
Models and Solution Methods for the Pooling Problem

Mohammed Alfaki



Dissertation for the degree of Philosophiae Doctor (PhD)

UNIVERSITY OF BERGEN
Department of Informatics
March 2012

ISBN: 978-82-308-2051-3
University of Bergen, Norway
printed version March 29, 2012

*To my parents, wife,
brothers, and sisters*

Scientific Environment

The research done in this thesis has been accomplished while I was a PhD-student in the Optimization group at the Department of Informatics, University of Bergen, where I have also been associated with the ICT Research school.

This thesis is part of the RAMONA project, which concerns the regularity and uncertainty analysis and management for the Norwegian gas processing and transportation system. This project is financed by the Norwegian Research Council (80%), StatoilHydro (10%), and Gassco (10%). It is a joint project between three Norwegian universities (University of Stavanger, Norwegian University of Science and Technology, and University of Bergen). One of the sub-projects, devoted to the development of flow allocation models and algorithms under complex constraints, was assigned to University of Bergen, and this thesis is part of this sub-project.

UNIVERSITY OF BERGEN
Department of Informatics



ICT

**Research School in
Information and Communication Technology**

Acknowledgements

First and foremost, all praise is due to Allah, who gave me the ability to complete this work. While completing my PhD thesis, I have been fortunate to work with many skilled and encouraging people. It gives me great pleasure to acknowledge those many people who have influenced my thinking in some way and contributed to my often inadequate knowledge. I have invariably learned from all my supervisors, colleagues and friends.

I am very grateful to my supervisor, Professor Dag Haugland, for his excellent guidance and the time he afforded me to comment on my ideas and writing. My knowledge has benefited greatly from his expertise, enthusiasm and encouragement. I was fortunate to work under his guidance during my MSc and PhD studies. I also wish to express my warmest gratitude to my co-supervisor, Professor Trond Steihaug, for his numerous invaluable suggestions during this work. Special thanks go to my co-authors, Dr. Lennart Frimannslund and Dr. Mohamed El Ghami, for sharing valuable information and the discussions that contributed in various ways to this work. Warm thanks go to my father-in-law, Dr. Tagelsir Mohammed Suleiman, for reading the thesis and making valuable suggestions regarding the manuscript. I would also like to thank the former members of the Optimization group, Dr. Conrado Borraz-Sánchez and Dr. Geir Gundersen, with whom I have shared offices; they have been wonderful friends. My sincere gratitude goes to Professor Emeritus Sverre Storøy and Dr. Joanna Bauer. My thanks also go to the administration staff at the Department of Informatics, especially to the former Heads of Administration, Signe Knappskog and Ida Holen, for their help in many practicalities. I would like to extend my gratitude to Marta Lopez, Tor Bastiansen, Liljan Myhr and Steinar Heldal. I am enormously grateful to the Norwegian Research Council, Gassco, and Statoil for funding this thesis through the RAMONA project.

Last but not least, on a more personal note, I would like to thank my sisters,

Acknowledgements

Tahani and Mona, and my brothers, Mubark, Bashir and Abdallah, for their care, constant support and prayers. My thanks extended also to all my friends in Bergen for making life easier and more enjoyable. Above all, I would like to thank my parents, Altoma and Ali, and my wife, Sara, without whose love, support and sacrifices I could not have succeeded; my thesis is dedicated to all my family members.

UNIVERSITY OF BERGEN
Department of Informatics

Models and Solution Methods for the Pooling Problem

Dissertation for the degree of Philosophiae Doctor (PhD)

Abstract

Pipeline transportation of natural gas is largely affected by restrictions regarding gas quality imposed by the market and the actual quality of the gas produced at sources. From the sources, gas flow streams of unequal compositions are mixed in intermediate tanks (pools) and blended again in terminal points. At the pools and the terminals, the quality of the mixture is given as volume-weighted average of the qualities of each mixed gas flow stream. The optimization problem of allocating flow in pipeline transportation networks at minimum cost is referred to as the pooling problem. Such problem is frequently encountered not only in gas transportation planning, but also in the process industries such as petrochemicals.

The pooling problem is a well-studied global optimization problem, which is formulated as a nonconvex (bilinear) problem, and consequently the problem can possibly have many local optima. Despite the strong \mathcal{NP} -hardness of the problem, which is proved formally in this thesis, much progress in solving small to moderate size instances to global optimality has recently been made by use of strong formulations. However, the literature offers few approaches to approximation algorithms and other inexact methods dedicated for large-scale instances. The main contribution of this thesis is the development of strong formulations and efficient solution methods for the pooling problem. In this thesis, we develop a new formulation that proves to be stronger than other formulations based on proportion variables for the standard pooling problem. For the generalized case, we propose a multi-commodity flow formulation, and prove its strength over formulations from the literature.

Regarding the solution methods, the thesis proposes three solution approaches to tackle the problem. In the first methodology, we discuss solving a simplified

Abstract

version of the standard pooling problem using a solution strategy that based on a sequence of semidefinite programming relaxations. The second approach is based on discretization method in which the pooling problem is approximated by a mixed-integer programming problem. Finally, we give a greedy construction method especially designed to find good feasible solutions for large-scale instances.

March 2012 – Bergen, Norway,
Mohammed Alfaki

Contents

Abstract	ix
List of publications	xv
I Overview	1
1 Introduction	3
1.1 Background	3
1.2 Optimization – in brief	8
1.3 Problem statement	12
1.3.1 The standard pooling problem	12
1.3.2 The blending problem	13
1.3.3 The generalized pooling problem	14
1.4 Structure of the thesis	15
2 Model formulations for pooling problems	17
2.1 Introduction	17
2.2 Formulations for the standard pooling problem	19
2.2.1 The quality formulation	19
2.2.2 Proportion formulations	20
2.3 Formulations for the generalized pooling problem	23
2.3.1 A hybrid formulation	23
2.3.2 A multi-commodity flow formulation	25
2.4 Formulations for extensions	26

3	Solution methods	29
3.1	Inexact/heuristic techniques	29
3.1.1	Improvement heuristics	29
3.1.2	A construction heuristic	31
3.1.3	Successive linear programming	31
3.1.4	Benders decomposition	32
3.1.5	Discretization approaches	32
3.2	Exact solution techniques	33
3.2.1	Branch-and-bound algorithms	33
3.2.2	Semidefinite programming relaxations	36
3.3	Summary	37
4	Summary of papers	39
4.1	Strong formulations	39
4.1.1	The standard pooling problem (Paper A)	39
4.1.2	The generalized pooling problem (Paper B)	40
4.2	New solution methods	41
4.2.1	LMI relaxations (Paper C)	41
4.2.2	A discretization approach (Paper D)	42
4.2.3	A construction heuristic method (Paper E)	43
5	Conclusion and future work	45
5.1	Conclusion	45
5.2	Future work	46
	Bibliography	49
II	Scientific contributions	57
	Paper A Strong formulations for the pooling problem	59
	Paper B A multi-commodity flow formulation for the generalized pooling problem	93

Paper C	Solving the pooling problem with LMI relaxations	123
Paper D	Comparison of discrete and continuous models for the pooling problem	145
Paper E	Computing feasible solutions to the pooling problem	163

List of publications

This thesis presents the results of my research performed at Department of Informatics, University of Bergen, for the degree of Philosophiae Doctor. The contributions of this thesis are based on the following publications:

- A. Alfaki, M. and Haugland, D. (2012). **Strong formulations for the pooling problem.** In: *Journal of Global Optimization*, doi: [10.1007/s10898-012-9875-6](https://doi.org/10.1007/s10898-012-9875-6).
- B. Alfaki, M. and Haugland, D. (2012). **A multi-commodity flow formulation for the generalized pooling problem.** In: *Journal of Global Optimization*, doi: [10.1007/s10898-012-9890-7](https://doi.org/10.1007/s10898-012-9890-7).
- C. Frimannslund, L., El Ghami, M., Alfaki, M. and Haugland, D. (2010). **Solving the pooling problem with LMI relaxations.** In: S. Cafieri, B. G.-Tóth, E. Hendrix, L. Liberti and F. Messine (Eds.), *Proceedings of the Toulouse Global Optimization Workshop* (pp. 51–54).
- D. Alfaki, M. and Haugland, D. (2011). **Comparison of discrete and continuous models for the pooling problem.** In: A. Caprara and S. Kontogiannis (Eds.), *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (Vol. 20, pp. 112–121). OpenAccess Series in Informatics (OASICS). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, doi: [10.4230/OASIcs.ATMOS.2011.112](https://doi.org/10.4230/OASIcs.ATMOS.2011.112).
- E. Alfaki, M. and Haugland, D. (2011). **Computing feasible solutions to the pooling problem.** Submitted to: *Annals of Operations Research*.

Part I

Overview

Chapter 1

Introduction

Mathematical programming, or optimization, models and techniques have been extensively used to optimize the entire supply chain in the petroleum industry since the mid of the last century. The downstream part of the petroleum industry, which is concerned with turning crude petroleum into finished usable products that are delivered to consumers, is more concerned with mathematical programming than other parts of the industry. In the downstream part, the decision makers examine every possibility to make good decisions regarding a complex system composed of several operations. The operations include: crude purchasing, processing the crude into a variety of products, and transporting the products to the consumers. The primary goal of this business is to minimize the operational costs and maximize the profit, while satisfying customers needs. High economic value and operability benefits, associated with integrating mathematical programming software in the petroleum industry, are the driving forces for theoretical and commercial development of the field.

1.1 Background

Natural gas, which is a subcategory of petroleum, is one of the most widely used energy sources in the world. Its usage has been increasing in recent years. This is due to the facts that it is an efficient fossil fuel, with low cost and relatively low pollutant emissions to the environment. Geologists and chemists agree that petroleum was formed when the remains of organisms that accumulated in the past are compressed under the earth at very high pressure for millions of years.

Section 1.1. Background

Since petroleum derivatives are depleted faster than its formation, natural gas is generally considered a nonrenewable source of energy.

