.5	DWD with the best heuristic	112
9.5.1	Production analysis	114
9.6	Number of SOS2 breakpoints	116
9.7	Further Improvements	120
9.7.1	Sequential versus Parallel solving	120
9.7.2	More columns per subproblem	121
9.8	Applicability in practical planning	123
10	Conclusion	125
11	Further work	127
	Bibliography	128
A	SOS2 formulation	137
A.1	The well model	137
A.2	Riser pipe pressure drop	139
A.3	Gas lift pressure drop	141
B	Implemented MILP - model	145
B.1	Sets and Indices	146
B.2	Parameters	147
B.3	Variables	147
B.4	Objective function	149
B.5	Constraints	149
C	Dantzig- Wolfe decomposed model	157
C.1	Decomposing on manifolds	157
C.1.1	Restricted master problem	157
C.1.2	Subproblems	158
C.2	Decomposing on topside wells and subsea manifolds	162
C.2.1	Restricted master problem	162
C.2.2	Subproblems	163
D	Electronical documentation	167

List of Figures

1.1	Global oil production in millions of Barrels per day (ExxonMobile 2012)	4
1.2	Oil and gas value chain	4
1.3	Planning in the Petroleum Industry	5
1.4	Overview of the Marlim Field (Offshore Technology 2012)	6
1.5	Example of an FPSO (RigZone 2012 <i>a</i>)	8
1.6	Production asset to be optimized	8
1.7	The gas lift system	11
3.1	Production asset with parameters and variables	19
3.2	Gas injection system with variables	20
3.3	Lift gas system	24
3.4	Routing of flow from subsea manifolds to pipelines	26
3.5	Routing of flow to separators	27
4.1	Flow rates for liquid, oil, gas and water as a function of BHP.	33
4.2	Screenshot of a well model in PIPESIM	34
4.3	Well 6 at manifold 3: Wellhead pressure as a function of liquid flow for different lift gas rates	34
4.4	Wellhead pressure as a function of lift gas rates and BHP	35
4.5	Model of the riser pipe system in PIPESIM	36
4.6	Pressure drop across a riser pipeline as a function of gas- and liquid (oil + water) flow rates. Shown from different angles.	37
4.7	Outlet pressure at the gas lift pipe from the compressor to subsea manifold 1	38
4.8	Well 2 at subsea manifold 2: Outlet pressure at the gas lift pipe from the wellhead to the injection point in the tubing	39
5.1	An example of a Branch & Bound tree	43
5.2	Piecewise linearization curve for $y = x^2$	45
5.3	Grid points for the non-linear function $z=g(x,y)$	45
5.4	Routing of flow to separators with the new variables	48
5.5	Routing of flow from wells to subsea manifolds with the new variable	50

5.6	Eight neighbours of an interpolation point in three dimensions . . .	52
5.7	SOS2 function approximation of manifold pressure in the gas lift pipe at subsea manifold 1, with 5 breakpoints	56
5.8	SOS2 function approximation of the riser pipe pressure drop at man- ifold 1	57
6.1	Illustration of the primal angular block structure	60
6.2	Flow of information between RMP and the subproblems	61
6.3	Dantzig-Wolfe algorithm	62
6.4	Decomposition strategies	63
7.1	Gap between UB and LB	72
7.2	Search tree for the DepthFirst heuristic	79
7.3	SlackBoundsLP heuristic example	80
7.4	SlackBoundsIP heuristic example	83
8.1	Implementation of the original model	89
8.2	Implementation of DWD	90
8.3	Sequential solving of the subproblems	92
9.1	Comparison of computational time between using a DWD heuristic and the non-decomposed model on three cases.	114
9.2	Duality gap with N subproblem solutions	122
9.3	Computational time when obtaining N columns per subproblem . . .	122

List of Tables

4.1	Pressure units	29
4.2	Flow units (RigZone 2012 <i>b</i>)	30
8.1	Wells and pipelines in Case 1	93
8.2	Separator and compressor capacities in Case 1	93
8.3	Wells and pipelines in Case 2	94
8.4	Separator and compressor capacities in Case 2	94
8.5	Wells and pipelines in Case 3	94
8.6	Separator and compressor capacities in Case 3	95
9.1	Summary of the results	98
9.2	Production and allocation of lift gas in wells for case 1	99
9.3	Routing from wells to pipelines for Case 1	100
9.4	Routing from pipelines to separators for Case 1	100
9.5	Routing from satellite wells to separators for Case 1	101
9.6	Capacity slack for separators and compressor in Case 1	101
9.7	Sensitivity analysis of total gas capacity on the separators	102
9.8	Sensitivity analysis of total water capacity on the separators	103
9.9	Sensitivity analysis of total lift gas capacity on the compressor	104
9.10	Decomposing on manifolds	106
9.11	Decomposing on subsea manifolds and topside wells	106
9.12	Summary of the heuristics	108
9.13	Combinations of the heuristics	108
9.14	Comparison of the heuristics	109
9.15	Comparison of the heuristics	110
9.16	Slack before and after SlackBoundsLP&IP	111
9.17	Optimal slack for separators and compressor in Case 2	111
9.18	Summary of the results from the best DWD heuristic	112
9.19	Production and allocation of lift gas to wells for Case 1 for the best DW heuristic	115
9.20	Routing from wells to pipelines for Case 1 for the best DW heuristic	115
9.21	Routing from pipelines to separators for Case 1 for the best DW heuristic	116

9.22 Routing from satellite wells to separators for Case 1 for the best DW heuristic	116
9.23 Overview of the complexity	117
9.24 Breakpoint sensitivity in the Original model	117
9.25 Breakpoint sensitivity in the best DW heuristic	119
9.26 Computaion time for each of the subproblems	120

List of Algorithms

6.1	Decomposing on manifolds	65
6.2	Decomposing on subsea manifolds and topside wells	68
7.1	NewSepRouting	76
7.2	DepthFirst	78
7.3	SlackBoundsLP	81
7.4	SlackBoundsIP	84

Nomenclature

B&B	-	Branch and Bound
BD	-	Benders Decomposition
B&P	-	Branch and Price
BHP	-	Bottom Hole Pressure
BOPD	-	Barrels of Oil Per Day
DW	-	Dantzig - Wolfe
DWD	-	Dantzig - Wolfe Decomposition
FPSO	-	Floating Production, Storage and Offloading
GOR	-	Gas - to - Oil Ratio
IP	-	Integer Program
LB	-	Lower Bound
LP	-	Linear Program
LR	-	Lagrangean Relaxation
MILP	-	Mixed Integer Linear Program
MINLP	-	Mixed Integer Non-Linear Program
MP	-	Master Problem
QP	-	Quadratic Programming
RMP	-	Restricted Master Problem
RTO	-	Real - Time Optimization
SOS2	-	Special Ordered Sets of Type 2
UB	-	Upper Bound

Chapter 1

Introduction

This opening chapter presents the background for this master thesis and provides an introduction to the production planning problem at the Marlim field in Brazil. Section 1.1 provides an overview of the petroleum industry today. In Section 1.2, planning in the petroleum industry is shortly described. Information about the Marlim field is presented in Section 1.3, while the production asset to be optimized is introduced in Section 1.4. Section 1.5 describes the purpose and scope of this thesis.

1.1 Petroleum Industry

Large scale oil production is believed to have started in the USA in the beginning of the 20th century (Maugeri 2007). Since then, the petroleum industry has experienced large growth, and there are several global players in the market. Figure 1.1 shows how global oil production has increased over the last decades. Today, the top three oil producing countries are Russia, Saudi Arabia and the United States (CIA 2012). Oil has a major role in today's society as it covers the high demand for vehicle fuel and is used in industrial processing.

Among the countries experiencing greatest development within the oil industry is Brazil. Brazil is currently among the top fifteen oil producing countries (CIA 2012), and it is expected to be one of the world's key supply sources of oil in the coming years (OPEC 2012). Petrobras is the leading oil company in Brazil and was the 3th largest energy company in the world in 2011, measured on market capitalization (Petrobras 2012). Petrobras has been investing heavily in Research and Development the past few years, and the company currently has the largest applied research center on the Southern Hemisphere. New research projects within the field of optimization applied to the oil industry, have been initiated by the

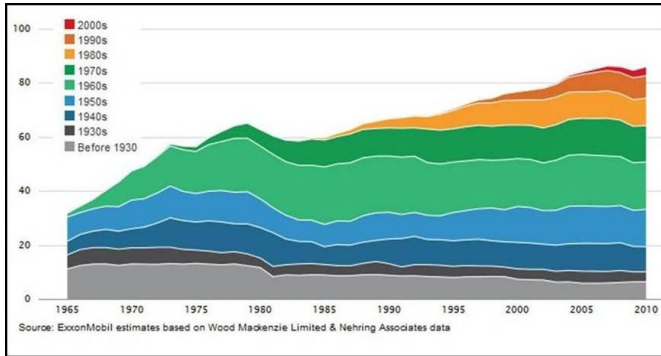


Figure 1.1: Global oil production in millions of Barrels per day (ExxonMobile 2012)

company and industrial partners. This master thesis is a contribution to the research on optimization methods applied to the Marlim field in Brazil, owned by Petrobras.

1.2 Planning within the Petroleum Industry

The value chain in the petroleum industry consists of several processes, such as exploration, production, refining, transporting (often with oil tankers and pipelines) and marketing of petroleum products (PetroStrategies Inc 2012). The industry is usually divided into three major components: upstream, midstream and downstream. Figure 1.2 illustrates how the different activities within a value chain for oil and gas are linked together and classified as upstream-, midstream- or downstream activities.



Figure 1.2: Oil and gas value chain

The red circle in Figure 1.2 indicates that the focus of this thesis is on optimization related to production, which is an upstream activity.

Planning within the petroleum industry can be divided into strategic-, tactical- and operational planning (Fleischmann et al. 2004, Anthony 1965). An overview of the planning divisions is given in Figure 1.3. Strategic planning covers long term decision making, such as investments in new field activity, equipment, technology and projects (Nygreen et al. 1998). Tactical planning has a medium-term planning horizon (3 months - 2 years) (Gunnerud 2011). The decisions include for example

aggregated production levels to meet customer demand, assuming that production capacity is decided on the strategic level, as described in Ulstein et al. (2005). Other decisions at this stage can involve allocation of resources and determining production patterns.

On an operational level, production plans are made in the short term. At this stage, the challenge is to find an optimal system state, while minimizing production costs or maximizing oil production rates (Wang 2003). Bieker et al. (2007) describe how operational decisions are based on weekly or monthly production plans. Decisions are typically related to determining optimal oil production at each well, and adjustment of valves to control the pressure. In Figure 1.3, this corresponds to the lowest level of planning, called operational production planning. Within the operational planning horizon, a sub-level called real-time production optimization (RTPO) is defined. RTPO operates on a very short time horizon. The idea is to operate the production system as close to the desired optimum as possible, at every instant of time (Bieker et al. 2007). Bieker et al. (2007) describe how this could be done by continuously collecting and analyzing data from the production system. RTPO uses real time measurements and information about physical properties from the reservoir to operate the control settings for the wells and other aspects of the system.



Figure 1.3: *Planning in the Petroleum Industry*

An important note to Figure 1.3, is that decisions at a lower planning level are constrained by decisions made at a higher level. Additionally, an optimal operational solution might not be optimal in strategic perspective. To avoid sub-optimal solutions and achieve an overall good production plan, planning at the different levels must be coordinated.

In this thesis, upstream processes in an RTPO problem are studied. Production optimization is not a trivial task to the operational planner, who is faced with many operational challenges. Decisions related to rate allocation, pressure control and flow routing between wells, pipelines and separators, must often be adjusted

on a daily basis (Private conversation January 3, 2012). Evaluating the potential in mathematical programming for this purpose is therefore important.

Optimization techniques have been applied to the upstream petroleum industry for several decades, and the methods have advanced rapidly along with technological and algorithmic developments. Investment in high technology equipment and expensive exploration and development projects, makes petroleum production a highly capital-intensive industry. At the same time, market competition among the oil companies continues to increase. As a result, focus on improved planning across all levels becomes more important.

1.3 The Marlim Field

The Marlim field was discovered in February 1985 and is located Northeast of Campos Basin, about 110km offshore Rio de Janeiro in Brazil (Lorenzatto et al. 2004). The oil field has been in production since 1998 and consists of 102 production wells, 50 injection wells, more than 80km of rigid pipelines and 400km of flexible lines (Subsea Oil & Gas Directory 2012). An overview of the field is given in Figure 1.4.

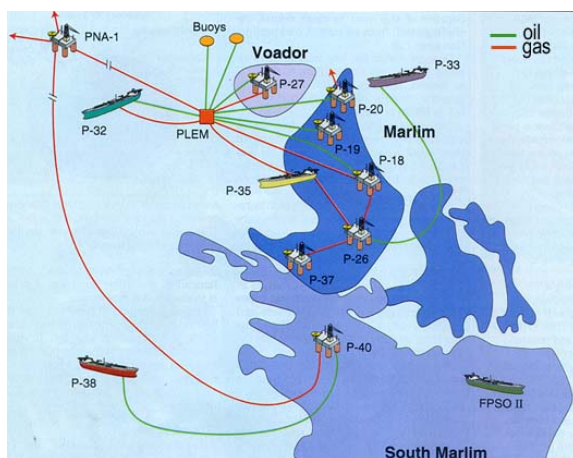


Figure 1.4: Overview of the Marlim Field (*Offshore Technology 2012*)

The field is considered to be the largest producing oil field in Brazil and is operated and owned 100% by Petrobras (Subsea Oil & Gas Directory 2012). The Marlim field produces around 390000 Barrels of oil per day (bopd), equivalent to a daily production rate of 22.4 million standard cubic meters (Sm^3/day).

The Marlim field can be classified as a brown field. The production at such fields has reached a mature stage or is even declining (Schlumberger 2012b). Cost-effective,

low-risk technologies are often installed at these fields to maintain acceptable production levels. It is increasingly important to operate brown fields in a more efficient and intelligent way to extend their economic life. More accurate optimization models, taking new technologies into consideration, can result in better decision making and lead to higher profit. One technology commonly applied to brown fields is artificial lift by injecting gas. This technology is in use at the Marlim field today (Lorenzatto et al. 2004).

Currently, Petrobras uses an in-house simulation tool to optimize lift gas allocation at the Marlim field. The optimization is based on results from well tests, updated well models and reservoir simulation (Lorenzatto et al. 2004). At the Marlim field the production is stable and the well tests are only made once every three months (Private conversation January 3, 2012). The well production rates are dynamically calculated based on the optimized gas lift allocation (Pinto et al. 2001).

1.4 The Production Asset

In this section, the production asset at the Marlim field is introduced. An explanation of the terminology used in this thesis is provided, followed by a description of important technologies installed at the production asset.

1.4.1 FPSO

This thesis addresses a floating production, storage and offloading (FPSO) unit placed at the Marlim Field. FPSO units are offshore production facilities that house both processing equipment and storage for oil, gas and water (RigZone 2012a). These units are designed to receive and process hydrocarbons produced from nearby platforms, and production wells directly coupled to the FPSO unit. Oil and gas is periodically offloaded to shuttle tankers or transmitted through pipelines. An example of an FPSO unit is illustrated in Figure 1.5.

According to Dias & da Silva (1999), Petrobras has utilized several different types of FPSO units during the development of the Marlim field. FPSO units have great advantages in locations where seabed pipelines are not cost effective. They eliminate the need of expensive long-distance pipelines from production wells to onshore terminals (FPSO 2012). FPSO units can also be used in smaller and more marginal oil fields, which can be exhausted in a few years and therefore do not justify the expense of installing a fixed oil platform. The FPSO unit can be moved to a new location when the field is depleted, and therefore there are major cost savings related to investments (Dantas et al. 1995).



Figure 1.5: Example of an FPSO (RigZone 2012a)

1.4.2 System description

The petroleum production asset optimized in this thesis is illustrated in Figure 1.6. The terminology in the figure, as defined in Schlumberger (2012b), will be used throughout the report and is explained in more detail in this section.

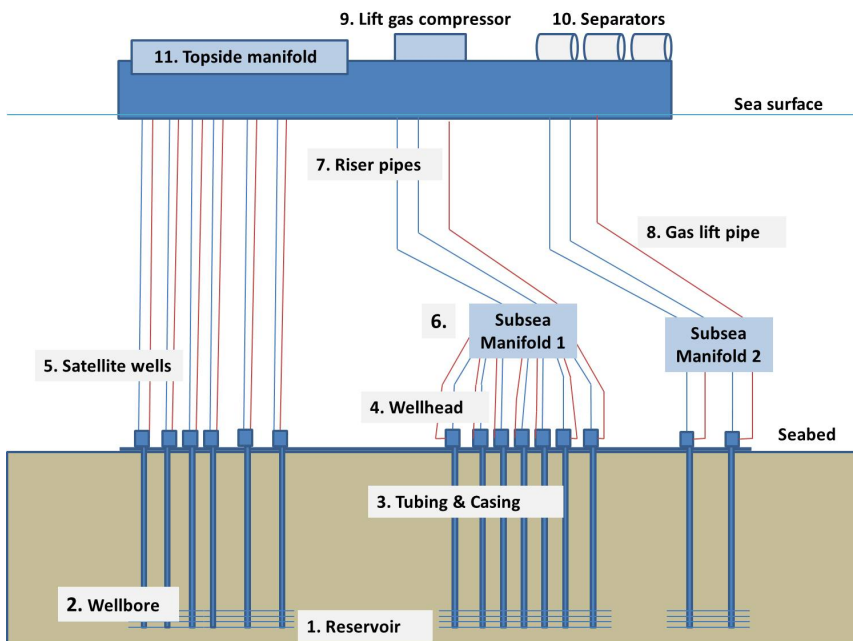


Figure 1.6: Production asset to be optimized

1. **Reservoir**

The reservoir is defined as a subsurface body of porous rock and high enough permeability to transport and store fluids and/or gas.

2. **Wellbore**

The term wellbore defines the bottom part of the well, which is partly uncased (see point 3 for the term casing). At the bottom the well is perforated, which means that there are communication tunnels in the casing through which fluids and gas can flow from the reservoir and into the well. The pressure at the wellbore is called bottom hole pressure.

3. **Tubing and Casing**

The well itself is drilled into the rock and consists of an inner pipe called tubing, where the production of fluids from the reservoir can flow. The tubing connects the wellbore to the wellhead (see points 2 and 4). The outer layer of the well piping system consists of a steel pipe called casing, which covers and protects the tubing. Casing and tubing are illustrated and explained further in relation to Figure 1.7.

4. **Wellhead**

The wellhead is a component at the surface of a well that provides a structural interface for the drilling and production equipment. One of the primary functions of the wellhead is to provide a suspension point and pressure seal for the tubing and casing. The pressure at the wellhead is regulated with a choke valve, by controlling the inlet of fluid and gas flow.

5. **Satellite wells**

Satellite wells are single wells installed to exploit marginal reservoirs that cannot be produced from wells on the permanent drilling structure (ENCYCLO 2011). Satellite wells are directly connected to a topside manifold at the platform.

6. **Manifolds**

Manifolds are arrangements of valves which control, distribute and monitor fluid flow. In the production asset in Figure 1.6, groups of wells must be routed to a pipeline for further transportation of the production flows. This routing happens through a manifold, and the manifold can be placed on both topside and subsea level.

7. **Riser pipes**

Riser pipes establish a connection between the wells and the platform. For the subsea wells, these are the pipelines between the manifolds and the platform. In the report, the terms riser pipe and pipeline are used interchangeably.

8. **Gas lift pipe**

The gas lift pipes are part of the gas lift system described in further detail later in this section. Only gas for the purpose of gas lift flows through these pipes.

9. Lift gas compressor

The compressor is found on the platform and the lift gas is stored there. Lift gas is the term used for gas which is injected as part of the gas lift process.

10. Separators

The separators are vessels placed on the platform and used to separate oil, gas and water from the multiphase flow which is produced.

11. Topside manifold

For the satellite well system, the manifold is placed on the platform.

To summarize, the production asset addressed in this report consists of 3 manifolds, where 2 are subsea- and 1 is a topside manifold. There are in total 15 wells; 6 satellite wells connected to the topside manifold, 7 wells connected to subsea manifold 1 and 2 wells connected to subsea manifold 2.

1.4.3 Gas Lift

As mentioned in Section 1.3, the production asset in Figure 1.6 is placed on a brown field, which means that production is stagnating. Some form of artificial lift system is required to help raise the well fluid to the surface. The most common methods are gas lift, sucker rod pumps, hydraulic pumps and submersible pumps (Mian 1992). At the Marlim field, gas lift is installed. Injecting lift gas into the tubing of the well reduces the fluid density, leading to a reduction in bottom hole pressure. This increases the pressure drop between the reservoir and the wellbore, and more oil can be extracted.

Figure 1.7 illustrates the gas lift process. Lift gas is transported from the sea surface through separate pipelines and is injected into each well at the wellhead. The gas flow enters the casing and is injected into the tubing near the bottom of the well, where it blends with the fluid from the reservoir (Schlumberger 2012*b*). As seen in Figure 1.7, a choke valve is used to regulate the injection of lift gas into the casing, while a check valve is installed to prevent fluid from the tubing to flow back into the casing. A check valve opens in only one direction.

There is usually a capacity limit on the amount of gas available for gas lift, and the challenge is to allocate the lift gas in an optimal way to maximize oil production. A portion of the gas that reaches the surface is re-injected into the gas lift process. The remaining gas is either exported for sale or used for electricity. The utilization of gas after it reaches the platform is not considered as part of the upstream production optimization problem studied in this thesis and will not be further discussed.

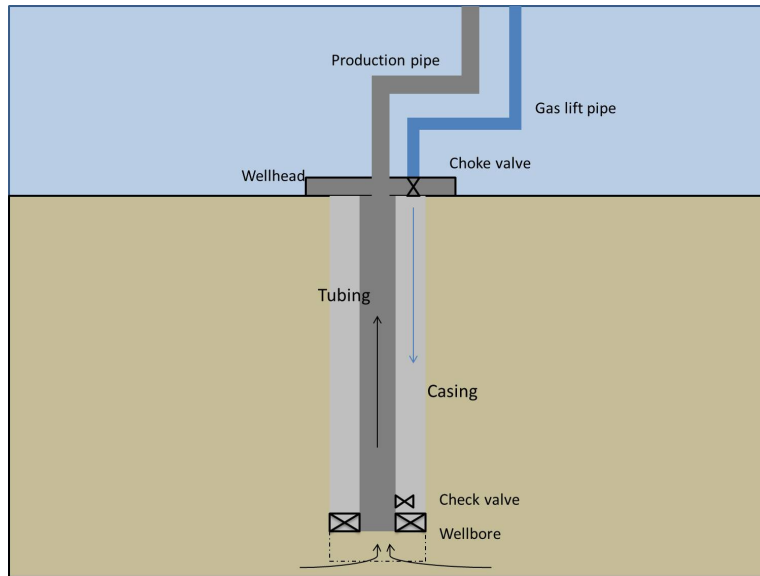


Figure 1.7: *The gas lift system*

1.5 Purpose and scope

There are several purposes with the work presented in this thesis. First, we wish to identify the important decision variables related to short term production planning at an FPSO unit at the Marlim field.

As mentioned in Section 1.4, gas lift is the only part of the production asset being optimized at Marlim today. After lift gas is allocated to wells, this determines the oil production rates at each well. However, since all decisions in the system are interrelated, there is a great potential in optimizing the production asset with regards to multiple decision variables simultaneously. For example, by combining production- and gas lift rate allocation into one optimization problem, an overall better solution might be found.

An important scope of this thesis is therefore to develop accurate, flexible and robust optimization models for the production planning problem at Marlim. In the general model formulation, an attempt will be made to model all relevant physical aspects of the real system. In addition to modelling pressure and flow rates, this work includes lift gas rates and routing of flow between wells, pipelines and separators. The objective of the problem is to maximize the oil extracted from the reservoir, assuming it is always economically preferable to produce as much oil as possible in a short term context.

Further, this thesis presents several solution strategies for the formulated models, and implements a subset of these with appropriate optimization software. Special

attention is devoted to the development of alternative solution methods exploiting the structure of the RTPO problem to improve solution quality and computational time.

Finally, some emphasis is placed on discussing the applicability of the optimization methods and solutions in practical production planning. The result of the work is not intended as a direct guideline to production engineers on how to operate the production asset at hand. Instead, the goal is to demonstrate the potential in using optimization methods as a decision support tool for the production problem being studied.

In this thesis, only optimization theory relevant to the work is discussed. The theory is explained, and methods are introduced, in the beginning of each chapter where this is relevant. The readers of this work are assumed to be familiar with basic optimization, therefore only descriptions of more advanced optimization theory are provided.

The contents of this thesis are summarized in the following. Chapter 2 gives a review of relevant literature related to optimization of oil production and optimization methods applied in this thesis. In Chapter 3, the mathematical formulation of the problem is presented. Chapter 4 contains information about the simulation software applied to generate data and a discussion of the simulation results. Linearization of the mathematical model is discussed in Chapter 5, while Chapter 6 contains a Dantzig-Wolfe decomposition of the model formulation. In Chapter 7, different methods of improving the solutions from the Dantzig-Wolfe decomposition are provided. Chapter 8 describes the implementation of the models, while Chapter 9 contains the computational results. A discussion of the optimization methods applied is presented together with the results. In Chapter 10 a conclusion of the work is provided. Suggestions for further work are found in Chapter 11.

Chapter 2

Literature Review

This chapter provides an overview of the literature related to optimization within the petroleum industry, with the aim of placing the work in this thesis into a wider context. Section 2.1 gives a general review of optimization methods developed and used across the industry, before focusing on previous work related to the real-time production optimization problem. In Section 2.2, research and articles modelling the lift gas rate allocation problem are summarized, while Section 2.3 presents a selection of literature related to optimization applied to the Marlim field.

2.1 Petroleum production planning

Existing literature related to optimization within the petroleum industry covers a wide range of topics, from strategic planning to short term rate allocation problems. Bodington & Baker (1990) report on the application of mathematical programming techniques in the industry from as early as the 1940s. They explain how the development in algorithms and computer power has lead to a widespread use of optimization methods in practical planning problems.

An example from the Norwegian industry is found in Nygreen et al. (1998), presenting a mixed-integer linear program (MILP) for long term planning of petroleum production and transportation. At the time the report was published, the model had been in use by the Norwegian Petroleum Directorate (NPD) and several Norwegian oil companies for more than fifteen years. The model uses a discrete and deterministic formulation to find the optimal time to invest in projects. Further, Ulstein et al. (2005) give a contribution to tactical planning of Norwegian petroleum production. They use a network flow formulation to describe the main processes related to production and transportation of multi-component flows, such as splitting and blending of components, to meet market demand. An MILP formulation with special ordered sets of type 1 (SOS1) is used.

The term real-time production optimization (RTPO) is widely used in the petroleum industry and in various literature. Bieker et al. (2007) place the RTPO in a practical context by describing the information flow related to optimization of an offshore petroleum production plant. A reservoir management problem is modelled in Saputelli et al. (2005), integrating business objectives and physical constraints. Kosmidis et al. (2005) develop a model for the production rate allocation problem based on the outer approximation algorithm.

An important source of inspiration to the work presented in this thesis is found in Gunnerud (2011), where the RTPO problem is addressed in several papers. Gunnerud & Foss (2010) suggest a model formulation including manifolds, wells, pipelines and routing decisions for multiphase flow at a petroleum production asset. A piecewise linear approximation with special ordered sets is used to transform the nonlinear problem into an MILP, in contrast to Wang (2003), who also considers nonlinear solution techniques.

To exploit the structure of the RTPO model and reduce computational time, Gunnerud & Foss (2010) consider decomposing the problem using Lagrangean relaxation (LR) and Dantzig-Wolfe decomposition (DWD). Foss et al. (2009) introduce the use of LR for petroleum production problems on a more conceptual level, while Gunnerud et al. (2010) apply DWD with the Branch & Price algorithm to solve problems of practical size to optimality. Hagem & Torgnes (2009) use a similar approach to solve their RTPO problem, investigating possibilities for parallelization with DWD to improve algorithm efficiency.

Glæserud & Syrdalen (2009) study a production rate allocation problem at the Troll C field, owned by Statoil. Their optimization model takes uncertainties into account by applying stochastic programming methods. The uncertainties are related to the separator capacities and well flow streams.

2.2 Gas Lift

Most of the literature covering RTPO problems mentioned in the previous section leave out the modelling of gas lift. Many mature fields use gas lift technology to maintain high production levels, and lift gas allocation to wells could beneficially be integrated as part of an RTPO model.

The lift gas allocation problem deals with deciding how to allocate the available lift gas between wells in an optimal way. An appropriate amount of lift gas injected into a well increases oil production, while excessive amounts lead to reduced production. Early literature addressing the lift gas allocation problem introduce the use of gas lift performance curves, showing oil production rate as a function of lift gas rate (Nishikiori et al. 1989). Kanu et al. (1981) present a heuristic approach based on the equal-slope principle, stating that, in the optimal solution, the slope of the gas lift performance curve should be equal for all wells.

Linear programming (LP) has frequently been used when solving lift gas allocation problems. Fang & Lo (1996) use LP to solve a combined oil production- and lift gas allocation problem maximizing daily oil production rate. Only maximizing oil rates is more recently criticized in Camponogara & Nakashima (2006). They present a model including economics in the objective function where costs can be associated with the use of lift gas. The gas lift performance curve is approximated using piecewise linearization methods, resulting in an MILP problem.

Examples of nonlinear approaches have also appeared in literature. Wang (2003) efficiently solves the lift gas allocation problem as a nonlinear programming problem, using the sequential quadratic programming (SQP) algorithm. Additionally, it should be mentioned that Wang (2003) solves an integrated production- and lift gas allocation problem, modelling flow and pressure relationships throughout the system. His contribution is among the most comprehensive for production systems using gas lift. However, routing decisions for flow between wells, pipelines and separators are not included in his study. This simplifies the problem considerably when using nonlinear solution methods, as there are no integer variables in the problem.

Codas & Camponogara (2012) include the routing decision for directing flow between a number of separators in their production rate allocation problem, in addition to modelling gas lift. Pressure relationships, on the other hand, are not modelled for the system. Eikrem (2006) addresses stabilization of gas lift wells with feedback control. He describes how oil production becomes unstable when the lift gas rate increases to a certain level.

2.3 Optimization at Marlim

Previous work on optimization at the Marlim field is limited to the use of simulation tools and trial-and-error methods, leading to sub-optimal decisions. Bampi & Costa (2010) address an optimization problem at the field development stage for Marlim, considering optimal well placement decisions. The work is based on the use of global metaheuristics and reservoir flow simulations. Optimization at the development stage of the Marlim field is also studied in Corbishley et al. (1999). They show how reservoir simulation tools are used to minimize economic risk and maximize oil production, while dynamically optimizing lift gas allocation to wells.

Several decades of experience within operational planning, drilling and reservoir development at the Marlim field is described in numerous reports, such as Awad et al. (1995), Acosta et al. (2005) and Pinto et al. (2001). However, none of these works can document the use of optimization tools integrating all the key decision variables related to short term production planning.

To summarize this chapter, a considerable amount of literature has been published within optimization of petroleum production. In the case of RTPO problems,

different elements have been highlighted separately in the literature, such as production allocation and lift gas optimization. Many elements from previous works are applicable to the Marlim field, while some aspects are less relevant. The RTPO model in this thesis does not include economics in the objective function, as seen in Saputelli et al. (2005), since it is considered unnecessary in short term production planning. Costs related to the use of lift gas are left out for the same reason. Further, uncertainty in the well flows and system capacities covered in Glæserud & Syrdalen (2009) for the Troll C field, are not as relevant for the Marlim field, which has a much more stable flow dynamics.

Similar to the work of Gunnerud (2011), an MILP formulation is used. Our model is further extended to include gas lift. In addition, the production asset at Marlim has three different separators at the platform, and parts of the formulation used by Cudas & Camponogara (2012) is appropriate for the Marlim case. To our knowledge, no one has treated the complete RTPO problem taking into account multiphase flow, pressure, routing between wells, pipelines and separators, combined with the lift gas allocation problem.

Chapter 3

Model formulation

In this chapter the petroleum production asset in Figure 1.6, presented in Chapter 1, is modeled as an optimization problem. The objective of the problem is to maximize the total amount of oil extracted from the reservoir. The outline for this chapter is as follows. First, in Section 3.1 the main decisions related to production planning at Marlim are presented. Then in Section 3.2, the assumptions behind the model are discussed. The mathematical model is described in Section 3.3. First a presentation and explanation of indices, sets, parameters and variables used in the model are provided. Finally, a presentation of the objective function and all the constraints is given.

3.1 Identifying relevant decision variables

As mentioned in Section 1.3, it can be challenging for the operational planner to optimize the whole production asset manually with regards to all decision variables. As a result, sub-optimal decisions are often made. One important aspect in this chapter is to include the decision variables having a considerable impact on total oil production. Based on conversations with production engineers at Petrobras (Private conversation January 3, 2012), the following key decision variables related to production planning at an FPSO unit are identified:

- Oil production at each well
- Flow routing between wells, pipelines and separators
- Lift gas rate allocation between wells
- Flow of gas and water from each well

Pressure levels through the production system are also treated as decision variables in this thesis, and can be regarded as secondary decisions in this context. In prac-

tical production optimization, it is generally not common to consider pressure as a variable, and the literature is divided on whether pressure should be included in the modeling or not. Gunnerud & Foss (2010), Wang et al. (2002) and Kosmidis et al. (2005) are including pressure in their optimization models, while Camponogara & Nakashima (2006) and Cudas & Camponogara (2012) are not. Pressure variables are included in this thesis to make the model as realistic and accurate as possible.

3.2 Assumptions

In this thesis, time discretization is not modeled. The output of the optimization model gives a steady state solution to the production system. In reality, fluid flow rates tend to have a dynamic behaviour. This thesis assumes steady state, due to the flow behaviour at the oil field being studied. The Marlim Field has a slow dynamic structure in the reservoirs and it takes months and years between changes. Additionally, the problem in this report is a short term production planning problem, making steady state a realistic assumption.

The optimization model maximizes total oil production instead of profit. This simplification is widely used in literature, including Gunnerud & Foss (2010), Wang (2003) and Foss et al. (2009). It is assumed that the system is designed in a way that makes it profitable to produce on full capacity. Another assumption, supporting the choice of objective function, is that there is always enough demand for oil in the market. This makes it always economically preferable to produce as much oil as possible. Further, maximizing oil production and not profit, may involve a much less complex objective function. An analysis of costs is also a huge and time consuming study (Wang 2003). In addition there is easier for petroleum engineers to understand and relate to oil production instead of profit. One challenge with the choice of objective function can be costs related to gas lift, since these costs are not included. The model does therefore not consider the trade-off between costs and benefits related to the use of gas lift, as is the case in practical production planning at Marlim (Private conversation January 3, 2012).

When modelling the gas lift system, it is assumed that the point where gas is injected is at the same height level as the wellbore itself. In reality the place of this injection point can vary from close to the bottom of the wells to near the wellhead. Since it is more common with an injection point near the wellbore, the distance is assumed to be relatively small, and is therefore neglected. The pressure at the injection point is assumed to be the same as the wellbore pressure in all the wells.