Natural gas is extracted from deep rock reservoirs in the Earth's crust, where it exists under high pressure, either alone or associated with heavier hydrocarbons and water. It is produced from the reservoir in the same manner as the crude oil. In general, the gas associated with heavier hydrocarbons and water is found in rock reservoirs at depths ranging between 1000 and 6000 meters, while deeper rock reservoirs produce mainly dry gas. Natural gas is mainly produced from three types of reservoirs:

1. Crude oil reservoir wells, where the gas is produced as a by-product and it is referred to as *associated gas*.
2. Dry gas wells, which typically do not contain any hydrocarbon liquids. The produced gas is called *non-associated gas*.
3. Condensate wells, in which the extracted gas is also non-associated, but contains hydrocarbon liquids. This type of gas is referred to as *unconventional gas* or wet gas.

As a consequence of natural gas market developments and the advances in production technology, the natural gas industry has begun to explore for more challenging condensate reservoirs, which have a high percentage of impurities. Examples of such condensate reservoirs are tight gas which exists in low permeability¹ rock formations, shale, coalbed methane, natural gas hydrates and deep gas. For more detailed treatments of natural gas geological formation and characteristic, the reader is referred to the survey by [Mokhatab et al. \(2006\)](#).

Due to the distinct characteristics of each well, all natural gas produced is not of the same quality. Even gas produced from a particular well may over-time vary in component percentages. Raw natural gas is mainly composed of methane (CH_4) with varying amounts of heavier gaseous hydrocarbons (e.g. ethane (C_2H_6), propane (C_3H_8), butane (C_4H_{10})), acid gases (e.g. carbon dioxide (CO_2), hydrogen sulfide (H_2S)), other gases (e.g. nitrogen (N_2), helium (He)), liquid hydrocarbons, water vapor, mercury, and radioactive gas. Generally, natural gas is classified into two main categories: If the natural gas contains small

¹Permeability is the measure of the ability of a material to transmit fluids.

amounts of H₂S and CO₂ it is commonly referred to as *sweet* gas, and otherwise it is called *sour* gas. Table 1.1 shows typical chemical components in mole percentages in natural gas extracted from three different sources (Wardzinski et al., 2004). In addition to the chemical components, the gas is also described by its physical properties such as the heating value, which is the amount of energy in mega joule (MJ) per cubic meter, and the Wobbe index², which is used to compare the combustion energy output.

Table 1.1: Example of natural gas composition in mole percent.

Composition	Associated	Non-associated	Unconventional
Carbon-Dioxide	0.63	–	–
Nitrogen	3.73	1.25	0.53
Hydrogen-Sulfide	0.57	–	–
Methane	64.48	91.01	94.87
Ethane	11.98	4.88	2.89
Propane	8.75	1.69	0.92
Iso-Butane	0.93	0.14	0.31
n-Butane	2.91	0.52	0.22
iso-Pentane	0.54	0.09	0.09
n-Pentane	0.80	0.18	0.06
Hexanes	0.37	0.13	0.05
Heptanes-plus	0.31	0.11	0.06

Since natural gas is available with relatively affordable prices and low pollutant emissions, it is used as a source of energy as well as raw material in manufacturing. It has been used for heating space and water, air conditioning and cooking, especially in seasonal months. At the industrial level, it is used for example in power generation, hydrogen production, vehicles, and fertilizers. In addition, natural gas is an important raw material in manufacturing of fabrics, glass, and other products. Figure 1.1 shows the natural gas global demand by sector.

In order for the consumers to use the natural gas safely and efficiently in their equipment, it must be within specified quality (we simply refer to the relative content of a component or a physical property as quality) range. Otherwise, serious problems may occur, such as the flame lifting when the Wobbe index is not in its correct range. Table 1.2 shows examples of natural quality ranges

²The Wobble index is measured in Btu (British thermal unit of energy), 1 Btu \approx 1055 joules.

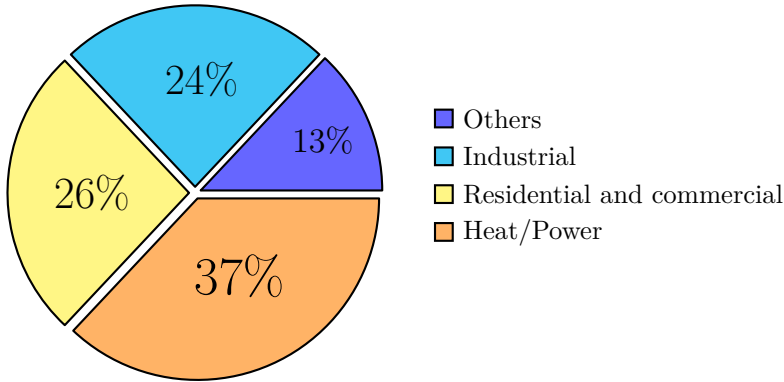


Figure 1.1: Global natural gas demand. The figure is based on data from (Simmons et al., 2006).

in North America and Europe³. The quality standards can vary greatly from country to country, and even from consumer to consumer in the same country.

Table 1.2: Typical examples of quality requirements demanded by the markets in North America and Europe (Hubbard, 2009). The table is divided into two groups of rows: composition and physical property requirements.

Quality	North America	Europe
CO ₂ concentration	1–3 mol%	2–3 mol%
N ₂	2–3 mol%*	2–3 mol%*
Total inerts	3–5 mol%*	NA
H ₂ S	0.25–1.0 grain/100 scf	5–7 mg/Nm ³
Total S	0.5–20 grain/100 scf	120–150 mg/Nm ³
Mercaptans	0.25–1.0 grain/100 scf*	6–15 mg/Nm ³
Oxygen	10–2000 ppm (mol)	1000–5000 ppm (mol)
Water dew-point	4–7 lbm H ₂ O/MMscf of gas	-10 to -12°C at 7000 kPa
Hydrocarbon dew-point	14–40°F at specified P	0 to -5°C at P < 7000 kPa
Heating value	950–1200 Btu/scf	40–46 MJ/Nm ³
Wobbe index	NA	51–56 MJ/Nm ³

NA = not applicable
 * = often not specified

³1 grain = 64.79891 milligrams (mg), scf ≡ standard cubic feet, Nm³ ≡ normal cubic meter, ppm ≡ parts-per-million, the pound-mass (lbm) is a unit of mass, 1 MMscf = 10⁶ scf, kPa ≡ kilo-pascals is a unit for pressure (P).

As soon as the gas is extracted from the wellheads, it goes through a number of processing operations to remove undesired impurities such as water, liquid hydrocarbons and sulfur. Removing impurities through processing operations is an important step in the natural gas journey to the end consumer. Processing operations can be divided into two major processes: *separation* and *blending*. In the separation process, natural gas is converted into intermediate products known as *pipeline-quality* dry gas, where the compositions of the product are given in fixed proportions of the original ones. In this process, natural gas can go through up to four processing facilities depending upon the level of impurities present in the gas. These processing facilities apply chemical and physical technologies. The main purpose of the separation process is to purify the natural gas to facilitate its transportation through the pipelines or the ship vessels. For more information about the separation process the reader is directed to (Guo and Ghaleb, 2005).

Blending is the physical mixture of different flow gas streams, which takes place in the so-called *pools* (tanks, vessels or through injection), where the quality of this mixture is given as volume-weighted average of the qualities of each mixed gas flow stream. In the blending operation, as opposed to the separation process, the quality of the final product depends on *both* the volume and the quality of the entering natural gas stream. In comparison to the separation process, blending is cheaper, and therefore it can be used to reduce the level of impurities prior to the separation process. If the natural gas is sweet, it can be transported directly to the end consumers, but it may still require further blending to match the consumer quality requirements.

Transporting the natural gas from the production sources to the consumers is a complex process on a transportation network of large number of pipelines and processing units. In this process, a blend of gases from different sources is formed in order to meet the end consumers' requirements, while taking into account the network configuration and its capacity. Optimization models are used to efficiently and effectively to allocate natural gas flow in this pipeline transportation network. Optimal flow allocations require that the constraints imposed by the system are modeled at an appropriate level of detail, and that corresponding solution procedures are available.

1.2 Optimization – in brief

This section briefly introduces some mathematical optimization concepts that are useful in this thesis. For a comprehensive treatment of this subject, the interested reader is referred to [Boyd and Vandenberghe \(2004\)](#) for convex optimization, [Nocedal and Wright \(2000\)](#) and [Floudas \(2000\)](#) or [Hendrix and G.-Tóth \(2010\)](#) for nonconvex optimization, and [Wolsey \(1998\)](#) for integer optimization.

The origin of optimization can be traced back to the work of Euler and Lagrange in the calculus of variations. In the 1940s, the invention of *linear programming* by [Kantorovich \(1940\)](#) and [Dantzig \(1949\)](#), and the subsequent theoretical and practical developments have further shaped the field. An *optimization problem* is to find the best solution for minimizing (or maximizing) an *objective function* subject to inequality and/or equality constraints. Suppose that $x \in \mathbb{R}^n$ is a vector of n variables, and $f_i : \mathbb{R}^n \mapsto \mathbb{R}$, where $i = 0, 1, 2, \dots, m$, are the objective ($i = 0$) and the constraint functions ($i \geq 1$), respectively. The optimization problem can be written as:

$$\left. \begin{array}{ll} \min_{x \in \mathbb{R}^n} & f_0(x) \\ \text{s.t.} & f_i(x) \leq 0, \quad i \in \mathcal{I}, \\ & f_i(x) = 0, \quad i \in \mathcal{E}, \end{array} \right\} \quad (1.1)$$

where the sets \mathcal{I} and \mathcal{E} consist of indices for inequality and equality constraints, respectively. If the sets \mathcal{I} and \mathcal{E} are empty, the optimization problem (1.1) is called an *unconstrained optimization* problem, otherwise the problem is referred to as a *constrained optimization* problem. Since the variables x are assumed to take real values in (1.1), the problem is called a *continuous optimization* problem. However, in many modeling situations, the variables make sense only if they take discrete values, e.g. $x \in S \subseteq \mathbb{Z}^n$ instead of $x \in \mathbb{R}^n$ in (1.1), in which we refer to the problem as an *integer optimization* problem. When some, but not all, of the variables are restricted to be integers the problem is known as a *mixed integer programming* problem.

The *feasible region* Ω of the problem (1.1) is defined as the set of all vectors satisfying the constraints. That is, $\Omega = \{x \in \mathbb{R}^n : f_i(x) \leq 0, i \in \mathcal{I}; f_i(x) = 0, i \in \mathcal{E}\}$.

We refer to a vector x^* as a *local optimum*, if it has the smaller objective function value among all vectors in a neighborhood of x^* . A local optimum is called a *global optimum*, if it has the smallest objective function value among all local optima. Global optimal solutions are important in many practical applications. Nonetheless, the task of finding them is challenging in many problems.

Optimization problems can be classified depending on the form of the objective and constraint functions. An important concept is the notion of *convexity*, which is a property that makes the optimization problem efficiently solvable in both theory and practice.

Before going further, let us informally explain what we mean by efficiently solvable and hard problems. In the computational complexity theory, we say that a problem is efficiently (polynomially) solvable, if there exists an algorithm which computes its exact solution in a number of arithmetic operations that is bounded above by a polynomial in the instance size, for any instance of the problem. The algorithm is said to have polynomial running time, or to be a *polynomial time algorithm* in short. Here, the instance size of a problem is the number of bits needed to represent the instance on the computer.

Given the optimization problem (1.1), an associated *decision problem* is, for a given number $z \in \mathbb{R}$, to answer ‘yes’ or ‘no’ to the question: Is there an $x \in \Omega$ such that $f_0(x) \leq z$?⁴ The class \mathcal{NP} (non-deterministic polynomial time) contains decision problems where a given ‘yes’-answer can be *verified* in polynomial time. The class of all polynomially solvable decision problems in \mathcal{NP} is referred to as \mathcal{P} . The decision problem $\pi_1 \in \mathcal{NP}$ is *polynomially reducible* to $\pi_2 \in \mathcal{NP}$, if we can convert any instance of π_1 to an instance of π_2 in polynomial time. A decision problem $\pi_1 \in \mathcal{NP}$ is \mathcal{NP} -complete, if all $\pi \in \mathcal{NP}$ are polynomially reducible to π_1 . In other words, if a polynomial time algorithm for any \mathcal{NP} -complete problem exists, then all problems in \mathcal{NP} can be solved in polynomial time. No \mathcal{NP} -complete problem is currently known to have a polynomial solution algorithm, and a big question in computer science that remains unsolved is the \mathcal{P} versus \mathcal{NP} question, i.e. is $\mathcal{P} = \mathcal{NP}$ or $\mathcal{P} \neq \mathcal{NP}$? Resolution to the question, either way, will have important theoretical and computational consequences. An

⁴By solving the decision problem a number of times and using the bisection on the objective function value, we can find the optimal solution to the original optimization problem.

optimization problem for which its decision problem is \mathcal{NP} -complete, is referred to as an \mathcal{NP} -hard problem.

Definition 1.1. A set $S \subseteq \mathbb{R}^n$ is convex if the entire line segment between any two points of S lies in S . That is, for all $x, y \in S$ we have $\alpha x + (1 - \alpha)y \in S$ for all $\alpha \in [0, 1]$.

The notion of convexity also applies to functions. We say that a function f is convex if and only if the set of points above the graph of f is convex. That is, for any $x, y \in S$ (S is convex) we have,

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y), \quad \forall \alpha \in [0, 1].$$

A function f is said to be concave if $-f$ is convex. A class of optimization problems in which the objective function and the feasible region are convex is referred to as *convex optimization* problems. One important property of this class is that any local optimum is also a global optimum, which means that it is sufficient to apply any local optimization algorithm in order to compute the global optimum. An advantage of recognizing a problem as a convex optimization problem, is that the problem can be solved in polynomial time, for example using *interior-point* methods. This means that the convex optimization problems are in \mathcal{P} . One of the most widely used subclasses of convex optimization problems is known as the *linear programming* problem, in which all the constraints and the objective function are linear. That is, in problem (1.1), the objective and constraint functions satisfy

$$f_i(\alpha x + \beta y) = \alpha f_i(x) + \beta f_i(y),$$

for all $x, y \in \mathbb{R}^n$ and for all $\alpha, \beta \in \mathbb{R}$. Several effective methods for solving the linear programming problem are used in many practical applications. Among these is the simplex method developed by [Dantzig \(1949\)](#), and the interior-point method introduced by [Karmarkar \(1984\)](#), which later has been extended by [Nesterov and Nemirovskii \(1994\)](#) for solving general convex optimization problems.

The class of problems where the objective function or the feasible region is not necessarily convex is referred to as *nonconvex optimization* problems. As

opposed to convex problems, nonconvex problems may have several local optima with unequal objective function values, and there is no polynomial time algorithm (unless $\mathcal{P} = \mathcal{NP}$) to find a global optimum in general. Computation and characterization of global optima are the subjects of *global optimization*.

Global optimization has traditionally attracted far less attention than local optimization and solution methods for convex problems. Over the last few decades, however, research in the field has emerged, and several textbooks devoted to the subject have been published. Noteworthy among these are (Horst et al., 2000), (Floudas, 2000) and (Hendrix and G.-Tóth, 2010).

Global optimization approaches are typically based upon *relaxation* problems of the original nonconvex problem to provide lower bounds on the optimal objective function value. The relaxation problem is a modification of the objective function and/or the feasible region giving a new problem that is easier to solve, and it is formally defined as follows:

Definition 1.2. A problem $\min_{x \in \tilde{\Omega}} \tilde{f}_0(x)$ is a relaxation of the problem $\min_{x \in \Omega} f_0(x)$ if: (i) $\Omega \subseteq \tilde{\Omega}$, and (ii) $\tilde{f}_0(x) \leq f_0(x)$, for all $x \in \Omega$.

Different types of relaxation techniques are used to compute lower bounds for the nonconvex problems, for example convex relaxations and Lagrangian relaxations. One way to construct convex relaxations of problems with some nonconvex constraint function f can be based on the *convex* and *concave envelopes* of the function f . Let S be a convex set, and let $\mathcal{L}(f, S)$ be the set of convex functions $g : \mathbb{R}^n \mapsto \mathbb{R}$ that everywhere in S lie below f , i.e. for all $x \in S$ we have $g(x) \leq f(x)$. Then the convex envelope of f is formally defined as follows:

Definition 1.3. The convex envelope of a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ on a convex set S is defined as a function $\text{vex}_S f : \mathbb{R}^n \mapsto \mathbb{R}$ such that for all $x \in S$

$$\text{vex}_S f(x) = \sup_g \{g(x) : g \in \mathcal{L}(f, S)\}.$$

Hence, $\text{vex}_S f(x)$ is the pointwise supremum of $\mathcal{L}(f, S)$. The concave envelope of f is defined as $\text{cav}_S f(x) = -\text{vex}(-f)_S(x)$, which thus becomes the smallest concave function which throughout S lies above f .

1.3 Problem statement

In this work, we consider an abstraction of natural gas transportation networks consisting of supply nodes, intermediate nodes and terminal nodes, as well as links representing pipelines used to transport the gas between two nodes. The gas enters the transportation network at the supply nodes and flows through the intermediate nodes, and finally, it leaves the network through the terminal nodes. In this network, blending operations occur in the intermediate and terminal nodes.

1.3.1 The standard pooling problem

Because of the blending operations and the quality constraints at the terminals, the problem of allocating gas flow to the network is equivalent to a problem frequently occurring in planning oil refinery operations. [Haverly \(1978\)](#) defined the *pooling problem* in terms of the instance depicted in [Figure 1.2](#). A problem instance of the pooling problem is characterized by a network with, at the reception side, source streams with different qualities that can enter the network. Flow from the sources is fed into a limited number of available storage tanks (pools), where the entering flow is mixed to form intermediate blends with new qualities. The pool contents are subsequently used to form final blends at the terminals, where specific quality requirements to the blend are imposed by the market. Due to the blending operations, the optimization model for this problem involves nonconvex constraints, and consequently the pooling problem can possibly have many local optima.

In the Haverly example, we have three types of oils (the source nodes) denoted s_1 , s_2 and s_3 with different concentrations of sulfur contents (for simplicity, assume that sulfur is the only quality parameter) given as 3%, 1% and 2%, respectively. The oils s_1 and s_2 are blended in an intermediate node (pool) denoted p_1 , whereas the oil in s_3 is transported directly to two consumers (terminal nodes) denoted t_1 and t_2 . The output of the pool p_1 is transported to the same terminals. Consumers t_1 and t_2 will only buy the oil if it contains no more than

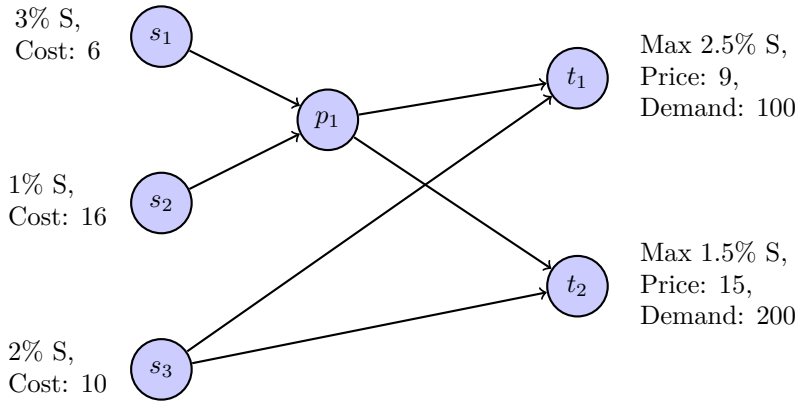


Figure 1.2: Haverly's pooling problem instance.

2.5% and 1.5% of sulfur, respectively. The oil price at the sources, the consumer demand, and sale prices are given in Figure 1.2.

The objective function is to minimize the total cost, while satisfying the consumers quality standard and demand. Dealing with the general instantiations of the pooling problem is the subject of this thesis. Its translation to transportation networks for natural gas is quite direct: Wells and processing units correspond to the sources, junction points are represented by pools, and reception units are modeled as terminals. Although our motivation is the natural gas industry, we will study the pooling problem and its extensions independently of their particular applications.