The separator pressure is in reality often equal for all separators, but in this model flexibility is added by letting the parameter values vary for the different separators

The temperature in the pipelines might have an impact of the pressure and flow rates in the system, but this impact is considered negligibly small. Temperature has most impact on the formation of wax and hydrates in the pipelines, which is an

issue related to fluid management. This does not belong to a production planning problem, and temperature is therefore left out of the model formulation in this thesis.

3.3 Mathematical Model

This section presents the mathematical model for the RTPO problem at Marlim. The notation is based on the use of lower-case letters to represent subscripts and decisions variables, and capital letters to represent sets and parameters.

3.3.1 Declarations

Figure 3.1 and Figure 3.2 show excerpts from the production asset and the gas lift system. The figures give a location overview of some of the variables and parameters used in the model formulation.

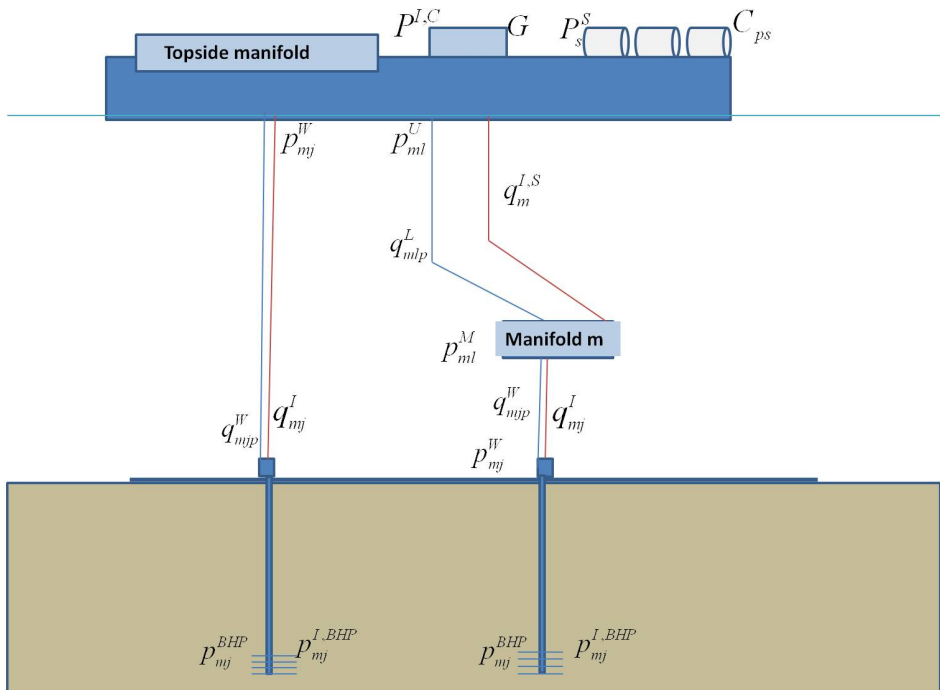


Figure 3.1: Production asset with parameters and variables

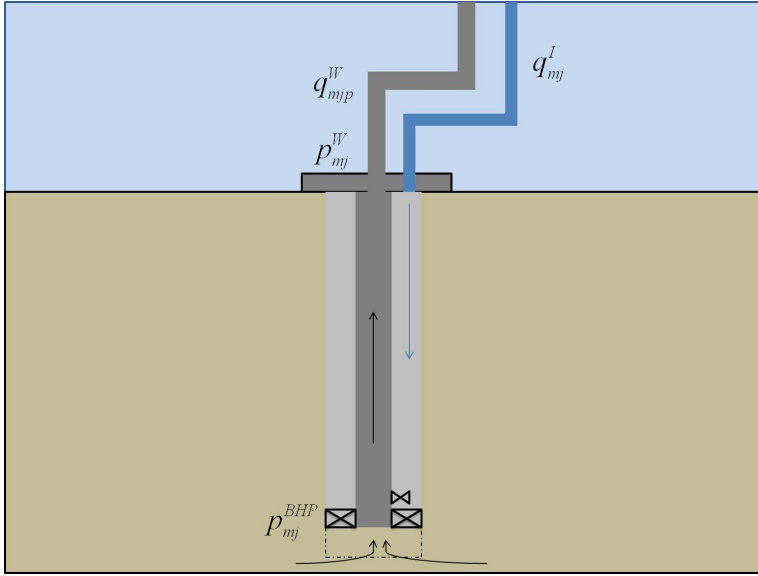


Figure 3.2: Gas injection system with variables

Sets and Indices

- \mathcal{P} - Set of phases (g for gas, o for oil, w for water)
- \mathcal{M} - Set of manifolds, indexed by $m \in 1 \dots M$
- \mathcal{M}^S - Set of subsea manifolds, indexed by $m \in 1 \dots M^S$
- \mathcal{M}^T - Set of topside manifolds, indexed by $m \in 1 \dots M^T$
- \mathcal{J}_m - Set of wells connected to manifold m , indexed by $j \in 1 \dots J_m$
- \mathcal{L}_m - Set of pipelines connected to subsea manifold m , indexed by $l \in 1 \dots L_m$
- \mathcal{S} - Set of separators, indexed by $s \in 1 \dots S$

Parameters

- C_{ps} - Capacity limit on flow of phase p in separator s
 G - Capacity limit on lift gas compressor
 P_s^S - Pressure at separator s
 $P^{I,C}$ - Pressure at the gas lift compressor

Variables

- q_{mj}^W - Well flow of phase p , from well j , connected to manifold m
 q_{mlp}^L - Pipe flow of phase p , in pipeline l , connected to subsea manifold m
 q_{mj}^I - Flow of lift gas injected to well j , connected to manifold m
 $q_m^{I,S}$ - Flow of lift gas injected to subsea manifold m
 p_{mj}^W - Wellhead pressure at well j , connected to manifold m
 $p_{mj}^{I,W}$ - Pressure in the gas lift pipe at the wellhead for well j , connected to manifold m
 p_{ml}^M - Pressure at subsea manifold m , connected to pipeline l
 $p_m^{I,M}$ - Pressure in the gas lift pipe at subsea manifold m
 p_{ml}^U - Pressure before routing to separators, from pipeline l , connected to subsea manifold m
 p_{mj}^{BHP} - Bottom hole pressure for well j , connected to manifold m
 y_{mjl} - Equals 1 if flow from well j , connected to subsea manifold m , is routed to pipeline l , 0 otherwise
 w_{mls} - Equals 1 if flow from pipeline l , connected to subsea manifold m , is routed to separator s , 0 otherwise
 u_{mjs} - Equals 1 if flow from well j , connected to topside manifold m , is routed to separator s , 0 otherwise
 x_{mj} - Equals 1 if lift gas is injected to well j , connected to manifold m , 0 otherwise

3.3.2 Formulation

Objective function

As mentioned in Section 3.2 the objective function maximizes total oil production. This means maximizing the oil flow from all pipelines connected to the subsea manifolds, and the oil flow from all satellite wells.

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlo}^L + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjo}^W \quad (3.1)$$

Constraints

The well model Output from a well depends on the properties of the reservoir. The well model consists of two parts, the multiphase flow in the wells from the reservoir and the wellhead pressure. The well flow rates are dependent on the bottom hole pressure. Total well flow of gas is the sum of gas flow from the reservoir and the lift gas injected. These relationships are represented by Constraints (3.2) and (3.3).

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{w, o\} \quad (3.2)$$

$$q_{mjg}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (3.3)$$

In addition to the bottom hole pressure, the wellhead pressure is dependent on the amount and lift gas injected, as seen in Constraints (3.4).

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (3.4)$$

Note that the "wellhead" pressure for the satellite wells is defined as the pressure on the topside manifold. This is shown in Figure 3.1.

Pressure relationships A requirement for the flow to move upwards, is that there is a pressure drop in the direction of the flow. Constraints (3.5) state the necessary relationship between wellhead and subsea manifold pressure.

$$p_{mj}^W \geq p_{ml}^M y_{mjl} \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m \quad (3.5)$$

Whenever $y_{mjl} = 1$ the wellhead pressure must be higher than or equal to the manifold pressure. When $y_{mjl} = 0$, no restriction is imposed on the wellhead pressure.

Constraints (3.6) describe the routing of flow from pipeline l , connected to subsea manifold m , to separator s .

$$p_{ml}^U \geq P_s^S w_{mls} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, s \in \mathcal{S} \quad (3.6)$$

When $w_{mls} = 1$, the flow from pipeline l , connected to subsea manifold m , is routed to separator s . In that case the pressure at the end of riser pipe l must be higher than or equal to the pressure on separator s . When $w_{mls} = 0$, no restriction is imposed on p_{ml}^U .

As mentioned before, for the satellite wells, the wellhead pressure is the same as the pressure at the topside manifold. Therefore, the wellhead pressure is used in the routing Constraints (3.7) for flow from well j , connected to topside manifold m , to separator s .

$$p_{mj}^W \geq P_s^S u_{mjs} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, s \in \mathcal{S} \quad (3.7)$$

When $u_{mjs} = 1$, the flow from satellite well j , is routed to separator s . The pressure at wellhead j , connected to topside manifold m , must be higher than or equal to the pressure on separator s . When $y_{mj} = 0$, no restriction is imposed on p_{mj}^W .

The pressure drop across the production pipelines between the subsea manifolds and the platform, is a complex property. It is dependent on several variables in the system, including temperature, absolute pressure and flows. Gunnerud (2011) formulated this pressure drop as a function of only oil, gas and water flow rates. This is considered to be a realistic assumption because the flow rates affect the pressure drop the most. The same assumption is therefore made in Constraints (3.8) in this thesis.

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L) \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (3.8)$$

The pressure drop between the compressor and the manifold in the gas lift system, is represented in Constraints (3.9) for the subsea manifolds. This is illustrated in Figure 3.3a.

$$P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,S}) \quad m \in \mathcal{M}^S \quad (3.9)$$

Constraints (3.10) represent the difference between the wellhead pressure and the pressure at the lift gas injection point in the gas lift system. The pressure at the point where lift gas is injected, corresponds to p_{mj}^{BHP} . The constraints are valid for all manifolds. Figure 3.3a shows this relationship for the subsea manifolds, and Figure 3.3b illustrates the same for the topside manifold.

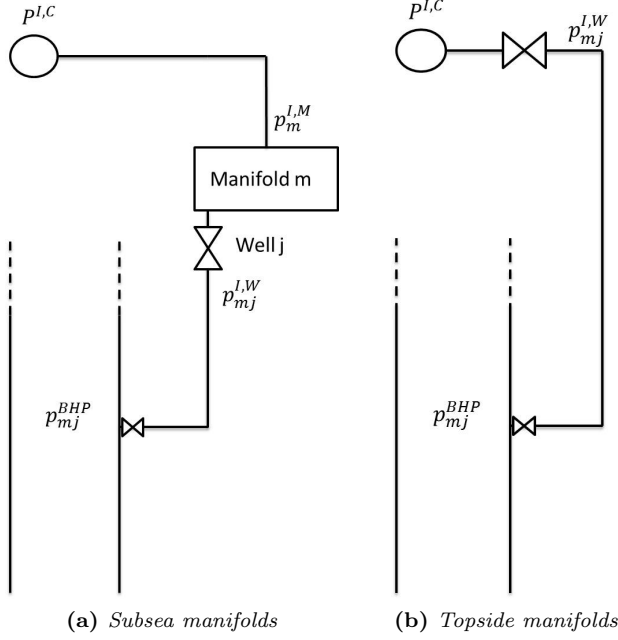


Figure 3.3: Lift gas system

$$p_{mj}^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (3.10)$$

Constraints (3.11) and (3.12) describe the pressure relationships at the choke valves shown in Figure 3.3. When $x_{mj} = 1$ the pressure conditions in Constraints (3.12) must hold. Otherwise, no restriction is imposed on the manifold pressure ($p_m^{I,M}$). Constraints (3.11) are not controlled by the binary variable as the restrictions will hold regardless of the value of x_{mj} .

$$p_{mj}^{I,W} \leq P^{I,C} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (3.11)$$

$$p_m^{I,M} \geq p_{mj}^{I,W} x_{mj} \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m \quad (3.12)$$

Capacity constraints Constraints (3.13) state that the sum of all flows of gas and water from the pipelines and satellite wells, routed to separator s , should not

exceed the capacity limit of the separators.

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlp}^L w_{mls} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjp}^W u_{mjs} \leq C_{ps} \quad p \in \{g, w\}, s \in \mathcal{S} \quad (3.13)$$

When $w_{mls} = 1$ ($u_{mjs} = 1$), the flow variables in Constraint (3.13) give a contribution to the left hand side. In case the binary variables are 0, the flows are not included in the summation.

Constraints (3.14) describe the total lift gas capacity at the compressor. The sum of lift gas injected into all the wells should not exceed the total capacity, G .

$$\sum_{m \in \mathcal{M}} \sum_{j \in \mathcal{J}_m} q_{mj}^I \leq G \quad (3.14)$$

Constraints 3.15 place an upper bound on the lift gas rate injected into well j , connected to manifold m .

$$q_{mj}^I \leq G x_{mj} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (3.15)$$

If the binary variable $x_{mj} = 0$, the lift gas rate is forced to 0. Otherwise, the amount of lift gas injected into well j , connected to manifold m , is only constrained by the total lift gas capacity. The purpose of the binary variable x_{mj} , is to control the pressure conditions in Constraints (3.12) when lift gas is injected into the wells.

The mass balance Constraints (3.16) consider the flows between wells and pipelines for the subsea manifolds:

$$\sum_{j \in \mathcal{J}_m} q_{mjp}^W y_{mjl} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \mathcal{P} \quad (3.16)$$

When $y_{mjl} = 1$, the sum of the well flow rates equal the pipe flow rate for each phase p . However, when $y_{mjl} = 0$, the pipeline flow is forced to 0, but nothing is said about the well flow rate. The well flow rate is not included in the objective function for the subsea manifolds and will therefore have no impact on the optimization result.

Constraints (3.17) make sure that the total injected lift gas into the subsea wells, should equal the amount of lift gas into the subsea manifold connected to the wells.

$$\sum_{j \in \mathcal{J}_m} q_{mj}^I = q_m^{I,S} \quad m \in \mathcal{M}^S \quad (3.17)$$

Routing constraints For the subsea manifolds, each well can only be routed to at most one pipeline. This is illustrated in Figure 3.4.

$$\sum_{l \in \mathcal{L}} y_{mj l} \leq 1 \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m \quad (3.18)$$

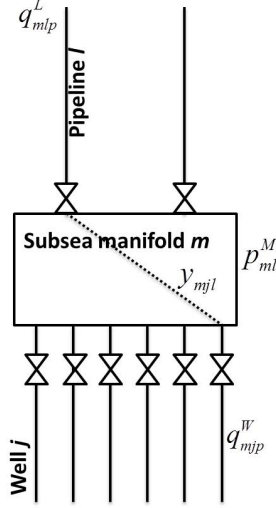


Figure 3.4: Routing of flow from subsea manifolds to pipelines

As illustrated in Figure 3.5a, the flow in each pipeline l , connected to subsea manifold m , can only be routed to at most one of the separators, s .

$$\sum_{s \in \mathcal{S}} w_{m l s} \leq 1 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (3.19)$$

Similarly for the topside manifolds, flow is routed directly from each satellite well j to at most one of the separators s . This is shown in Figure 3.5b.

$$\sum_{s \in \mathcal{S}} u_{m j s} \leq 1 \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (3.20)$$

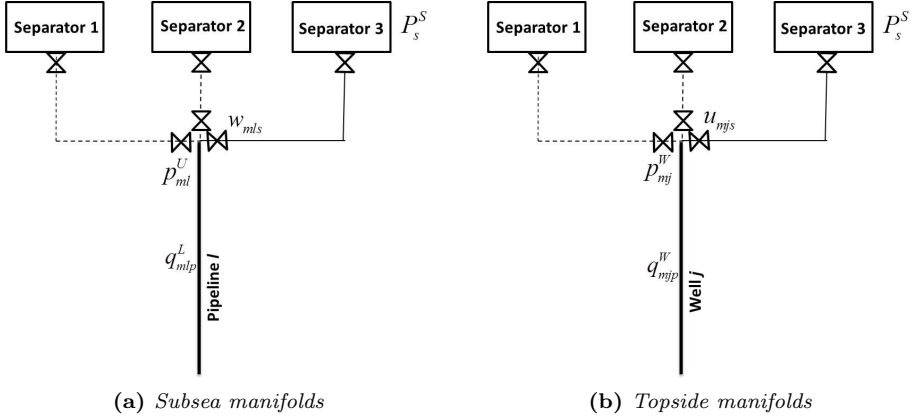


Figure 3.5: Routing of flow to separators

Binary requirements The variables listed in Constraints (3.21) can either be zero or one.

$$\begin{aligned}
 y_{mjl} &\in \{0, 1\} & m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m \\
 w_{mls} &\in \{0, 1\} & m \in \mathcal{M}^S, s \in \mathcal{S}, l \in \mathcal{L}_m \\
 u_{mjs} &\in \{0, 1\} & m \in \mathcal{M}^T, s \in \mathcal{S}, j \in \mathcal{J}_m \\
 x_{mj} &\in \{0, 1\} & m \in \mathcal{M}, j \in \mathcal{J}_m
 \end{aligned} \tag{3.21}$$

Non-negativity All the variables in the mathematical model presented in this section are non-negative.

Chapter 4

Data simulation

Some of the constraints presented in Chapter 3 describe complex thermodynamic relationships, for which exact analytical expressions often do not exist. In this chapter the main goal is to identify the behaviour and shape of these functions by applying a simulator. Section 4.1 gives a short overview of the units used in this thesis. Section 4.2 gives an overview of the functions to be simulated. Further, the simulation and relevant settings related to the PIPESIM multiphase flow simulator are described in more detail. Assumptions behind the methods and tools used are mentioned where it is relevant. Finally, the main results from the simulations are presented.

4.1 Units

The pressure unit, as defined by the SI Unit system, is Pascal (Pa). In this thesis, pressure values are given in pounds per square inch absolute (psia). The word "absolute" indicates that the atmospheric pressure¹ is included in the pressure value. Table 4.1 shows the conversion of 1 psia to the two other commonly used pressure units, pascal and bar.

Table 4.1: *Pressure units*

	Pascal	Bar
	[Pa]	[bar]
1 psia	6894.7	68.948×10^{-3}

Flow rates of oil, gas and water are given in standard cubic metres per day [Sm^3/d]. A standard cubic metre is a measure of volume under certain temperature and pres-

¹1 standard atmosphere = $1.01325 \times 10^5 Pa$ (Moran & Shapiro 2006)

sure conditions, which may vary. In the petroleum industry, standard conditions are commonly defined at 15°C and 14.7psia (ISO 5024 1999). Standard cubic feet per day [scf/d] is a unit of measurement for liquids and gases (ENCYCLO 2012). This unit is often used in millions (mmscf/d) for lift gas values in PIPESIM, to avoid dealing with large numbers. Another commonly used unit in the Petroleum industry is Barrels of Oil per Day [bopd] (Schlumberger 2012b). Table 4.2 gives an overview of these volume flow units and conversion between them.

Table 4.2: Flow units (RigZone 2012b)

	Standard Cubic Feet per Day	Barrels of Oil Per Day
	[scf/d]	[bopd]
1 Sm^3/d	35.315	6.2898

4.2 Simulation

The optimization model in Chapter 3 includes several functions describing important physical properties that define the production asset and affect the oil production. Understanding how these properties are related to each other and formulating them analytically, is not a trivial task. The following functions are studied in more detail in this chapter:

- Liquid well flow (3.2) : $q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP})$
- Gas well flow (3.3) : $q_{mjp}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I$
- Wellhead pressure (3.4) : $p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I)$
- Riser pipe pressure drop (3.8) : $p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L)$
- Pressure drop, gas lift system (3.9) : $P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,S})$
- Pressure drop, gas lift system (3.10) : $P^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I)$

To get a better understanding of the behaviour of the functions listed here, an appropriate simulation software was used. This was done in order to maintain a certain level of realism in the input data used in the optimization. At the same time it is important to keep in mind that the degree of data realism is restricted by the accuracy of the simulator.

4.2.1 Software

The simulator used in this thesis is called PIPESIM (version 2010.1). PIPESIM is a production system analysis software owned by Schlumberger and is best suited for steady-state, multiphase flow simulation. At Petrobras, PIPESIM is mainly

used at Cenpes, the research department. The company has a similar, in-house simulator available at the Marlim field, but this was not made accessible to us due to confidentiality issues. According to Private conversation (January 3, 2012) the in-house simulator has many of the same functionalities as PIPESIM; it simulates pressure drop across pipelines, well behaviour and it handles multiphase flow. PIPESIM has the additional functionality of connecting wells and pipelines into networks. It should be mentioned that the simulator at Marlim is not actively used for production optimization today.

4.2.2 Well flow

Constraints (3.2) and (3.3) describe the well flow rates for oil, gas and water as a function of bottom hole pressure (BHP):

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{w, o\}$$

$$q_{mjp}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{g\}$$

For Constraints (3.3), handling gas flow from the wells, only the first part of the expression is simulated. This represents the gas flow extracted from the reservoir. The second part is the lift gas rate included in the expression for the total gas rate flowing in the wells.

To summarize briefly what is described in Chapter 3, m is the manifold index, j is the well index and p represents flow phase (oil, gas or water). These constraints were not simulated with PIPESIM, but calculated using Vogel's Equation (Schlumberger 2010). Vogel's equation is one of several possible methods used for determining the inflow performance relationship (IPR) of a well completion. The IPR is a model for fluid flow from reservoirs to wells. According to Vogel's equation, the liquid flow rate is a function of reservoir pressure and BHP. In this thesis, reservoir pressure is assumed constant, which makes BHP the only variable when determining the flow rates:

$$Q_{liq} = Q_{max} \left(1 - (1 - C) \left(\frac{p^{Res}}{p^{BHP}} \right) - C \left(\frac{p^{Res}}{p^{BHP}} \right)^2 \right) \quad (4.1)$$

- p^{Res} - Reservoir pressure
- p^{BHP} - Bottom hole pressure (BHP)
- C - Vogel coefficient

The Vogel coefficient is a difficult parameter to predict and is usually determined through well testing. In this study, the Vogel coefficient is set equal to 0.80, based on information provided by Petrobras for the Marlim field. The reservoir pressure is set to 3800 psia.

It should also be noted that Vogel's equation is a model developed for saturated systems. Saturation is a term from thermodynamics, where gas is dissolved into oil at certain pressure and temperature conditions. More information on this topic can be found in Towler (1989). As confirmed by Petrobras, saturation is an accurate assumption for the fluids at the Marlim field. For other petroleum fields Vogel's equation might not fit the characteristics of the reservoir and wells. Different simulation tools or equations would then be necessary.

Further, Vogel's equation determines liquid flow, which is the sum of oil and water flow rates. To derive the oil-, gas- and water flows, information about gas-to-oil ratio (GOR) and water cut is necessary. GOR and water cut are commonly used terms in the oil industry and Petrobras has provided this information for each well at the production asset being studied. The ratios are defined as in Equations (4.2) and (4.3).

$$GOR = \frac{q_{gas}}{q_{oil}} \quad (4.2)$$

$$WaterCut = \frac{q_{water}}{q_{oil} + q_{water}} \quad (4.3)$$

Flow rates for liquid, oil, gas and water as a function of BHP, calculated using Vogel's equation, are shown in Figure 4.1. It is clear that the well flow for all phases decrease when increasing the BHP.

4.2.3 Wellhead pressure

Pressure at the wellhead is a function of BHP and lift gas rate injected into the wells, as stated in Constraints (3.4):

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m$$

This pressure relationship was simulated with the PIPESIM multiphase flow simulator. The following paragraph gives a brief description of how to set up PIPESIM for this purpose and how the simulations are done.

A new file in PIPESIM is created by choosing the option "Well Performance Analysis" under the Files tab. This will create a new window with a meshed grid, where

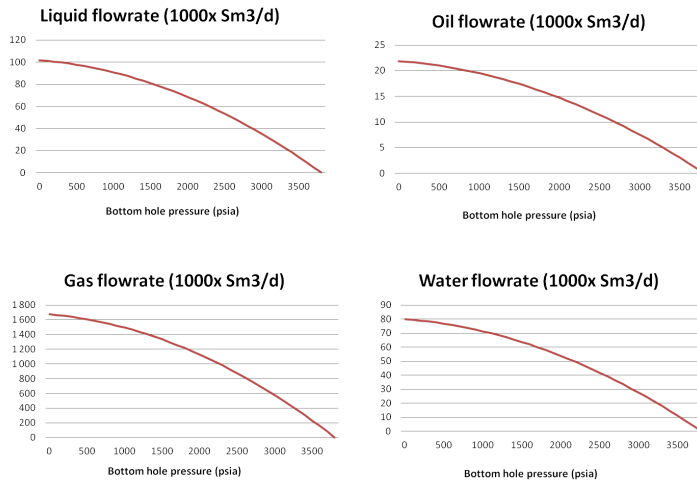


Figure 4.1: Flow rates for liquid, oil, gas and water as a function of BHP.

a well can be designed. Designing a well is done by connecting different well components, or objects, found under the Tools tab. First, a vertical well completion is added, and input data such as static reservoir pressure, temperature and choice of IPR model must be given. PIPESIM has the option of using Vogel's equation as an IPR model, which is the preferred choice, as described in the previous section.

A tubing object must be connected to the vertical well completion before the well design is completed. For the tubing object it is necessary to set the True Vertical Depth (TVD), inner diameter, ambient temperature and depth at which gas lift is injected. A screen-shot of the well model in PIPESIM is shown in Figure 4.2. Input data for all wells are based on real production data from the Marlim Field. This is confidential information and is therefore not attached to the thesis.

Before simulations can be made, it is necessary to define the fluid specifications. This is done by choosing "Black oil" under the Setup tab, and giving information about GOR and water cut for the particular well being modeled.

Simulation set up is done by opening "System Analysis" in the Operations tab. The option to calculate outlet pressure is chosen, which is the pressure at the top of the tubing object (corresponding to wellhead pressure). Simulating wellhead pressure for different BHP values is done by first manually calculating liquid flow rates in Excel, using Vogel's equation for different BHP values. Each flow rate then corresponds to one BHP value. Liquid flow rates are inserted, indirectly representing BHP, and lift gas rates are inserted for the desired range. Running the simulator generates the graph shown in Figure 4.3. The graph shows simulation results for well 6 at manifold 3 (topside manifold), and similar simulations are done for all 15 wells.

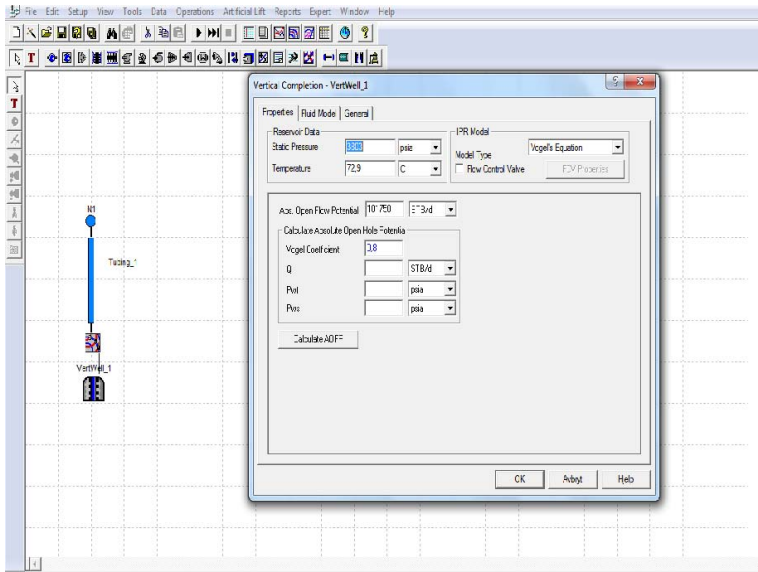


Figure 4.2: Screenshot of a well model in PIPESIM

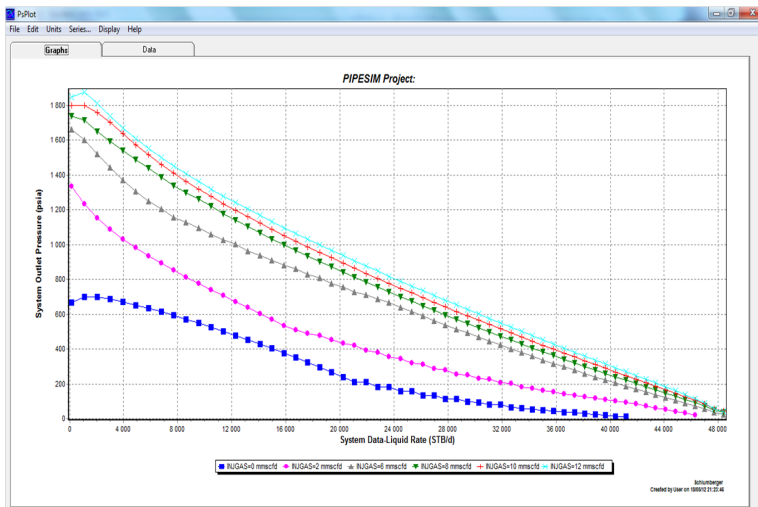


Figure 4.3: Well 6 at manifold 3: Wellhead pressure as a function of liquid flow for different lift gas rates

The curves in Figure 4.3 were generated for liquid flow rates along the x-axis, corresponding to BHP values in the range of [2000, 3800] psia with increments of 20 psia. Graphs are plotted for six lift gas rates from 0 to 12 mmcsf/d, as the figure indicates. The curves are decreasing with increased liquid flow rate. If

BHP were given directly along the x-axis, the curves would be increasing, since Vogel's equation shows that higher BHP leads to lower flow rates. This means that $p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I)$ is an increasing function of BHP, for a fixed lift gas rate.

In Figure 4.4, the graph shows the wellhead pressure in a three-dimensional plot. Wellhead pressure increases with increased BHP and lift gas rate, and the dependency is clearly non-linear. In general, BHP is preferred low in order to maximize the pressure drop between the reservoir and the bottom hole, leading to more production. At the same time, it is beneficial to have a high wellhead pressure in order to lift more flow from the well up to the platform. These two objectives are contradicting, but Figure 4.4 shows that lift gas can be used to increase wellhead pressure without changing the BHP.

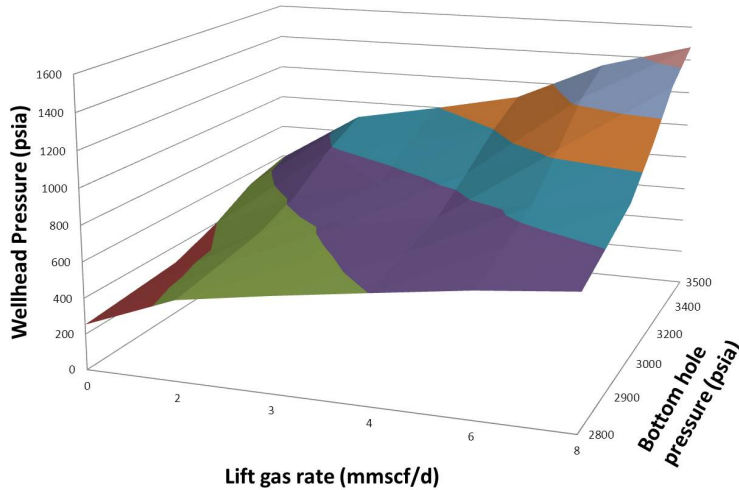


Figure 4.4: Wellhead pressure as a function of lift gas rates and BHP

4.2.4 Pressure drop across riser pipes

Constraints (3.8) represent the pressure drop across riser pipes (Section 1.4.2), defined as functions of oil-, gas- and water flow rates:

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{ml,o}^L, q_{ml,g}^L, q_{ml,w}^L) \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m$$

PIPESIM was used to run simulations in this case as well, but the simulator was run through a Macro in Excel. With PIPESIM, it is not a trivial task to simulate for different oil-, gas and water flow rates directly, as PIPESIM operates with GOR,

water cut and liquid rate. This requires some extra pre- and post calculations when simulating. To avoid this, the instances tested in our problem were generated in an Excel Macro, in a similar way as in Silva et al. (2012).

The Excel macro requires some input specifications, such as range and step size of the flow variables. System outlet pressure and black oil data must also be specified. In addition, it is necessary to create a model of the riser pipe in a PIPESIM file, and the file path should be specified in Excel. By running the Excel Macro, the PIPESIM simulator is activated. The simulation results are written to a .txt file, where pressure drop values are presented for each combination of oil-, gas- and water flow rate specified in the input.

Creating a riser model in PIPESIM is done in a similar way as the well completion model described earlier in this chapter. The riser pipe is created by connecting a flow line object to a riser object. A source object is placed before the flow line and represents the system boundary. Flow from the manifold enters the riser pipes here and moves up through the pipe system. Vertical and horizontal length, and inner diameters, must be specified for the riser and flow line.

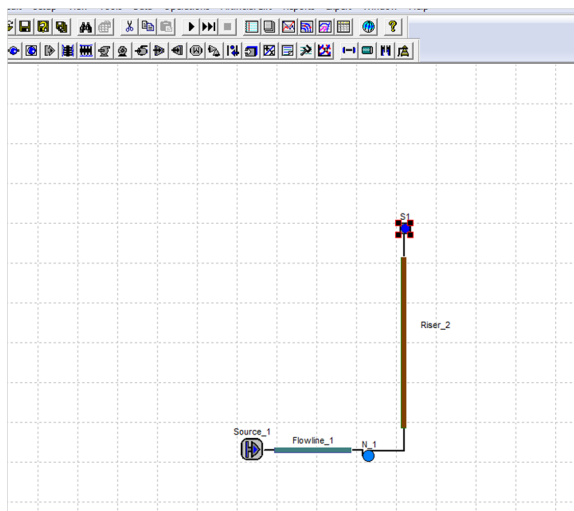


Figure 4.5: Model of the riser pipe system in PIPESIM

The riser pipe pressure drop is difficult to illustrate graphically, since it is a four dimensional plot. In Figure 4.6 the pressure drop is plotted against gas- and liquid flow rates at a fixed water cut of 50%. Liquid rates are commonly used in the industry (Private conversation January 3, 2012). Simulations were made with 10 breakpoints for each of the three flow variables. Oil- and water flow rates were chosen in the range $[0, 30]$ 1000x Sm³/d, while the gas range is $[0, 1240]$ 1000x Sm³/d.

This pressure relationship has a non-linear behaviour, and it is difficult to say

something in general about the isolated effect on pressure by any of the flow rates alone.

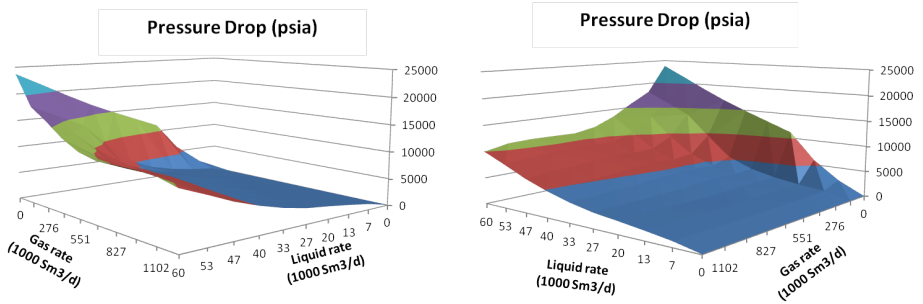


Figure 4.6: Pressure drop across a riser pipeline as a function of gas- and liquid (oil + water) flow rates. Shown from different angles.