1.3.2 The blending problem

A special case of the problem under study occurs when the intermediate nodes are not needed, in other words, the flow streams are directly blended at the terminal nodes, leading to a problem known as the *blending problem*. This problem can be modeled as a linear program and can hence be solved fast. An example of this problem is shown in Figure 1.3(a), which is a modification of Haverly's instance illustrated in Figure 1.2, where the pool p_1 is removed.

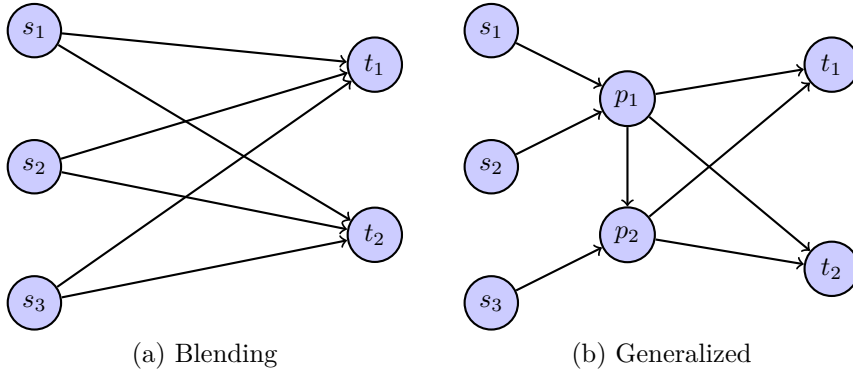


Figure 1.3: Examples of the blending problem, and the generalized pooling problem.

1.3.3 The generalized pooling problem

In the literature, when referring to the pooling problem, it is frequently assumed that the only connections allowed in the network are the connections from sources to pools, from sources to terminals, and from pools to terminals. We refer to this as the *standard pooling problem* in this thesis. An example is given in Figure 1.2. Audet et al. (2004) introduced the *generalized* pooling problem, where connections between pools are allowed as well. Figure 1.3(b) shows an example constructed by modification of Haverly’s instance depicted above.

Driven by increased consumption of energy in the world, coupled with the introduction of unconventional gas in the transportation network, pipeline transportation of natural gas has become a complex system. These conditions have posed several challenges that require more accurate optimization models and more efficient solution methods for the pooling problem. Although a number of optimization models and solution approaches have been applied to the pooling problem, solving large-scale instances in reasonable time with acceptable accuracy is still challenging. This thesis contributes to both modeling and algorithmic methods in order to approach this problem.

1.4 Structure of the thesis

This thesis is divided into two parts. Part I provides motivations, overview of the field, related works, and a summary of the scientific contributions. In Part II, the five publications documenting the results of the thesis are provided in separate attachments in their published form, with the exception of their style which is changed to suite this thesis format.

Part I is further composed of five chapters, including this introductory chapter, and the remaining chapters are organized as follows: In Chapter 2 and 3, we give, along with the thesis contributions, a literature review of model formulations and solution methods for pooling problems, respectively. In Chapter 4, summaries of the included papers are given. Finally, Chapter 5 gives some concluding remarks and proposes possible future research directions.

Chapter 2

Model formulations for pooling problems

The pooling problem is an important optimization problem that has been extensively studied mainly because of its industrial applications. In addition to the applications in the transportation of natural gas and in oil refining discussed in the previous chapter, it is frequently encountered in waste-water treatment and general petrochemicals industries. The research on pooling problems focuses on two directions: The first is developing mathematical formulations that exhibit favorable properties, and the second is designing efficient solution methods (see Chapter 3).

This chapter reviews the different formulations to our problem and its extensions. They are categorized into models for the standard and the generalized pooling problems, which in their turn can also be classified as quality and proportion based models. All of these have bilinear constraints. In a global optimization context, pooling problems are often approached by branch-and-bound algorithms (studied in more details in Section 3.2.1), which rely mainly on the construction of linear relaxations. Building stronger/tighter formulations to provide strong lower bounds on the optimal solution are crucial to the convergence of the algorithms.

2.1 Introduction

The *minimum-cost flow problem* with node capacities is defined on a directed graph $G = (N, A)$ where N is the set of nodes and A is the set of arcs, where each node $i \in N$ has a known *capacity* b_i , and each arc $(i, j) \in A$ has a unit *cost* c_{ij} . For any node $i \in N$, let $N_i^+ = \{j \in N : (i, j) \in A\}$ and $N_i^- = \{j \in N : (j, i) \in A\}$ denote the set of out- and in-neighbors of i , respectively. We assume that G has

non-empty sets $S, T \subseteq N$ of *sources* and *terminals*, respectively, where $N_s^- = \emptyset$, $\forall s \in S$ and $N_t^+ = \emptyset$, $\forall t \in T$. The optimization problem is to determine the minimum-cost plan for sending flow through the network to satisfy supply and demand requirements. The arc flows must be nonnegative and respect the node capacities, and they must satisfy conservation of flow at the intermediate nodes. The minimum-cost flow problem is a transportation model that allocates *single* commodity flow in the network, in other words, the flow entering the network at different sources has the same quality. As already known, there are efficient algorithms to solve this problem which can be modeled as a linear programming problem.

In natural gas transportation networks, the gas flow is coming from several sources that have different qualities, which means that we have *multi-commodity* flow in the network. Consequently, the minimum-cost flow model in this situation needs to be extended to handle this kind of flow. The pooling problem can be viewed as an extension of the minimum-cost flow problem where the quality of the flow depends on the sources from which it originates. At each source, the quality is known, whereas in all other nodes, the quality of the flow blends linearly (see Definition 2.1). In addition to the minimum-cost flow problem parameters, let K be the set of all *quality attributes*. With each $i \in S \cup T$, we define a real constant q_i^k for each $k \in K$. If $s \in S$, q_s^k is referred to as the *quality parameter* of attribute k at that source, and if $t \in T$, q_t^k is referred to as the *quality bound* of attribute k at terminal t . We refer to all nodes in $I = N \setminus (S \cup T)$ as *pools*.

Definition 2.1 (*Linear blending*). The quality at node $i \in I \cup T$ is defined as a weighted average of the qualities at entering arcs, where the corresponding arc flows constitute the weights. The quality at any arc $(i, j) \in A$ is defined as the quality at node i .

Definition 2.2. The *pooling problem* is to assign flow values to all arcs $(i, j) \in A$ such that, in addition to the constraints of the minimum-cost flow problem, the quality bounds at the terminals are respected while the total flow cost is minimized.

In the literature, when referring to the pooling problem, it is frequently assumed that all maximal paths in G have exactly one source and one terminal,

and at most one pool. We refer to this as the *standard pooling problem*, which means that G is a tripartite graph, i.e. $A \subseteq (S \times I) \cup (I \times T) \cup (S \times T)$. The optimization problem arising when no longer assuming the tripartite network structure is referred to (Audet et al., 2004; Misener and Floudas, 2009) as the *generalized pooling problem*.

Different variants of optimization models (or formulations) for the pooling problem and its extensions exist in the literature. Generally, one can divide these formulations into two main categories. The first one consists of flow and quality variables, whereas the other uses flow proportions instead of quality variables. In the next sections, we discuss these formulations and their extensions in more details.

2.2 Formulations for the standard pooling problem

2.2.1 The quality formulation

The most straightforward formulation is achieved by spelling out the definition given in Section 2.1, which in the literature is commonly referred to as the P-formulation. Define f_{ij} as the flow along the arc $(i, j) \in A$, and w_i^k ($k \in K$) as the quality of the flow leaving node $i \in S \cup I$ (if $i \in S$ let $w_i^k = q_i^k$). Then the P-formulation can be written as:

$$[\text{P}] \quad \min_{f,w} \quad \sum_{(i,j) \in A} c_{ij} f_{ij}, \quad (2.1)$$

$$\text{s.t.} \quad \sum_{j \in N_i^+} f_{ij} \leq b_i, \quad i \in N \setminus T, \quad (2.2)$$

$$\sum_{j \in N_i^-} f_{jt} \leq b_t, \quad t \in T, \quad (2.3)$$

$$\sum_{j \in N_i^-} f_{ji} - \sum_{j \in N_i^+} f_{ij} = 0, \quad i \in I, \quad (2.4)$$

$$\sum_{j \in N_i^-} w_j^k f_{ji} - \sum_{j \in N_i^+} w_i^k f_{ij} = 0, \quad i \in I, k \in K, \quad (2.5)$$

$$\sum_{j \in N_t^-} w_j^k f_{jt} - q_t^k \sum_{j \in N_t^-} f_{jt} \leq 0, \quad t \in T, k \in K, \quad (2.6)$$

$$f_{ij} \geq 0, \quad (i, j) \in A. \quad (2.7)$$

Constraints (2.2)–(2.3) and (2.4) express the flow capacity bound at all nodes and the flow conservation around pool nodes, respectively. Constraint (2.5) is the result of direct application of Definition 2.1 to all pool nodes, whereas constraint (2.6) follows by the application of the same definition to terminal $t \in T$, and the quality bound constraint, $w_t^k \leq q_t^k$ for all $k \in K$, from the problem definition. The number of bilinear terms in the P-formulation is proportional to the number of quality attributes. The P-formulation is originally derived for the standard pooling problem. Nevertheless, it can easily be generalized to handle more general pooling networks than the formulation (2.1)–(2.7) does.

Haverly (1978) is the first to use the P-formulation to model the pooling problem, and from that time, many researchers have used it. Among them are Lasdon et al. (1979), Floudas and Aggarwal (1990), Foulds et al. (1992) and Fieldhouse (1993). A practical application of the P-formulation has been shown by Baker and Lasdon (1985) and Amos et al. (1997), who use this formulation at Exxon refineries and New Zealand Refining company, respectively.

2.2.2 Proportion formulations

An alternative formulation relies on variables that represent proportions (fractions) of flow instead of explicit quality variables. However, these types of formulation are applicable only for the standard pooling problem. We can use two types of proportion variables, source or terminal proportions, to replace the quality variables. The idea of using proportion variables was first suggested by Ben-Tal et al. (1994) who derived a formulation that relies on source proportion variables, and they referred to it as the Q-formulation. Recently, by building on the same idea, we gave two new formulations based on terminal proportions (see Paper A).