4.2.5 Gas lift pipes

In this thesis, pressure drop is modelled for the gas lift pipe system, as shown in Figure 3.2 in Chapter 3. Constraints (3.9) define the pressure drop between the compressor to the subsea manifolds in the gas lift system:

$$P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,S}) \quad m \in \mathcal{M}^S$$

Simulations of the pressure relationship above are also done in PIPESIM. It is possible to do simple gas lift analyses in PIPESIM, simulating gas lift rates against casing (wellhead) pressure. For this purpose, however, the built-in functionality for gas lift was not used. Instead, a simple pipe system was modelled with identical components as the riser pipe system (see Figure 4.5). This is because the gas lift pipelines are parallel with, and have the same height as the riser pipes, but have a much smaller inner diameter. In this case the inner diameter of the gas lift pipelines is 3 inches. Another difference is that the source object is placed on top, and pipe height is given with a negative sign to indicate downward flow direction since gas is injected into the system.

As described earlier in this chapter, it is necessary to specify flow information before simulating. Black oil has so far been used, but in this analysis, only gas is flowing in the pipes. Therefore, the option "Compositional Template" was chosen in the Setup tab. With this option it is possible to customize the flow composition to be analyzed by setting the type of hydrocarbons and corresponding mol value. Lift gas may consist of different gas compositions and is commonly made up of ethane

(Majoni & Hamouda 2011). In this thesis it is assumed that the lift gas consists of 100% mol ethane.

Figure 4.7 shows some simulation results for the gas lift pipe to subsea manifold 1.

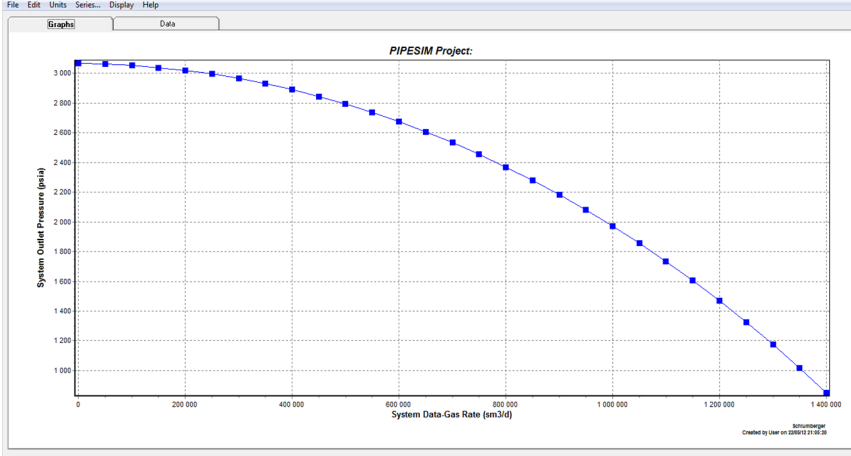


Figure 4.7: Outlet pressure at the gas lift pipe from the compressor to subsea manifold 1

The y -axis shows outlet pressure, which corresponds to $p_m^{I,M}$. To find the pressure drop, this number is simply subtracted from the constant compressor pressure ($P^{I,C}$) equal to 2500 psia. The range of lift gas rates is $[0, 1400] 1000 \times \text{Sm}^3/\text{d}$ with a step size of $50000 \text{ Sm}^3/\text{d}$. As the figure shows, the curve is non-linear but has a smooth and convex shape. Manifold pressure decreases when more lift gas is injected.

Constraints (3.10) define the second type of pressure relationship at the gas lift system:

$$p^{I,W} - p_{m_j}^{BHP} = f_{m_j}^I(q_{m_j}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m$$

Constraints (3.10) describe the difference between the pressure in the gas lift pipe at the wellhead, and the BHP. The location of the wellhead is slightly different for topside and subsea wells, as Figures 3.3a and 3.3b in Chapter 3 illustrate. Modelling these pipelines in PIPESIM is done by connecting flowline objects that stretch from the source (wellhead) to the gas lift injection point inside the tubing. At the point where lift gas is injected, there is a check valve. The valve is modelled as a very small and narrow pipe, with a length of 20 cm and a 1 inch inner diameter.

When simulating with PIPESIM it is necessary to fix at least one of the boundary conditions, in this case the pressure in the gas lift pipe at the wellhead (inlet pressure). With inlet pressure fixed, the outlet pressure, or BHP, is simulated for

different lift gas rates. The inlet pressure ($p^{I,W}$) is a variable in the optimization model. Due to simulator limitations, an assumption is made that the pressure drop as a function of lift gas rate will remain the same regardless of the boundary conditions. This is not entirely true, but the error is negligibly small. At the subsea manifolds, inlet pressure is set to 2700 psia. For the topside manifold it is fixed at 2300 psia. Since the topside wellhead is on a higher elevation than the subsea wellheads the pressure should be lower there. The pressure drop is obtained by subtracting these fixed inlet pressures from the simulated outlet pressure, as defined in Constraints (3.10).

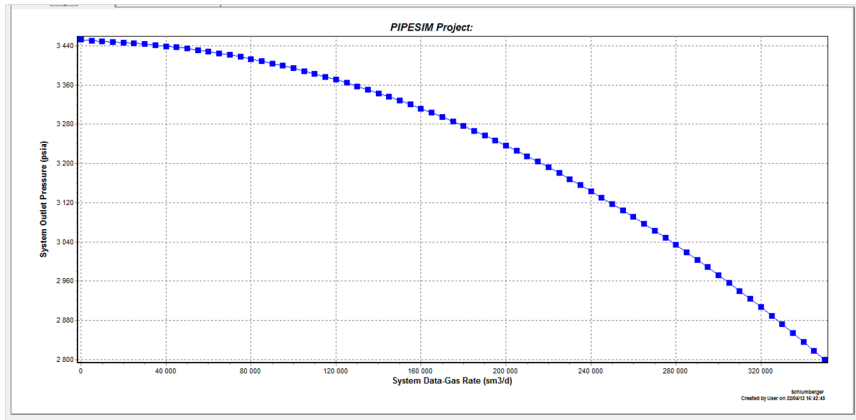


Figure 4.8: Well 2 at subsea manifold 2: Outlet pressure at the gas lift pipe from the wellhead to the injection point in the tubing

Simulation results for well 2 at manifold 2 are presented in Figure 4.8. The graph shows the simulated outlet pressure (BHP) for lift gas rates in the range $[0, 350] \times 1000 \text{ Sm}^3/\text{d}$ with a step size of $5000 \text{ Sm}^3/\text{d}$. As the previous gas lift simulation shows, this curve is also non-linear and convex. The BHP is decreasing with increased lift gas rates.

Based on the simulation results, it is clear that all the functions studied in this chapter have a non-linear behaviour. Two obvious options are available when formulating a mathematical model in this case. One option is to implement a non-linear optimization model and approximate the relationships mathematically by using curve fitting methods based on the simulation results. In this thesis, an MILP model is formulated instead, and linear approximation methods are used to describe the non-linear functions. This is explained in more detail in Chapter 5.

Chapter 5

Linear formulation

This chapter describes how the model presented in Chapter 3, is linearized and transformed into a mixed-integer linear programming (MILP) problem. The linearization methods applied when reformulating the model are the Big M method, and piecewise linearization with special ordered sets of type 2 (SOS2). A general description of MILP problems and relevant theory is presented in Section 5.1. Then, Section 5.2 shows how the Big-M method is used to linearize some of the constraints in the model. In Section 5.3, SOS2 is applied to the remaining non-linear expressions of the original optimization model.

5.1 Mixed-Integer Linear Programming

The optimization model formulated in Chapter 3 is a mixed integer non-linear programming (MINLP) problem. Binary routing variables make the problem mixed integer. Different types of non-linear expressions are also found in the model. One type involves expressions where a binary routing variable is multiplied with a continuous variable, as found in the mass balance Constraints (3.16) for example. The second type of non-linearities are the functions simulated in Chapter 4, such as the well model (Constraints (3.2) and (3.4)), the riser pipe pressure drop (Constraints (3.8)) and the pressure drop across gas lift pipes (Constraints (3.9) and (3.10)).

The MINLP class of problems is among the most challenging, due to the combined integer and non-linear characteristics. Algorithms than can be applied to practical MINLP, have mainly been developed during the last two decades. Increase in computational power has also contributed to the solvability of such problems. Examples of promising methods to solve MINLPs are found in Bonami et al. (2005). Among non-linear solvers are BONMIN, developed by the Carnegie Mellon University and IBM, and BARON. The solvability of MINLP problems to optimum is highly dependent on the problem characteristics, i.e whether the problems are

convex or not. More can be found on convexity related to non-linear programming in Kasana & Kumar (2004).

A simpler type of optimization problem is a mixed-integer linear programming problem (MILP), for which successful algorithms have existed since the 1960s, such as the Branch & Bound algorithm Lundgren et al. (2010). Further, an MILP formulation allows for simplicity in calculating bounds on the global optimum. Information about bounds can be useful when solving large scale MILP problems. This information is in fact exploited by the Branch & Bound algorithm, as described in Section 5.1.1. The most challenging feature of MILP models are the integer variables, otherwise these problems are generally easier to solve than non-linear problems with the methods and tools available today.

The MINLP problem studied in this thesis is reformulated into an MILP problem. This choice is mainly motivated by the possibility to exploit the advantages of MILP-modelling, mentioned in the previous paragraph. As highlighted by D'Ambrosio et al. (2010), the remarkable increase in efficiency of MILP software tools has also encouraged their use when solving non-linear problems. Successful results have been documented for similar production planning problems using MILP, for example in the works of Gunnerud (2011), Camponogara & Nakashima (2006) and Ulstein et al. (2005). This supports our belief that an MILP formulation should work well for the optimization problem being studied here. The performance and accuracy of an MILP model formulation is, however, highly dependent on finding efficient techniques to linearize non-linear functions.

5.1.1 Branch & Bound

Branch & Bound (B&B) is a solution method for solving integer programming (IP) problems. The algorithm splits the feasible region into smaller regions, and a relaxed problem with continuous variables, is solved in each subproblem. The procedure can be illustrated by a search tree, with nodes representing the subproblems. From each node, "branching" is done by constructing new subproblems, meaning new constraints are added. Throughout the search tree, pessimistic and optimistic bounds are generated by solving the relaxed subproblems. For a minimization problem, each feasible IP solution is an upper (pessimistic) bound to the original problem. Since more constraints are added to the problem by branching, all solutions to the subproblems at a deeper level in the search tree can only be equal to, or worse than the previous solutions. Knowing this fact, whenever an LP solution at a node is worse than the best pessimistic bound, no better solution can be found by branching, and the search from that node is terminated. When no further branching is necessary from any node, the best pessimistic bound is the optimal solution to the original problem (Lundgren et al. 2010). Figure 5.1 shows an example of a Branch & Bound tree.

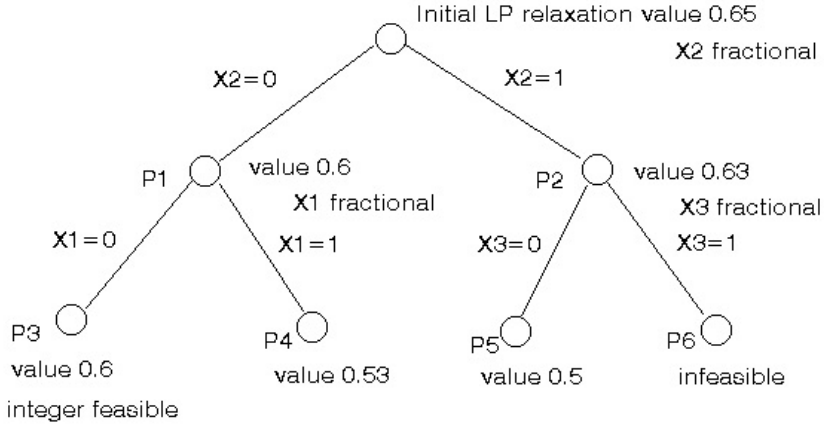


Figure 5.1: An example of a Branch & Bound tree

5.1.2 The Big-M method

Constraints that involve expressions where a continuous variable is multiplied by a binary variable are intended to indicate certain states. In this case, the binary variables are called indicator variables (Williams 1999). For example, if a binary variable equals 1, this could indicate that a continuous flow variable is added to a mass balance constraint, otherwise not.

These types of constraints make the optimization problem non-linear, which we are trying to avoid. Williams (1999) describes an alternative, linear method for linking decision variables with continuous variables to indicate certain states. Linearization is achieved using a well known optimization method where a constant coefficient, M , is introduced, representing a lower or upper bound to the constraint being linearized. Instead of multiplying two variables, thereby creating a non-linear term, the constant coefficient, M , can be multiplied with the indicator variable. An example is given below.

$$y \geq x\delta \quad (5.1)$$

Let x and y be two continuous, non-negative variables, and let δ be a binary variable. Constraint (5.1) is non-linear. δ equal to 1 indicates the state where $y \geq x$. When δ equals 0, no restriction is imposed on y . By introducing a constant coefficient, M , Constraint (5.1) can be reformulated into a linear constraint:

$$y - x \geq M(1 - \delta) \quad (5.2)$$

Whenever δ equals 1, Constraint (5.2) indicates the same state as Constraint (5.1), namely $y \geq x$. If δ equals 0, Constraint (5.2) gives that $y - x \geq M$.

When determining an appropriate M value, it is important to avoid imposing any additional restrictions on the problem when the value is present in the constraint (i.e. when the binary variable equals 0). An option is to assign an arbitrarily large number (absolute value) to the coefficient to be sure that it will not restrict the problem. However, there could be computational advantages in making the coefficient as realistic as possible (Williams 1999), by acquiring information about maximum (or minimum) values of the terms being bounded by M .

To avoid imposing any restriction on the left hand side of Constraint (5.2) from the example above, M must at least be as small as the minimum possible value of the left hand term:

$$M = \min\{y - x\} \tag{5.3}$$

5.1.3 Special Ordered Sets of type 2

A standard method for linearizing non-linear functions is by using piecewise linear approximation. The non-linear functions in the optimization model in Chapter 3 are approximated using piecewise linearization with special ordered sets (SOS). This method was first introduced in Beale & Tomlin (1970), and is extensively treated in literature, including Keha et al. (2004), Vielma et al. (2010) and Misener et al. (2009). The latter solves a lift gas allocation problem using four different linearization methods, with the conclusion that a type of SOS formulation is the most promising (type 2). SOS sets have been applied to several large-scale petroleum optimization problems, as seen in Gunnerud (2011), Kosmidis et al. (2005) and Bieker et al. (2006). The principle behind the method is explained in this section.

Taking the function $y = x^2$ as an example, a fixed number of breakpoint values for x and y , denoted by X_i and Y_i , are defined. Let I be the set of all breakpoints, indexed with i . Straight lines can be drawn between the points, as illustrated in Figure 5.2a. Nonnegative weight variables, λ_i , are associated with each breakpoint i . This gives rise to the following constraints:

$$x = \sum_{i \in I} X_i \lambda_i \tag{5.4}$$

$$y = \sum_{i \in I} Y_i \lambda_i \tag{5.5}$$

$$\sum_{i \in I} \lambda_i = 1 \tag{5.6}$$

If the breakpoint values associated with x and y can be sorted, for example in increasing or decreasing order, it is possible to define the weight variables as a special ordered set (SOS). It is common to distinguish between two types of sets, SOS1 and SOS2. SOS1 states that at most one of the weight variables can be non-zero. With an SOS2 formulation, at most two weight variables can be non-zero. The latter allows for interpolation between two breakpoints. As a result, fewer breakpoints are generally necessary with SOS2 compared to SOS1.

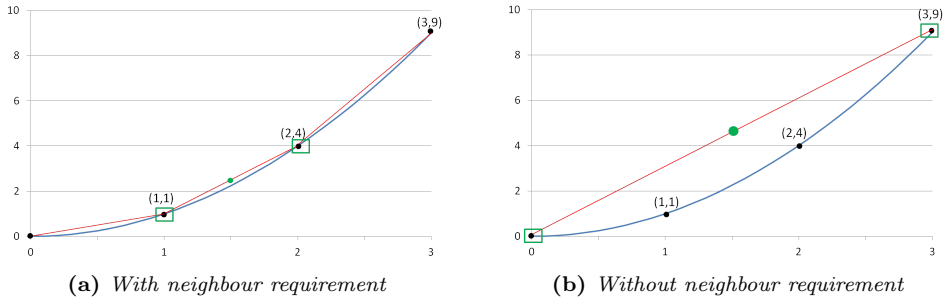


Figure 5.2: Piecewise linearization curve for $y = x^2$

Further, a SOS2 formulation requires that the non-zero weight variables are *adjacent*. This forces the solution to lie on one of the straight lines of the piecewise linear curve, as shown in Figure 5.2a. Relaxing the neighbour requirement might lead to poor function approximations, as seen in Figure 5.2b. In some cases the objective function can guarantee that the SOS2 neighbour requirement is always fulfilled, making it unnecessary to define variables as SOS2.

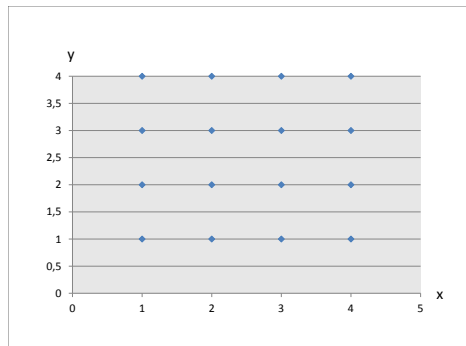


Figure 5.3: Grid points for the non-linear function $z=g(x,y)$

SOS2 can also be used when linearizing multi dimensional functions, as covered in Williams (1999). A non-linear function $z = g(x,y)$ is used as an example. In this case it is necessary to discretize in two dimensions, x and y , which results

in piecewise linear *surfaces* of z . Let I be a set of breakpoints for the variable x , indexed by i , and J the set of breakpoints for variable y , indexed by j . Grid points, not necessarily equidistant, (X_i, Y_j) can be defined as illustrated in Figure 5.3.

Nonnegative weight variables, λ_{ij} , are defined for each grid point. The formulation can be written as follows:

$$x = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} X_i \lambda_{ij} \quad (5.7)$$

$$y = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} Y_j \lambda_{ij} \quad (5.8)$$

$$z = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} Z_{ij} \lambda_{ij} \quad (5.9)$$

$$\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \lambda_{ij} = 1 \quad (5.10)$$

To make sure that the solution lies on the piecewise linear surface it is necessary to allow interpolation between at most four non-zero, neighbouring λ_s . This condition can be viewed as a generalization of a SOS2 set. To ensure that interpolation can be made between four neighbour points, new variables are introduced in the following way:

$$\delta_i = \sum_{j \in \mathcal{J}} \lambda_{ij}, \quad \gamma_j = \sum_{i \in \mathcal{I}} \lambda_{ij} \quad (5.11)$$

Instead of defining the weight variables, λ_{ij} , as SOS2 (as in the one dimensional case), the sets $\delta_i, i \in \mathcal{I}$, and $\gamma_j, j \in \mathcal{J}$ are defined as SOS2. In a similar way, this method can be used with an arbitrary number of variables.

5.2 Linearization using Big-M

In this section Constraints (3.5), (3.12), (3.13) and (3.16) are linearized using the Big-M method introduced in Section 5.1.2.

Manifold routing The pressure relationship in Constraints (3.5) is reformulated as follows:

$$p_{mj}^W \geq p_{ml}^M y_{mjl} \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m$$

$$p_{mj}^W \geq p_{ml}^M + M_{mjl}^1 (y_{mjl} - 1) \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m \quad (5.12)$$

From Constraints (5.12), it is clear that whenever $y_{mjl} = 1$, the wellhead pressure must be greater than or equal to the manifold pressure. When $y_{mjl} = 0$, the following term is obtained:

$$p_{ml}^M - p_{mj}^W \leq M_{mjl}^1 \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m$$

To avoid any additional restrictions on the pressure values, M_{mjl}^1 must be the largest difference between p_{ml}^M and p_{mj}^W for all subsea manifolds, wells and pipelines.

$$M_{mjl}^1 = \max_{m,j,l} \{p_{ml}^M - p_{mj}^W\}$$

Gas lift pressure The gas lift pressure Constraints (3.12), are reformulated in a similar way as the manifold routing:

$$p_m^{I,M} \geq p_{mj}^{I,W} x_{mj} \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m$$

$$p_m^{I,M} \geq p_{mj}^{I,W} + M_{mj}^2(1 - x_{mj}) \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m \quad (5.13)$$

Constraints (5.13) show that whenever $x_{mj} = 1$, the manifold pressure must be greater than or equal to the wellhead pressure in the gas lift system. When $x_{mj} = 0$, the following term is obtained:

$$p_m^{I,M} - p_{mj}^{I,W} \geq M_{mj}^2 \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m$$

M_{mj}^2 must in this case be the smallest difference between $p_m^{I,M}$ and $p_{mj}^{I,W}$ for all manifolds and wells, to avoid introducing any new constraints on the pressure variables.

$$M_{mj}^2 = \min_{m,j} \{p_m^{I,M} - p_{mj}^{I,W}\}$$

Separator capacity The separator capacity Constraints (3.13) for gas and water are also reformulated:

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlp}^L w_{msl} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjp}^W u_{msj} \leq C_{sp} \quad s \in \mathcal{S}, p \in \{g, w\}$$

It is necessary to introduce new flow variables:

- q_{mlps}^{LL} - Line flow of phase p , in pipeline l , at subsea manifold m to separator s
- q_{mjpg}^{WW} - Well flow of phase p , from well j at topside manifold m to separator s

The new variables are explained in Figure 5.4. For the subsea manifolds, Figure 5.4a shows that the new flow variable, q_{mlps}^{LL} , is the portion of the total pipeline flow, q_{mlp}^L , that is routed to separator s . The similar is done for the flow routing at the topside manifold. The new flow variable, q_{mjpg}^{WW} , is an amount of the total well flow, q_{mj}^W , routed to separator s , as seen in Figure 5.4b.

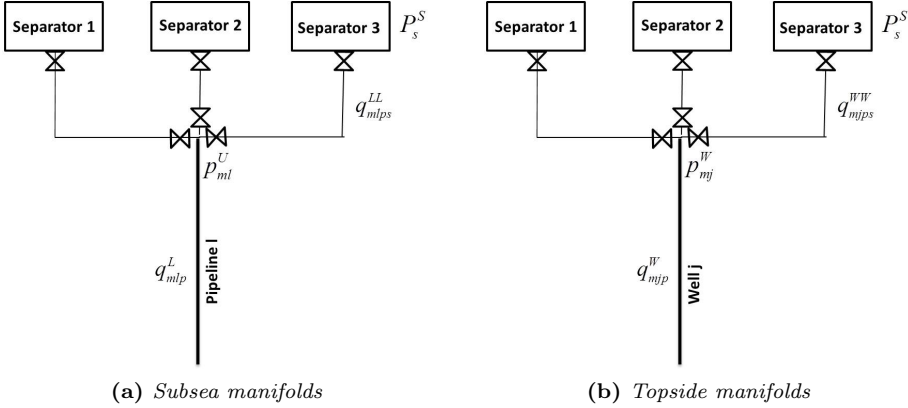


Figure 5.4: Routing of flow to separators with the new variables

Constraints (5.14) - (5.18) are included in the new MILP formulation.

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlps}^{LL} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjpg}^{WW} \leq C_{sp} \quad s \in \mathcal{S}, p \in \{g, w\} \quad (5.14)$$

$$q_{mlps}^{LL} \leq C_{sp} w_{msl} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (5.15)$$

$$q_{mjps}^{WW} \leq C_{sp} u_{msj} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (5.16)$$

$$\sum_{s \in \mathcal{S}} q_{mjps}^{WW} = q_{mjp}^W \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, p \in P \quad (5.17)$$

$$\sum_{s \in \mathcal{S}} q_{mlps}^{LL} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in P \quad (5.18)$$

C_{sp} represents the big-M parameter previously discussed. This is the capacity limit of separator s for phase p and the largest possible value any single flow rate variable can take. If $w_{msl} = 0$ (or $u_{msj} = 0$), the separator flow rates are forced to be 0 due to the combination of the above formulation and the non-negativity constraints. When $w_{msl} = 1$ (or $u_{msj} = 1$) the flow to separator s should not be restricted, and big-M must therefore equal the maximum capacity, C_{sp} .

Mass balance Finally, the mass balance Constraints (3.16) are linearized.

$$\sum_{j \in \mathcal{J}} q_{mjlp}^W y_{jml} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in P$$

A new variable is introduced:

$$q_{mjlp}^{WM} \quad - \quad \text{Well flow of phase } p, \text{ routed from well } j \text{ to pipeline } l \text{ at subsea manifold } m$$

The variable is illustrated in Figure 5.5. As the figure shows, the new flow variable, q_{mjlp}^{WM} , represents the portion of the total well flow rate, q_{mjp}^W , that is routed to pipeline l . Constraints (5.19) - (5.21) are included in the new MILP formulation.

$$\sum_{j \in \mathcal{J}_m} q_{mjlp}^{WM} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in P \quad (5.19)$$

$$q_{mjlp}^{WM} \leq M_{mjlp}^3 y_{jml} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, j \in \mathcal{J}_m, p \in P \quad (5.20)$$

$$\sum_{l \in \mathcal{L}_m} q_{mjlp}^{WM} = q_{mjlp}^W \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (5.21)$$

$$M_{mjlp}^3 = \max_{m,j,l,p} \{q_{mjlp}^{WM}\}$$

M_{mjlp}^3 is the value of the greatest possible flow from well j , to pipeline l , of phase p at manifold m . Whenever $y_{mjlp} = 0$, the flow rate q_{mjlp}^{WM} is forced to zero. If $y_{mjlp} = 1$, no restriction is imposed on the problem, due to the choice of parameter value M_{mjlp}^3 .

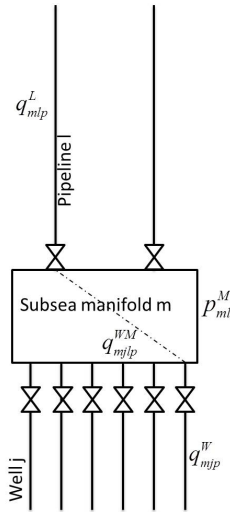


Figure 5.5: Routing of flow from wells to subsea manifolds with the new variable

5.3 Linearization using SOS2

According to the simulation results found in Chapter 4, the functions defined in Constraints (3.2), (3.3), (3.4), (3.8), (3.9) and (3.10) have a non-linear behaviour. These constraints define the well model, pressure drop across the gas lift pipes, and pressure drop across the riser pipes. From the simulation results, it is not obvious that the solution will lie between two neighbour points in the linear approximations, when the oil production is maximized. Similar to the production allocation problem in Torgnes et al. (2011), SOS2 requirements are therefore necessary in the optimization problem studied in this thesis.

5.3.1 SOS2 formulation

The functions are linearized using a SOS2 formulation, as described in Section 5.1.3. Linearization of Constraints (3.8) and (3.9) is presented in this section, to show the use of SOS2 in one and several dimensions. The complete SOS2 formulations are found in Appendix A.

Gas lift pressure drop linearization

Constraints (3.9) define the pressure drop across the gas lift pipes from the compressor to the subsea manifolds:

$$P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,M}) \quad m \in \mathcal{M}^S$$

The constraints are only dependent on one variable, and linearization using SOS2 is quite straight forward, as described in Section 5.1.3. Breakpoints for different lift gas rates are defined and a new weight variable, associated with each breakpoint, is introduced. The complete formulation is presented below, showing new sets, indices, parameters, variables and constraints related to the SOS2 formulation.

- \mathcal{R} - Set of breakpoints for values of gas injection to manifolds , indexed by $r \in 1 \dots R$
- $Q_{(m)r}^{I,S}$ - Value of lift gas rate at breakpoint r for manifold m .
- $F_{(m)r}^{I,S}$ - Value of pressure drop from manifold m to the compressor, at breakpoint r .
- $\mu_{(m)r}$ - Weight of breakpoint r .

Constraints (5.22) - (5.26) represent the SOS2 formulation, which replaces the non-linear Constraints (3.9) in the new MILP model.

$$q_m^{I,S} = \sum_{r \in \mathcal{R}} Q_{mr}^{I,S} \mu_{(m)r} \quad m \in \mathcal{M}^S \quad (5.22)$$

$$\sum_{r \in \mathcal{R}} F_{(m)r}^{I,S} \mu_{(m)r} = P^{I,C} - p_m^{I,M} \quad m \in \mathcal{M}^S \quad (5.23)$$

$$\sum_{r \in \mathcal{R}} \mu_{(m)r} = 1 \quad m \in \mathcal{M}^S \quad (5.24)$$

$$\mu_{(m)r} \geq 0 \quad m \in \mathcal{M}^S, r \in \mathcal{R} \quad (5.25)$$

$$\mu_{(m)r} \quad \text{is SOS2 for } r \quad m \in \mathcal{M}^S, r \in \mathcal{R} \quad (5.26)$$

Riser pipe pressure drop linearization

Constraints (3.8) define the pressure drop across the riser pipes as a function of oil-, gas- and water flowrates:

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L) \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m$$

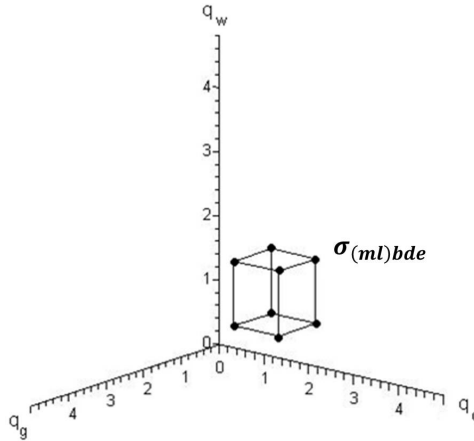


Figure 5.6: Eight neighbours of an interpolation point in three dimensions

The pressure drop function is dependent on three variables, making it less intuitive to reformulate using piecewise linearization. As described in Section 5.1.3, it is possible to formulate SOS2 in several dimensions. Gunnerud & Foss (2010) use SOS2 to linearize the same pressure drop in their production allocation problem, by introducing a weight variable for each breakpoint in a three dimensional grid where the axes represent oil-, gas- and water flowrates. The SOS2 requirement now

implies that at most eight neighbouring points can be non-zero. This is illustrated in Figure 5.6.

New sets and indices are formulated for the breakpoints related to each of the three flow variables:

- \mathcal{B} - Set of breakpoints for values of pipeline oil flow, indexed by $b \in 1 \dots B$
- \mathcal{D} - Set of breakpoints for values of pipeline gas flow, indexed by $d \in 1 \dots D$
- \mathcal{E} - Set of breakpoints for values of pipeline water flow, indexed by $e \in 1 \dots E$

Each breakpoint gives rise to a new parameter, representing flow or pressure values related to that specific breakpoint. A weight variable, $\sigma_{(ml)bde}$, is introduced for each breakpoint in the grid:

- $Q_{(ml)b}^O$ - Value of oil flow in pipeline l , connected to manifold m , at breakpoint b .
- $Q_{(ml)d}^G$ - Value of gas flow in pipeline l , connected to manifold m , at breakpoint d .
- $Q_{(ml)e}^W$ - Value of water flow in pipeline l , connected to manifold m , at breakpoint e .
- $F_{(ml)bde}^L$ - Value of pressure drop over pipeline l , connected to manifold m , at breakpoint b , d and e .
- $\sigma_{(ml)bde}$ - Weight variable of breakpoint b , d and e for subsea manifold m and pipeline l .
- $\theta_{(ml)b}^B$ - Sum of weights of breakpoints d and e for breakpoints b for subsea manifold m and pipeline l .
- $\theta_{(ml)d}^D$ - Sum of weights of breakpoints b and e for breakpoints d for subsea manifold m and pipeline l .
- $\theta_{(ml)e}^E$ - Sum of weights of breakpoints b and d for breakpoints e for subsea manifold m and pipeline l .

Constraints (5.27) - (5.36) define the new SOS2 formulation, replacing Constraints (3.8) in the MILP problem. Constraints (5.32) - (5.34) together with Constraints (5.36), ensure that interpolation can be made between eight neighbour points in a three dimensional grid.

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)b}^O \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in o \quad (5.27)$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)d}^G \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in g \quad (5.28)$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)e}^W \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in w \quad (5.29)$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} F_{(ml)bde}^L \sigma_{(ml)bde} = p_{ml}^M - p_{ml}^U \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (5.30)$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} = 1 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (5.31)$$

$$\theta_{(ml)b}^B = \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \quad (5.32)$$

$$\theta_{(ml)d}^D = \sum_{b \in \mathcal{B}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \quad (5.33)$$

$$\theta_{(ml)e}^E = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \quad (5.34)$$

$$\begin{aligned} \sigma_{(ml)bde} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, b \in \mathcal{B}, d \in \mathcal{D}, e \in \mathcal{E} \\ \theta_{(ml)b}^B &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \\ \theta_{(ml)d}^D &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \\ \theta_{(ml)e}^E &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \end{aligned} \quad (5.35)$$

$$\begin{aligned}
\theta_{(ml)b}^B & \quad \text{is SOS2 for } b & \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \\
\theta_{(ml)d}^D & \quad \text{is SOS2 for } d & \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \\
\theta_{(ml)e}^E & \quad \text{is SOS2 for } e & \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E}
\end{aligned} \tag{5.36}$$

5.3.2 Defining breakpoint intervals

Several important factors must be considered when constructing breakpoint intervals for SOS2. First, the breakpoint resolution must be decided. Second, the maximum and minimum breakpoint values, defining the interval, must be chosen. Then, it might also be relevant to consider how to optimally place the breakpoints over the chosen interval, i.e with equal distance, or having regions with denser and coarser resolution. These factors have an impact on the solvability of the optimization problem, as well as the accuracy of a solution.

Interval boundaries

When selecting data points for the piecewise linearization of a function it is important to choose an interval that will not restrict the optimization problem. Therefore, it could be useful to have some idea of the region in which the solution is expected to lie. If the minimum value of the defined interval is too high, the problem might not find the optimal solution because it is not included in the interval. The same is true if the maximum value is too low. On the other hand, the interval should not be made unnecessarily large because it would increase the problem complexity and computational time.

Number of data points

Problem complexity introduces the next important decision, namely deciding how many data points to include in the interval. To obtain the best possible function approximation, it is obviously always good to include as many data points as possible. From an optimization point of view, however, this is not a wise strategy, because the problem rapidly increases in size when adding data points, as pointed out in Gunnerud & Foss (2010). Taking the three dimensional case as an example, 10 breakpoints in each dimension gives $10^3 = 1000$ possible interpolation points, each associated with a weight variable. When this number of variables is added for each subsea manifold-pipeline combination (ml), as in the riser pipe pressure drop formulation, the problem expands dramatically. In addition, a SOS2 formulation comes with several new constraints which add to the problem complexity. A trade-off must often be made between solution accuracy and model size when determining the number of data points in the piecewise linearization.

Placements of data points

With a limited number of breakpoints, it is important to place the points wisely across the interval to approximate the function as well as possible. In this work, the

data points are mostly equidistant, with a few exceptions. When modelling gas lift for example, it is necessary to include the possibility of zero lift gas rate, but most of the remaining data points are chosen much higher since the solution is more likely to be in the higher region. Optimizing the number and placement of data points for SOS2 is not the main scope of this thesis, but some analysis has been made, for example related to the choice of interval extremes. Some adjustments were made after solving the optimization model, to make sure the solution remained inside, and not on the boundaries, of the SOS2 dataset.