2.2.2.1 Formulations with source proportions

Define the proportion variables y_i^s ($s \in S$, $i \in I$) as the fraction of the flow through pool i that originates from source s . That is, if the flow through i is non-zero, we have $y_i^s = f_{si} / \sum_{t \in N_i^+} f_{it}$. We keep variable f_{ij} as the flow along the arc $(i, j) \in A$ as in Section 2.2.1. We observe that, due to the introduction of the new variables, the flow along arc (s, i) , where $s \in S$ and $i \in I$, can be represented by $\sum_{t \in N_i^+} y_i^s f_{it}$. Using this observation in constraint (2.5), the quality variables can be expressed as $w_i^k = \sum_{s \in N_i^-} q_s^k y_i^s$ for all $i \in I$, $k \in K$. Combining these observations and the proportion variables with the flow variables, we arrive at the Q-formulation written as:

$$\begin{aligned}
 \text{[Q]} \quad & \min_{f, y} \sum_{i \in I} \sum_{s \in N_i^-} c_{si} y_i^s \sum_{t \in N_i^+} f_{it} + \sum_{t \in T} \sum_{j \in N_t^-} c_{jt} f_{jt} \\
 \text{s.t.} \quad & \sum_{i \in I \cap N_s^+} y_i^s \sum_{t \in N_i^+} f_{it} + \sum_{t \in T \cap N_s^+} f_{st} \leq b_s, \quad s \in S, \\
 & \sum_{t \in N_i^+} f_{it} \leq b_i, \quad i \in I, \quad (2.8) \\
 & \sum_{j \in N_t^-} f_{jt} \leq b_t, \quad t \in T,
 \end{aligned}$$

$$\begin{aligned}
 & \sum_{s \in S \cap N_t^-} q_s^k f_{st} + \sum_{i \in I \cap N_t^-} \sum_{s \in N_i^-} q_s^k y_i^s f_{it} \leq q_t^k \sum_{j \in N_t^-} f_{jt}, \quad t \in T, k \in K, \\
 & \sum_{s \in N_i^-} y_i^s = 1, \quad i \in I, \quad (2.9) \\
 & f_{jt} \geq 0, \quad t \in T, j \in N_t^-, \\
 & 0 \leq y_i^s \leq 1, \quad i \in I, s \in N_i^-.
 \end{aligned}$$

The number of nonlinear variables in the Q-formulation is independent of the number of quality attributes, making this formulation more practical as the number of quality attributes increases. [Tawarmalani and Sahinidis \(2002\)](#) extended the Q-formulation by applying the reformulation-linearization technique (see Section 3.2.1) to constraints (2.8) and (2.9). That is, multiplying (2.9) by

f_{it} yields (2.10), similarly, (2.11) is obtained by multiplying (2.8) by y_i^s . The new formulation is referred to as the PQ-formulation. The new constraints,

$$f_{it} - \sum_{s \in N_i^-} y_i^s f_{it} = 0, \quad i \in I, t \in N_i^+, \quad (2.10)$$

$$\sum_{t \in N_i^+} y_i^s f_{it} - b_i y_i^s \leq 0, \quad i \in I, s \in N_i^-, \quad (2.11)$$

were already derived by Quesada and Grossmann (1995). Tawarmalani and Sahinidis (2002) proved that the linear programming relaxation (constructed by the McCormick envelopes, see Section 3.2.1) of the PQ-formulation dominates the linear programming relaxation of both the P- and the Q-formulation.

2.2.2.2 Formulations with terminal proportions

The PQ-formulation has proportion variables corresponding to sources, and flow variables on arcs entering terminals. Symmetric to the PQ-formulation, we suggest in Paper A (Alfaki and Haugland, 2012b) a formulation with proportion variables corresponding to terminals, and flow variables on arcs leaving sources.

Define for all pools $i \in I$, y_i^t as the proportion of the flow at i destined for terminal $t \in T$. That is, we let $y_i^t = f_{it} / \sum_{s \in S} f_{si}$ when the latter sum is positive. Hence, the new formulation, referred to as the TP-formulation, is given as follows:

$$\begin{aligned}
 \text{[TP]} \quad & \min_{f,y} \sum_{s \in S} \sum_{j \in N_s^+} c_{sj} f_{sj} + \sum_{i \in I} \sum_{t \in N_i^+} c_{it} y_i^t \sum_{s \in N_i^-} f_{si} \\
 \text{s.t.} \quad & \sum_{j \in N_s^+} f_{sj} \leq b_s, & s \in S, \\
 & \sum_{s \in N_i^-} f_{si} \leq b_i, & i \in I, \\
 & \sum_{s \in S \cap N_t^-} f_{st} + \sum_{i \in I \cap N_t^-} \sum_{s \in N_i^-} f_{si} y_i^t \leq b_t, & t \in T, \\
 & \sum_{s \in S \cap N_t^-} q_s^k f_{st} + \sum_{i \in I \cap N_t^-} \sum_{s \in N_i^-} q_s^k f_{si} y_i^t \leq q_t^k \sum_{j \in N_t^-} f_{jt}, & t \in T, k \in K, \\
 & \sum_{t \in N_i^+} y_i^t = 1, & i \in I,
 \end{aligned}$$

$$\sum_{t \in N_i^+} f_{si} y_i^t - f_{si} = 0, \quad s \in N_i^-, i \in I, \quad (2.12)$$

$$\sum_{s \in N_i^-} f_{si} y_i^t - b_i y_i^s \leq 0, \quad i \in I, s \in N_i^-, \quad (2.13)$$

$$\begin{aligned} f_{sj} &\geq 0, & s \in S, j \in N_s^+, \\ 0 \leq y_i^t &\leq 1, & i \in I, t \in N_i^+. \end{aligned}$$

The interpretation of the constraints is analogous to the PQ-formulation. Constraints (2.12) and (2.13) are redundant. Following the pattern of the PQ-formulation, they are included to strengthen the relaxation. A comparison to the PQ-formulation showed that the formulations do not in general have equal strength, but none dominates the other (see Paper A).

In Paper A (Alfaki and Haugland, 2012b), we develop a new proportion formulation by combining both source and terminal proportions in one model. It follows from the definition of y_i^s and y_i^t that $y_i^s f_{it}$ and $y_i^t f_{si}$ both can be interpreted as the flow along the unique path connecting source $s \in S$, pool $i \in I$ and terminal $t \in T$. Given this observation, the STP-formulation can be derived by combining the variables and the constraints from both models.

In the same paper, it has been shown that the linear relaxation of the STP-formulation is at least as tight as the relaxations of both the PQ-, and the TP-formulations. The experiments presented in the paper showed that the STP-formulation in some instances is tighter than both its competitors (see Paper A for complete details).

2.3 Formulations for the generalized pooling problem

2.3.1 A hybrid formulation

Audet et al. (2004) applied their branch-and-cut algorithm to both the P- and Q-formulation and found that the Q-formulation is the more favorable in their algorithm. They also observed that the Q-formulation is not applicable to networks where flow streams leaving one pool may be blended in some pools further downstream in the network. Therefore, the Q-formulation in such networks is no

Section 2.3. Formulations for the generalized pooling problem

longer a bilinear model. In fact, the model will contain terms that are products of two proportion variables. In order to avoid terms where proportion variables are squared, Audet et al. (2004) introduced such variables exclusively for pools that only have sources as in-neighbors, and quality variables for the remaining pools. Denote the former subset of pools I_1 . For each pool which in the sense defined above is close to the sources, the hybrid model thus makes use of a proportion variable y_i^s for each neighboring source s (as defined in Section 2.2.2.1). For other pools, a quality variable w_i^k for each attribute $k \in K$ (as defined in Section 2.2.1) is used. The hybrid formulation denoted [HYB] can be written as:

$$[\text{HYB}] \quad \min_{f,y,w} \sum_{i \in N} \sum_{j \in N_i^+ \setminus I_1} c_{ij} f_{ij} + \sum_{s \in S} \sum_{i \in N_s^+ \cap I_1} \sum_{j \in N_i^+} c_{si} y_i^s f_{ij}, \quad (2.14)$$

$$\text{s.t.} \quad \sum_{j \in N_s^+ \setminus I_1} f_{sj} + \sum_{i \in N_s^+ \cap I_1} \sum_{j \in N_i^+} y_i^s f_{ij} \leq b_s, \quad s \in S, \quad (2.15)$$

$$\sum_{i \in N_t^-} f_{it} \leq b_t, \quad t \in T, \quad (2.16)$$

$$\sum_{j \in N_i^+} f_{ij} \leq b_i, \quad i \in I, \quad (2.17)$$

$$\sum_{j \in N_i^+} f_{ij} - \sum_{j \in N_i^-} f_{ji} = 0, \quad i \in I \setminus I_1, \quad (2.18)$$

$$\begin{aligned} \sum_{j \in N_i^- \cap I_1} \sum_{s \in N_j^-} q_s^k y_j^s f_{ji} + \sum_{j \in N_i^- \cap (I \setminus I_1)} w_j^k f_{ji} \\ - \sum_{j \in N_i^+} w_i^k f_{ij} = 0, \quad i \in I \setminus I_1, k \in K, \end{aligned} \quad (2.19)$$

$$\begin{aligned} \sum_{j \in N_t^- \cap I_1} \sum_{s \in N_j^-} q_s^k y_j^s f_{jt} + \sum_{j \in N_t^- \cap (I \setminus I_1)} w_j^k f_{jt} \\ - q_t^k \sum_{j \in N_t^-} f_{jt} \leq 0, \quad t \in T, k \in K, \end{aligned} \quad (2.20)$$

$$\sum_{s \in N_i^-} y_i^s = 1, \quad i \in I_1, \quad (2.21)$$

$$f_{ij} \geq 0, \quad (i, j) \in A, j \notin I_1, \quad (2.22)$$

$$0 \leq y_i^s \leq 1, \quad (s, i) \in A, i \in I_1. \quad (2.23)$$

In Paper B (Alfaki and Haugland, 2012a), we have found a flaw in the original hybrid formulation given by Audet et al. (2004). The authors gave this formulation only in terms of an example of the generalized pooling problem. Therefore, (2.14)–(2.23) is not only a correction but also a generalization of their formulation.