All variables in the well model Constraints (3.2) and (3.4) and in the riser pipe pressure drop Constraints (3.8) are approximated with SOS2 sets containing 6 data points. Variables related to the gas lift Constraints (3.9) and (3.10) have 5 data points across the interval. According to the simulation results in Chapter (4), this choice of datapoint resolution seemed sufficiently accurate for most of the functions. Figure 5.7 gives an example of how the pressure drop across the gas lift pipe in Constraints (3.9) is linearized using SOS2.

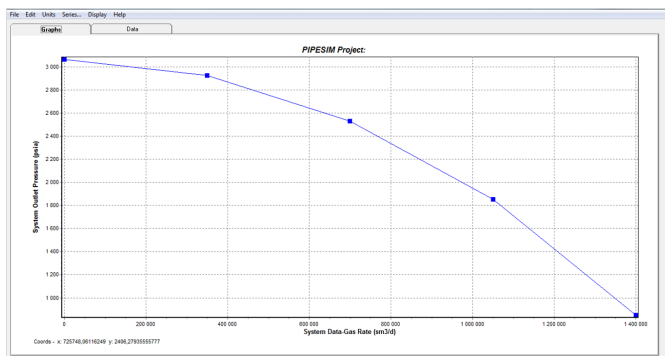


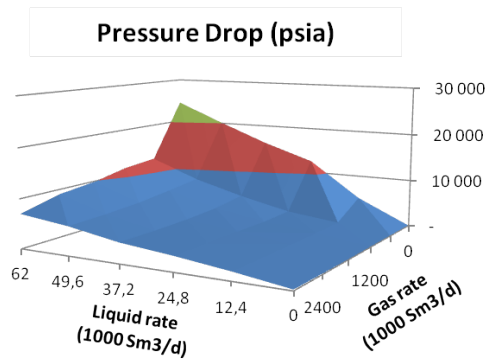
Figure 5.7: *SOS2 function approximation of manifold pressure in the gas lift pipe at subsea manifold 1, with 5 breakpoints*

The four-dimensional function describing riser pipe pressure drop (Constraints (3.8)) is the most challenging to linearize accurately. Hagem & Torgnes (2009) use as many as 30 data points for oil-, gas and water flowrates to approximate this function in their optimization model of the Statoil Troll West C field. However, they conclude that the large number of breakpoints represents a key challenge when attempting to solve the petroleum production problem using an MILP formulation.

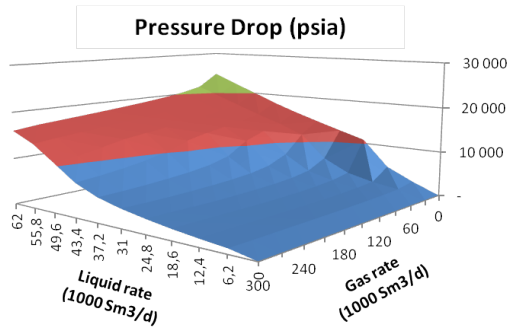
In this thesis, another instance of SOS2 sets is created for the pipeline pressure drop, where an additional datapoint is added between the existing 6, resulting in 11 data points in all three dimensions. A computational study is made where the optimization problem is solved with both 6 and 11 breakpoints to compare the impact on solution time and accuracy by increasing the resolution here. This particular function is chosen because we believe that better accuracy in the pipeline pressure drop could contribute the most to total model realism. The SOS2 sets

for the other functions remain unchanged in the two test cases. The results of the analysis are presented in Chapter 9.

Figures 5.8a and 5.8b show the riser pipe pressure drop as a function of gas- and liquid flowrates, approximated in each dimension with 6 and 11 data points, respectively. The total linearized and implemented model is shown in Appendix B.



(a) 6 breakpoints



(b) 11 breakpoints

Figure 5.8: SOS2 function approximation of the riser pipe pressure drop at manifold 1

Chapter 6

Dantzig-Wolfe decomposition

In this chapter, the optimization model introduced in Chapter 3, is decomposed using Dantzig-Wolfe decomposition (DWD). This method involves separating the problem into several smaller subproblems, and a masterproblem linking the subproblems together. In Section 6.1 relevant decomposition methods are presented and discussed. Section 6.2 describes the principles behind the DWD method. Several alternative ways of decomposing the optimization problem are evaluated in Section 6.3. Finally, Section 6.4 provides the mathematical formulations of the two decomposition methods applied to the optimization problem in this thesis.

6.1 Decomposition methods

Decomposition methods have been developed to solve large optimization problems that are too challenging to solve within reasonable time. Certain structural forms of these problems reappear frequently in various applications (Bradley et al. 1977). Problems with a structure consisting of many individual subproblems, only having a few common constraints, can easily be decomposed into smaller and easier individual problems (Elster 1993). The subproblems can be solved sequentially or in parallel. When decomposing, a few variables and constraints connect the masterproblem and the subproblems to ensure optimality for the problem as a whole.

Separation into smaller and independent subproblems, often has several important implications (Bradley et al. 1977). First, it can provide significant computational savings for LP problems, since problems with that structure are quite sensitive to the number of constraints. Second, each of the subproblems can be solved separately and only data related to the individual problems need to be analyzed and stored. The problems can be solved simultaneously, but since an iterative process requires the problems to be solved more than once, decomposition has no guarantee of reducing total computational time.

Due to the features mentioned above, decomposition is widely used when problems become too large for commercial software. The three most common decomposition methods are Lagrangian relaxation (LR), Dantzig-Wolfe decomposition (DWD) and Benders decomposition (BD). All three methods were considered in this thesis, but only DWD is applied to the problem. According to Gunnerud & Foss (2010), who also study an MILP production planning problem, BD is not a good choice because the method does not exploit the structure of the problem effectively. BD has only one master- and subproblem, and in addition, the method does not handle complicated and tough constraints in a good way. Further, Gunnerud & Foss (2010) conclude that DWD gives better performance and is more stable with respect to computational time than LR, making it the preferred decomposition method. As opposed to LR and BD, the DWD method has the advantage of being able to store, and make use of, solutions from each iteration, and in that sense it has a memory feature. This is useful when solving integer programs with decomposition, as will be explained in Chapter 7. The reader is referred to Beasley (1993) and Benders (1962) for more information about LR and BD.

6.2 Dantzig-Wolfe Decomposition

Dantzig-Wolfe Decomposition (DWD) was introduced by George B. Dantzig and Philip Wolfe in 1960 in Dantzig & Wolfe (1960). DWD has shown greatest potential when applied to LP problems that have a certain structure, as shown in Figure 6.1. This structure is, according to Bradley et al. (1977), named primal angular block structure. Problems with this structure consist of some common constraints linking several variables together. Other constraints ($A_1 - A_n$) apply to each variable individually, and can be handled separately in n individual subproblems. Each subproblem i includes an objective function and the constraint matrix A_i .

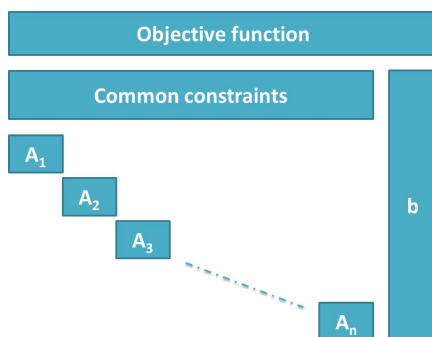


Figure 6.1: *Illustration of the primal angular block structure*

The common constraints are placed in a master problem (MP), together with the original objective function. The MP contains fewer constraints than the original

problem, but an increase in the number of variables may be balancing this advantage (Williams 2005). The MP is needed to coordinate the subproblems. Instead of solving with respect to the original variables, the variables in MP are formulated as a weighted sum of all feasible columns from the subproblems, which are assumed to be known in advance. The notation *column*, relates to all information from a corner solution in the convex subproblems (Lundgren et al. 2010). A column can for example include flow rates and pressures. The MP tries to find the most optimal weights, and a fundamental assumption is that a convex combination of feasible solutions is also feasible (Lundgren et al. 2010).

Finding all feasible columns in advance is often very time-consuming. The master problem quickly increases in size with one weight variable for each column. Quite often, many of the pre-generated columns are not needed as only a few weight variables are non-zero in the optimal solution. Therefore, a restricted master problem (RMP) is solved instead, initially including only a subset of all possible columns (Lasdon & Tabak 1981). By solving the RMP and the subproblems, more columns can be added iteratively if necessary (Lundgren et al. 2010).

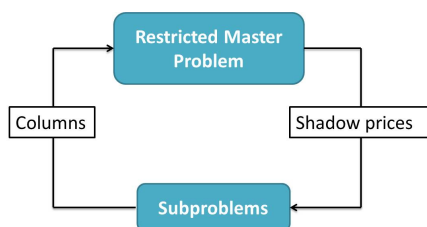


Figure 6.2: *Flow of information between RMP and the subproblems*

When a solution to the RMP is found, it is necessary to check if more candidate columns exist which could improve the objective value. This is done by sending updated shadow prices, or dual prices, associated with the RMP common constraints, to the subproblems each time RMP is solved. In the subproblems, dual prices appear as part of the expression for reduced cost in the objective function. Dual prices can be interpreted as costs of using the resources defined by the RMP common constraints, and are the only parameters coordinating the subproblems. Since no constraints are imposed on the common resources in the subproblems, the shadow prices are necessary to obtain a more correct use of them. Each subproblem is solved by maximizing the reduced cost, and new columns are sent to RMP until no new columns with positive reduced cost can be found (maximization problem) (Barnhart et al. 1998). For a minimization problem, the objective would be to find columns with negative reduced costs until no such column can be found. This terminates the column generation process, and an optimal solution to the original problem is found by solving the RMP one last time. The iterative process, named Column generation (Lasdon & Tabak 1981), is illustrated in Figure 6.2. The maximum number of iterations is the number of corner solutions in the subproblems. In practice, significantly fewer iterations are required (Lundgren et al. 2010).

The complete DWD algorithm for a maximization problem is illustrated in Figure 6.3.

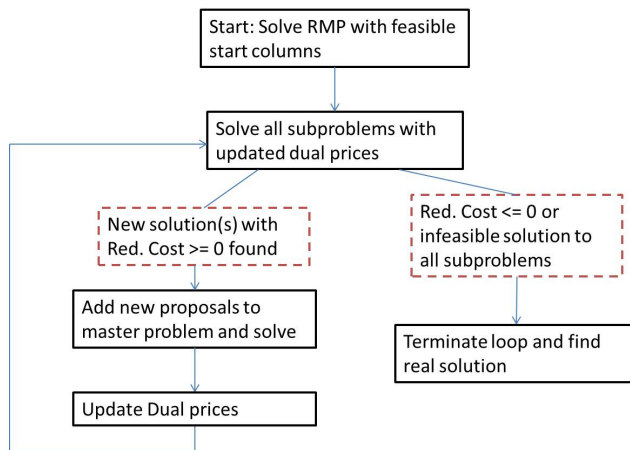


Figure 6.3: *Dantzig-Wolfe algorithm*

6.3 Different decomposition strategies

The problem described in this thesis has a great potential of being decomposed. The structure makes it easy to split the problem into smaller and simpler parts. When decomposing with the DW-method, an important decision is how to divide the model formulation between master- and subproblems. As mentioned earlier, the size of the masterproblems increases when one weight variable is added for each column. Huge masterproblems will therefore tend to have longer solution times. This is an argument supporting the choice of few common constraints, since it is important to be able to re-solve the master problem quickly. On the other hand, the subproblems are solved iteratively and should be well constructed to avoid computational problems (Colombani & Heipcke 2006). As a result, the distribution of constraints between the subproblems and the masterproblem should to some degree be balanced, but the main focus is on keeping the masterproblem relatively simple.

For the problem in this thesis, there are several possible decomposition strategies. One possibility is to decompose on separators. Using this strategy, the masterproblem would only contain the separator routing Constraints (3.19) and (3.20). All remaining constraints would be placed in the three subproblems, one for each separator. This decomposition strategy makes both the RMP and subproblems MILP, due to the presence of binary routing variables in both problems, and the SOS2 requirements in the subproblems. Although decomposing on separators might be an intuitive choice, the model complexity that follows from an MILP structure in

the RMP is a serious drawback, as it is not straight forward to obtain meaningful shadow prices.

The problem also has a potential of being decomposed on manifolds, resulting in three subproblems, as in the previous case. Gunnerud et al. (2010) suggest a similar decomposition approach in their petroleum production problem. With this strategy the master problem would contain separator capacity Constraints (3.13), in addition to the gas lift capacity Constraints (3.14). The subproblems would handle routing to separators and the remaining constraints. For this particular system, there would be two types of subproblems, one including the constraints related to the topside manifold, and the other handling constraints related to the subsea manifolds. With this strategy, the RMP is an LP problem, while the subproblems are MILP.

A similar way of splitting the system, is to decompose on subsea manifolds (as in the previous strategy), while a separate subproblem is defined for each well connected to the topside manifold. This is an extension of the decomposition method suggested by Gunnerud et al. (2010). A clear advantage of this strategy is less complex topside subproblems. As the previous decomposition method, the RMP would contain capacity constraints related to gas, water and gas lift, whereas the remaining constraints are placed in the subproblems.

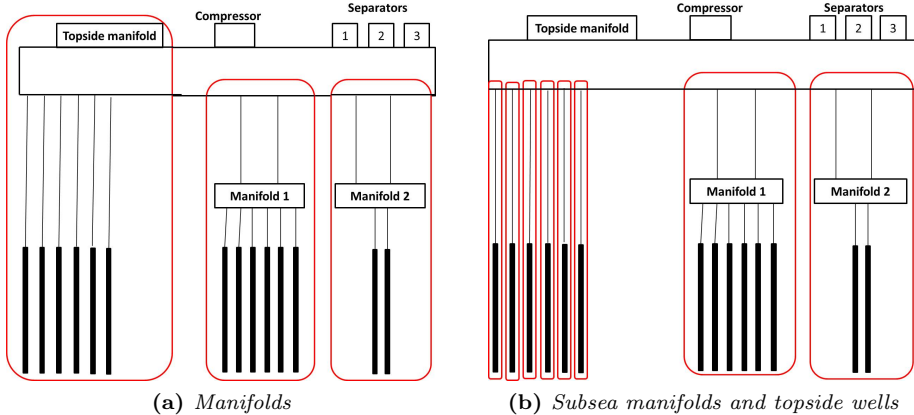


Figure 6.4: *Decomposition strategies*

As stated by Colombani & Heipcke (2006), it is often difficult to predict the performance of a decomposed model, and computational experiments for the particular application might be necessary. In this thesis, the latter two decomposition methods are applied to the optimization problem. Based on a total evaluation, these methods seem the most promising due to the appropriate distribution of constraints between subproblems and RMP. A major factor is the model simplicity that follows from an LP masterproblem. The two strategies are illustrated in Figure 6.4.

6.4 Decomposition using DWD

This section presents the decomposition strategies selected previously in this chapter. First, the optimization problem is decomposed on manifolds, followed by the decomposition on subsea manifolds and satellite wells.

6.4.1 DWD approach with MILP

As mentioned in Section 6.2, the DWD method assumes a bounded and convex optimization problem. In that case a solution can be expressed as a convex combination of the extreme points in the solution space, known as the convexification approach (Vanderbeck 2006). MILP problems, however, do not have a convex solution space. The MILP model in this thesis includes two aspects making the solution space non-convex, the binary routing variables and the SOS2 neighbour requirement. Routing makes the convex combination of columns from the subproblems challenging, because columns with different routing decisions cannot be weighted to give a feasible MILP solution. In addition, two columns interpolating between different SOS2 neighbour points cannot be combined to give a feasible solution to the MILP problem. In other words, the only way a convex combination of columns can result in a feasible MILP solution, is if the weighted columns have the same routing decisions *and* lie in the same SOS2 breakpoint interval across all the axes of the SOS2 formulations. This is highly unlikely to happen, and solving the RMP as an LP will in most cases generate infeasible MILP solutions.

To find a feasible MILP solution, the RMP is solved with binary requirements on the weights after column generation. This is referred to as RMP-IP, while solving the RMP with continuous weight variables is from now on referred to as RMP-LP. More advanced methods to generate feasible MILP solutions are discussed in Chapter 7.

6.4.2 Decomposing on manifolds

When decomposing on manifolds, each manifold is a separate subproblem. Since the topside manifold and the subsea manifolds have different structure, there is, as mentioned in Section 6.3, a need for two types of subproblems. The subproblems will consist of constraints specific to each subsea or topside manifold. The RMP handles constraints common to all manifolds, linking the whole system together. The solution process for this decomposition is described in Algorithm 6.1.

Algorithm 6.1: Decomposing on manifolds

forall the common constraints in RMP do
 └ set the dual values to zero
repeat
 └ send dual values to the subproblems
 forall the manifolds do
 └ solve manifold subproblems
 if column has positive reduced cost then
 └ send new column to RMP
 solve RMP-LP
until no new columns with positive reduced cost can be found
 solve RMP-IP

Common constraints

The constraints linking all manifolds together are the separator capacity Constraints (3.13), and the gas lift capacity Constraints (3.14), as given below:

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlps}^{LL} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjps}^{WW} \leq C_{sp} \quad s \in \mathcal{S}, p \in \{g, w\}$$

$$\sum_{m \in \mathcal{M}} \sum_{j \in \mathcal{J}_m} q_{mj}^I \leq G$$

These constraints represent the RMP common constraints, and must be reformulated using the convexification approach before being added to the RMP. The following reformulations are necessary:

$$\sum_{h \in \mathcal{H}^S} Q_{mlpsh}^{LL} \phi_{mh} = q_{mlps}^{LL} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \mathcal{P}, s \in \mathcal{S}$$

$$\sum_{h \in \mathcal{H}^T} Q_{mjpsh}^{WW} \omega_{mh} = q_{mjps}^{WW} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, p \in \mathcal{P}, s \in \mathcal{S}$$

$$\sum_{h \in \mathcal{H}^S} Q_{mjh}^I \phi_{mh} = q_{mj}^I \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m$$

$$\sum_{h \in \mathcal{H}^T} Q_{mjh}^I \omega_{mh} = q_{mj}^I \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m$$

The arrays Q_{mlpsh}^{LL} , Q_{mjps}^{WW} and Q_{mjh} , are values of q_{mlps}^{LL} , q_{mjps}^{WW} and q_{mj}^I in a corner solution h , of subproblem m . H is the set of all columns in the RMP. ϕ_{mh} and ω_{mh} are continuous weight variables for the subsea and topside subproblems, respectively.

Restricted Master Problem

The objective function in the RMP is a convex combination of oil production from all columns, indexed by h , sent from the subproblems.

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^S} Q_{mlpsh}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^T} Q_{mjps}^{WW} \omega_{mh} \quad (6.1)$$

Constraints (6.2) og (6.3) make sure that total capacities on the separators and the compressor are not exceeded.

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{h \in \mathcal{H}^S} Q_{mlpsh}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjps}^{WW} \omega_{mh} \leq C_{ps} \quad p \in \{g, w\}, s \in \mathcal{S} \quad (6.2)$$

$$\sum_{m \in \mathcal{M}^S} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^S} Q_{mjh}^I \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjh}^I \omega_{mh} \leq G \quad (6.3)$$

Due to the convex formulation and introduction of weight variables, additional convexity constraints are necessary. These are formulated as follows:

$$\sum_{h \in \mathcal{H}^S} \phi_{mh} \leq 1 \quad m \in \mathcal{M}^S \quad (6.4)$$

$$\sum_{h \in \mathcal{H}^T} \omega_{mh} \leq 1 \quad m \in \mathcal{M}^T \quad (6.5)$$

The convexity Constraints (6.4) and (6.5), are usually formulated with an equality sign. However, Barnhart et al. (1998) present a number of applications where an

inequality sign is used instead. In this particular model it is possible to formulate these constraints with inequality. The objective function prefers the weight variables as large as possible and will make sure equality is achieved if possible. Relaxing the constraints this way is beneficial when searching for a feasible initial solution to the RMP, as explained in Chapter 8. Finally, non-negativity is imposed on the weight variables.

$$\phi_{mh} \geq 0 \quad m \in \mathcal{M}^S, h \in \mathcal{H}^S \quad (6.6)$$

$$\omega_{mh} \geq 0 \quad m \in \mathcal{M}^T, h \in \mathcal{H}^T \quad (6.7)$$

Subproblems

As previously discussed, it is necessary to define two different types of subproblems, one for subsea- and one for topside manifolds. π_{sg}^G , π_{sg}^W and π^I are shadow prices associated with the RMP common constraints for separator capacity on gas and water, and gas lift capacity, respectively. The shadow prices, or dual prices, appear in the objective function of the subproblems.

A column generated in the subproblems is only sent to RMP if the reduced cost, \bar{c} , is positive, thereby having the potential to improve the RMP objective value. In accordance with the principle behind column generation, reduced costs define the objective functions of the subproblems and are maximized. The reduced costs for this optimization problem are shown in Equations (6.8) and (6.9).

$$\bar{c}_m^S = \sum_{l \in \mathcal{L}_m} (q_{mlo}^L - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mlgs}^{LL} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mlws}^{LL}) - \sum_{j \in \mathcal{J}_m} \pi^I q_{mj}^I - \pi^{C,S} \quad (6.8)$$

$$\bar{c}_m^T = \sum_{j \in \mathcal{J}_m} (q_{mjo}^W - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mjgs}^{WW} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mjws}^{WW} - \pi^I q_{mj}^I) - \pi^{C,T} \quad (6.9)$$

The last part of the above formulations, $\pi^{C,S}$ and $\pi^{C,T}$, are shadow prices associated with the convexity constraints. These are left out of the subproblem objective functions used in this thesis, because constants do not affect the optimization. The resulting subproblem objective functions for the subsea- and topside manifold subproblems, are respectively shown in Equations (6.10) and (6.11).

$$\max w_m^S = \sum_{l \in \mathcal{L}_m} (q_{mlo}^L - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mlgs}^{LL} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mlws}^{LL}) - \sum_{j \in \mathcal{J}_m} \pi^I q_{mj}^I \quad (6.10)$$

$$\max w_m^T = \sum_{j \in \mathcal{J}_m} (q_{mjo}^W - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mjgs}^{WW} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mjws}^{WW} - \pi^I q_{mj}^I) \quad (6.11)$$

All constraints in the original MILP model, valid for each individual subsea or topside manifold, are a part of the subproblem constraints. The total decomposition on manifolds can be found in Appendix C.1.

6.4.3 Decomposing on subsea manifolds and topside wells

Decomposing on subsea manifolds and topside wells, is quite similar to decomposing only on manifolds. The only difference is that there is one subproblem for each well connected to the topside manifold, instead of treating the topside manifold as a whole. It is still relevant to distinguish between two types of subproblems, one for subsea manifolds and one for the topside wells. This decomposition strategy is described in Algorithm 6.2. The common constraints are the same as in Section 6.4.2.

Algorithm 6.2: Decomposing on subsea manifolds and topside wells

```

forall the common constraints in RMP do
  | set the dual values to zero
repeat
  | send dual values to the subproblems
  | forall the subsea manifolds do
  |   | solve subsea manifold subproblem
  |   | if column has positive reduced cost then
  |   |   | send new column to RMP
  |   | forall the topside wells do
  |   |   | solve topside well subproblem
  |   |   | if column has positive reduced cost then
  |   |   |   | send new column to RMP
  |   | solve RMP-LP
  | until no new columns with positive reduced cost can be found
  | solve RMP-IP

```

The rest of this section presents the formulation of the decomposition method described here. Detailed explanations of the constraints and equations are left out, to avoid repeating what is discussed in the previous section. Differences between the two decomposition methods will be commented where this is relevant.

Restricted Master Problem

The RMP objective function is given below:

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^S} Q_{ml\text{osh}}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^T} Q_{mj\text{osh}}^{WW} \omega_{mjh}$$

Total oil production is, as previously, a convex combination of all columns, indexed by h , sent from the subproblems. The only difference compared to the formulation in Section 6.4.2 is in the definition of weights variables for the topside well subproblems (ω_{mjh}). These now have a well index, j , in addition to the manifold- and column index, m and h .

Constraints (6.12) - (6.17) define the RMP. As mentioned, the only difference compared to the previous decomposition method, is in the indexation of weight variable ω_{mjh} .

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{h \in \mathcal{H}^S} Q_{ml\text{psh}}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mj\text{psh}}^{WW} \omega_{mjh} \leq C_{ps} \quad p \in \{g, w\}, s \in \mathcal{S} \quad (6.12)$$

$$\sum_{m \in \mathcal{M}^S} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^S} Q_{mj\text{h}}^I \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mj\text{h}}^I \omega_{mjh} \leq G \quad (6.13)$$

$$\sum_{h \in \mathcal{H}^S} \phi_{mh} \leq 1 \quad m \in \mathcal{M}^S \quad (6.14)$$

$$\sum_{h \in \mathcal{H}^T} \omega_{mjh} \leq 1 \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (6.15)$$

$$\phi_{mh} \geq 0 \quad m \in \mathcal{M}, h \in \mathcal{H}^S \quad (6.16)$$

$$\omega_{mjh} \geq 0 \quad m \in \mathcal{M}, j \in \mathcal{J}_m, h \in \mathcal{H}^T \quad (6.17)$$

Subproblems

The objective values in the subproblems are formulated in Equations (6.18) and (6.19) for the subsea and topside well subproblems, respectively.

$$\max w_m^S = \sum_{l \in \mathcal{L}_m} (q_{mlo}^L - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mlgs}^{LL} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mlws}^{LL}) - \sum_{j \in \mathcal{J}_m} \pi^I q_{mj}^I \quad (6.18)$$

$$\max w_{mj}^T = q_{mjo}^W - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mjgs}^{WW} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mjws}^{WW} - \pi^I q_{mj}^I \quad (6.19)$$

A complete decomposed model is presented in Appendix C.2.

Chapter 7

Introducing heuristics

In the previous chapter, the MILP problem presented in Chapter 3 is decomposed using the Dantzig-Wolfe decomposition (DWD) method. This chapter extends to discussing the implications of using DWD on an MILP problem, as the convexity assumptions behind the DWD-method are not fulfilled for such problems. The sections of this chapter suggest different solution strategies to solve the MILP problem based on DWD. An exact solution method is briefly explained in Section 7.1, but not implemented in this thesis. The remainder of this chapter focuses on heuristics to solve the MILP problem.

7.1 DWD applied to MILP problems

As explained in Chapter 6, the DWD method assumes convex, bounded optimization problems. MILP problems are not in this category due to the integer properties and SOS2 requirements. This section introduces the concept of bounds when solving MILP problems, and briefly describes an exact algorithm for solving such problems.

7.1.1 Duality gap

Applying DWD directly to an MILP problem using the convexification approach described in Section 6.4.1, results in an optimistic bound to the MILP solution. Since the problem addressed in this thesis is a maximization problem, the optimistic bound corresponds to an upper bound (UB). The UB solution will most likely not be feasible in the original MILP problem, but it provides a good starting point in the search for feasible solutions. A simple heuristic approach ensuring feasibility is to enforce binary requirements on the weight variables in the restricted master

problem (RMP) after basic column generation, as described in Chapter 6. This results in a lower bound (LB) solution to the original MILP. The LB is integer feasible because it can only consist of whole (not fractional) columns generated from the subproblems, but it might not be the optimal solution to the MILP. The reason is that the number of feasible integer columns to choose from in the RMP is only a subset of all possible columns. Basic column generation only generates the subset of columns necessary to find the optimal solution to a convex, bounded optimization problem.

Figure 7.1 illustrates the so called *duality gap* (Beasley 1996), representing the distance between the UB and LB obtained by solving a DW decomposed MILP problem. In this thesis, two additional gap definitions are introduced. The distance between global optimum and the UB will be referred to as *Optimistic gap*, because all solutions in this region are optimistic, infeasible solutions to the MILP. The gap between the LB and global optimum is named *Pessimistic gap*, since integer feasible solutions, representing pessimistic bounds to global optimum, exist in this region.

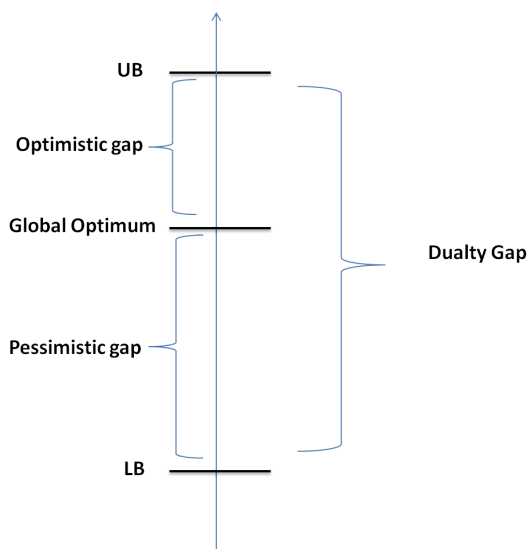


Figure 7.1: Gap between *UB* and *LB*

Obtaining information about bounds when solving MILP problems is very useful when evaluating the potential for improvement of an existing solution. As Figure 7.1 shows, the global optimum exists somewhere between the UB and LB. Reducing the gap can be done by either increasing the LB or decreasing the UB. Any solution above global optimum is infeasible, whereas any improvement in the LB up to global optimum is feasible.

7.1.2 Branch and Price

A well known optimization algorithm for solving large scale integer programs is called Branch & Price (B&P). The method was developed to solve integer problems with a huge number of variables (Barnhart et al. 1998). B&P combines ideas from the Branch & Bound (B&B) algorithm with basic column generation described in Chapters 5 and 6, respectively. A MILP is decomposed using a DWD formulation, including an RMP and several subproblems. After column generation, if the final RMP solution, solved as an LP, does not satisfy the integrality conditions of the original problem, branching is initiated. Branching is done by splitting the feasible region into smaller parts, i.e. adding more constraints to the optimization problem. There are several branching options. One strategy is to branch on the RMP weight variables by fixing some variables. A tree structure similar to that of the B&B algorithm, is formed during the process.

A consequence that follows from branching is that the RMP problem is changed. Re-optimizing the problem as an LP will most likely lead to new dual prices in the RMP constraints, making it possible to find more columns from the subproblems. In the B&P method, column generation is therefore repeated in each node of the search tree. An important note is that the B&P method is an exact optimization method, guaranteed to eventually find the optimal solution (Danna & Pape 2005), in the same way as B&B.

7.2 Heuristics

Heuristics are solution methods that generate feasible solutions to optimization problems without providing any quality guarantee of the solutions found. A heuristically obtained solution can be close to optimum, but it is not possible to say with certainty whether optimum is reached. Some heuristics try to prove distance to optimum, but these guarantees are generally poor (Lundgren et al. 2010). Knowledge of the application is therefore an advantage when developing efficient heuristics.

Heuristics are often used to solve difficult optimization problems within a limited amount of time. Quite often heuristics are preferred because they are more intuitive for "non-expert" users to understand and accept (Lundgren et al. 2010). These solution methods have been applied successfully within many fields of engineering and other sciences for more than two decades (Winker & Gilli 2004). Ronen (1982) and Martin et al. (1988) provide examples of large integer programs within shipping optimization solved with heuristics.

Heuristic based optimization methods can be divided into two main categories: *constructive methods* and *local search methods* (Winker & Gilli 2004). Constructive heuristics start with "nothing" and successively construct feasible solutions. A common type of such methods is called *greedy algorithms* (Lundgren et al. 2010).

Local search methods focus on iteratively improving already existing solutions. For combinatorial optimization problems, the search process is often expressed as a state space search because the algorithm moves from one solution to another. Central to this method is the definition of a *search neighbourhood*, N , defined by a subset of states (feasible solutions). A search strategy should consist of an evaluation criterion for each state, together with some rules for selecting the next move. The local search heuristic needs an initial solution, which can be either feasible or infeasible, depending on how the search neighbourhoods and rules are defined (Løkketangen 1995).

An obvious weakness with local search methods is that they tend to get stuck in local optima. So-called *metaheuristics* have been developed to improve the performance of local search heuristics by managing the search more efficiently. A local search method can for example be directed to different parts of the search region, thereby scanning a larger area (Lundgren et al. 2010). Examples of efficient metaheuristics for combinatorial problems are *tabu search* and *simulated annealing*. More can be read about these methods in Løkketangen (1995) and Kirkpatrick et al. (1983), respectively.

7.2.1 Motivation and Assumptions

As seen in Figure 7.1, there are two options for reducing the duality gap and moving closer to global optimum: either by reducing the UB or increasing the LB. From a practical point of view it makes sense to try to improve the current LB solution, as any LB improvement is feasible and can be implemented at the production asset, leading to immediate production gain. In contrast, reducing the UB does not generate feasible solutions. Information about UB is primarily helpful for guaranteeing global optimum or distance to it, as exact solution methods like B&B and B&P algorithms do. Heuristics are only able to find solutions that increase the LB. Only working with the LB does not guarantee an optimal solution, but it might be possible to find good solutions within reasonable time.

Gunnerud et al. (2011) implement a production rate allocation problem with B&P, applied to the Troll west field owned by Statoil. They are able to solve the problem with a duality gap of 0.01%, within 3.5 hours. It is however argued that there are situations in practical production planning where faster solution times are needed. Quick decision making is necessary for example in case of sudden production problems at different parts of the system.

As a solution method, B&P has the advantage of being able to guarantee global optimum or find solutions within a predefined gap specified by the user. However, the method can lead to high solution times in cases where the search tree becomes large, especially since column generation must be repeated in each node. Additionally, the B&P algorithm is considered complicated and time consuming to implement. If a simpler heuristic can generate good, feasible solutions fast, this might be more valuable in the short term than spending hours trying to find the

optimal solution. Considering all these factors concerning exact versus heuristic optimization methods, the choice is made to primarily focus on using heuristics in this thesis.

The remainder of this chapter describes different heuristics developed for solving the MILP problem presented Chapter 3 and 5. All heuristics are combined with a DWD formulation, and use the LP relaxed RMP solution as a basis when searching for good feasible MILP solutions. The idea is to find clever ways to generate additional columns from the subproblems, that are not added to the RMP during basic column generation. This way, better feasible integer solutions might be obtained after the RMP is re-optimized with binary requirements on the weight variables.

The DWD model from Section 6.4.2, decomposing on manifolds, is from now on referred to as DWD-Man, while the second DWD model, decomposing on manifolds and satellite wells (Section 6.4.3) is named DWD-ManWell. Recall from Chapter 6 that the solution to the RMP with continuous weight variables, is named RMP-LP, while a solution to the RMP with binary weight variables is referred to as RMP-IP in the rest of the thesis.

All heuristics presented in the following sections of this chapter are decomposed using DWD-ManWell. This is the preferred decomposition method because it appears to be more flexible and will most likely result in more column combinations from the satellite well system. The wells connected to the topside manifold are completely independent of each other and splitting them into single subproblems may lead to computational savings. Technically, the topside manifold connects the wells together, but from an optimization point of view it has no particular purpose, i.e. flow from the wells is not blended before it is directed to the separators.

7.2.2 Adding new columns with different routing

Each column generated from a subproblem constraints information about flow rates and routing decisions, among other things. While routing decisions are done in each subproblem, the separator capacity constraints are placed in the RMP. Changing the routing decision of a column would result in a new column, with identical flow rates, pressures values etc., as the original one. The routing decision is to a certain degree moved to the RMP. This might add more flexibility to the RMP-IP problem, as it allows for more options when distributing flow between the three separators to fulfill the common capacity constraints. The method will be named *NewSepRouting* in the rest of the thesis.

Since there are three separators, flow from each pipeline and satellite well has three routing options. This means that up to three columns can be added to the RMP for every column generated by basic column generation. More columns in the final RMP is beneficial when searching for good RMP-IP solutions.

Before a new column can be created with the method described here, certain pressure conditions must be fulfilled to guarantee that the new routing decision is

feasible. The pressure at the end of each pipeline must be greater than, or equal to the separator pressure before any flow can be directed to that particular separator. This condition is formulated in Constraints (3.6) and (3.7) of the optimization model found in Chapter 3, but since the problem is not re-solved, a manual check is made instead.

Algorithm 7.1: NewSepRouting

```

forall the common constraints in RMP do
  | set the dual values to zero
repeat
  | send dual values to the subproblems
  | forall the subsea manifolds do
  |   | solve subsea manifold subproblem
  |   | if column has positive reduced cost then
  |   |   | send new column to RMP
  |   | forall the topside wells do
  |   |   | solve topside well subproblem
  |   |   | if column has positive reduced cost then
  |   |   |   | send new column to RMP
  |   | solve RMP
  | until no new columns with positive reduced cost can be found

  | forall the columns do
  |   | create new feasible columns with same values, but different routing to
  |   | separators
  |   | send new columns to RMP

  solve RMP-IP
  
```

New routing columns are added right after column generation is terminated. This way it is possible to guarantee that the new RMP-IP solution must be at least as good as the RMP-IP solution before adding new columns. The problem will still contain the same columns as before, in addition to the new ones that could potentially improve the IP solution. Algorithm 7.1 describes the method in pseudo code.

7.2.3 Depth first

The term *depth first* is usually associated with a branching strategy in the B&B algorithm (Lundgren et al. 2010). This section describes a proposed heuristic with the same name, DepthFirst, since it is inspired by the branching rule but must not be mistaken for it. A search tree, similar to a B&B tree with nodes and branches, can be used to describe the heuristic. This will be explained later in the section.

After solving the final RMP-LP in the first node, called the root node, some weight variables will most likely be fractional. The principle behind the DepthFirst heuristic is to fix the weight variable with the highest fractional value to 1. By doing this, the column associated with that particular weight variable is included in the solution when the RMP-LP problem is re-solved. A feasibility check must be made, to make sure that the common capacity constraints in the RMP are not violated when fixing a column. If infeasible, the heuristic searches for the second highest fractional weight, and so on.

Fixing a weight variable changes the RMP-LP problem and re-optimizing might lead to new dual prices. Therefore, it becomes necessary to repeat the column generation process, as more columns with positive reduced cost might exist, that can be added to the RMP. After new columns are added, the constraints fixing the weight variables are removed before RMP-IP is solved. Note that fixing weight variables in between each column generation process is only done to generate more columns in the RMP. The fixed columns might not be used at all in the final RMP-IP solution.

Termination criteria

The process of fixing fractional weight variables to 1 after each column generation continues until fixing additional columns makes the RMP infeasible, or until a predefined gap is achieved between the RMP-LP root node solution (UB) and the best RMP-IP solution. Gap termination criteria are commonly used in the B&P algorithm, where the user might have a solution quality requirement. Note that the heuristic suggested here does not guarantee to fulfill any predefined gap criteria. Due to the limited tree search performed by the DepthFirst heuristic, it is less likely that the method will produce solutions satisfying a low gap criterion. The heuristic is expected to terminate due to infeasibility in most cases. Algorithm 7.2 summarizes the DepthFirst heuristic in a pseudo-code.

It is clear that the DepthFirst heuristic suggested here, can be compared with the B&P algorithm described in Section 7.1.2. Elements from B&P are included, such as branching and repeated column generation in each node, as the tree illustration in Figure 7.2 shows. Column generation is initiated in each node of the search tree, and the RMP is solved both as LP and IP. Branching is associated with the process of fixing weight variables to 1.

An important difference between the heuristic and B&P is that DepthFirst does not continue branching on the second branch going out from a node, as indicated with

Algorithm 7.2: DepthFirst

```

forall the common constraints in RMP do
  | set the dual values to zero

repeat
  | repeat
  | | send dual values to the subproblems
  | | forall the subsea manifolds do
  | | | solve subsea manifold subproblem
  | | | if column has positive reduced cost then
  | | | | send new column to RMP
  | | |
  | | | forall the topside wells do
  | | | | solve topside well subproblem
  | | | | if column has positive reduced cost then
  | | | | | send new column to RMP
  | | |
  | | solve RMP
  | until no new columns with positive reduced cost can be found

  solve RMP-IP

  if gap between the first LP and last IP is acceptable then
  | gap is ok
  else
  | fix the highest feasible, fractional weight variable to 1
  | solve RMP

until gap is ok or no feasible fixing of weights

```

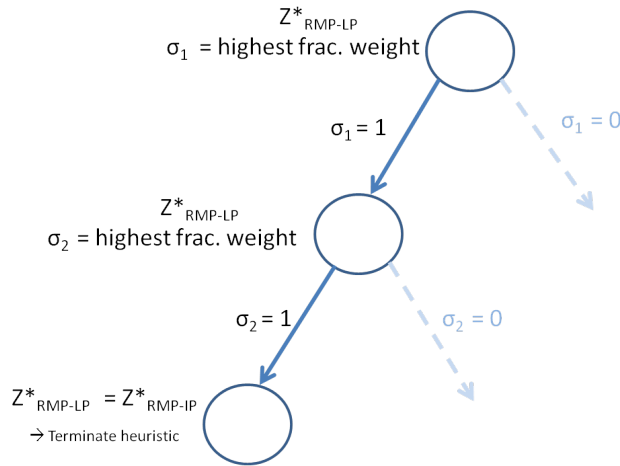


Figure 7.2: Search tree for the DepthFirst heuristic

pale stippled lines in Figure 7.2. It only searches in the depth of one branch until it terminates according to the rules explained earlier. This way, it is impossible to guarantee an optimal solution. B&P, on the other hand, would continue branching and make use of upper and lower bounds to prune nodes and guarantee optimum.

7.2.4 Reducing the slack based on RMP-LP

The RMP-LP solution interpolates between the columns present in the final RMP to make optimal use of the available capacity, so that total oil production is maximized. When solving RMP-IP, the capacity will most likely not be optimally utilized, and the slack is expected to be higher. The idea behind the heuristic described here, is to try to replicate the RMP-LP solution by constructing an RMP-IP solution with the same slack in the common constraints. Better use of the available resources might lead to more production. This method will be named *SlackBoundsLP* in the rest of the thesis. The method is best described with an example.

Figure 7.3 gives a simple example of an RMP-LP solution where columns from 3 different subproblems are interpolated to satisfy the common constraint. Interpolation results in a right hand side value of $14 + 18 + 12 = 44$, which gives a slack of 1 in the optimal LP solution. Note that the above constraint is not binding in optimum. This is a conscious choice to illustrate important features of the *SlackBoundsLP* heuristic.

If the problem is solved with binary weight requirements, the optimal RMP-LP solution is not possible. The RMP-IP problem is forced to choose at most one column from each subproblem, resulting in a higher slack.



Subproblem:	1	2	3		
Weights:	0.4	0.6	0.8	0.2	1
Columns:	20 + 10	15 + 30	+ 12		≤ 45
					
	14				
					
	18				

Figure 7.3: *SlackBoundsLP* heuristic example

From now on it is assumed that the example represents a separator capacity constraint for water, where the numbers in the columns are water flow rates. The objective is to maximize oil production. The *SlackBoundsLP* heuristic is divided into two steps:

Method 1 The principle behind the first method is to try to generate new columns from the subproblems with water flow rates equal to the interpolated sum of flow rates from the RMP-LP solution, thereby allowing RMP-IP to find the LP solution without the need to interpolate. Taking subproblem 1 as an example, a new column with an exact water flow rate of 14 is desired so that this rate is achieved without having to interpolate between the two existing columns.

Method 2 The first method would have been sufficient if the new generated column had resulted in the same oil production as achieved by interpolating between the two existing columns. However, the new column might give a slightly lower oil flow rate, otherwise it would have been found during the basic column generation. Method 2 is therefore included in an attempt to increase production even further, by generating additional columns from the subproblems that make use of the RMP-LP slack. From the example in Figure 7.3, a new column with water flow rate of $14 + \text{slack} = 15$ is wanted from subproblem 1 to increase oil production as much as possible.

Algorithm 7.3: SlackBoundsLP

```

forall the common constraints in RMP do
  | set the dual values to zero
repeat
  | repeat
  | | send dual values to the subproblems
  | | forall the subsea manifolds do
  | | | solve subsea manifold subproblem
  | | | if column has positive reduced cost then
  | | | | send new column to RMP
  | | | forall the topside wells do
  | | | | solve topside well subproblem
  | | | | if column has positive reduced cost then
  | | | | | send new column to RMP
  | | | solve RMP
  | | until no new columns with positive reduced cost can be found
  | | compute new bounds with Method 1
  | | send new bounds to subproblems and change the subproblem objective
  | until added new column from subproblems with new bounds
  solve RMP

  compute new bounds with Method 2
  send new bounds to subproblems and change the subproblem objective

  forall the subsea manifolds do
  | solve subsea manifold subproblem
  | if column has positive reduced cost then
  | | send new column to RMP

  forall the topside wells do
  | solve topside well subproblem
  | if column has positive reduced cost then
  | | send new column to RMP

  solve RMP-IP

```

Generating new columns

Having described the main idea behind the heuristic, it remains to explain how the new column is generated. The interpolated water flow rate of 14 from Method 1 in the example above, is sent to subproblem 1 as an upper bound on the water flow, adding a \leq constraint to the subproblem. The subproblem is then re-solved disregarding dual prices and only maximizing oil rates. The purpose of including dual prices is to find the RMP-LP optimal flow rates. In this case, however, the RMP-LP optimal flow rates are known in advance, making dual prices unnecessary. Assuming more water leads to more oil, the subproblem will find a new solution (column) where the water flow rate equals its maximum possible value, i.e. 14, if not restricted by other constraints. The same procedure is used for generating columns in Method 2, the only difference being the value of the upper bounds added to the subproblems.

Each subproblem is re-optimized as described in the previous paragraph, with an objective function disregarding dual prices. Only the subproblems that have weighted columns in the RMP-LP solution get bounds on the flow variables. The remaining subproblems are re-optimized without defining new constraints, for the purpose of generating as many new columns as possible, increasing the chances of finding good RMP-IP solutions. Algorithm 7.3, gives a detailed description of each step in the heuristic.

Challenges related to routing

The SlackBoundsLP method seems promising, but there are some challenges making it difficult to predict whether the method will work as intended. The fact that there are three separator constraints in the RMP for both gas- and water flow, complicates the example in Figure 7.3.

The subsea subproblems generate columns including information about flow rates and routing decisions for two riser pipes. Both of these can be routed to any of the three separators. This means that one column can send flow to two different separators. There might be scenarios where two columns with different pipeline routing are interpolated in the RMP-LP solution, giving rise to the question of how the pipelines in the new column should be routed. This is handled by imposing a bound on the sum of flow from both pipelines to one separator, to make sure the desired flow is achieved without directly placing any restriction on the routing decision. A bound is generated for each separator and flow phase. There might be scenarios where it is impossible for one column to reproduce the RMP-LP solution, for example if the LP solution suggests that one subproblem should send flow to all three separators. This is not possible to achieve from only one column, making it difficult to predict how the heuristic will perform.

7.2.5 Reducing the slack based on RMP-IP

The final heuristic described in this section, is to some degree related to SlackBoundsLP. Constraints imposing an upper bound on flow variables are added to the subproblems, and they are re-solved disregarding dual prices in the objective function. Gunnerud (2011) also applied this heuristic when solving an oil production planning problem.

The heuristic introduced here will be named *SlackBoundsIP* in the rest of the thesis. The main difference between SlackBoundsIP and SlackBoundsLP is in the way bounds are calculated. Similar to the example in Figure 7.3, the next example represents a common constraint in an RMP where columns from the different subproblems are included.

Subproblem:	1	2	3		
Weights:	0	1	0	1	0
Columns:	40 + 50	80 + 35	+ 12	\leq	100

Figure 7.4: *SlackBoundsIP* heuristic example

Figure 7.4 represents the solution to an RMP-IP problem. At most one column per subproblem can be included in the IP solution, as interpolation between columns is forbidden. This will most likely result in more slack in the RMP common constraints, compared to solving the RMP-LP problem. Available resources are left unused since there are not enough good columns in the RMP to combine when solving the IP problem.

The example in Figure 7.4 will from now on represent another separator capacity constraint, and the numbers are water flow rates routed to that separator. The left hand side adds up to $50 + 35 = 85$, resulting in a slack of 15. The principle behind SlackBoundsIP is to try to find a new column, with a higher water flow rate, including the slack value from the RMP-IP constraint. In that case, the new column would eliminate the slack, resulting in better capacity use and more oil production. Considering, for instance, subproblem 1 in the example above, the column with a flow rate of 50 is currently used. If the subproblem could generate a new feasible column with a water flow rate of $50 + \text{slack} = 65$, the new column is expected to be selected in the optimal RMP-IP solution instead, thereby eliminating the slack. An additional constraint placing an upper bound of 65 on the water flow variable is included in subproblem 1, and the subproblem is re-solved by maximizing oil production without considering dual prices in the same way as SlackBoundsLP.

New constraints are added to all subproblems where a column is used in the RMP-IP solution, as summarized in the pseudo code in Algorithm 7.4. In the example above, new bounds are added for subproblem 1 and 2, whereas subproblem 3 is

merely re-optimized without dual variables in the objective function and additional constraints on the flow variables.

Algorithm 7.4: SlackBoundsIP

```

forall the common constraints in RMP do
  | set the dual values to zero
repeat
  | repeat
  | | send dual values to the subproblems
  | | forall the subsea manifolds do
  | | | solve subsea manifold subproblem
  | | | if column has positive reduced cost then
  | | | | send new column to RMP
  | | | forall the topside wells do
  | | | | solve topside well subproblem
  | | | | if column has positive reduced cost then
  | | | | | send new column to RMP
  | | solve RMP
  | until no new columns with positive reduced cost can be found

  solve RMP-IP

  compute new bounds from slack in the IP solution
  send new bounds to subproblems and change the subproblem objective
until added new column from subproblems with new bounds

```

The challenge related to routing decisions mentioned for SlackBoundsLP, is not as problematic for SlackBoundsIP. In the latter, if a column flow is routed to a particular separator in the RMP-IP solution, the heuristic tries to generate a new column with the same routing as the existing column. This means that some additional constraints are added to the subproblems, where routing decisions are fixed. Note that the main difference between SlackBoundsLP and SlackBoundsIP is that the former is based on the RMP-LP solution, while the latter uses the RMP-IP solution as a starting point.

7.2.6 Obtaining N solutions from the subproblems

A final option considered in this thesis for reducing the duality gap and solution time, is to generate multiple columns each time a subproblem is solved. Tebbboth (2001) can document successful results using this method, and the implementation is not expected to require much extra computational time since the additional integer solutions must only be stored and transferred between files.

Each time a subproblem is solved, the solution with the maximum reduced cost is added to the RMP, but all other feasible solutions with a positive reduced cost have the potential to improve the RMP-LP solution. Therefore it might make sense to add a number of these columns at once with the aim of improving computational efficiency. However, the method might end up with a different set of columns in the final RMP, because the dual prices in the column generation process are affected. For this reason, there is no guarantee that this method will give an improved RMP-IP solution.

In this thesis, the approach described here is not treated as one of the main heuristics. The method will be included to the most promising heuristic to see if improved solutions can be found. The results of this analysis are found in Chapter 9.

Chapter 8

Implementation

Implementation of an optimization model requires collection of input data, formulating a mathematical model and using appropriate optimization software to solve the problem. This chapter covers the implementation strategies used for solving the model presented in Chapter 3. All strategies involve the use of the Mosel modelling language with the Xpress Optimizer. Generation of input data is mainly done in the PIPESIM simulator, as described in Chapter 4.

In Section 8.1, the software and hardware applied in this thesis are discussed. The pre-processing of data is presented in Section 8.2. Implementation of the original model and the DWD model is described in Section 8.3 and Section 8.4, respectively. A presentation of data sets is given in Section 8.5, while removal of symmetry in the model is described in Section 8.6. Complete Mosel codes for the original model and the DWD strategies can be found in the electronic Appendix D.

8.1 Software and hardware

The mathematical model presented in Chapter 3 was implemented in the Mosel modelling language and solved with Xpress-MP v7.3.0. Xpress-MP is a software solving linear, continuous, quadratic, integer, and mixed-integer programs (FICO 2009*b*). The solver applies the Simplex and B&B algorithms to solve optimization problems to optimality. These algorithms are embedded into the Xpress-MP software and do not have to be programmed manually (Ashford 2007). The Mosel language is a high-level, procedural programming and modelling language, allowing formulations close to the algebraic notation (Columbani & Heipcke 2008). The language was designed to be easy-to-learn and to interact robustly with the Xpress solver engines.

Xpress-MP is mainly a linear optimization software, which implies that it can only handle linear expressions (FICO 2009b). In this thesis, the solution approach is to linearize the initial MINLP model, as explained in Chapter 5, and solve it with Xpress-MP. A SOS2 formulation is used in the linearization process. According to Ashford (2007) variables declared as SOS2 are treated efficiently by Xpress-MP. When declaring variables to be SOS2, Xpress-MP recognizes the SOS2 structure during the presolve procedure (Ashford 2007). Xpress-MP has been used in several optimization courses at the department of Industrial Economics at the Norwegian University of Science and Technology (NTNU). Glæserud & Syrdalen (2009) and Hagem & Torgnes (2009) conclude in their master theses that Xpress-MP is well suited for MILP problems. It is therefore assumed that Xpress-MP is sufficient for solving the optimization problem studied in this thesis.

All the models are solved on a node of a remote cluster environment, named *Solstorm*. This node is a Hewlett Packard dl 165 G5 PC, with 24GB RAM memory and 2x Intel AMD Opteron 2431, 2,4 GHz central processor units (NTNU 2012).

8.2 Data pre-processing

The input data for the optimization problem is stored into .txt and Microsoft Excel¹ files. Xpress-MP can easily read data from .txt files, whereas reading from Excel files requires the inclusion of additional Mosel *modules*. A module includes its own sets of procedures and functions that directly extend the vocabulary and capabilities of the Mosel language. Modules provide access to Xpress optimizer solvers for LP, MIP and QP problems, in addition to data handling facilities (FICO 2009a). Necessary solver modules must be included, because Mosel does not integrate any solver by default (Columbani & Heipcke 2008). The *mmodbc* Mosel module provides a software specific interface for Excel and allows the user to access databases and spreadsheets (Columbani & Heipcke 2008).

As mentioned in Section 8.1, all models were run on a remote cluster of computers through the Linux Operating System (OS). This creates problems when reading data from Excel, as the use of Microsoft software with Linux OS is not trivial. Therefore, a separate Mosel code was written to generate .dat files from Excel files, as .dat files are compatible with Linux. The code reads data from Excel through the *mmodbc* module and writes the same data to .dat files, later used as input to the optimization model. The data transfer code must be run on a Microsoft Operating System in the pre-processing stage, before the optimization problem can be solved.

¹Excel 2010 (version 14)

8.3 Implementation of the original model

The term original model is a reference to the non-decomposed MILP model presented in Chapter 3 with the linearizations made in Chapter 5. Implementation with Xpress Mosel includes reading of information from .dat files, containing production data and simulated values from PIPESIM. In addition, defining sets, parameters, variables and constraints, and solving the problem with respect to an objective function is a part of the implementation process. In this case the objective function, representing oil production, is maximized. The original model is solved using the Simplex- and B&B-methods embedded into the Xpress-MP optimizer. Xpress-MP first solves an LP-relaxed version of the MILP problem with Simplex. If the solution to the relaxed problem does not satisfy the integer requirements, B&B is invoked to try to find an integer feasible solution. When the optimal integer solution is found, if any, the results containing objective value, computational time and production data are sent to .dat files. An overview of the implementation process is shown in Figure 8.1.

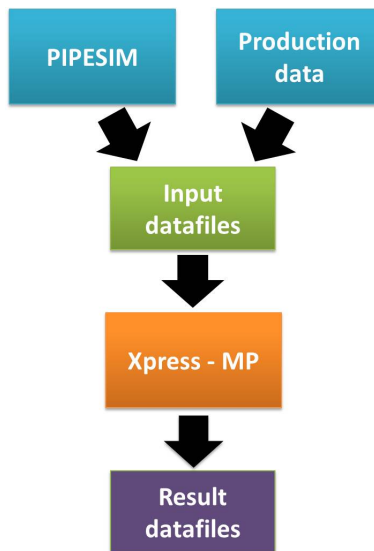


Figure 8.1: *Implementation of the original model*

When implementing the original model in Xpress-MP with Mosel, the three modules *mmodbc*, *mmxprs* and *mmsystem*, must be included. The first module, *mmodbc*, allows communication with Excel, as explained in Section 8.2. *mmxprs* gives access to the Xpress Optimizer for solving LP, MIP and QP problems, while *mmsystem* provides a set of useful procedures and functions related to the operating system (FICO 2012). The *mmxprs* and *mmsystem* modules provide the most standard functionalities, and are included in all models in this thesis.

Sometimes the solution history of the B&B tree is interesting to study, as in cases where no optimal solution is found. The solution status at each B&B node of the search tree is therefore logged and written to a result file. This way, the user can keep an overview of the solutions and bounds found in the solution process.

8.4 Implementation of DWD

When solving the optimization models decomposed with DWD, it is necessary to run several models that can communicate with each other. For the MILP problem in this thesis, three Mosel models are necessary; one model for the RMP, another for the subsea manifold subproblems and the third for the topside manifold subproblems. These models need to communicate in the way that shadow prices are sent from the RMP to the subproblems, and columns with solution information are sent from each of the subproblems to the RMP. The implemented RMP should also be able to load and run all the subproblems. To make this possible, the *mmjobs* Mosel module is added, in addition to the modules already mentioned in Section 8.3. The *mmjobs* module implements facilities for handling multiple models, including mechanisms for data exchange in memory. This module makes it possible to load several models in memory and execute them concurrently (Columbani & Heipcke 2008).

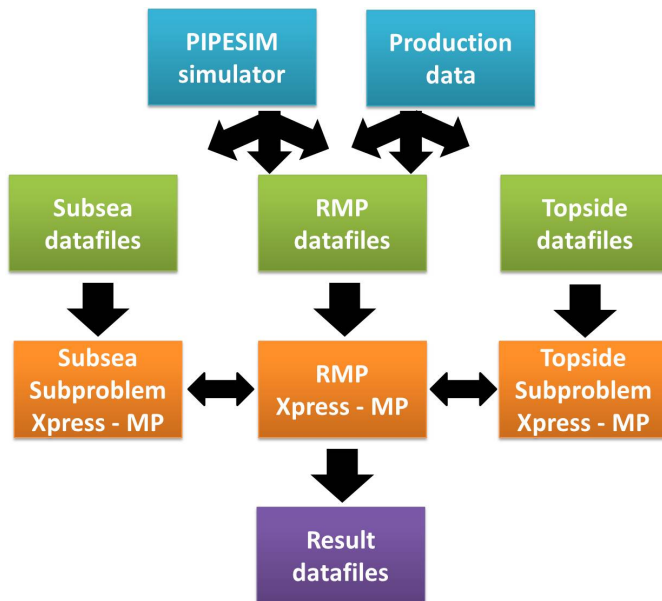


Figure 8.2: Implementation of DWD

Due to the decomposition of the original MILP model into multiple smaller models, the input data must be restructured and correctly distributed between master- and submodels. Separate data files are necessary for each of the three models mentioned above. The RMP model controls the running process and writes the final results of the optimization to a .dat file. The implementation structure and the interaction between the files are illustrated in Figure 8.2.

8.4.1 Finding an initial solution

The DW solution procedure normally requires an initial set of columns in the RMP, to form an initial feasible RMP-LP solution before column generation can start (as seen in Figure 6.3 in Chapter 6). In this thesis, column generation is started without first solving the RMP. Instead, the subproblems are solved with initial dual prices of zero. To guarantee that a feasible solution is found the first time RMP is solved, = is replaced with \leq in the convexity constraints of the RMP. This can be found in Sections 6.4.2 and 6.4.3.

At most one column is generated from each subproblem in the first iteration of the column generation process. Using equality signs in the convexity constraints, would demand that all columns are included in the first RMP solution. This might not be feasible according to the RMP common constraints. By relaxing the convexity constraints as mentioned, fractional columns are allowed. This will always produce a feasible RMP solution, as zero production is also a possible solution. The nature of the RMP maximization function will give solutions where the convexity constraints are binding whenever possible.

8.4.2 Column generation

As described in Section 6.2, column generation is used to generate new columns to the RMP. This is done by solving the subproblems to optimum and sending the solutions to RMP. As explained in Chapter 7, more columns in the column "pool" of the RMP may improve the IP solution of the problem.

The column generation process can be done in two different ways, sequentially or in parallel. When using the first approach, the subproblems are solved one at a time. While the first subproblem is being solved, the RMP is paused on a so called *wait event*. When including the *mmjobs* module in Xpress-MP, an event queue is attached to each submodel to collect all events sent to RMP. This queue is managed with a first in- first out policy (FICO 2012). Therefore, when RMP is paused on wait, it should not do anything until it receives a new event. When the subproblem that was started terminates, a *send event* is used to continue the RMP. Information about the column from the subproblem is then stored and the process is repeated for the next submodel. This sequential solving process is illustrated in Figure 8.3.

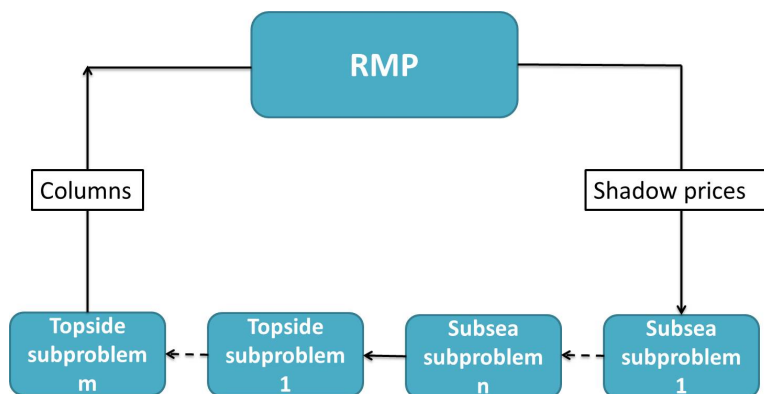


Figure 8.3: *Sequential solving of the subproblems*

When applying the parallel method, all subproblems are solved simultaneously. The RMP waits for the slowest subproblem to finish, before it is solved and the next iteration is started. With parallel solving, the solution time in every iteration should therefore theoretically be equal to the solution time of the slowest subproblem. In practice, software challenges related to memory and data transfer between files might additionally increase computational time. With sequential running of the subproblems, the solution time should equal the total time of running all the subproblems. In this thesis, sequential running of the subproblems is applied in the column generation process.

8.4.3 Termination criteria

The column generation is terminated if no subproblems are able to find a solution with positive reduced cost. In the implementation of the model, a tolerance limit at slightly above zero is used to avoid entering a loop that does not terminate due to data noise. There is also a possibility to terminate if the number of iterations exceeds a predefined number. Based on computational experiments, the models terminated relatively fast without any need for a limit on the number of iterations. The DWD heuristics could additionally be terminated using a gap criterion. If the duality gap is below a predefined limit, the process could be terminated. This criterion is implemented for the DepthFirst heuristic, as explained in Section 7.2.3. For the remaining heuristics, termination on duality gap is not necessary, due to the structure and size of the algorithms.

8.5 System data

For the purpose of analyzing the optimization methods developed in this thesis, three case instances based on the production asset from Chapter 1 are constructed. The first case represents the production system today, whereas the two other cases are different expansion opportunities. According to production engineers at the Marlim field, a natural expansion of the asset is to install more subsea manifolds and wells connected to these manifolds. Adding more topside manifolds is not of current interest (Private conversation January 3, 2012). Six wells are connected to subsea manifold 1 at the current production asset, which is considered an appropriate number. It is therefore only relevant to expand the number of wells on subsea manifold 2, which today has only two wells.

The production data presented in this section is based on real data at Marlim. For the expanded, fictive cases, separator- and lift gas capacities are increased approximately in proportion to the size of the system.

8.5.1 Case 1

Case 1 represents the real production asset. The system consists of two subsea manifolds and one topside manifold. Number of wells and pipelines connected to each of the manifolds is shown in Table 8.1. The separator and compressor capacities are shown in Table 8.2.

Table 8.1: *Wells and pipelines in Case 1*

	Subsea manifold		Topside manifold
	1	2	
Wells	7	2	6
Pipelines	2	2	-

Table 8.2: *Separator and compressor capacities in Case 1*

	Water	Gas
	[sm ³ /d]	[sm ³ /d]
Separator 1	1600000	12000
Separator 2	1600000	8000
Separator 3	1200000	4000
Compressor	-	850000

8.5.2 Case 2

The second case consists of four subsea manifolds and one topside manifold. As shown in Table 8.3, there has been an increase in the number of wells connected to subsea manifold 2. The number of wells and pipelines connected to the additional subsea manifolds, 3 and 4, is the same as for subsea manifolds 1 and 2. To avoid symmetrical solutions, the extra added manifolds and wells are assigned slightly different physical properties, such as pipe thickness and well depth. The separator and compressor capacities are presented in Table 8.4.

Table 8.3: *Wells and pipelines in Case 2*

	Subsea manifold				Topside manifold
	1	2	3	4	
Wells	7	4	7	4	6
Pipelines	2	2	2	2	-

Table 8.4: *Separator and compressor capacities in Case 2*

	Water	Gas
	[sm ³ /d]	[sm ³ /d]
Separator 1	3200000	22000
Separator 2	3200000	15000
Separator 3	2400000	8000
Compressor	-	1700000

8.5.3 Case 3

Case 3 consists of six subsea manifolds and one topside manifold. The number of wells and pipelines connected to each of the manifolds is shown in Table 8.5. The increase in wells and manifolds follows the same pattern as in Case 2. Separator and compressor capacities are shown in Table 8.6.

Table 8.5: *Wells and pipelines in Case 3*

	Subsea manifold						Topside manifold
	1	2	3	4	5	6	
Wells	7	5	7	5	7	5	6
Pipelines	2	2	2	2	2	2	-

Table 8.6: Separator and compressor capacities in Case 3

	Water	Gas
	[sm ³ /d]	[sm ³ /d]
Separator 1	4800000	36000
Separator 2	4800000	24000
Separator 3	3600000	12000
Compressor	-	2500000

8.6 Removal of symmetry

There is symmetry in the solution space of the optimization model in Chapter 3, because it is possible to find several equal model solutions for each real solution. This increases the complexity of the problem because the optimizer will most likely consume time and computer memory searching through a large number of solutions that are identical for the practical case. As mentioned in Baricelli et al. (1998), removing some of these solutions in the preprocessing could therefore be of great computational advantage.

In Constraints (3.18), symmetry related to routing of wells to pipelines at the subsea manifolds is found. Each well can be routed to at most one of two pipelines, and a binary variable is defined for each routing option, y_{mjl} . m , j and l are indices for manifold, well and pipeline, respectively:

$$\sum_{l \in \mathcal{L}} y_{mjl} \leq 1 \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m$$

As both pipelines connected to a manifold have the exact same physical properties, using either of the two is practically indifferent. To clarify the difference between model solutions and real solutions, an example is given below.

As seen in the illustration of the production asset in Figure 1.6 in the introductory chapter, subsea manifold 2 has two wells and two pipelines connected to it. Considering a case where only well 1 is producing, two identical model solutions for that scenario exist:

- Routing well 1 to pipeline 1
- Routing well 1 to pipeline 2

Since the two pipelines are identical, the above routing decisions are equal to the production engineer. The only real solutions for subsea manifold 2 are:

- No wells producing - No routing
- One well producing - Routing to one pipeline

- Two wells producing - Both routed to the same pipeline
- Two wells producing - Both routed to different pipelines

A simple preprocessing of the model is done to remove equivalent solutions. We choose one well for each subsea manifold and fix one of the corresponding routing variables to 0. For example, if it is optimal for well 1 to be producing, it must be routed to a specific pipeline. But no real solution is removed because the well still has the option of being closed (no routing) or open, which is the essential decision. To remove this symmetry, Constraints (8.1) and (8.2) are added to the model.

$$y_{m21} = 0 \qquad m \in \{1, 3, 5\} \qquad (8.1)$$

$$y_{m12} = 0 \qquad m \in \{2, 4, 6\} \qquad (8.2)$$

In the expanded cases, subsea manifolds with even numbers are structurally the same (same number of wells and pipelines). The same is true between odd numbered subsea manifolds. Therefore two distinct constraints are defined for the two manifold categories.

Chapter 9

Computational study

This chapter presents the main results and analyses from the computational study of the optimization models implemented in this thesis. Two distinct studies are made, one analyzing the performance of the methods used, and the other studying the operational results related to the production asset. Section 9.1 covers the evaluation of the original MILP model described in Chapter 3 and 5, followed by a consideration of the importance of using gas lift technology at the Marlim field, found in Section 9.2. The DWD heuristics described in Chapter 7 are evaluated in Section 9.3, and the most promising heuristic is selected for further analysis. In the final section, the breakpoint resolution is increased for some SOS2 sets and a performance comparison is made between the original model and the most promising DWD heuristic.

9.1 Original model

This section outlines the main computational results of the study made on the implemented original MILP model found in Appendix B. An evaluation is made on the performance of the model. For this purpose, the model is tested on three cases, representing petroleum production assets of different size. The test cases are described in Section 8.5. This section also presents the main results of a sensitivity analysis for the separator capacity constraints for water- and gas flow rates. The second part of the study gives a more detailed overview of the optimization output for the optimal solution to Case 1. Optimal values of the decision variables for the physical system are presented and analysed.

9.1.1 Technical analysis

Users of optimization tools prefer realistic models that can be solved fast, but these objectives are often conflicting. Nygreen et al. (1998) discuss the classical trade-off between *reality representation* and *computational time* in the field of optimization, arising from the fact that an increase in model reality most often leads to longer solution time. Two important measures used in this thesis for performance evaluation of the implemented optimization models are duality gap, and computational time.

In this section, some important performance results from implementing and testing Case 1, 2 and 3 with the original MILP model are presented. Table 9.1 shows total oil production and computational time for the three tests. As the production asset is expanded from Case 1 to Case 3, the number of variables and constraints is increased since new manifolds and wells are added. Therefore it is natural to expect a longer solution time, but also an increase in oil production.

Table 9.1: *Summary of the results*

	Oil production [Sm ³ /d]	Comp.Time [s]
Case1	40765.8	1.62
Case2	66318.6	3652.03
Case3	–	–

As Table 9.1 shows, computation time increases considerably from Case 1 to Case 2. Case 3 became too complex for the Xpress-MP and the running of the model was terminated after about 2.5 hours, because the 24GB computer memory limit was exceeded. A note should be made that even for the unsolved case, problem initialization took only a few seconds, and the Simplex algorithm found an LP solution quite fast. The search for feasible integer solutions, however, proved to be extremely memory consuming for Case 3. The model was terminated while Xpress-MP was still searching in the B&B tree, without finding any feasible integer solutions.