2.3.2 A multi-commodity flow formulation

In order to extend the PQ-formulation to the generalized pooling problem, we suggest in Paper B (Alfaki and Haugland, 2012a) a multi-commodity flow formulation. We associate a flow *commodity* with each source $s \in S$, where at most b_s units of the commodity can enter the network. The commodity can leave the network at any $t \in T$. At all other nodes, the commodity neither enters nor leaves the network. Now, the variable f_{ij} defines the total flow of all commodities along arc $(i, j) \in A$. For each $i \in N$, let S_i be the set of sources from which there exists a path to i in G (let $S_s = \{s\} \forall s \in S$). Relative to the total flow leaving node $i \in S \cup I$, let the variable y_i^s (this is a generalization of the proportion variable introduced in Section 2.2.2.1) denote the proportion of commodity s . Define $y_i^s = 0$ if $s \notin S_i$ and $y_i^s = 1$ for all $s \in S$. Therefore, the quantity $y_i^s f_{ij}$ defines the flow of commodity s (meaning the commodity associated with source s , we simply refer to s as a commodity whenever convenient) along the arc (i, j) . Based on this multi-commodity flow idea, we have the following formulation:

$$\begin{aligned}
 \text{[MCF]} \quad & \min_{f, y, x} \sum_{(i, j) \in A} c_{ij} f_{ij} \\
 \text{s.t.} \quad & \sum_{j \in N_i^+} f_{ij} \leq b_i, & i \in N \setminus T, \\
 & \sum_{j \in N_t^-} f_{jt} \leq b_t, & t \in T, \\
 & \sum_{j \in N_i^-} y_j^s f_{ji} - \sum_{j \in N_i^+} y_i^s f_{ij} = 0, & i \in I, s \in S_i, \quad (2.24)
 \end{aligned}$$

$$\sum_{j \in N_t^-} \sum_{s \in S_j} (q_s^k - q_t^k) y_j^s f_{jt} \leq 0, \quad t \in T, k \in K, \quad (2.25)$$

$$\sum_{s \in S_i} y_i^s = 1, \quad i \in I,$$

$$\sum_{s \in S_i} y_i^s f_{ij} - f_{ij} = 0, \quad (i, j) \in A, i \in I, \quad (2.26)$$

$$\sum_{j \in N_i^+} y_i^s f_{ij} - y_i^s b_i \leq 0, \quad i \in I, s \in S_i, \quad (2.27)$$

$$f_{ij} \geq 0, \quad (i, j) \in A,$$

$$0 \leq y_i^s \leq 1, \quad i \in I, s \in S_i.$$

Constraints (2.25) express the quality bound at the terminals, and the constraints (2.24) impose the flow proportions y_i^s on all arcs with start node i . Analogous to (2.10)–(2.11), constraints (2.26)–(2.27) are redundant, but are added for the same reason as (2.10)–(2.11) were added to the PQ-formulation. The linear relaxation of the MCF-formulation dominates the corresponding linear relaxations of the HYB-formulation and the generalized version of the P-formulation. In Paper B (Alfaki and Haugland, 2012a), we present computational experiments with this formulation and the P- and the HYB-formulation applied to 40 instances of the generalized pooling problem. Experiments demonstrate that the suggested formulation enables faster computation of the global optimum.

2.4 Formulations for extensions

Meyer and Floudas (2006) and Misener and Floudas (2010) introduced an extension of the pooling problem where the network topology is treated as decision variables: There is a fixed charge for opening the arcs and activating the treatment plants. Therefore, binary variables are needed and the model becomes a mixed integer nonlinear program (MINLP). Such a problem has applications to the design of wastewater treatment networks (Takama et al., 1980). The authors use the P-formulation, since the flow of water may undergo reduction of contamination through several stages of treatment plants. In general, the model for the *wastewater treatment problem* defers from the pooling problem in two aspects: First, there is a fixed cost for opening arcs and installing treatment plants.

Second, each treatment plant has a removal ratio, which represents the removal technologies on this plant, for each contamination (quality) parameter.

Consider the parameters defined in Section 2.1. Let the set of sources represent the effluent streams which usually come from industrial plants. The pools no longer play the role of storage tanks or mixers only, in addition they may be used to reduce the contaminant levels in the wastewater streams. For each treatment plant $i \in I$, define the constant r_i^k as the *removal ratio* of quality (contaminant) $k \in K$. The terminals represent the exit side into which the treated wastewater flows. Define c_{ii} as the unit cost of the flow going through treatment plant $i \in I$. For each arc $(i, j) \in A$ define d_{ij} as the *fixed cost* for opening this arc and d_{ii} as the fixed cost of using the treatment plant $i \in I$. Define the binary variable z_{ij} indicating whether arc $(i, j) \in A$ is used, and z_{ii} as binary variable for using plant $i \in I$. The formulation for the wastewater treatment problem given in (Meyer and Floudas, 2006) can be written as:

$$\min \sum_{i \in I} \left(\sum_{j \in N_i^+} c_{ii} f_{ij} + d_{ii} z_{ii} \right) + \sum_{(i,j) \in A} (c_{ij} f_{ij} + d_{ij} z_{ij}), \quad (2.28)$$

$$\text{s.t.} \quad \sum_{j \in N_s^+} f_{sj} \leq b_s, \quad s \in S,$$

$$\sum_{j \in N_i^+} f_{ij} \leq b_i z_{ii}, \quad i \in I,$$

$$\sum_{j \in N_t^-} f_{jt} \leq b_t, \quad t \in T,$$

$$\sum_{j \in N_i^+} f_{ij} - \sum_{j \in N_i^-} f_{ji} = 0, \quad i \in I,$$

$$(1 - r_i^k) \sum_{j \in N_i^-} w_j^k f_{ji} - \sum_{j \in N_i^+} w_i^k f_{ij} = 0, \quad i \in I, k \in K, \quad (2.29)$$

$$\sum_{j \in N_t^-} w_j^k f_{jt} - q_t^k \sum_{j \in N_t^-} f_{jt} \leq 0, \quad t \in T, k \in K,$$

$$w_i^k \leq (1 - r_i^k) \max_{s \in S} q_s^k, \quad i \in I, k \in K, \quad (2.30)$$

$$z_{ij}, z_{ii} \in \{0, 1\}, \quad i \in I, j \in N_i^+,$$

$$f_{ij} \geq 0, \quad (i, j) \in A.$$

The objective (2.28) is minimization of the total fixed and variable cost associated with use of the pipelines and installation of the treatment plants. We observe that each quality $k \in K$ of the flow leaving treatment plant $i \in I$ will be reduced by r_i^k percent, and hence the quality balance constraint (2.29) follows. Constraint (2.30) has been added to strengthen the formulation.

Another interesting extension of the standard pooling problem was proposed by Misener and Floudas (2010). The purpose of their model is to maximize the profit of blending reformulated gasoline, subject to environmental standards that involve complex emission constraints.

In an oil and gas production planning context, sometimes the decision makers must take into account newly discovered and developed oil and gas fields, and consider these in the pipeline transportation system. However, knowledge of different quality levels and capacities of the wells is needed in advance. To handle such situations, Armagan (2009) and Li et al. (2011b) proposed the *stochastic pooling problem*, which is an extension of the pooling problem accounting for uncertain parameters present in the planning model. Examples of such uncertain parameters are the quality parameters in the raw gas, the capacity bounds of production sources, and the demands at the consumers side. Li et al. (2011a) presented a stochastic MINLP formulation for this generalization of the pooling problem, where the uncertainty in the parameters is given by a limited number of scenarios.

Chapter 3

Solution methods

Many solution techniques have been suggested for the pooling problem. They merely vary in how they deal with the bilinear terms that appear in tracking of the regulated qualities. We have divided the solution methods proposed in the literature into inexact and exact solution approaches.

3.1 Inexact/heuristic techniques

Heuristic algorithms are targeting large problem instances to find good solutions at reasonable computational cost without guaranteeing global optimality. Usually, these solutions are found by iteratively trying to improve a candidate solution with regard to a given measure.

3.1.1 Improvement heuristics

One of the earliest heuristic algorithm proposed to solve the pooling problem is the iterative method proposed by [Haverly \(1978\)](#). This method starts by estimating and fixing the pool qualities, and then the resulting linear programming is solved. The new qualities are calculated using the flow values from the solution of the linear program (LP). If the new and the old qualities coincide the method stops, otherwise it constructs a new LP using the new qualities and repeats these steps until it converges.

The solution returned by the iterative method depends on the initial guess of the values of the quality variables. Moreover, as pointed out in ([Haverly](#),

1979, 1980), this method may not provide a feasible solution, and if it does, the solution it provides is not always the global optimum. Main (1993) observed that the iterative method is unstable in large instances. Practical implementations of the iterative method have been discussed by White and Trierwiler (1980) who used the distributive recursion, which is an improved version of Haverly's iterative method, at SoCal¹, where they managed to model and solve practical instances. As shown by Lasdon and Joffe (1990), the distributive recursion in some sense is more closely related to the successive linear-programming technique, a method that will be discussed in Section 3.1.3.

A more general heuristic for the iterative method has been suggested by Audet et al. (2004), and is referred to as the alternate heuristic (ALT). This heuristic is a two step algorithm that, starting from a feasible point, the first step freezes one set of the variables appearing in the bilinear terms, and solves the resulting for the remaining variables in the model. For example in the P-formulation (see Section 2.2.1), it fixes w_i^k ($i \in I, k \in K$), and solves for f_{ij} for all $(i, j) \in A$. In the second step, the flow variables on arcs leaving the pools are fixed to the values given by the solution to the LP solved in the first step. The resulting LP is then solved for the quality variables and the flow variables on arcs leaving the sources. These two steps are repeated until a fixed point is reached.

The ALT heuristic hence alternates between two linear programs, each of which corresponds to fixing one set of variables occurring in bilinear terms. This contrasts the method of Haverly (1978), which corresponds to fixing the flow on all arcs in the second step.