9.1.2 Production analysis

This section gives a more detailed overview of the output data from the optimization. There are three main decision variables in the short term production planning problem studied here:

- Oil production at each well
- Allocation of lift gas to wells
- Flow routing

For Petrobras, the information listed above is most interesting for Case 1, as it represents the real production asset. Therefore, detailed output data will only be presented for Case 1. However, the implemented model generates the same output for any case where a feasible solution is found, and the information can easily be retrieved from the output files.

A final remark is that the input data used in the optimization is not completely realistic, but only *based* on real production data from the Marlim field. The results presented in this section are therefore not meant to provide an exact solution to how the production asset should be optimized. Instead, the results demonstrate how the optimization methods developed in this thesis can be used as a decision support tool when fictive data is replaced with real data.

Oil production and gas lift

The optimal solution to Case 1, according to the optimization model, suggests that 7 out of 15 wells are producing, while the rest should be closed down. Table 9.2 shows oil production rates and lift gas rates at each well in the optimal solution.

Table 9.2: *Production and allocation of lift gas in wells for case 1*

Manifold	Well	Oil production [Sm ³ /d]	Lift gas injected [Sm ³ /d]
Subsea 1	1	6201.3	276606.0
	2	–	–
	3	–	–
	4	911.2	96504.8
	5	–	–
	6	–	–
	7	3829.7	269988.0
Subsea 2	1	–	–
	2	–	–
Topside 1	1	–	–
	2	12150.1	206901.0
	3	1348.6	–
	4	11306.1	–
	5	5018.9	–
	6	–	–
Sum		40765.8	850000.0

According to the results, the wells connected to subsea manifold 2 are not producing oil at all, and most of the production comes from the satellite wells at the topside manifold. Lift gas is only distributed between 4 wells, and all of the available capacity is utilized in the optimal solution. This indicates the importance of using lift gas to maintain high production levels, as further discussed in Section 9.2.

Flow routing

Routing problems arise several places throughout the production system. At the subsea manifolds, several wells must be routed to one of two possible riser pipes. When the flow reaches the platform, a routing decision must again be made since the flow can be directed to three different separators. Tables 9.3, 9.4 and 9.5 present the optimal routing decisions according to the optimization model. The results are presented in matrices where all possible routing combinations are shown. The number 1 is used to indicate that flow is routed for a particular combination, whereas 0 corresponds to no routing.

Table 9.3: *Routing from wells to pipelines for Case 1*

Manifold	Pipe/ Well	1	2
Subsea 1	1	1	0
	2	0	0
	3	0	0
	4	0	1
	5	0	0
	6	0	0
	7	0	1
Subsea 2	1	0	0
	2	0	0

Table 9.4: *Routing from pipelines to separators for Case 1*

Manifold	Separator / Pipe	1	2	3
Subsea 1	1	0	1	0
	2	0	0	1
Subsea 2	1	0	0	0
	2	0	0	0

Table 9.5: Routing from satellite wells to separators for Case 1

Separator / Well	1	2	3
1	0	0	0
2	1	0	0
3	0	1	0
4	0	1	0
5	1	0	0
6	0	0	0

Capacity utilization

An important objective when making short term production plans is to utilize the available capacity in the best possible way to maximize total oil production. The most important capacity constraints when optimizing production are the separator capacities for water and gas and lift gas capacity at the compressor (Constraints (3.13) and (3.14)). In this section, we introduce a common term within optimization, namely the *slack*, representing the amount of unutilized capacity in the optimal solution. Table 9.6 shows the slack in all the previously mentioned capacity constraints for the optimal solution to Case 1.

Table 9.6: Capacity slack for separators and compressor in Case 1

Constraint	Phase	Slack [Sm ³ /d]
Compressor	lift gas	0
Separator 1	gas	0
	water	0
Separator 2	gas	0
	water	0
Separator 3	gas	287 430
	water	0

A slack equal to zero means that the installed capacity is fully utilized. In that case, the capacity constraints are said to be *binding*. Table 9.6 shows that all constraints are binding, except for an excessive amount of gas capacity on separator 3. This could indicate that the existing system is properly dimensioned, apart from separator 3, which appears to have too much gas capacity. It should be mentioned that making all capacity constraints binding might not always be possible or even desirable due to the separator routing flexibility in the MILP. From a practical point of view it often makes sense to install extra capacity so that production is not interrupted during maintenance or in case of unexpected breakdown of a separator, for example. Determining optimal capacities is a decision related to

medium-term production planning. This is not in the scope of our work and will not be discussed any further.

Sensitivity analysis

A sensitivity analysis is made to study the impact on total oil production when changing the existing capacity limits on the separators. The analysis is done for Case 1, where two separate studies are made for gas and water handling capacities. Such analyses can provide valuable information in the process of making investment decisions related to capacity expansion. Several other factors should be taken into account when making investment plans, which are not in the scope of this thesis.

Tables 9.7 and 9.8 present the main results from the sensitivity analyses for gas and water handling capacities, respectively. The tables show percentage change in capacity compared with Case 1, representing the Base Case, the corresponding oil production and production change as a result of the new capacities. In addition, slack and computational time are given for each of the tests. The percentage capacity change represents total change across all three separators, and the given slack value is the total slack from all separator constraints.

Table 9.7: *Sensitivity analysis of total gas capacity on the separators*

	Oil production [Sm ³ /d]	Prod. change [%]	Slack [Sm ³ /d]	Comp. time [s]
Base Case	40765.8	–	0	1.62
-10 %	39955.4	–1.99	0	2.25
10 %	40794.9	0.07	726214.1	1.57
20 %	40902.5	0.34	1167508.5	1.34
50 %	40929.9	0.40	2481824.7	2.13
100 %	40929.9	0.40	4681824.7	1.38

Table 9.7 shows that the gain in oil production rate is relatively modest as a result of increased gas capacity. This matches the results presented earlier in this section, showing that the current system already has excessive gas capacity at one of the separators. Small percentage changes in production can still have a considerable impact on income. The results show that a reduction in total gas capacity of 10% reduces oil production with almost 2%. A production loss of 2% equals 810.4 [Sm³/d] or 5097.3 [bopd] (see Chapter 4.1 for units conversion). Considering an oil price of 108 \$ per oil Barrel ¹, this would lead to an income loss of 5097.3 [Sm³/d] x 108 \$/Barrel = 550504 dollars per day.

¹Monthly average OPEC Basket Price for May 2012 (OPEC 2012)

According to Table 9.7 there is an upper limit to how much production gain can be made by adding more gas capacity at the separators. In fact, expanding capacity from 50% to 100% above the Base Case leads to no further increase in oil production.

Table 9.8: *Sensitivity analysis of total water capacity on the separators*

	Oil production [Sm ³ /d]	Prod. Change [%]	Slack [Sm ³ /d]	Comp. Time [s]
Base Case	40765.8	–	0	1.62
-10 %	38223.1	–6.24	0	1.06
10 %	42629.8	4.57	0	4.57
20 %	44809.8	9.92	0	3.33
50 %	48080.7	17.94	0	97.74
100 %	50677.8	24.31	0	692.05
200 %	52833.5	29.60	9622.1	47.60
250 %	52833.5	29.60	21770.3	136.22

Table 9.8 shows that water capacity has a much greater impact on total oil production than gas capacity. Reducing the total water capacity by 10%, leads to a production loss of more than 6%. By increasing total water handling capacity at the separators, it is possible to achieve a production gain of almost 30% compared with the Base Case.

It is difficult to make general conclusions about the slack for non-convex optimization problems, such as the MILP problem studied here. If the problem were an LP, it would have been impossible to increase production by increasing the capacity for a constraint with positive slack (Lundgren et al. 2010). Since the problem is an MILP, increasing capacity at an already unutilized separator might suddenly lead to a re-routing of flow between the separators that gives higher objective value. This is reflected in the results from the sensitivity analyses presented above, where the slack does not seem to follow a clear pattern.

One general statement is that increasing capacity leads to an relaxed solution space, which cannot make the solution worse than before. This is valid for both LP and MILP problems, and is confirmed by the results. In the same way, restricting the solution space by a capacity reduction can cut off the currently optimal solution and lead to worse solutions.

It is perhaps intuitive to expect the computational time to increase when the solution space is expanded, since the B&B algorithm might need to search through a larger set of feasible solutions before the optimum is found. From Table 9.8, showing water sensitivity, it is observed that solution time increases considerably as capacity is increased up to 100%. Increasing capacity above this value, however, leads to a remarkable reduction in solution time. The B&B algorithm can sometimes find the optimal integer solution early in the tree search and in that case

terminate fast, which could be the case here. For the gas sensitivity analysis, Table 9.7 shows that solution times are very low and global optimum is found quickly for all capacity values tested.

9.2 The importance of Gas lift

Gas lift technology is used to maintain high production levels at the Marlim field. The study presented in this section considers production sensitivity to gas lift and the importance of using this technology. Similar to the analyses presented so far, the study presented here is based on Case 1, because it has most relevance to Petrobras.

The original MILP model is solved with several different lift gas capacities. The results are compared with the optimal solution of Case 1, or the Base Case, where the lift gas capacity is 850000 [Sm³/d].

Table 9.9: *Sensitivity analysis of total lift gas capacity on the compressor*

Capacity [Sm ³ /d]	Oil production [Sm ³ /d]	Prod. change [%]	Slack [Sm ³ /d]	Comp. time [s]
Base Case	40765.8	–	0	1.62
0	36779.8	–9.78	0	0.98
500000	40437.9	–0.80	0	2.56
1000000	40781.6	0.04	76584.7	1.55
2000000	40781.0	0.04	1026584.7	1.47

Table 9.9 shows that removing lift gas capacity completely leads to a production loss of almost 10%. With an oil price of 108 \$ per oil Barrel, also used in the sensitivity analysis in Section 9.1.2, a 10% production loss would lead to a dramatic income loss of 2.7 million dollars per day. This example clearly shows the importance of using gas lift technology at the Marlim field.

Expanding capacity from 0 to 500000 [Sm³/d] increases production significantly. A slack value of 0 means that all available capacity is utilized, indicating that more capacity could possibly increase production even further. A capacity of 1000000 [Sm³/d] confirms this, as production is increased with 0.04% compared with the Base Case. This increase is however small, and there is some slack at this level. Adding even more capacity does not lead to any further production gain, as can be seen when capacity is set to 2000000 [Sm³/d]. To summarize, the maximum possible production gain is relatively small by expanding lift gas capacity, which is an indication that the system is reasonably well dimensioned.

As mentioned in Section 9.1.2, a sensitivity analysis can be used in the process of making investment decisions regarding capacity expansion. The analysis presented

here shows how optimization theory and methods can be interpreted and used for this purpose. However, as previously mentioned, an investment analysis is outside the scope of this work.

A final remark is that costs related to gas lift are left out of the optimization problem, as discussed in Section 3.2. Including costs in the objective function related to the use of gas lift could have an affect on optimal allocation of the available lift gas. The choice to disregard these costs is made based on discussions with production engineers at the Marlim field, stating that a cost consideration for gas lift is not common practice (Private conversation January 3, 2012).

9.3 Dantzig-Wolfe decomposition

This section presents the main results from tests done on the DWD heuristics described in Chapter 7. First, performance results for the two DWD methods described in Chapter 6 are presented and discussed. Then, the DWD heuristics are evaluated with the aim of determining the most successful one(s).

9.3.1 DWD test cases

All DWD heuristics are tested on 10 datasets based on Case 2, defined in Section 8.5. Using the production data for Case 2, 9 additional test cases were created by increasing or decreasing all simulated breakpoint data with 2% – 10%. This can be done without compromising physical realism, since the new test cases can be viewed as systems with slightly different flow properties, and pipeline and well design. In some test cases, separator capacities for water and gas were also changed to create even more variation in the systems. The reason for testing the methods on 10 datasets is to be able to make more robust conclusions regarding the performance of the heuristics developed.

The choice to use Case 2 as a basis for testing, is because Case 2 represents a production asset of considerable size in terms of number of wells and manifolds. This makes decomposition more relevant, and the heuristics produce more stable results. As previously mentioned, the main focus of the tests performed on the different heuristics, is to evaluate the methods from an optimization perspective, making the input data and solution values less relevant.

9.3.2 Comparison of decomposition methods

The optimization problem is decomposed in two slightly different ways, as described in Chapter 6, and the results from implementing both methods are presented here. Finally, a comparison of the performance for the two methods is made.

Decomposing on manifolds

The original model was initially decomposed on manifolds, where each manifold represents a separate optimization problem (subproblem) in the DWD formulation. This formulation is presented in Section 6.4.2, and the method will here be referred to as DWD-Man. Table 9.10 shows the average results from testing the decomposed model with 10 datasets.

Table 9.10: *Decomposing on manifolds*

RMP-IP solution [Sm ³ /d]	Comp. time [s]	Duality gap [%]	Iterations [#]	Columns [#]
54195.4	49.55	25.76	15	54.4

Each of the test instances generate an upper bound to the original MILP problem, which is obtained by solving the RMP-LP, as described in Chapter 7. A feasible IP solution, and a lower bound, is found by placing binary requirements on the weights (RMP-IP). As seen in Table 9.10, the duality gap between the lower and upper bound is on average 25.76% using this decomposition method.

The decomposition method made on average 15 column generation iterations before solving the RMP the last time. When the final RMP was solved it had an average of 54.4 columns, and an equal number of variables.

Decomposing on subsea manifolds and topside wells

Additional decomposition of the problem was done by splitting the topside manifold subproblem into several smaller subproblems for each satellite well connected to it. This is possible because the topside manifold subproblem has no common constraints for the wells. The model formulation is described in Section 6.4.3. To summarize, the decomposition method is a combination of decomposing on subsea manifolds and topside wells, and it is referred to as DWD-ManWell. The average results from tests done on this model are found in Table 9.11.

Table 9.11: *Decomposing on subsea manifolds and topside wells*

RMP-IP solution [Sm ³ /d]	Comp. time [s]	Duality gap [%]	Iterations [#]	Columns [#]
64 487.2	42.36	2.16	12.1	58.7

Comparison

A comparison of the results from the two decomposition methods shows that decomposing on topside wells gives the lowest average gap, with a value of 2,16%. In comparison, decomposing on manifolds resulted in an average gap of 25.76%.

Further, the DWD-Man method performed worst in all 10 tests in terms of duality gap. Sometimes the two methods produced similar solutions, while other times DWD-Man had a very poor duality gap performance. In 3 of the tests it produced a duality gap of more than 40%.

The results show that slightly more RMP columns were generated using DWD-ManWell, it converged after fewer iterations and the computational time was a little shorter on average. This could be a logical consequence of the fact that fewer iterations means less time spent solving the sequential subproblems. It is, however, important to note that several other factors might have an impact on the RMP-IP solution and computational time. A large number of columns will not always lead to better RMP-IP solutions if the columns are bad, for example. Computational time is not only dependent on the number of subproblems but also on the complexity of the sub- and master problems. Sensitivity to input data could also have a major impact on the results.

As seen in Tables 9.10 and 9.11, average computational time does not differ much between the two methods. This is therefore not a crucial factor in the evaluation of the two methods. Basing our evaluation on duality gap performance alone, is perhaps not sufficient to determine which is the most appropriate decomposition approach to use. However, the overall results from the 10 data tests presented here, support the choice to base the remaining heuristics on DWD-ManWell, as argued in Chapter 7.2.1. An important factor, reflected by the duality gap, is obviously the ability for the method to generate good columns, especially since the column generation process might be viewed as the core feature of all heuristics.

LP upper bound

An interesting observation was made for some tests regarding the upper bound from the two decomposition methods. The DWD-Man method resulted in lower LP relaxed solutions than DWD-ManWell in 4 out of 10 tests. For the remaining 6 tests the UB is equal. It is not surprising for the two methods to generate different columns during column generation, since dual prices from solving the RMPs will most likely be different. Interpolation between columns can still give the same RMP-LP solution. The fact that DWD-ManWell in some cases gives a higher LP solution, means that the solution space is relaxed compared to using DWD-Man. The former method clearly uses a combination of columns from the topside wells which is not possible to generate when all topside wells are solved as one subproblem, as in the latter method. DWD-Man considers separator capacity and routing on all topside wells simultaneously, and some combinations of solutions for the topside wells are not feasible. DWD-ManWell, however, has the option to interpolate between each topside well column.

9.4 Heuristics for improving the IP solution

Solving the RMP-IP problem right after column generation is the simplest heuristic that finds a feasible solution to the MILP problem. In this section, more advanced heuristics have been tested with the aim of adding more "good" columns to the RMP, which are not found in the initial column generation process. As mentioned in Section 7.2.1, this will not affect the RMP-LP solution, but hopefully improve the the solution of RMP-IP. A complete description of the heuristics is found in Chapter 7, and Table 9.12 gives a short summary.

Table 9.12: *Summary of the heuristics*

NewSepRouting	Creates new columns with the same flow and pressure values, but different routing, for all columns after the column generation is finished
DepthFirst	Fixes the largest feasible weight to 1 and generates new columns until the duality gap is small enough
SlackBoundsLP	Computes new bounds from the LP slack in two different ways and generates new columns by solving the subproblems with the new bounds.
SlackBoundsIP	Computes new bounds from the IP slack and generates new columns by solving the subproblems with the new bounds.

The methods can be combined in several different ways to produce new heuristics potentially improving the performance. The focus has been on implementing the combinations with the greatest potential. Table 9.13 gives an overview of the additional, implemented methods resulting from a combination of the main heuristics presented in Table 9.12.

Table 9.13: *Combinations of the heuristics*

DepthFirstNC	The original DepthFirst heuristic, in combination with NewSepRouting.
DepthFirstNCend	The original DepthFirst heuristic, with NewSepRouting at the end.
SlackBoundsLP&IP	Generates new columns by first solving SlackBoundsLP and then solving SlackBoundsIP.

DepthFirstNC is a combination of the DepthFirst and NewSepRouting heuristics. The difference between this heuristic and DepthFirstNCend, is that the former adds new routing columns in each node after column generation, while the latter only adds new routing columns in the last node right before RMP-IP is solved the last time.

SlackBoundsLP&IP uses the methods from SlackBoundsLP and SlackBoundsIP to generate additional columns from the subproblems. Each method generates columns independent of the other, before the new columns from both methods are combined and RMP-IP is solved one last time.

Table 9.14: *Comparison of the heuristics*

Heuristic	Duality gap [%]	Gap red. [%]	Comp. Time [s]	Columns [#]	Iterations [#]
NewSepRouting	2.16	0.00	42.72	312.8	12.1
DepthFirst	1.73	20.86	71.07	79.1	32.8
DepthFirstNC	1.79	15.96	78.91	3257.4	34.8
DepthFirstNCend	1.73	20.86	71.26	454.8	32.8
SlackBoundsLP	1.95	6.95	52.38	76.8	14.1
SlackBoundsIP	1.74	16.64	45.64	68.6	13.1
SlackBoundsLP&IP	1.69	18.26	54.80	86.5	15.1

Table 9.14 presents the average results from testing each heuristic on the 10 different test cases described in the introduction to this section. Each heuristic is evaluated based on average reduction in duality gap compared with the gap obtained from solving the RMP-IP after first column generation. The table also shows the average solution time for the different heuristics. Table 9.15 shows the number of times a heuristic has given the largest and smallest gap reduction, together with best and worst case reduction percentage.

Gap performance

From Table 9.14, it is clear that DepthFirst gives the highest average gap reduction, with a 20.86% decrease. Adding new routing columns before RMP-IP is solved the last time in DepthFirstNCend, did not lead to any further improvement for the 10 test cases, even though the method generated on average over 5 times more columns than pure DepthFirst.

The results show that the DepthFirstNC heuristic generates over 3000 columns on average, which is significantly more than the other methods. This method adds extra columns in each node of the DepthFirst search tree shown in Section 7.2.3, not only in the very last node, as the DepthFirstNCend does. Therefore it should not be surprising that this method generates more columns. Compared to the related DepthFirst and DepthFirstNCend heuristics, the DepthFirstNC results in a different, poorer gap performance. This can be expected since adding

extra columns throughout the search tree changes the RMP problems in the nodes. Different dual prices and columns might therefore be found in the subproblems, and the quality of the solution cannot be guaranteed.

The results indicate that adding extra routing columns is not as efficient as anticipated. However, the total number of columns increases, so there is still a theoretical possibility that it might reduce the gap in other cases. Adding extra columns in the end, right before solving RMP-IP the last time, does in any case not result in worse solutions than before.

Table 9.15: *Comparison of the heuristics*

Heuristic	BEST [#]	Best Case Red [%]	No Improve [#]	Worst case red. [%]
NewSepRouting	0	0.00	10	0.00
DepthFirst	4	63.10	4	0.00
DepthFirstNC	2	63.10	5	0.00
DepthFirstNCEnd	4	63.10	4	0.00
SlackBoundsLP	1	27.00	7	0.00
SlackBoundsIP	4	38.45	0	2.04
SlackBoundsLP&IP	5	41.75	0	2.04

Another heuristic that stands out is SlackBoundsLP&IP, with an average gap reduction of 18.26%. According to Table 9.15 this method performed best in terms of gap in 5 out of 10 test cases. The method seems to be the most stable, as it lead to a gap improvement in every test case. Apart from the similar SlackBoundsIP heuristic, no other method had a gap improvement in all 10 tests. The worst case gap reduction was 2.04%, while the gap was reduced with 41.75% in the best case. Only the DepthFirst (and DepthFirstNCEnd) had a better best case performance, reducing gap with an impressive 63.1%. It is not surprising that combining SlackBoundsLP and SlackBoundsIP into one heuristic, where both methods are used independent of the other, performs at least equally good, or better than any of the methods alone. This is observed in all tests. SlackBoundsIP seems to be the most effective of the two.

Table 9.16 gives an example of how the slack in the RMP common constraints is changed when using the SlackBoundsLP&IP method on Case 2. The idea behind the method is to reduce the slack, but according to the results, slack is also increased for some constraints. An explanation could be that it was not possible to generate all the desired columns as described in Sections 7.2.4 and 7.2.5. As a result, the RMP-IP solution tries to find the best way of distributing capacity with the columns found, which leads to solutions similar to the one in Table 9.16. Similar results are observed for all 10 tests.

Table 9.17 shows the slack in the capacity constraints for Case 2 when using the original, non-decomposed model to find the optimal solution. Comparing the op-

Table 9.16: *Slack before and after SlackBoundsLP&IP*

Constraint	RMP-LP slack	RMP-IP slack	RMP-IP slack	Change
	[sm ³ /d]	Initial [sm ³ /d]	Final [sm ³ /d]	
Sep 1 - gas	0.0	5535.5	38503.5	595.6
Sep 1 - water	0.0	99.1	205.8	107.7
Sep 2 - gas	310825.0	530766.0	371804.0	-30.0
Sep 2 - water	0.0	359.3	9.4	-97.4
Sep 3 - gas	1548860.0	1500970.0	1500970.0	0.0
Sep 3 - water	0.0	954.6	954.0	0.0
Compressor- lift gas	0.0	109518.0	0.0	-100.0

timal slack to the final slack in Table 9.16, shows that the values for separator capacity for gas are in the same order of magnitude in both cases. In the optimal solution, separator capacity for water is 0 on all three separators. Using the heuristic gives a negligibly low slack value for water on the separators. An optimal slack of 0 on the compressor is also achieved with the heuristic method, as Table 9.16 shows. In conclusion, the heuristic solution is relatively close to the optimal solution obtained with the original model when comparing the slack in the capacity constraints discussed here. Reducing the slack to 0 in all constraints is not necessarily always desirable or optimal.

Table 9.17: *Optimal slack for separators and compressor in Case 2*

Constraint	Phase	Slack [Sm ³ /d]
Compressor	lift gas	0
Separator 1	gas	50804.76
	water	0
Separator 2	gas	381657.69
	water	0
Separator 3	gas	1429972.32
	water	0

Computational time

Table 9.14 show that average computational times are in the range of 40 – 80 seconds, which is quite low for all heuristics. The variations in computational time across the different methods are considered negligible, and time is not of significant importance when comparing total performance of the heuristics. If anything should be noted from the results in Table 9.14, the DepthFirst heuristics have the highest computational time, probably related to the fact that column generation is repeated

in several nodes. Since the subproblems are solved sequentially, more iterations might lead to higher solution time.

Generating new columns by changing the separator routing decision in DepthFirstNC and DepthFirstNCEnd is computationally cheap since the subproblems must not be re-solved to obtain the new columns. As Table 9.14 shows, the increase in solution times for DepthFirstNC and DepthFirstNCEnd compared to the basic DepthFirst heuristic, is negligibly small.

Data sensitivity

An important observation after testing the DWD heuristics on several datasets is the sensitivity of results to input data. DepthFirst, for example, finds some very good extra columns in the best case, resulting in a gap reduction over 60%. On 4 other, slightly different datasets, the heuristic does not give any gap improvement. As mentioned before, the RMP-LP solution is unaffected by adding extra columns to the RMP, since column generation finds all the necessary columns to achieve the optimal LP solution. Finding good RMP-IP solutions is much more dependent on having a good selection of columns to choose between. If some good columns are not generated from the subproblems due to data sensitivity or small changes in dual prices, the DWD heuristic might suddenly perform poorly.

Based on a total evaluation of the results presented in Table 9.14, the SlackBoundsLP&IP combination heuristic seems to be the most promising and is therefore used in the rest of the analyses of this chapter.

9.5 DWD with the best heuristic

In this section, the most promising heuristic, SlackBoundsLP&IP, is tested on Case 1, 2 and 3. A complete description of the cases is found in Section 8.5. Reduction in duality gap and computational time are studied for all cases. The results are compared with the performance of the original, non-decomposed model.

Table 9.18: *Summary of the results from the best DWD heuristic*

	Oil production	LP Solution	Comp. Time	Duality Gap
	[Sm ³ /d]	[Sm ³ /d]	[s]	[%]
Case1	39332.7	40985.8	21.78	4.03
Case2	65516.2	66361.6	56.13	1.27
Case3	91720.8	92634.7	103.45	0.99

Gap performance

Table 9.18 shows that the DWD heuristic finds a feasible integer solution in all three cases. An interesting observation is that the gap is reduced as the problem becomes larger. The duality gap is reduced from 4% in Case 1 to less than 1%

in Case 3. However, more extensive testing should be done before making general conclusions regarding this.

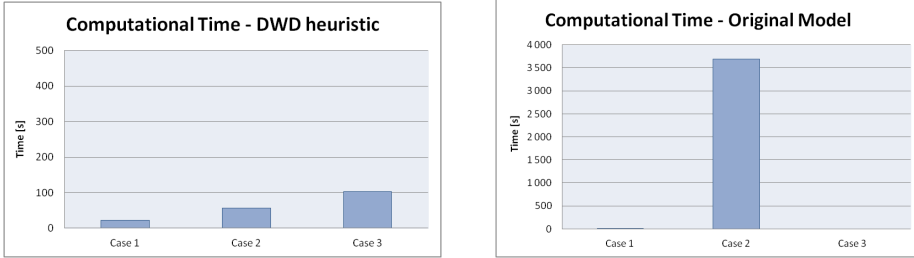
The duality gap provides an indication of the quality of the solution, as global optimum is found somewhere between the upper and lower bounds defining the gap. For example, in Case 1, no solution can be found that is more than 4.03% better than the current solution. 4.03% is an optimistic, upper limit, and the real attainable, optimum is most likely lower. In fact, since the optimal solution to Case 1 is known from solving the exact, non-decomposed model, the real distance to optimum is also known. The optimal solution is found in Table 9.1, and the heuristic solution is 3.52% lower than optimum. In Case 1 it therefore makes little sense to use decomposition methods to solve the model, as the optimal solution can be found within less than 2 seconds with the non-decomposed model.

Case 3 has a duality gap of 0.99%, obtained within less than 2 minutes, whereas the original model did not find any feasible solution and terminated after 2.5 hours due to the lack of computer memory. For this case it is impossible to know exactly how far the heuristically obtained solution is from global optimum, but the UB tells us that it is impossible to increase production by more than 0.99%.

According to the results in Table 9.18, computational time increases from around 20 seconds to a little over 100 seconds in Case 3. From Case 1 to Case 3, there has been an increase in the number of subsea manifolds, in addition to a few more subsea wells. As a consequence, the decomposed model has more subproblems. Further, the additional subsea wells make some of the subproblems more complex. Both of these factors have an impact on computational time. Since the subproblems are solved sequentially, the increase in solution time due to more subproblems is expected.

Computational time

Figure 9.1 compares the computational times for the three test cases implemented with the DWD heuristic and the original model. For the DWD heuristic, solution times are increasing almost linearly from Case 1 to Case 3. In comparison, the solution time using the original model increases dramatically for Case 2, whereas Case 3 is unsolved.



(a) Computational time using the DWD heuristic

(b) Computational time using the original, non-decomposed model

Figure 9.1: Comparison of computational time between using a DWD heuristic and the non-decomposed model on three cases.

9.5.1 Production analysis

Similar to Section 9.1.1, this section presents more detailed production results from using a DWD heuristic method to solve the optimization problem in Case 1. The results are compared with the original model, that was able to find the optimal solution. The DWD heuristic does not guarantee an optimal solution, and it is therefore interesting to look at the difference in production decisions between the two solutions.

Table 9.19 shows optimal oil production rates and lift gas rates for each well, according to the heuristic solution. Compared with the optimal solution presented in Section 9.1.2, this solution suggests that 6, instead of 7, wells should be producing oil. As a result, total oil production is 3.52 % lower than in the optimal solution. Gas lift capacity is fully utilized, as in the optimal solution.

Table 9.19: *Production and allocation of lift gas to wells for Case 1 for the best DW heuristic*

Manifold	Well	Oil production Sm ³ /d	Lift gas injected Sm ³ /d
Subsea 1	1	6139.1	273270.0
	2	–	–
	3	–	–
	4	881.4	56633.7
	5	–	–
	6	–	–
	7	3888.6	322362.0
Subsea 2	1	–	–
	2	–	–
Topside 1	1	–	–
	2	12134.9	197735.0
	3	–	–
	4	11327.6	–
	5	4961.2	–
	6	–	–
Sum		39332.7	850000.0

Tables 9.20, 9.21 and 9.22 give information about flow routing decisions. The results show that routing decisions remain unchanged in the heuristic solution. The only difference compared with the optimal solution from the original model is that satellite well 3 at the topside manifold is closed down and therefore not routed to separator 3.

Table 9.20: *Routing from wells to pipelines for Case 1 for the best DW heuristic*

Manifold	Pipe/ Well	1	2
Subsea 1	1	1	0
	2	0	0
	3	0	0
	4	0	1
	5	0	0
	6	0	0
	7	0	1
Subsea 2	1	0	0
	2	0	0

Table 9.21: Routing from pipelines to separators for Case 1 for the best DW heuristic

Manifold	Separator / Pipe	1	2	3
Subsea 1	1	0	1	0
	2	0	0	1
Subsea 2	1	0	0	0
	2	0	0	0

Table 9.22: Routing from satellite wells to separators for Case 1 for the best DW heuristic

Separator / Well	1	2	3
1	0	0	0
2	1	0	0
3	0	0	0
4	0	1	0
5	1	0	0
6	0	0	0

9.6 Number of SOS2 breakpoints

The MILP model studied in this thesis is formulated using piecewise linear approximation with SOS2, as described in Chapter 5.3, making it highly relevant to discuss the trade-off between model realism and solution time. Increasing the breakpoint resolution when approximating a non-linear function increases model accuracy, but makes the problem more complex.

In this section, an analysis is presented where the breakpoint resolution for the riser pipe pressure drop (Constraints (3.8)) is increased. The initial resolution, used in the computational analyses so far, has 6 breakpoints for each of the three flow rates. A new SOS2 set is simulated using 11 data points in each dimension, as described in Chapter 5.3. Figure 5.8 in Chapter 5 gives a comparison in function approximation between the two SOS2 sets. The aim of this analysis is to study the effect on computational time and solution quality when increasing the SOS2 breakpoint resolution. The performance of the DWD heuristic is compared with the original model.

In Table 9.23, an overview of the number of constraints, variables, SOS2 sets and total number of SOS2 variables is shown for Case 1, 2 and 3 with the two different SOS2 resolutions. The table shows the complexity of the original optimization model. The SOS2 sets using 6 data points per flow rate are named *Small*, and *Large* represents the sets with 11 data points.

Table 9.23: *Overview of the complexity*

	Constraints	Variables	SOS2 sets	Set elements
Case1 Small	707	2047	59	337
Case1 Large	767	6567	59	397
Case2 Small	1335	3950	112	640
Case2 Large	1455	12990	112	760
Case3 Small	1999	5922	168	960
Case3 Large	2179	19482	168	1140

As Table 9.23 shows, increasing the SOS2 breakpoint resolution from Small to Large results in a dramatic increase in the total number of variables. This is caused by the additional weight variables associated with each new data point in the SOS2 formulation. Table 9.23 also shows that there is an increase in model complexity as the production unit is expanded from Case 1 to Case 3. More constraints and variables are added in this process. Overall, increasing the number of SOS2 data points has the greatest impact on model complexity.

Original model results

Table 9.24 presents the main performance results from testing the original, non-decomposed MILP model for Case 1, 2 and 3 with both breakpoint resolutions. As the table shows, the large SOS2 set leads to increased solution time for Case 1, although both solution times are less than 10 seconds. The impact on solution time is more noticeable for Case 2. In this case the solver uses approximately one hour to solve the model with the small SOS2 set, whereas the large set remains unsolved, i.e. no feasible solution is found by Xpress-MP. The running is terminated due to the lack of computer memory. The same happens with Case 3 for both the small and large SOS2 sets. Reading of input data and initializing the problems is done relatively fast, but Xpress-MP consumes all available memory during the search for feasible integer solutions in the B&B tree.

Table 9.24: *Breakpoint sensitivity in the Original model*

	Oil production [Sm ³ /d]	LP solution [Sm ³ /d]	Comp. Time [s]
Case1 Small	40765.8	42763.0	1.62
Case1 Large	40628.3	42763.0	6.60
Case2 Small	66318.6	74021.4	3652.03
Case2 Large	–	74021.4	–
Case3 Small	–	112206.0	–
Case3 Large	–	112206.0	–

The LP solution in Table 9.24 is obtained by solving the LP relaxation of the MILP problem. This means that all integer variables, binary in this case, are made continuous. In addition, the SOS2 neighbour point requirements described in Chapter 5 are relaxed. This creates a convex, linear solution space. In the cases where an optimal solution is not found, the LP solution gives the user information about the upper bound on total oil production. For example, for Case 2 Large, the optimal oil production rate cannot be better than the LP upper bound of 74021.4 [Sm³/d]. The maximum attainable production rate, or global optimum, is most likely lower.

An interesting observation from Table 9.24 is that the LP solution remains unchanged for all three cases when increasing the SOS2 breakpoint resolution. SOS2 Large is created by adding one new data point between each of the existing points in SOS2 Small for the pipeline pressure drop relationship, as explained in Chapter 5. In the LP relaxed problem, the SOS2 neighbour point requirements are relaxed, as previously mentioned, making it possible to interpolate between data points that are not adjacent. As a result, no feasible LP solutions are removed when using SOS2 Large instead of SOS2 Small, which explains the results in Table 9.24.