All heuristic algorithms discussed so far are improvement heuristics, which, based on the fact that freezing one set of the variables that participate in the bilinear terms, results in a linear program (LP). Audet et al. (2004) also suggested a variable neighborhood search (VNS) heuristic, where the local search procedure is provided by the ALT heuristic. VNS initially defines a set of pre-selected neighborhood structures by modifying the feasible extreme points of the LP resulting from ALT. Starting with one neighborhood, this method moves from the current solution by finding a new solution using local search, where the starting point is drawn randomly within a neighborhood of the current solution. If the

¹Standard oil Company of California (SoCal) is the old name of Chevron U.S.A. Corporation.

new solution does improve the current solution, it becomes the new current solution. Otherwise, it selects the next neighborhood and proceeds with the current solution. The algorithm repeats this until the maximum number of iterations is reached.

3.1.2 A construction heuristic

In Paper E (Alfaki and Haugland, 2011b), we propose a construction heuristic for the pooling problem. The heuristic considers a sequence of subgraphs, each of which contains a single terminal, and an associated bilinear program for optimizing the flow to the terminal. The optimal solution to each subproblem serves as a feasible augmentation of the total flow accumulated so far. Experimental results on 20 large-scale standard pooling problem instances indicate that, in large instances, our heuristic algorithm outperforms multi-start local optimization techniques provided by commercially available software. Our heuristic can also easily be extended to tackle generalized pooling problem instances.

3.1.3 Successive linear programming

Successive linear programming (SLP) has traditionally been used to solve the pooling problem in the petrochemical industries. This technique is also referred to as the method of approximate programming (MAP) by Griffith and Stewart (1961) of Shell oil company, who originally proposed and tested the approach on petroleum refinery optimization. Perhaps the major reasons for this popularity are its ability to employ available linear programming codes and solve large instances (Baker and Lasdon, 1985). The method starts with an initial guess of the variable values, approximates the bilinear terms using the Taylor's first order expansion at the initial guess, and then solves the resulting LP. The procedure is repeated with the LP solution as the new base of the Taylor expansion, until convergence to a fix point is obtained. Lasdon et al. (1979) applied SLP and the generalized reduced gradient algorithms to the pooling problem, where they showed some advantages over the iterative method of Haverly. Some improvements of the SLP are reported by Palacios-Gomez et al. (1982), Zhang et al. (1985), Baker and Lasdon (1985) and Sarker and Gunn (1997).

Many successful applications of the SLP technique in the leading oil and gas companies were reported in the literature. Among these is the work of [Simon and Azma \(1983\)](#) at Exxon, where the authors documented Exxon experience with the SLP technique implemented in the system PLATOFORM. Later, [Baker and Lasdon \(1985\)](#) described Exxon's attempt to unify the treatment of nonlinear functions appearing in their mathematical programming system, which was accomplished by the introduction of nonnegative deviation variables in the SLP linearized subproblem.

3.1.4 Benders decomposition

[Benders \(1962\)](#) has proposed a well-known and popular decomposition method for solving nonlinear optimization problems, where the variables are partitioned into complicating and non-complicating variables. The partition is made such that fixing the complicating variables reduces the problem to a linear program in the remaining variables, parametrized by the value of the complicating ones.

Based on Geoffrion's generalization of Benders decomposition ([Geoffrion, 1972](#)), [Floudas and Aggarwal \(1990\)](#) proposed a method that searches for a global solution to the pooling problem. Following the variable partition of Benders, the original problem can be partitioned into a subproblem where the complicating variables are fixed, and a master problem in the complicating variables. The method iterates between the subproblem and the master problem to identify an optimal solution. Despite the satisfactory behavior in some instances, this method could not guarantee convergence to a global solution.

3.1.5 Discretization approaches

To approximate the bilinear constraints, [Tomasgard et al. \(2007\)](#) and [Rømo et al. \(2009\)](#) discretized the quality variables which resulted in a mixed integer programming problem (MILP). A similar approach is used in ([Faria and Bagajewicz, 2008](#)) for the wastewater treatment problem, and replaced the bilinear constraints by "big M" constraints. Pushing in the same direction, [Pham et al. \(2009\)](#) and [Pham \(2007\)](#) eliminated the bilinear terms by discretizing the quality

variables. Consequently, the pooling problem is approximated by a mixed-integer programming problem.

In Paper D (Alfaki and Haugland, 2011a), we propose a method that linearizes the bilinear terms by discretizing the domain of the proportion variables into a fixed number of points. The resulting model serves as an approximation to the pooling problem. This approach is a generalization of the discretization approach proposed by Pham et al. (2009). Computational experiments on a set of large-scale generalized pooling problem instances show that this approach outperforms traditional solution methods where continuous models are used, even when a very coarse discretization is applied. With a fine discretization, however, the discretization approach implies a large computational effort.

3.2 Exact solution techniques

3.2.1 Branch-and-bound algorithms

Most exact global optimization methods are based on a branch-and-bound framework. In general, the branch-and-bound algorithm starts by partitioning the feasible region of the problem into two or more sub-regions (branching process), and constructs a relaxation for each sub-region. This yields a lower bound on the global minimum cost, possibly also an upper bound (bounding process). The branching process is then applied recursively, and defines a search tree in which the nodes represents the sub-regions. Nodes in the resulting search tree are pruned when its lower bounds exceed the best upper bound found so far. The algorithm halts when the tree is empty or the best lower and upper bounds are sufficiently close.

3.2.1.1 Primal-dual decomposition methods

In an effort to improve the method described in Section 3.1.4, Visweswaran and Floudas (1990) suggested the first global optimization algorithm based on a decomposition technique and branch-and-bound. The problem is decomposed into primal and dual subproblems to provide upper and lower bounds on the global solution. Gradient information of the Lagrange function is used to partition the

current domain into sub-domains, and the procedure is repeated until it converges to the global solution. Some improvements of this method are observed in e.g. (Visweswaran and Floudas, 1993) and (Androukakis et al., 1996).

3.2.1.2 Linear relaxation based algorithms

Linear relaxations of problems involving some bilinear function $f(x, y) = xy$, where $(x, y) \in D$, are obtained by the *convex* and *concave envelopes* (see Section 1.2) of f , denoted $\text{vex}_D f(x, y)$ and $\text{cav}_D f(x, y)$, respectively. It can be shown (see Al-Khayyal and Falk, 1983; McCormick, 1976) that the convex and concave envelopes of f on the rectangle $D = [\underline{x}, \bar{x}] \times [\underline{y}, \bar{y}]$ are given by, respectively,

$$\text{vex}_D f(x, y) = \max \{ \underline{y}x + \underline{x}y - \underline{x}\underline{y}, \bar{y}x + \bar{x}y - \bar{x}\bar{y} \}, \quad (3.1)$$

$$\text{cav}_D f(x, y) = \min \{ \underline{y}x + \bar{x}y - \bar{x}\underline{y}, \bar{y}x + \underline{x}y - \underline{x}\bar{y} \}. \quad (3.2)$$

Linear relaxations of the pooling problem formulations given in Section 2.2 are obtained by replacing all occurrences of the bilinear terms by new variables, and by bounding each new variable between its corresponding envelopes. A branch-and-bound algorithm based on such relaxations, where in each iteration the rectangle is divided into four sub-rectangles, was first applied to the pooling problem by Foulds et al. (1992). Audet et al. (2004) suggested a branch-and-cut algorithm, which is an improvement of the above branch-and-bound technique.

The reformulation-linearization technique (RLT) (Sherali and Alameddine, 1992) is a methodology for constructing tight linear relaxations of a nonconvex problem. The second step, linearization, was already discussed above. The first step, reformulation, is to add new valid constraints obtained by multiplying two original constraints.

It is interesting to note that one can arrive to the McCormick's convex and concave envelopes (3.1)–(3.2) by applying the RLT to the bound constraints of x and y . For example, multiplying $(x - \underline{x}) \geq 0$ and $(\bar{y} - y) \geq 0$ yields the constraint $xy \leq \bar{y}x + \underline{x}y - \underline{x}\bar{y}$, which is one of the constraints suggested by (3.2).

Quesada and Grossmann (1995) applied the RLT to obtain a relaxation which is used within a spatial branch and bound algorithm that uses a nonlinear solver

to provide upper bounds. The result in several instances showed that a few branch-and-bound nodes were needed to verify the global solutions.

Sahinidis and Tawarmalani (2005) applied their branch-and-reduce algorithm, which uses the McCormick’s relaxation as lower-bounding and local and random search as upper-bounding techniques. In addition, it uses various range reduction techniques. This algorithm is implemented in the generic global optimization code, BARON (Sahinidis, 1996), by use of the PQ-formulation. When applying their code to standard instances from the literature, they were able to reduce the running time and the size of the search tree significantly.

Liberti and Pantelides (2006) proposed an improved relaxation technique referred to as the reduced reformulation linearization technique (RRLT), and incorporated it in a spatial branch-and-bound algorithm. The authors also suggested an algorithm that automatically constructs this relaxation for large and sparse NLPs such as the pooling problem. They applied their algorithm to common pooling instances where the results showed that tight linear relaxations and hence faster convergence are provided.

Piecewise-linear relaxations have been proposed by Wicaksono and Karimi (2008) and Gounaris et al. (2009), who utilize piecewise linearization schemes by partitioning the original domain of the variables involved in the bilinear terms into smaller sub-domains. Applying the McCormick relaxation for each of the resulting sub-domains, and using binary variables to select the optimal sub-domain, resulted in an efficient relaxation that can be used in the branch-and-bound framework to accelerate convergence.

3.2.1.3 Lagrangian relaxation based algorithms

The Lagrangian relaxation is a useful technique when the problem’s constraints can be decomposed into “difficult” and “easy” ones. The difficult constraints are relaxed by adding them to the objective with weight (*Lagrange multipliers*), and thereby the solution provides a lower bound on the global solution of the original problem. In the pooling problem, the difficult constraints are the bilinear ones. As shown in Chapter 2, these constraints arise from quality balances around pools and the quality bounds at terminals.