DWD heuristic results

The same case instances are tested on the most promising DWD heuristic, and the results can be found in Table 9.25. This method finds a feasible solution to the MILP problem for all cases and SOS2 sets. For each case, solution time is increased when using the large SOS2 set. Increasing the SOS2 breakpoint resolution leads to more variables and additional constraints, as seen in Table 9.23. In the DWD model, this increases the subproblem complexity. In this case, only the subsea subproblems are affected because the riser pipe function is only defined for subsea manifolds. As a result, the subsea subproblems take longer to solve, which is the main reason for the increase in total computational time. Note the difference between increasing the number of subproblems from Case 1 to Case 3, as opposed to making each subproblem more complex with higher SOS2 data point resolution. Both changes lead to higher computational time, but for different reasons, as previously explained.

Table 9.25: Breakpoint sensitivity in the best DW heuristic

	Oil production [Sm ³ /d]	LP solution [Sm ³ /d]	Comp. Time [s]	Pessimistic Gap [%]	Duality Gap [%]
Case1 Small	39332.7	40985.8	21.78	3.52	4.03
Case1 Large	36069.4	40823.2	51.53	11.22	11.64
Case2 Small	65516.2	66361.6	56.13	1.21	1.27
Case2 Large	66590.6	68101.7	159.31	–	2.22
Case3 Small	91720.8	92634.7	103.45	–	0.99
Case3 Large	91300.6	91912.0	557.33	–	0.67

Table 9.25 shows that the LP solution, obtained from solving the RMP-LP, changes when using different SOS2 sets. Note that the LP solution from the DWD heuristic is not calculated in the same way as the LP solution from the original model. In the RMP-LP, the SOS2 requirements are not relaxed, and therefore the LP solutions change when increasing the SOS2 breakpoint resolution. This is not surprising as the solution space is different. Some solutions might be cut off, and other possible solutions added. This could therefore both increase or decrease the real solutions and upper bounds. For Case 1 and Case 3, both the MILP solutions and the LP solutions are decreased when using the Large SOS2 sets. The opposite happens in Case 2. For Case 2 Large, the oil production rate is higher than the LP upper bound for Case 2 Small, as Table 9.25 shows. This indicates that adding more SOS2 data points resulted in new, and better solutions in the solution space of the MILP for Case 2.

Comparison

Comparing the original model with the DWD heuristic, an increase in SOS2 points clearly shows the advantage of decomposing the problem for Case 2 and 3. Even the largest system (Case 3), is solved within less than 10 minutes and with a duality gap of 0.67%, whereas the original model does not provide any feasible solutions at all. For Case 1 however, the original model outperforms the DWD heuristic, as it finds the optimal solution very fast with both SOS2 resolutions. This indicates that problems must be of a certain size before it makes sense to use decomposition. Continuing to increase breakpoint resolution even further, and on more functions, could make even Case 1 unsolvable at some stage with the original model. Since increasing data resolution does not have the same dramatic impact on computational time for the DWD heuristic, this method would most likely find a feasible solution within reasonable time. An accurate, feasible solution might be more valuable than solving a less realistic model to optimum, in which case a heuristically solution approach is more appropriate.

The results from Tables 9.24 and 9.25 show further that the LP relaxed solution, corresponding to an upper bound (UB), is lower for the DWD heuristic compared with the original model in all cases. A lower UB is preferred for maximization

problems since it is closer to the global optimum. In cases where global optimum cannot be found or guaranteed, good bounds provide important information about the distance to optimum. There might be situations where the non-decomposed model cannot guarantee optimum either. In that case, bounds information is an important measure for solution quality.

9.7 Further Improvements

This section discusses further options for improvement in the most promising DWD heuristic.

9.7.1 Sequential versus Parallel solving

As mentioned in Chapter 8, the column generation processes in the DWD heuristics developed in this thesis, is implemented and solved sequentially. Table 9.26 presents solution times for all subproblems in the first iteration of the column generation process for the SlackBoundsLP&IP heuristic, solved for Case 2. Solution times are given for both the Small and Large SOS2 sets, analyzed in Section 9.6.

Table 9.26: *Computation time for each of the subproblems*

Manifold Type	Subproblem	Solution time	Solution time
		SOSsmall [s]	SOSlarge [s]
Subsea	1	2.036	8.086
	2	1.107	3.301
	3	1.716	8.293
	4	1.162	2.947
Topside	1	0.083	0.008
	2	0.008	0.007
	3	0.005	0.047
	4	0.005	0.055
	5	0.069	0.037
	6	0.065	0.009
Total		6.256	22.79

Table 9.26 shows that increasing the SOS2 breakpoint resolution for some SOS2 sets in the subsea subproblems leads to increased solution times. Since no changes in the breakpoint resolution are made in any SOS2 set for the topside subproblems, there is no significant change in complexity or solution time there.

When the subproblems are solved sequentially, total solution time per iteration in the column generation, equals the sum of solution times from solving the subproblems. Instead, it is possible to implement a parallel solution strategy. In that case the subproblems are solved simultaneously and total computational time would ideally equal the time it takes to solve the slowest subproblem. In the example from Table 9.26, the problem using the large SOS2 sets would have a total solution time of 8.086 seconds for this particular iteration, instead of 22.79 seconds with the sequential method.

Parallel solving has a great potential of reducing the computational time when data point resolution is increased. This way, higher model accuracy could be achieved and larger MILP problems could be solved more efficiently. In this thesis, a parallel implementation was tested with the *mmjobs* module in Xpress-MP, but the expected improvement in solution time was not achieved. An explanation could be that this Xpress-MP functionality faces some challenges related to solving several subproblems at once, and handling data transfer to memory for several models simultaneously. This assumption is not further studied in the work presented here and remains an interesting topic for further research.

9.7.2 More columns per subproblem

In an attempt to improve efficiency and reduce the duality gap even further with the most promising heuristic, SlackBoundsLP&IP, the method described in Section 7.2.6 was implemented. The idea is that, instead of adding only the column with the highest reduced cost from each subproblem, N feasible columns with positive reduced cost are added to the RMP at the same time. For large case instances the method could lead to a significant reduction in solution time (Barnhart et al. 1998). Therefore, the analysis is done for the most complex case in this thesis, namely Case 3 with the large SOS2 set.

The graph in Figure 9.2 shows the duality gap when obtaining N solutions from each subproblem. Tests are made for N values in the interval $[1, 10]$.

Adding N columns to the RMP each time a subproblem is solved affects the column generation process. Different N values result in different RMP problems, and the dual prices are most likely changed. This has an impact on the columns found in the subproblems. Therefore it is difficult to say whether increasing N will lead to better or worse RMP-IP solutions. As Table 9.2 shows, the optimal number for Case 3 Large seems to be $N = 5$ in the interval tested.

Table 9.3 gives an overview of the solution time as N increases. Solution time decreases at first, but starts to increase as N becomes higher. In the interval tested, $N = 2$ gives the lowest solution time.

Several factors could have an impact on the solution time and it is not possible to guarantee efficiency improvement by increasing N . The column generation process is, as previously mentioned, affected by changing N in a way that is difficult to

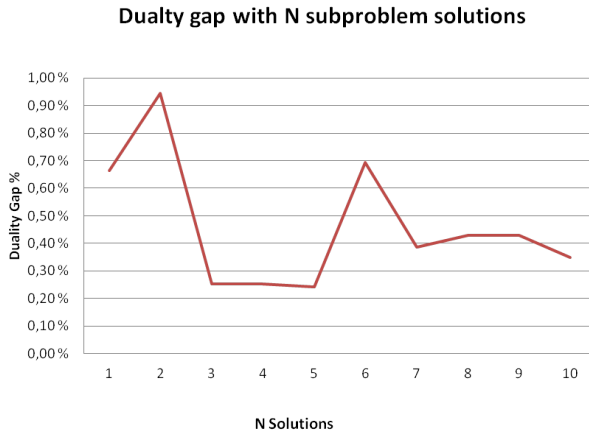


Figure 9.2: *Duality gap with N subproblem solutions*

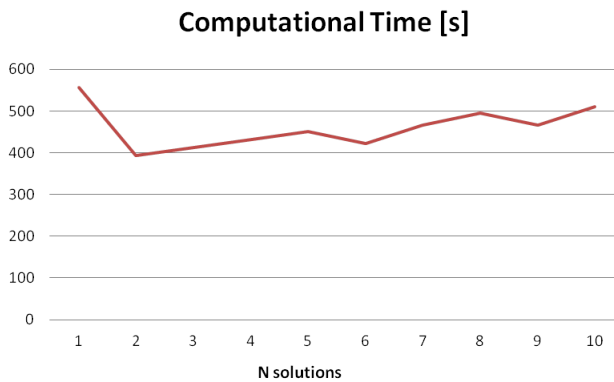


Figure 9.3: *Computational time when obtaining N columns per subproblem*

predict. If the number of iterations is not reduced, or the subproblems take longer to solve, this could lead to higher solution times. We have also experienced that data transfer between files in Xpress-MP is considerably time consuming for very large values of N , i.e. when $N = 100$.

The main conclusion from this study is that the method clearly has a potential for improving the solution time and duality gap, but it is difficult to make general conclusions regarding the optimal N . Finding an appropriate N should ideally be based on a similar performance evaluation as part of a post-processing analysis for the each individual problem being studied.

9.8 Applicability in practical planning

Several factors must be considered when evaluating the applicability of the optimization models in practical production planning. Piecewise linearization of the non-linear functions described in Chapter 5, leads to modeling errors. In particular, the riser pipe pressure drop as a function of oil, gas and water flow rates is a difficult relationship to approximate. Verifying the pressure results from the optimization model with the PIPESIM simulator for Case 1 and the small SOS2 sets, gave an error of 11.3% in the worst case. To increase model realism, it might therefore be necessary to improve the function approximations.

The assumptions made in Chapter 3.2 are simplifications of the physical system making the model less realistic. However, it is important to keep in mind that all models are simplifications of the real system, and a model is not necessarily good if it includes as many details as possible. Determining whether a model is good or bad is dependent on the purpose of the model (Wallace 2003).

In the optimization problem studied in this thesis, the purpose is to provide decision support related to short-term petroleum production planning. Some decisions, such as flow routing and amount of lift gas to be injected into each well, are directly applicable by the operational engineer. Other decisions, such as oil production rates in the wells, are consequences of optimal pressure values determined by the model. Before the models developed in this thesis can be applied to the real production asset, a verification of the results is necessary to make conclusions regarding the quality of the models.

Chapter 10

Conclusion

The work presented in this thesis has several purposes. First, it suggests solution strategies based on optimization methods for a petroleum production planning problem for the Brazilian energy company, Petrobras. Several model formulations are considered and a subset of these is implemented with appropriate software. The structure of the problem gives rise to a consideration of decomposition approaches to improve solution efficiency. Therefore, another important aspect of this thesis is the evaluation of decomposition methods to solve large scale mixed integer linear programming (MILP) problems.

The optimization problem studied in this thesis has integer requirements, in addition to several non-linear flow and pressure relationships. These are approximated using piecewise linearization with SOS2, resulting in an MILP model. Two main types of model formulations are implemented. The first is a standard linearized optimization model, implemented and solved in its complete form. It is referred to as the original model. The second is a decomposed model formulation, where the overall problem is divided into smaller subproblems, solved sequentially, and a master problem handling the common constraints. The decomposed model is combined with several different heuristics to find a feasible integer solution. Using heuristics, good solutions can be found fast but global optimum is not guaranteed.

Dantzig-Wolfe decomposition with different heuristics finding feasible integer solutions are tested and evaluated based on gap performance and solution time. The aim has been to generate additional columns from the subproblems in a clever way, increasing the chances of finding good integer solutions efficiently. Tests showed that the performance is highly sensitive to input data. Small variations in the numbers transferred between the subproblems and masterproblem have a great impact on the columns generated and the resulting MILP solution. It is not possible to guarantee that one heuristic will perform well in every test case, although some methods proved more stable than others.

Using the original optimization model, the optimal solution to the production planning problem is efficiently found. Solving the decomposed problem with the most promising heuristic, generates a feasible integer solution with a gap of 4.03 % and a longer solution time than the former method. Based on the performance results, it makes more sense to apply the original, non-decomposed model, as it finds the globally optimal solution. Solution accuracy should, however, be questioned. Crude approximations of the non-linear functions are used, resulting in a less accurate model. This motivates another important focus area of this thesis, namely studying the effect of model complexity on the two solution methods.

Three different test cases are defined, based on the actual production asset, representing Case 1. Case 2 and 3 consider possible expansions of the current system, where more wells and manifolds are added. This increases the model complexity in terms of number of variables and constraints. Complexity is further increased by increasing the data point resolution in the piecewise linear formulation of the most challenging function, namely the pipeline pressure drop as a function of oil, gas and water flow rates.

Results show that increasing complexity has a much more dramatic effect on the original model compared to the decomposed model. With the crude linear approximation, the original model spends more than one hour before optimum is found for the medium sized production asset in Case 2, whereas the decomposed model finds a solution in less than a minute. For the largest case, Case 3, the original model does not find any feasible solutions, and the computer capacity of 24 GB is reached. In contrast, the heuristic is able to find a feasible integer solution for all cases, even using a higher data point resolution. The most complex case, Case 3, combined with the more accurate linear approximation, is solved in less than 10 minutes with a duality gap of 0.67 % with the most promising heuristic.

Based on these results, it can be concluded that the optimization model should be of a certain size before it makes sense to use decomposition methods. At the same time, decomposition methods allow for more accurate modelling, resulting in more realistic solutions, although optimum cannot be guaranteed. Solving the production planning problem with the original model presents major limitations concerning solution accuracy, as increasing realism quickly makes the problem unsolvable. In the choice between an exact solution method versus a heuristic, the trade-off must be made between model realism and solution time.

There are still challenges to overcome before the models can be applied to the production planning problem at Marlim. Modelling error must be reduced to increase solution realism. This is mainly related to the function approximation of difficult non-linear pressure and flow relationships. In this context, decomposition methods provide great opportunities for efficiently solving more complex MILP models. As a conclusion, promising solution strategies have been developed for the petroleum production problem moving us closer to an applicable solution by providing a solid basis for further research.

Chapter 11

Further work

This last chapter provides ideas and thoughts about improvement and expansion of the work presented in this thesis. Several areas with potential for enhancement related to modelling, solution methods and data realism, are discussed.

As mentioned in Chapter 3, the mathematical model in this thesis maximizes oil production. To make the model more realistic and accurate, the model could be extended to maximize profit. In that case, an analysis of costs and income related to the production process has to be made. An overview of all costs and incomes requires time and effort, and this should be considered before deciding to expand the model. In this thesis, costs related to gas lift are also disregarded. According to the objective function used, it is always profitable to use more lift gas, as long as more oil is extracted from the reservoir. Including costs related to gas lift in the model, could make the resulting use of lift gas more realistic.

Another expansion possibility is to include investment decisions in the model. The investment decisions could be investing in more lift gas capacity or more gas and water capacity on the separators. In this context, the model should be made dynamic by including time discretization, since investment decisions are usually made for a longer time horizon. The model would therefore no longer be a short term production planning problem. This expansion of the model would increase the complexity, by introducing more decision variables.

Available gas capacity is continuously changing in real production planning problems. Therefore, it might be necessary to consider the uncertainty related to this parameter, by introducing stochastic programming methods. A stochastic formulation is also highly relevant when maximizing profit and including investment decisions in the mathematical model, because parameters such as demand and oil price are uncertain and should be implemented as stochastic parameters to obtain a reliable model.

In this thesis, heuristics that find feasible integer solutions have been applied with

a DWD formulation to reduce the duality gap. Instead of a heuristic solution, an exact solution approach, such as Branch& Price (B&P), can be implemented. Unlike heuristic methods, the B&P algorithm is able to guarantee global optimum. However, since implementing B&P is a quite time consuming process, the gain of implementing it should be weighted against the effort necessary.

The DWD algorithm was implemented with sequential running of the subproblems. An alternative is to use a parallel solution strategy. As discussed in Section 9.7.1, this has the potential to reduce the computational time considerably.

Xpress-MP, the optimization solver used in this thesis, proved to be relatively sensitive to input data. This is discussed in Chapter 9. Therefore, an interesting study could be to implement and solve the optimization problem with alternative software for comparison. Since this thesis only considers the use of MILP modelling, another possibility is to formulate this problem as an MINLP, and use a non-linear optimization software to solve the problem. An interesting study would be to compare the solution methods and results of these two solution approaches.

There is potential for improvement in the approximation of non-linear functions present in this model. In this thesis, a piecewise linearization method with special ordered sets of type 2 (SOS2) is used. Related to this formulation, an increase in data point resolution is necessary to increase model accuracy. It is also relevant to consider better placement of data points for this purpose. Alternatively, other function approximation methods than piecewise linearization exist, that could be interesting to implement with this optimization problem to improve accuracy.

Before applying this model in the industry, it might be necessary develop a better user interface. The system should be made easy for the production planner to understand. Today the simulation of data and the running of an optimization model are two separate processes. An attempt should be made to combine these into one program. Ideally the user should only need to insert input data into a model and push a *run* button.

Bibliography

- Acosta, M., Farias, R., Mendez, A., Pineda, F., Montanha, R., Garcia, E., Calderon, A., Magno, C. & Germino, M. (2005), Deepwater Horizontal Open-hole Gravel Packing in Marlim Sul Field, Campos Basin, Brazil - Completion Project Learning Curve and Optimization, *in* 'SPE Annual Technical Conference and Exhibition, 9-12 October 2005, Dallas, Texas', Society of Petroleum Engineers.
- Anthony, R. N. (1965), 'Planning and Control Systems: A Framework for Analysis'.
- Ashford, R. (2007), 'Mixed Integer Programming: A Historical Perspective with Xpress-MP', *Annals of Operations Research* **149**.
- Awad, S. P., Piazza, M. R. & Nogueira, E. F. (1995), Drilling Optimization in Deepwater Field Development Offshore Brazil, *in* 'Offshore Technology Conference, 1 May-4 May 1995, Houston, Texas', Offshore Technology Conference.
- Bampi, D. & Costa, O. J. (2010), Marlim Field: An Optimization Study on a Mature Field, *in* 'SPE Latin American and Caribbean Petroleum Engineering Conference, 1-3 December 2010, Lima, Peru', Society of Petroleum Engineers.
- Baricelli, P., Mitra, G. & Nygreen, B. (1998), 'Modelling of Augmented Makespan Assignment Problems (AMAPS): Computational Experience of Applying Integer Presolve at the Modelling Stage', *Annals of Operations Research* **82**, 269–288.
- Barnhart, C., Johnson, E. L., Nemhauser, G., Savelsbergh, M. W. P. & Vance, P. (1998), 'Branch and Price: Column Generation for Solving Huge Integer Programs', *Operations Research* **46**, 316–329.
- Beale, E. M. L. & Tomlin, J. A. (1970), 'Special Facilities in a General Mathematical Programming System for Non-Convex Problems Using Ordered Sets of Variables', *J. Lawrence OR 69:Proceedings of the Fifth International Conference on Operational Research* pp. 447–454.
- Beasley, J. (1993), Lagrangian relaxation, *in* C. Reeves, ed., 'Modern Heuristic Techniques for Combinatorial Problems', Blackwell Scientific Publications, Oxford, pp. 243–303.

- Beasley, J. (1996), *Advances in Linear and Integer Programming*, 'Oxford Science Publications.
- Benders, J. F. (1962), 'Partitioning Procedures for Solving Mixed-Variables Programming Problems', *Numerische Mathematik* **4**, 238–252.
- Bieker, H., O.Slupphaug & Johansen, T. (2006), Global Optimization of Multiphase Flow Networks in Oil and Gas Production Systems., in 'AIChE Annual Meeting, San Francisco'.
- Bieker, H., Slupphaug, O. & Johansen, T. (2007), 'Real-Time Production Optimization of Offshore Oil and Gas Production Systems: A Technology Survey', *SPE Production & Operations* **22**.
- Bodington, C. & Baker, T. (1990), 'A History of Mathematical Programming in the Petroleum Industry', *Interfaces* **20**, 117–127.
- Bonami, P., Biegler, L., Conn, A., Cornuejols, G., Grossmann, I., Laird, C., Lee, J., Lodi, A., Margot, F., Sawaya, N. & Wachter, A. (2005), 'An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs', *Discrete Optimization* **5**, 186–204.
- Bradley, S., Hax, A. & Magnanti, T. (1977), *Applied Mathematical Programming*, Addison-Wesley.
- Camponogara, E. & Nakashima, P. (2006), 'Optimizing gas-lift production of oil wells: piecewise linear formulation and computational analysis', *IIE Transactions* **38**, 173–182.
- CIA (2012), 'The World Factbook - Country Comparison Oil - Production'. Visited January 13, 2012.
URL:<https://www.cia.gov/library/publications/the-world-factbook/rankorder/2173rank.html>
- Codas, A. & Camponogara, E. (2012), 'Mixed-Integer Linear Optimization for Optimal Lift-Gas Allocation with Well-Separator Routing', *European Journal of Operation Research* **217**, 222–231.
- Colombani, Y. & Heipcke, S. (2006), 'Multiple Models and Parallel Solving with Mosel'. Downloaded March 14, 2012.
URL:http://www.ic.unicamp.br/~lee/mc548/trabalho/xpress/docs/mosel/mosel_parallel/dhtml/moselparcover.html
- Columbani, Y. & Heipcke, S. (2008), 'Mosel: An Overview'. Downloaded January 15, 2012.
URL:<http://www.fico.com/en/Products/DMTools/xpress-overview/Pages/Xpress-Documentation.aspx>
- Corbishley, D. W., Guedes, S. S., Pinto, A. C. C. & Cora, C. A. (1999), Optimizing the Development of a Giant Deepwater Multireservoir Oil Field through

- Reservoir Simulation, *in* ‘SPE Reservoir Simulation Symposium, 14-17 February 1999, Houston, Texas’, Society of Petroleum Engineers.
- D’Ambrosio, C., Lodi, A. & Martello, S. (2010), ‘Piecewise Linear Approximation of Functions of Two Variables in MILP Models’, *Operations Research Letters* **38**, 39–46.
- Danna, E. & Pape, C. L. (2005), *Column Generation*, Springer.
- Dantas, A. C., Kac, A. & Rosolem, C. R. (1995), ‘Development of Floating Production Storage and Offloading Systems to be Installed in Marlim Field, Campos Basin, Brazil’.
- Dantzig, G. & Wolfe, P. (1960), ‘Decomposition Principle for Linear Programs’, *Operation Research* **8**, 101–111.
- Dias, J. W. M. & da Silva, R. A. R. (1999), Marlim field: The Utilization of Floating Production Units, Offshore Technology Conference.
- Dzubur, L. & Langvik, A. (2011), Optimization of Oil Production. Specialization Project, Norwegian University of Science and Technology.
- Eikrem, G. (2006), Stabilization of Gas Lift Wells by Feedback Control, PhD thesis, The Norwegian University of Science and Technology, Trondheim, Norway.
- Elster, K. H. (1993), *Modern Mathematical Methods of Optimization*, Akademie Verlag.
- ENCYCLO (2011), ‘Look up: Satellite well’. Visited October 30, 2012.
URL:<http://www.encyclo.co.uk/define/satellite%20well>
- ENCYCLO (2012), ‘Look Up: mmscfd’. Visited June 5, 2012.
URL:<http://www.encyclo.co.uk/define/MMscfd>
- ExxonMobile (2012), ‘Energy & Technology’. Visited Mai 5, 2012.
URL:http://www.exxonmobil.com/corporate/energy_outlook_eoiloilproduction.aspx
- Fang, W. & Lo, K. (1996), ‘A Generalized Well-Management Scheme for Reservoir Simulation’, *SPE Reservoir Engineering* **11**, 116–120.
- FICO (2009a), ‘Getting Started with Xpress’. Downloaded February 10, 2012.
URL:http://brblog.typepad.com/files/getting_started-1.pdf
- FICO (2009b), ‘Xpress-Optimizer Reference Manual’. Downloaded February 10, 2012.
URL:<http://brblog.typepad.com/files/optimizer-1.pdf>
- FICO (2012), ‘Mosel Reference Manual’. Visited June 10, 2012.
URL:http://www.ic.unicamp.br/~lee/mc548/trabalho/xpress/docs/mosel/mosel_lang/dhtml/mmjobs.html

- Fleischmann, B., Meier, H. & Wagner, M. (2004), *Supply Chain Management and Advanced Planning*, Vol. 4, Springer, Chapter Advanced Planning, pp. 71–96.
- Foss, B., Gunnerud, V. & Diez, M. D. (2009), ‘Introduction to Decomposition of Petroleum Production Optimization Problems’, *SPE Journal* .
- FPSO(2012), ‘FPSO’. Visited Mai 5, 2012.
URL:<http://fпсо.com/>
- Glæserud, P. & Syrdalen, J. A. (2009), Optimization of Petroleum Production under Uncertainty - Applied to the Troll C Field, Master’s thesis, Norwegian University of Science and Technology.
- Gunnerud, V. (2011), On Decomposition and Piecewise Linearization in Petroleum Production Optimization, PhD thesis, The Norwegian University of Science and Technology, Trondheim, Norway.
- Gunnerud, V. & Foss, B. (2010), ‘Oil Production Optimization - A Piecewise Linear Model, Solved with Two Decomposition Strategies’, *Computers & Chemical Engineering*. **34**, 1803–1812.
- Gunnerud, V., Nygreen, B., McKinnon, K. I. M. & Foss, B. (2011), ‘Oil Production Optimization Solved by Piecewise Linearization in a Branch & Price Framework’, *Submitted to Computers and Operations Research Journal* .
- Gunnerud, V., Torgnes, E. & Foss, B. (2010), ‘Parallel Dantzig-Wolfe Decomposition for Real-Time Optimization - Applied to a Complex Oil Field’, *Journal of Process Control* .
- Hagem, E. & Torgnes, E. (2009), Petroleum Production Planning Optimization - Applied to the StatoilHydro Offshore Oil and Gas Field Troll West, Master’s thesis, Norwegian University of Science and Technology.
- ISO 5024 (1999), Petroleum Liquids and Liquefied Petroleum Gases - Measurement - Standard Reference Condition, International Standard, International Organization for Standardization. Downloaded June 2, 2012.
URL:<http://www.standard.no/>
- Kanu, E., Mach, J. & Brown, K. (1981), ‘Economic Approach to Oil Production and Gas Allocation in Continuous Gas Lift.’, *Journal of Petroleum Technology* pp. 1887–1892.
- Kasana, H. & Kumar, K. (2004), *Introductory Operations Research*, first edn, Springer.
- Keha, A. B., de Farias Jr., I. & Nemhauser, G. (2004), ‘Models for Representing Piecewise Linear Cost Functions’, *Operations Research Letters* **38**, 44–48.
- Kirkpatrick, S., Jr., C. D. G. & Vecchi, M. (1983), ‘Optimization by Simulated Annealing’, *Science* **220**, 671–680.

- Kosmidis, V., Perkins, J. & Pistikopoulos, E. (2005), 'A Mixed Integer Optimization for the Well Scheduling Problem on Petroleum Fields', *Computers and Chemical Engineering Journal* **29**, 1523–1541.
- Lasdon, L. & Tabak, D. (1981), 'Optimization theory of large systems', *Mathematical Programming* **20**, 303.
- Løkketangen, A. (1995), Tabu Search as a Metaheuristic Guide for Combinatorial Optimization Problems, PhD thesis, University of Bergen.
- Lorenzatto, R. A., R. Juiniti, Gomez, J. A. T. & Martins, J. A. (2004), The Marlim Field development: Strategies and Challenges, Offshore Technology Conference.
- Lundgren, J., Rönnqvist, M. & Våbrand, P. (2010), *Optimization*, first edn, Studentlitteratur.
- Maijoni, A. & Hamouda, A. A. (2011), Effect of Gas Lift Gas Composition on Production Stability / Instability by Dynamic and Steady State Simulation for Continues Gas Lift Injection Mode, in 'SPE Asia Pacific Oil and Gas Conference and Exhibition, 20-22 September 2011, Jakarta, Indonesia', Society of Petroleum Engineers.
- Martin, G., Randhava, S. & McDowell, E. (1988), 'Computerized Container-Ship Load Planning: A Methodology and Evaluation', *Computers & Industrial Engineering* **14**, 429–440.
- Maugeri, L. (2007), *The Age of Oil*, Lyons Press.
- Mian, M. A. (1992), *Petroleum engineering, Handbook for the Practical Engineer*, PennWell Publishing Company, Oklahoma, United States of America.
- Misener, R., Gounaris, C. & Floudas, C. (2009), 'Global Optimization of Gas Lifting Operations: A Comparative Study of Piecewise Linear Formulations', *Industrial & Engineering Chemistry Research* **48**, 6098–6104.
- Moran, M. J. & Shapiro, H. N. (2006), *Fundamentals of Engineering Thermodynamics*, John Wiley & Sons Ltd.
- Nishikiori, N., Redner, R., Doty, D. & Schmidt, Z. (1989), An Improved Method for Gas Lift Allocation Optimization, in 'SPE Annual Technical Conference and Exhibition, 8-11 October 1989', Society of Petroleum Engineers.
- NTNU (2012), 'Solstorm cluster'. Visited March 1, 2012.
URL:<https://solstorm.iot.ntnu.no/wordpress/>
- Nygreen, B., Christiansen, M., Haugen, K., Björkvoll, T. & Kristiansen, O. (1998), 'Modeling Norwegian Petroleum Production and Transportation', *Annals of Operations Research* **82**, 251–267.
- Offshore Technology (2012), 'Marlim Field'. Visited January 30, 2012.
URL:<http://www.offshore-technology.com/projects/marlimpetro>

- OPEC (2011), 'World Oil Outlook'. Downloaded January 30, 2012.
URL:http://www.opec.org/opec_web/en/
- OPEC (2012), 'Opec basket price'. Downloaded June 1, 2012.
URL:http://www.opec.org/opec_web/en/data_graphs/40.htm
- Petrobras (2012), 'About Us'. Visited Mai 5, 2012.
URL:<http://www.petrobras.com>
- PetroStrategies Inc (2012), 'Oil and Gas Value Chains'. Visited March 6, 2012.
URL:http://www.petrostrategies.org/Learning_Center/oil_and_gas_value_chains.htm
- Pinto, A. C. C., Guedes, S. S., Bruhn, C. H. L., Gomes, J. A. T., de SÃ, A. N. & Netto, J. R. F. (2001), Marlim Complex Development: A Reservoir Engineering Overview, *in* 'SPE Latin American and Caribbean Petroleum Engineering Conference, 25-28 March 2001, Buenos Aires, Argentina', Society of Petroleum Engineers.
- Private conversation (January 3, 2012). Meeting with Alex Teixeira, R&D Coordinator and Production Engineer at Petrobras.
- RigZone (2012a), 'Fpso'. Visited Mai 5, 2012.
URL:http://www.rigzone.com/training/insight.asp?insight_id=299&c_id=12
- RigZone (2012b), 'Oil and Gas Conversion Calculator'. Visited January 30, 2012.
URL:<http://www.rigzone.com/calculator/default.asp#calc>
- Ronen, D. (1982), 'The Effect of Oil Price on the Optimal Speed of Ships', *The Journal of Operational Research Society* **33**, 1035–1040.
- Saputelli, L., Nikolaou, M. & Economides, M. (2005), 'Self-Learning Reservoir Management', *SPE Reservoir Evaluation & Engineering* **8**, 534–547.
- Schlumberger (2010), 'PIPESIM User Guide'.
- Schlumberger (2012b), 'Oilfield Glossary'. Visited February 12, 2012.
URL:<http://www.glossary.oilfield.slb.com/search.cfm>
- Silva, T. L., Codas, A. & Camponogara, E. (2012), A Computational Analysis of Convex Combination Models for Multidimensional Piecewise-Linear Approximation in Oil Production Optimization.
- Subsea Oil & Gas Directory (2012), 'Marlim Facts'. Visited January 5, 2012.
URL:<http://www.subsea.org/projects/listdetails.asp?ProjectID=60>
- Tebboth, J. (2001), A Computational Study of Dantzig-Wolfe Decomposition, PhD thesis, University of Buckingham.

- Torgnes, E., Gunnerud, V., Hagem, E., RÅnnqvist, M. & Foss, B. (2011), ‘Parallel dantzig-wolfe decomposition of petroleum production allocation problems’, *Journal of the Operational Research Society* **63**, 1–19.
- Towler, B. F. (1989), *Reservoir Engineering Aspects of Bottomhole Sampling of Saturated Oils for pvt Analysis*.
- Ulstein, N., B.Nygreen & Sagli, J. (2005), ‘Tactical Planning of Offshore Petroleum Production’, *European journal of Operational Research* **176**.
- Vanderbeck, F. (2006), ‘A Generic View of Dantzig-Wolfe Decomposition in Mixed Integer Programming’, *Operations Research Letters* **34**, 296–306.
- Vielma, J. P., Ahmed, S. & Nemhauser, G. (2010), ‘Mixed Integer Models for Nonseparable Piecewise Linear Optimization: Unifying Framework and Extensions’, *Operations Research* **58**, 303–315.
- Wallace, S. W. (2003), *Decision Making Under Uncertainty: The art of Modelling*, Molde University College, Molde, Norway. Unpublished course text.
- Wang, P. (2003), *Development and Applications of Production Optimization Techniques for Petroleum Fields*, PhD thesis, Stanford University.
- Wang, P., Litvak, M. & Aziz, K. (2002), Optimization of Production Operations in Petroleum Fields, in ‘SPE Annual Technical Conference and Exhibition, 29 September-2 October 2002, San Antonio, Texas’, Society of Petroleum Engineers.
- Williams, H. P. (1999), *Model Building in Mathematical Programming*, fourth edn, John Wiley and Sons, pp. 153–164.
- Williams, H. P. (2005), *Model Building in Mathematical Programming*, fourth edn, Wiley.
- Winker, P. & Gilli, M. (2004), ‘Applications of Optimization Heuristics to Estimation and Modelling Problems’, *Computational Statistics & Data Analysis* **47**, 211–223.

Appendix A

SOS2 formulation

In this appendix all the SOS2 formulations are given.

A.1 The well model

Using a SOS2 formulation to linearize Constraints (3.2), (3.3) and (3.4).

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{w, o\}$$

$$q_{mjg}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad m \in \mathcal{M}, j \in \mathcal{J}_m$$

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m$$

Sets and indices:

- \mathcal{K} - Set of breakpoints for values of p_{mj}^{BHP} , indexed by $k \in 1 \dots K$
- \mathcal{N} - Set of breakpoints for values of q_{mj}^I , indexed by $n \in 1 \dots N$

Variables and parameters:

- $P_{(mj)k}^{BHP}$ - Value of p_{mj}^{BHP} at breakpoint k .
 $Q_{(mj)n}^I$ - Value of q_{mj}^I at breakpoint n .
 $Q_{(mjp)k}$ - Value of q_{mjp}^W at breakpoints k .
 $P_{(mj)kn}^W$ - Value of p_{mj}^W at breakpoints k and n .
 $\lambda_{(mj)kn}$ - Weight of breakpoints k and n for manifold m and well j .
 $\eta_{(mj)k}^K$ - Sum of weights of breakpoints n , for breakpoints k , manifold m and well j .
 $\eta_{(mj)n}^N$ - Sum of weights of breakpoints k , for breakpoints n , manifold m and well j .

The new formulation is:

$$p_{mj}^{BHP} = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} P_{(mj)k}^{BHP} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.1})$$

$$q_{mj}^I = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} Q_{(mj)n}^I \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.2})$$

$$q_{mjp}^W = \sum_{k \in \mathcal{K}} Q_{(mjp)k} \eta_{(mj)k}^K \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{w, o\} \quad (\text{A.3})$$

$$q_{mjg}^W = \sum_{k \in \mathcal{K}} Q_{(mjg)k} \eta_{(mj)k}^K + q_{mj}^I \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.4})$$

$$p_{mj}^W = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} P_{(mj)kn}^W \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.5})$$

$$\sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} \lambda_{(mj)kn} = 1 \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.6})$$

$$\eta_{(mj)k}^K = \sum_{n \in \mathcal{N}} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K} \quad (\text{A.7})$$

$$\eta_{(mj)n}^N = \sum_{k \in \mathcal{K}} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m, n \in \mathcal{N} \quad (\text{A.8})$$

$$\lambda_{(mj)kn}, \eta_{(mj)k}^K, \eta_{(mj)n}^N \geq 0 \quad m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K}, n \in \mathcal{N} \quad (\text{A.9})$$

$$\eta_{(mj)k}^K \quad \text{is SOS2 for } k \quad m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K} \quad (\text{A.10})$$

$$\eta_{(mj)n}^N \quad \text{is SOS2 for } n \quad m \in \mathcal{M}, j \in \mathcal{J}_m, n \in \mathcal{N} \quad (\text{A.11})$$

A.2 Riser pipe pressure drop

Using the SOS2 formulation to linearize Constraints (3.8)

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L) \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m$$

Sets and indices:

- \mathcal{B} - Set of breakpoints for values of $q_{mlp}^L, p \in o$, indexed by $b \in 1 \dots B$
- \mathcal{D} - Set of breakpoints for values of $q_{mlp}^L, p \in g$, indexed by $d \in 1 \dots D$
- \mathcal{E} - Set of breakpoints for values of $q_{mlp}^L, p \in w$, indexed by $e \in 1 \dots E$

Variables and parameters:

- $Q_{(ml)b}^O$ - Value of q_{mlp}^L , $p \in o$ at breakpoint b .
 $Q_{(ml)d}^G$ - Value of q_{mlp}^L , $p \in g$ at breakpoint d .
 $Q_{(ml)e}^W$ - Value of q_{mlp}^L , $p \in w$ at breakpoint e .
 $F_{(ml)bde}^L$ - Value of f_{ml}^L at breakpoint b , d and e .
 $\sigma_{(ml)bde}$ - Weight of breakpoint b , d and e for subsea manifold m and pipeline l .
 $\theta_{(ml)b}^B$ - Sum of weights of breakpoints d and e , for breakpoints b , for subsea manifold m and pipeline l .
 $\theta_{(ml)d}^D$ - Sum of weights of breakpoints b and e , for breakpoints n , for subsea manifold m and pipeline l .
 $\theta_{(ml)e}^E$ - Sum of weights of breakpoints b and d , for breakpoints e , for subsea manifold m and pipeline l .

The new formulation is:

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)b}^O \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in o \quad (\text{A.12})$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)d}^G \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in g \quad (\text{A.13})$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)e}^W \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in w \quad (\text{A.14})$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} F_{(ml)bde}^L \sigma_{(ml)bde} = p_{ml}^M - p_{ml}^U \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (\text{A.15})$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} = 1 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (\text{A.16})$$

$$\theta_{(ml)b}^B = \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \quad (\text{A.17})$$

$$\theta_{(ml)d}^D = \sum_{b \in \mathcal{B}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \quad (\text{A.18})$$

$$\theta_{(ml)e}^E = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \quad (\text{A.19})$$

$$\sigma_{(ml)bde}, \theta_{(ml)b}^B, \theta_{(ml)d}^D, \theta_{(ml)e}^E \geq 0 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B}, d \in \mathcal{D}, e \in \mathcal{E} \quad (\text{A.20})$$

$$\theta_{(ml)b}^B \quad \text{is SOS2 for } b \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \quad (\text{A.21})$$

$$\theta_{(ml)d}^D \quad \text{is SOS2 for } d \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \quad (\text{A.22})$$

$$\theta_{(ml)e}^E \quad \text{is SOS2 for } e \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \quad (\text{A.23})$$

A.3 Gas lift pressure drop

Using the SOS2 formulation to linearize Constraints (3.9).

$$P^{I,C} - p_m^M = f_m^{I,S}(q_m^{I,S}) \quad m \in \mathcal{M}^S$$

Sets and indices:

\mathcal{R} - Set of breakpoints for values of $q_m^{I,S}$, indexed by $r \in 1 \dots R$

Variables and parameters:

$Q_{(m)r}^{I,S}$ - Value of $q_m^{I,S}$ at breakpoint r .
 $F_{(m)r}^{I,S}$ - Value of $f_m^{I,S}$ at breakpoint r .
 $\mu_{(m)r}$ - Weight of breakpoint r .

The new formulation is:

$$q_m^{I,S} = \sum_{r \in \mathcal{R}} Q_{mr}^{I,S} \mu_{(m)r} \quad m \in \mathcal{M}^S \quad (\text{A.24})$$

$$\sum_{r \in \mathcal{R}} F_{(m)r}^{I,S} \mu_{(m)r} = P^{I,C} - p_m^{I,M} \quad m \in \mathcal{M}^S \quad (\text{A.25})$$

$$\sum_{r \in \mathcal{R}} \mu_{(m)r} = 1 \quad m \in \mathcal{M}^S \quad (\text{A.26})$$

$$\mu_{(m)r} \geq 0 \quad m \in \mathcal{M}^S, r \in \mathcal{R} \quad (\text{A.27})$$

$$\mu_{(m)r} \quad \text{is SOS2 for } r \quad m \in \mathcal{M}^S, r \in \mathcal{R} \quad (\text{A.28})$$

Using the SOS2 formulation to linearize Constraints (3.10).

$$p_{mj}^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I) \quad m \in \mathcal{M}, j \in \mathcal{J}_m$$

Defining new indices, sets, parameters and variables.

Sets and indices:

\mathcal{A} - Set of breakpoints for values of q_{mj}^I , indexed by $a \in 1 \dots A$

Variables and parameters:

$Q_{(mj)a}^{I,MW}$ - Value of q_{mj}^I at breakpoint a .
 $F_{(mj)a}^{I,MW}$ - Value of f_{mj}^I at breakpoint a .
 $\gamma_{(mj)a}$ - Weight of breakpoint a for well j .

The new formulation is:

$$q_{mj}^I = \sum_{a \in \mathcal{A}} Q_{(mj)a}^I \gamma_{(mj)a} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.29})$$

$$\sum_{a \in \mathcal{A}} F_{(mj)a}^I \gamma_{(mj)a} = p_{mj}^{I,W} - p_{mj}^{BHP} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.30})$$

$$\sum_{a \in \mathcal{A}} \gamma_{(mj)a} = 1 \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{A.31})$$

$$\gamma_{(mj)a} \geq 0 \quad m \in \mathcal{M}, j \in \mathcal{J}_m, a \in \mathcal{A} \quad (\text{A.32})$$

$$\gamma_{(mj)a} \quad \text{is SOS2 for } a \quad m \in \mathcal{M}, j \in \mathcal{J}_m, a \in \mathcal{A} \quad (\text{A.33})$$

Appendix B

Implemented MILP - model

B.1 Sets and Indices

\mathcal{P}	-	Set of phases (g for gas, o for oil, w for water)
\mathcal{M}	-	Set of manifolds, indexed by $m \in 1 \dots M$
\mathcal{M}^S	-	Set of subsea manifolds, indexed by $m \in 1 \dots M^S$
\mathcal{M}^T	-	Set of topside manifolds, indexed by $m \in 1 \dots M^T$
\mathcal{J}_m	-	Set of wells connected to manifold m , indexed by $j \in 1 \dots J_m$
\mathcal{L}_m	-	Set of pipelines connected to subsea manifold m , indexed by $l \in 1 \dots L_m$
\mathcal{S}	-	Set of separators, indexed by $s \in 1 \dots S$
\mathcal{K}	-	Set of breakpoint for values of BHP , indexed by $k \in 1 \dots K$
\mathcal{N}	-	Set of breakpoint for values of lift gas to wells, indexed by $n \in 1 \dots N$
\mathcal{R}	-	Set of breakpoints for values of lift gas to manifolds , indexed by $r \in 1 \dots R$
\mathcal{A}	-	Set of breakpoints for values of lift gas to wells, indexed by $a \in 1 \dots A$
\mathcal{B}	-	Set of breakpoints for values of pipeline oil flow, indexed by $b \in 1 \dots B$
\mathcal{D}	-	Set of breakpoints for values of pipeline gas flow, indexed by $d \in 1 \dots D$
\mathcal{E}	-	Set of breakpoints for values of pipeline water flow, indexed by $e \in 1 \dots E$

B.2 Parameters

C_{ps}	-	Capacity limit on flow of phase p in separator s
G	-	Capacity limit on lift-gas compressor
P_s^S	-	Pressure at separator s
$P^{I,C}$	-	Pressure at the gas lift compressor
M_{mj}^1	-	The largest difference between manifold and wellhead pressure, for subsea manifold m , well j and pipeline l .
M_{mj}^2	-	The smallest difference between manifold and wellhead pressure in the gas lift system, for subsea manifold m and well j .
M_{mjlp}^3	-	The greatest possible flow from well j , to pipeline l , of phase p at manifold m .
$P_{(mj)k}^{BHP}$	-	Value of BHP at breakpoint k for manifold m and well j .
$Q_{(mj)n}^I$	-	Value of lift gas at breakpoint n for manifold m and well j .
$Q_{(mjp)k}$	-	Well flow at breakpoint k of phase p , for manifold m and well j .
$F_{(mj)kn}^W$	-	Value of wellhead pressure at breakpoints k and n for manifold m and well j .
$Q_{(m)r}^{I,S}$	-	Value of lift gas at breakpoint r for manifold m .
$F_{(m)r}^{I,S}$	-	Value of pressure drop from manifold m to the compressor, at breakpoint r .
$Q_{(mj)a}^{I,MW}$	-	Value of lift gas at breakpoint a for manifold m and well j .
$F_{(mj)a}^{I,MW}$	-	Value of pressure drop from BHP to wellhead for well j , connected to manifold m at breakpoint a .
$Q_{(ml)b}^O$	-	Value of oil flow in pipeline l , connected to manifold m , at breakpoint b .
$Q_{(ml)d}^G$	-	Value of gas flow in pipeline l , connected to manifold m , at breakpoint d .
$Q_{(ml)e}^W$	-	Value of water flow in pipeline l , connected to manifold m , at breakpoint e .
$F_{(ml)bde}^L$	-	Value of pressure drop over pipeline l , connected to manifold m , at breakpoint b , d and e .

B.3 Variables

$q_{m,jp}^W$	-	Well flow of phase p , from well j connected to manifold m
q_{mlp}^L	-	Pipe flow of phase p , in pipeline l connected to subsea manifold m
$q_{m,j}^I$	-	Flow of lift gas injected to well j connected to manifold m
$q_m^{I,S}$	-	Flow of lift gas injected to subsea manifold m
$p_{m,j}^W$	-	Wellhead pressure on well j connected to manifold m
$p_{m,j}^{I,W}$	-	Pressure in the gas lift pipe at the wellhead for well j connected to manifold m
p_{ml}^M	-	Pressure at subsea manifold m connected to pipeline l
$p_m^{I,M}$	-	Pressure in the gas injection pipe at subsea manifold m
p_{ml}^U	-	Pressure before routing to separators, from pipeline l connected to subsea manifold m
$p_{m,j}^{BHP}$	-	Bottom hole pressure for well j , connected to manifold m
$y_{mj}l$	-	Equals 1 if flow is routed from well j to pipeline l , connected to subsea manifold m , 0 otherwise
w_{mls}	-	Equals 1 if flow is routed to separator s , from pipeline l , connected to subsea manifold m , 0 otherwise
u_{mjs}	-	Equals 1 if flow is routed from well j , connected to topside manifold m , to separator s , 0 otherwise
x_{mj}	-	Equals 1 if lift gas is injected to well j , connected to manifold m , 0 otherwise
q_{mlps}^{LL}	-	Line flow of phase p , in pipeline l , at subsea manifold m to separator s
$q_{m,jps}^{WW}$	-	Well flow of phase p , from well j at topside manifold m to separator s
$q_{m,jlp}^{WM}$	-	Well flow of phase p , routed from well j to pipeline l at subsea manifold m
$\lambda_{(mj)kn}$	-	Weight of breakpoints k and n , for manifold m and well j .
$\eta_{(mj)k}^K$	-	Sum of weights of breakpoints n , for breakpoint k , manifold m and well j .
$\eta_{(mj)n}^N$	-	Sum of weights of breakpoints k , for breakpoint n , manifold m and well j .
$\mu_{(m)r}$	-	Weight of breakpoint r .
$\gamma_{(mj)a}$	-	Weight of breakpoint a for well j .
$\sigma_{(ml)bde}$	-	Weight of breakpoint b , d and e for subsea manifold m and pipeline l .
$\theta_{(ml)b}^B$	-	Sum of weights of breakpoints d and e , for breakpoints b , for subsea manifold m and pipeline l .
$\theta_{(ml)d}^D$	-	Sum of weights of breakpoints b and e , for breakpoints n , for subsea manifold m and pipeline l .
$\theta_{(ml)e}^E$	-	Sum of weights of breakpoints b and d , for breakpoints e , for subsea manifold m and pipeline l .

B.4 Objective function

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlo}^L + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjo}^W \quad (\text{B.1})$$

B.5 Constraints

The well model

$$p_{mj}^{BHP} = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} P_{(mj)k}^{BHP} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.2})$$

$$q_{mj}^I = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} Q_{(mj)n}^I \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.3})$$

$$q_{mjp}^W = \sum_{k \in \mathcal{K}} Q_{(mjp)k} \eta_{(mj)k}^K \quad m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \{w, o\} \quad (\text{B.4})$$

$$q_{mjg}^W = \sum_{k \in \mathcal{K}} Q_{(mjg)k} \eta_{(mj)k}^K + q_{mj}^I \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.5})$$

$$p_{mj}^W = \sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} P_{(mj)kn}^W \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.6})$$

$$\sum_{k \in \mathcal{K}} \sum_{n \in \mathcal{N}} \lambda_{(mj)kn} = 1 \quad m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K} \quad (\text{B.7})$$

$$\eta_{(mj)k}^K = \sum_{n \in \mathcal{N}} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.8})$$

$$\eta_{(mj)n}^N = \sum_{k \in \mathcal{K}} \lambda_{(mj)kn} \quad m \in \mathcal{M}, j \in \mathcal{J}_m, n \in \mathcal{N} \quad (\text{B.9})$$

Pressure relationships

$$p_{mj}^W \geq p_{ml}^M + M_{mj}^1(y_{mj} - 1) \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m \quad (\text{B.10})$$

$$p_{ml}^U \geq P_s^S w_{mls} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, s \in \mathcal{S} \quad (\text{B.11})$$

$$p_{mj}^W \geq P_s^S u_{mjs} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, s \in \mathcal{S} \quad (\text{B.12})$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)b}^O \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in o \quad (\text{B.13})$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)d}^G \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in g \quad (\text{B.14})$$

$$q_{mlp}^L = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} Q_{(ml)e}^W \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in w \quad (\text{B.15})$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} F_{(ml)bde}^L \sigma_{(ml)bde} = p_{ml}^M - p_{ml}^U \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (\text{B.16})$$

$$\sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} = 1 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (\text{B.17})$$

$$\theta_{(ml)b}^B = \sum_{d \in \mathcal{D}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \quad (\text{B.18})$$

$$\theta_{(ml)d}^D = \sum_{b \in \mathcal{B}} \sum_{e \in \mathcal{E}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \quad (\text{B.19})$$

$$\theta_{(ml)e}^E = \sum_{b \in \mathcal{B}} \sum_{d \in \mathcal{D}} \sigma_{(ml)bde} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \quad (\text{B.20})$$

$$q_m^{I,S} = \sum_{r \in \mathcal{R}} Q_{mr}^{I,S} \mu_{(m)r} \quad m \in \mathcal{M}^S \quad (\text{B.21})$$

$$\sum_{r \in \mathcal{R}} F_{(m)r}^{I,S} \mu_{(m)r} = P^{I,C} - p_m^{I,M} \quad m \in \mathcal{M}^S \quad (\text{B.22})$$

$$\sum_{r \in \mathcal{R}} \mu_{(m)r} = 1 \quad m \in \mathcal{M}^S \quad (\text{B.23})$$

$$q_{mj}^I = \sum_{a \in \mathcal{A}} Q_{(mj)a}^I \gamma_{(mj)a} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.24})$$

$$\sum_{a \in \mathcal{A}} F_{(mj)a}^I \gamma_{(mj)a} = p_{mj}^{I,W} - p_{mj}^{BHP} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.25})$$

$$\sum_{a \in \mathcal{A}} \gamma_{(mj)a} = 1 \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.26})$$

$$p_{mj}^{I,W} \leq P^{I,C} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (\text{B.27})$$

$$p_m^{I,M} \geq p_{mj}^{I,W} + M_{mj}^2(1 - x_{mj}) \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m \quad (\text{B.28})$$

Capacity constraints

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} q_{mlps}^{LL} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} q_{mjps}^{WW} \leq C_{sp} \quad s \in \mathcal{S}, p \in \{g, w\} \quad (\text{B.29})$$

$$q_{mlps}^{LL} \leq C_{sp} w_{msl} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (\text{B.30})$$

$$q_{m_jps}^{WW} \leq C_{sp} u_{msj} \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (\text{B.31})$$

$$\sum_{s \in \mathcal{S}} q_{m_jps}^{WW} = q_{m_jp}^W \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{B.32})$$

$$\sum_{s \in \mathcal{S}} q_{mlps}^{LL} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{B.33})$$

$$\sum_{m \in \mathcal{M}} \sum_{j \in \mathcal{J}_m} q_{mj}^I \leq G \quad (\text{B.34})$$

$$q_{mj}^I \leq G x_{mj} \quad m \in \mathcal{M}, j \in \mathcal{J}_m \quad (\text{B.35})$$

The mass balance

$$\sum_{j \in \mathcal{J}_m} q_{mjlp}^{WM} = q_{mlp}^L \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{B.36})$$

$$q_{mjlp}^{WM} \leq M_{mjlp}^3 y_{jml} \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{B.37})$$

$$\sum_{l \in \mathcal{L}_m} q_{mjlp}^{WM} = q_{mj}^W \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{B.38})$$

$$\sum_{j \in \mathcal{J}_m} q_{mj}^I = q_m^{I,S} \quad m \in \mathcal{M}^S \quad (\text{B.39})$$

Routing constraints

$$\sum_{l \in \mathcal{L}} y_{mjl} \leq 1 \quad m \in \mathcal{M}^S, j \in \mathcal{J}_m \quad (\text{B.40})$$

$$\sum_{s \in \mathcal{S}} w_{mls} \leq 1 \quad m \in \mathcal{M}^S, l \in \mathcal{L}_m \quad (\text{B.41})$$

$$\sum_{s \in \mathcal{S}} u_{mjs} \leq 1 \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (\text{B.42})$$

Nonnegativity requirements

$$\begin{aligned}
q_{mjp}^W &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, p \in \mathcal{P} \\
q_{mlp}^L &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, p \in \mathcal{P} \\
q_{mj}^I &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m \\
q_m^{I,S} &\geq 0 & m \in \mathcal{M}^S \\
p_{mj}^W &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m \\
p_{mj}^{I,W} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m \\
p_{ml}^M &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m \\
p_m^{I,M} &\geq 0 & m \in \mathcal{M}^S \\
p_{ml}^U &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m \\
p_{mj}^{BHP} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m \\
\lambda_{(mj)kn} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K}, n \in \mathcal{N} \\
\eta_{(mj)k}^K &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K} \\
\eta_{(mj)n}^N &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, n \in \mathcal{N} \\
\mu_{(m)r} &\geq 0 & m \in \mathcal{M}^S, r \in \mathcal{R} \\
\sigma_{(ml)bde} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, b \in \mathcal{B}, d \in \mathcal{D}, e \in \mathcal{E} \\
\theta_{(ml)b}^B &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \\
\theta_{(ml)d}^D &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \\
\theta_{(ml)e}^E &\geq 0 & m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E} \\
\gamma_{(mj)a} &\geq 0 & m \in \mathcal{M}, j \in \mathcal{J}_m, a \in \mathcal{A}
\end{aligned} \tag{B.43}$$

Binary requirements

$$\begin{array}{ll}
y_{m,jl} \in \{0,1\} & m \in \mathcal{M}^S, j \in \mathcal{J}_m, l \in \mathcal{L}_m \\
w_{m,ls} \in \{0,1\} & m \in \mathcal{M}^S, s \in \mathcal{S}, l \in \mathcal{L}_m \\
u_{m,js} \in \{0,1\} & m \in \mathcal{M}^T, s \in \mathcal{S}, j \in \mathcal{J}_m \\
x_{mj} \in \{0,1\} & m \in \mathcal{M}, j \in \mathcal{J}_m
\end{array} \tag{B.44}$$

SOS2 requirements

$$\begin{array}{lll}
\eta_{(mj)k}^K & \text{is SOS2 for } k & m \in \mathcal{M}, j \in \mathcal{J}_m, k \in \mathcal{K} \\
\eta_{(mj)n}^N & \text{is SOS2 for } n & m \in \mathcal{M}, j \in \mathcal{J}_m, n \in \mathcal{N} \\
\mu_{(m)r} & \text{is SOS2 for } r & m \in \mathcal{M}^S, r \in \mathcal{R} \\
\gamma_{(mj)a} & \text{is SOS2 for } a & m \in \mathcal{M}, j \in \mathcal{J}_m, a \in \mathcal{A} \\
\theta_{(ml)b}^B & \text{is SOS2 for } b & m \in \mathcal{M}^S, l \in \mathcal{L}_m, b \in \mathcal{B} \\
\theta_{(ml)d}^D & \text{is SOS2 for } d & m \in \mathcal{M}^S, l \in \mathcal{L}_m, d \in \mathcal{D} \\
\theta_{(ml)e}^E & \text{is SOS2 for } e & m \in \mathcal{M}^S, l \in \mathcal{L}_m, e \in \mathcal{E}
\end{array} \tag{B.45}$$

Appendix C

Dantzig- Wolfe decomposed model

In this appendix the total decomposed models are given. As a simplification, the non-linear relationships linearized with SOS2 in Section 5.3, is formulated as the mathematical expressions introduced in Chapter 3.

C.1 Decomposing on manifolds

C.1.1 Restricted master problem

Objective function:

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^S} Q_{mlsh}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^T} Q_{mjsh}^{WW} \omega_{mh}$$

Constraints:

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{h \in \mathcal{H}^S} Q_{mlsh}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjsh}^{WW} \omega_{mh} \leq C_{ps} \quad p \in \{g, w\}, s \in \mathcal{S} \quad (\text{C.1})$$

$$\sum_{m \in \mathcal{M}^S} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^S} Q_{mjsh}^I \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjsh}^I \omega_{mh} \leq G \quad (\text{C.2})$$

$$\sum_{h \in \mathcal{H}^S} \phi_{mh} \leq 1 \quad m \in \mathcal{M}^S \quad (\text{C.3})$$

$$\sum_{h \in \mathcal{H}^T} \omega_{mh} \leq 1 \quad m \in \mathcal{M}^T \quad (\text{C.4})$$

$$\phi_{mh} \geq 0 \quad m \in \mathcal{M}^S, h \in \mathcal{H}^S \quad (\text{C.5})$$

$$\omega_{mh} \geq 0 \quad m \in \mathcal{M}^T, h \in \mathcal{H}^T \quad (\text{C.6})$$

C.1.2 Subproblems

Subsea manifolds subproblem:

$$\max w_m^S = \sum_{l \in \mathcal{L}_m} (q_{mlo}^L - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mlgs}^{LL} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mlws}^{LL}) - \sum_{j \in \mathcal{J}_m} \pi_j^I q_{mj}^I$$

Well model:

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad j \in \mathcal{J}_m, p \in \{w, o\} \quad (\text{C.7})$$

$$q_{mjg}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad j \in \mathcal{J}_m \quad (\text{C.8})$$

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad j \in \mathcal{J}_m \quad (\text{C.9})$$

Pressure balance:

$$p_{mj}^W \geq p_{ml}^M + M_{mj}(y_{mjl} - 1) \quad j \in \mathcal{J}_m, l \in \mathcal{L}_m \quad (\text{C.10})$$

$$p_{ml}^U \geq P_s^S w_{mls} \quad l \in \mathcal{L}_m, s \in \mathcal{S} \quad (\text{C.11})$$

Pressure balance, gas lift:

$$P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,S}) \quad j \in \mathcal{J}_m \quad (\text{C.12})$$

$$p_{mj}^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I) \quad j \in \mathcal{J}_m \quad (\text{C.13})$$

$$p_m^{I,M} \geq p_{mj}^{I,W} + M_{mj}(1 - x_{mj}) \quad j \in \mathcal{J}_m \quad (\text{C.14})$$

Gas lift capacity:

$$q_{mj}^I \leq Gx_{mj} \quad j \in \mathcal{J}_m \quad (\text{C.15})$$

Mass balance, gas lift:

$$\sum_{j \in \mathcal{J}_m} q_{mj}^I = q_m^{I,S} \quad (\text{C.16})$$

Pressure drop, pipelines:

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L) \quad l \in \mathcal{L}_m \quad (\text{C.17})$$

Separator capacity:

$$q_{mlps}^{LL} \leq C_{sp} w_{msl} \quad l \in \mathcal{L}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (\text{C.18})$$

Mass balances:

$$\sum_{s \in \mathcal{S}} q_{mlps}^{LL} = q_{mlp}^{LL} \quad l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{C.19})$$

$$\sum_{j \in \mathcal{J}_m} q_{mjlp}^{WM} = q_{mlp}^L \quad l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{C.20})$$

$$q_{mjlp}^{WM} \leq M_{mjlp} y_{jml} \quad l \in \mathcal{L}_m, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{C.21})$$

$$\sum_{l \in \mathcal{L}_m} q_{mjlp}^{WM} = q_{mj}^W \quad j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{C.22})$$

Routing:

$$\sum_{l \in \mathcal{L}} y_{mjl} \leq 1 \quad j \in \mathcal{J}_m \quad (\text{C.23})$$

$$\sum_{s \in \mathcal{S}} w_{mjs} \leq 1 \quad l \in \mathcal{L}_m \quad (\text{C.24})$$

Topside manifolds subproblem:

$$\max w_m^T = \sum_{j \in \mathcal{J}_m} (q_{mjo}^W - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mjgs}^{WW} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mjws}^{WW} - \pi^I q_{mj}^I)$$

Well model:

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad j \in \mathcal{J}_m, p \in \{w, o\} \quad (\text{C.25})$$

$$q_{mjg}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad j \in \mathcal{J}_m \quad (\text{C.26})$$

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad j \in \mathcal{J}_m \quad (\text{C.27})$$

$$p_{mj}^W \geq P_s^S u_{mjs} \quad j \in \mathcal{J}_m, s \in \mathcal{S} \quad (\text{C.28})$$

Pressure balance, gas lift:

$$p_{mj}^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I) \quad j \in \mathcal{J}_m \quad (\text{C.29})$$

$$p_{mj}^{I,W} \leq P^{I,C} \quad j \in \mathcal{J}_m \quad (\text{C.30})$$

Gas lift capacity:

$$q_{mj}^I \leq Gx_{mj} \quad j \in \mathcal{J}_m \quad (\text{C.31})$$

Separator capacity:

$$q_{mjps}^{WW} \leq C_{sp} u_{msj} \quad j \in \mathcal{J}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (\text{C.32})$$

Mass balance:

$$\sum_{s \in \mathcal{S}} q_{mjps}^{WW} = q_{mjp}^W \quad j \in \mathcal{J}_m, p \in P \quad (\text{C.33})$$

Routing to separator:

$$\sum_{s \in \mathcal{S}} u_{mjs} \leq 1 \quad j \in \mathcal{J}_m \quad (\text{C.34})$$

C.2 Decomposing on topside wells and subsea manifolds

C.2.1 Restricted master problem

Objective function:

$$\max Z = \sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^S} Q_{mlsh}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{s \in \mathcal{S}} \sum_{h \in \mathcal{H}^T} Q_{mjsh}^{WW} \omega_{mjh}$$

Constraints:

$$\sum_{m \in \mathcal{M}^S} \sum_{l \in \mathcal{L}_m} \sum_{h \in \mathcal{H}^S} Q_{mlps}^{LL} \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjps}^{WW} \omega_{mjh} \leq C_{ps} \quad p \in \{g, w\}, s \in \mathcal{S} \quad (\text{C.35})$$

$$\sum_{m \in \mathcal{M}^S} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^S} Q_{mjh}^I \phi_{mh} + \sum_{m \in \mathcal{M}^T} \sum_{j \in \mathcal{J}_m} \sum_{h \in \mathcal{H}^T} Q_{mjh}^I \omega_{mjh} \leq G \quad (\text{C.36})$$

$$\sum_{h \in \mathcal{H}^S} \phi_{mh} \leq 1 \quad m \in \mathcal{M}^S \quad (\text{C.37})$$

$$\sum_{h \in \mathcal{H}^T} \omega_{mjh} \leq 1 \quad m \in \mathcal{M}^T, j \in \mathcal{J}_m \quad (\text{C.38})$$

$$\phi_{mh} \geq 0 \quad m \in \mathcal{M}, h \in \mathcal{H}^S \quad (\text{C.39})$$

$$\omega_{mjh} \geq 0 \quad m \in \mathcal{M}, j \in \mathcal{J}_m, h \in \mathcal{H}^T \quad (\text{C.40})$$

C.2.2 Subproblems

Subsea manifolds subproblem:

$$\max w_m^S = \sum_{l \in \mathcal{L}_m} (q_{mlo}^L - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mlgs}^{LL} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mlws}^{LL}) - \sum_{j \in \mathcal{J}_m} \pi^I q_{mj}^I$$

Well model:

$$q_{mjp}^W = f_{mjp}^W(p_{mj}^{BHP}) \quad j \in \mathcal{J}_m, p \in \{w, o\} \quad (C.41)$$

$$q_{mjg}^W = f_{mjg}^W(p_{mj}^{BHP}) + q_{mj}^I \quad j \in \mathcal{J}_m \quad (C.42)$$

$$p_{mj}^W = f_{mj}^W(p_{mj}^{BHP}, q_{mj}^I) \quad j \in \mathcal{J}_m \quad (C.43)$$

Pressure balance:

$$p_{mj}^W \geq p_{ml}^M + M_{mj}(y_{mjl} - 1) \quad j \in \mathcal{J}_m, l \in \mathcal{L}_m \quad (C.44)$$

$$p_{ml}^U \geq P_s^S w_{mls} \quad l \in \mathcal{L}_m, s \in \mathcal{S} \quad (C.45)$$

Pressure balance, gas lift:

$$P^{I,C} - p_m^{I,M} = f_m^{I,S}(q_m^{I,S}) \quad j \in \mathcal{J}_m \quad (C.46)$$

$$p_{mj}^{I,W} - p_{mj}^{BHP} = f_{mj}^I(q_{mj}^I) \quad j \in \mathcal{J}_m \quad (C.47)$$

$$p_m^{I,M} \geq p_{mj}^{I,W} + M_{mj}(1 - x_{mj}) \quad j \in \mathcal{J}_m \quad (C.48)$$

Gas lift capacity:

$$q_{mj}^I \leq Gx_{mj} \quad j \in \mathcal{J}_m \quad (C.49)$$

Mass balance, gas lift:

$$\sum_{j \in \mathcal{J}_m} q_{mj}^I = q_m^{I,S} \quad (\text{C.50})$$

Pressure drop, pipelines:

$$p_{ml}^M - p_{ml}^U = f_{ml}^L(q_{mlo}^L, q_{mlg}^L, q_{mlw}^L) \quad l \in \mathcal{L}_m \quad (\text{C.51})$$

Separator capacity:

$$q_{mlps}^{LL} \leq C_{sp} w_{msl} \quad l \in \mathcal{L}_m, p \in \{g, w\}, S \in \mathcal{S} \quad (\text{C.52})$$

Mass balances:

$$\sum_{s \in \mathcal{S}} q_{mlps}^{LL} = q_{mlp}^{LL} \quad l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{C.53})$$

$$\sum_{j \in \mathcal{J}_m} q_{mjlp}^{WM} = q_{mlp}^L \quad l \in \mathcal{L}_m, p \in \mathcal{P} \quad (\text{C.54})$$

$$q_{mjlp}^{WM} \leq M_{mjlp} y_{jml} \quad l \in \mathcal{L}_m, j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{C.55})$$

$$\sum_{l \in \mathcal{L}_m} q_{mjlp}^{WM} = q_{mj}^W \quad j \in \mathcal{J}_m, p \in \mathcal{P} \quad (\text{C.56})$$

Routing:

$$\sum_{l \in \mathcal{L}} y_{mjl} \leq 1 \quad j \in \mathcal{J}_m \quad (\text{C.57})$$

$$\sum_{s \in \mathcal{S}} w_{mjs} \leq 1 \quad l \in \mathcal{L}_m \quad (\text{C.58})$$

Topside manifold subproblem:

$$\max w_{mj}^T = q_{mjo}^W - \sum_{s \in \mathcal{S}} \pi_{sg}^G q_{mjgs}^{WW} - \sum_{s \in \mathcal{S}} \pi_{sw}^W q_{mjws}^{WW} - \pi^I q_{mj}^I$$

Well model:

$$q_{mjp}^W = f_{mjp}^W(p_{m_j}^{BHP}) \quad p \in \{w, o\} \quad (C.59)$$

$$q_{mjg}^W = f_{mjg}^W(p_{m_j}^{BHP}) + q_{m_j}^I \quad (C.60)$$

$$p_{m_j}^W = f_{m_j}^W(p_{m_j}^{BHP}, q_{m_j}^I) \quad (C.61)$$

$$p_{m_j}^W \geq P_s^S u_{mjs} \quad s \in \mathcal{S} \quad (C.62)$$

Pressure balance, gas lift:

$$p_{m_j}^{I,W} - p_{m_j}^{BHP} = f_{m_j}^I(q_{m_j}^I) \quad (C.63)$$

$$p_{m_j}^{I,W} \leq P^{I,C} \quad (C.64)$$

Gas lift capacity:

$$q_{m_j}^I \leq Gx_{m_j} \quad (C.65)$$

Separator capacity:

$$q_{m_jps}^{WW} \leq C_{sp} u_{msj} \quad p \in \{g, w\}, S \in \mathcal{S} \quad (C.66)$$

Mass balance:

$$\sum_{s \in \mathcal{S}} q_{m_jps}^{WW} = q_{mjp}^W \quad p \in P \quad (C.67)$$

Routing to separator:

$$\sum_{s \in \mathcal{S}} u_{mjs} \leq 1 \quad (C.68)$$

Appendix D

Electronical documentation

The attached CD contains the following documentation:

1. PDF version of this project
2. Datafiles and Xpress-MP Files
 - Original MILP model
 - Decomposed model (DWD-Man)
 - Decomposed model (DWD-ManWell)
 - Heuristics