Ben-Tal et al. (1994) studied the Lagrangian relaxation of their Q-formulation (see Section 2.2.2.1 for details). The associated Lagrangian dual, which gives a lower bound on the minimum cost, is solved by analyzing the simplex $\{y \in \mathbb{R}^{S \times I} : \sum_{s \in N_i^-} y_i^s = 1\}$. This relaxation is integrated in a branch-and-bound algorithm that divides the simplex into smaller ones. Upper bounds on the global minimum cost are found by local search.

Adhya et al. (1999) introduced a Lagrangian relaxation by dualizing all the constraints in the P-formulation except for the variable bounds. The solution of the resulting Lagrangian subproblem is approximated by solving a sequence of MILPs. They also proved that the Lagrangian relaxation provides tighter lower bounds than standard linear relaxation does in the case of more than one quality parameter. A similar Lagrangian relaxation was suggested by Almutairi and Elhedhli (2009).

3.2.2 Semidefinite programming relaxations

In Paper C (Frimannslund et al., 2010), we suggest a technique based on a series of semidefinite programs (or linear matrix inequality (LMI) relaxations) to solve the pooling problem. LMI relaxations are used to turn general (nonconvex) optimization problems, where the objective and the constraints are polynomials, into a sequence of convex positive semidefinite programs (Lasserre, 2001a,b).

The general idea of this technique is as follows. Consider the optimization problem,

$$f^* = \min_{x \in \mathbb{R}^n} \{f_0(x) : x \in \Omega\}, \quad (3.3)$$

and assume for simplicity that Ω is compact and f_0 is continuous. Then the problem (3.3) can be turned into a convex problem by minimizing over the set, $\mathcal{B}(\Omega)$, of all Borel probability measures μ supported on Ω . The resulting optimization problem,

$$\mu^* = \min_{\mu \in \mathcal{B}(\Omega)} \int f_0(x) d\mu,$$

has the same global optimum value as the original problem. However, finding the probability distribution μ^* on the support Ω is done by characterizing its moment sequences, which is an infinite-dimensional convex optimization problem

known as the *moment problem* (Lasserre, 2010). Instead of solving an infinite-dimensional problem, a truncated moment sequences are determined, which can be cast as an LMI relaxation. By increasing the order of the moment sequences, a tighter relaxation is obtained.

By applying the above technique to the pooling problem with a single quality parameter, Frimannslund et al. (2010) show that if the feasible set has a nonempty interior, then we have a finite sequence of LMI relaxations with increasing order that converges to the global optimum. For a fixed relaxation order, this technique thus provides tight lower bounds for the global minimum cost. Based on the experiments, we show that for low order relaxations, the lower bound provided by this technique matches the true global optimum in several small instances.

3.3 Summary

A review of the literature on solution techniques proposed to solve the pooling problem is given in this chapter. These techniques are classified as improvement heuristics, successive linear programming, decomposition techniques and branch-and-bound algorithms. Global optimization algorithms are quite effective for instances of small to moderate size. In larger instances, however, global optimizers fail to converge in reasonable time, while existing local optimizers depend largely on good initial guesses.

Chapter 4

Summary of papers

In this chapter, we give an overview of the five papers constituting the thesis. Two of the papers are focused on modeling (see Section 4.1), while the topic of the others is solution methods (see Section 4.2). With the exception of Paper E, each paper was presented in at least one international conference. Papers A and C are direct extensions of conference papers (not included).

4.1 Strong formulations

4.1.1 The standard pooling problem (Paper A)

Paper A, entitled “Strong formulations for the pooling problem” and authored by Mohammed Alfaki and Dag Haugland, is published in the *Journal of Global optimization*, doi: 10.1007/s10898-012-9875-6. A short version of the paper is presented at *Toulouse Global Optimization Workshop* (TOGO 2010) in Toulouse, France, and published in the same conference proceedings (Alfaki and Haugland, 2010). The paper was also presented at the *4th Nordic Optimization Symposium*, 2010, in Århus, Denmark.

In this paper, we develop new formulations for the standard pooling problem based on terminal proportion variables (see Section 2.2.2.2). In the strongest model, we combine source and terminal proportion variables. This formulation is proved to be stronger than other formulations based uniquely on source proportions or quality variables. A new branching strategy that performs well with the strongest formulation is presented.

Main contributions:

- We give a formal proof of the *strong* \mathcal{NP} -hardness¹ of the pooling problem by constructing a polynomial reduction from the *maximum independent vertex set problem* (MIVS)². We also prove that the strong \mathcal{NP} -hardness persists in two interesting special cases: (i) If the networks have only one pool, (ii) If we consider maximizing the total flow instead of minimizing the total cost.
- We extend the idea of proportion variables in the PQ-formulation (see Section 2.2.2.1), and give two new formulations. The first is the TP-formulation (see Section 2.2.2.2), which contrasts the PQ-formulation in that it uses terminal proportions instead of source proportions. This formulation is comparable to the PQ-formulation in terms of strength. The second new formulation is the STP-formulation (see Section 2.2.2.2), which combines source and terminal proportions, and proves to have stronger linear relaxation than both the PQ-, and TP-formulations.
- The strength of the STP-formulation is tested on a set of well-studied pooling problem instances from the literature, and on large randomly generated instances. Computational experiments have confirmed the strength of the STP-formulation. In most of the instances, the STP-formulation turned out to yield stronger lower bounds than the competing formulations.
- We develop a special branching rule suitable for the STP-formulation, which is incorporated in a branch-and-bound algorithm. Computational experiments with an implementation of the algorithm indicate that this branching rule helps to improve the lower bound on the global optimum.

4.1.2 The generalized pooling problem (Paper B)

Paper B, entitled “A multi-commodity flow formulation for the generalized pooling problem” and authored by Mohammed Alfaki and Dag Haugland, is published in *Journal of Global optimization*, doi: 10.1007/s10898-012-9890-7. A prelimi-

¹A problem is a *strongly* \mathcal{NP} -hard if it remains \mathcal{NP} -hard even when all of its parameters are bounded by a polynomial in the input size.

²The MIVS problem is to find a largest possible subset of the vertices in a graph, such that all pairs of vertices in the subset are non-neighbors.

nary version was presented at the *3rd Nordic Optimization Symposium*, 2009, in Stockholm, Sweden.

In this paper, we develop a multi-commodity flow formulation for the generalized pooling problem (see Section 2.3.2). This formulation is a generalization of a well-established formulation for the standard problem, and we demonstrate that it is stronger than alternative, popular formulations from the literature.

Main contributions:

- We develop a multi-commodity flow formulation for the generalized pooling problem. When applied to the standard pooling problem, this formulation coincides with PQ-formulation.
- We prove that our new formulation is stronger than both the P-formulation (see Section 2.2.1) and the HYB-formulation (see Section 2.3.1).
- The paper also presents computational experiments of 40 instances with up to 35 nodes and 12 quality attributes confirming that the suggested formulation performs better. They confirm that the suggested formulation enables faster computation of (strong bounds on) the global optimum.

4.2 New solution methods

4.2.1 LMI relaxations (Paper C)

A short version of Paper C, which is entitled “Solving the pooling problem with LMI relaxations” and authored by Lennart Frimannslund, Mohamed El Ghami, Mohammed Alfaki, and Dag Haugland, is presented at *Toulouse Global Optimization Workshop* (TOGO 2010) in Toulouse, France, and published in the conference proceedings (Frimannslund et al., 2010) (reviewed). In the thesis, we include the complete version of the paper.

In this paper, we suggest a solution framework based on a sequence of LMI relaxations (see Section 3.2.2) to solve standard pooling problems with a single quality parameter. We have considered both the maximum flow and the minimum cost versions of the problem. Based on our experiments, we show that standard pooling instances with a single quality parameter can be solved at low LMI

relaxation orders. However, solving such relaxation implies a large computational effort, which for large instances makes the method impractical to use.

Main contributions:

- We suggest a technique that provides tight lower bounds for the global objective function value, which, under certain conditions, converges monotonically to the global optimum. The method applies to the standard pooling problem with a single quality parameter.
- The experiments show that, in several small instances, the maximum flow and minimum cost versions of the problem can be solved at LMI relaxation order 2 and 3, respectively.

4.2.2 A discretization approach (Paper D)

Paper D, entitled “Comparison of discrete and continuous models for the pooling problem” and authored by Mohammed Alfaki and Dag Haugland, is presented at the *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS’11)*, 2011, in Saarbrücken, Germany. The paper is reviewed and published in the conference proceedings.

In this paper, we generalize the discretization approach proposed in (Pham et al., 2009) to the generalized pooling problem. This method approximates the problem by discretizing the proportion variables into a given number of points, and consequently, the resulting model is a mixed-integer programming problem (see Section 3.1.5). Through numerical experiments on large scale instances, we compare our discrete formulation with a continuous formulation. The purpose of this is to investigate whether discrete models are more suitable for finding good solutions when the global optimum is out of reach. By lower bounding techniques, we also aim to estimate the error introduced by discretizing the solution space.

Main contributions:

- We propose a method that linearizes the bilinear terms at the cost of introducing binary variables. The resulting model serves as an approximation to the generalized pooling problem.

- Computational experiments on a set of large-scale instances show that this approach is superior to approaches that use continuous variables, even when a very coarse discretization is applied.

4.2.3 A construction heuristic method (Paper E)

Paper E, entitled “Computing feasible solutions to the pooling problem” and authored by Mohammed Alfaki and Dag Haugland, is submitted to *Annals of Operations Research* (the first review is currently in progress).

In this paper, we give a construction heuristic for the standard pooling problem. It constructs a sequence of subgraphs, each of which contains a single terminal, and associated bilinear programs for optimizing the flow to the terminal. The optimal solution to each bilinear program serves as a feasible augmentation of the total flow accumulated so far. The suggested method is designed to give good feasible solutions, particularly in large instances, and does not guarantee to find the optimal solution. In order to keep the work focused, we confine the study to the standard version of the problem, but it is straightforward to extend the proposed method to the generalized version.

Main contributions:

- We develop a greedy construction heuristic method for the pooling problem (see Section 3.1.2).
- Experimental results on 20 large-scale instances indicate that, in large instances, our heuristic algorithm outperforms multi-start local optimization techniques provided by commercially available software.

Chapter 5

Conclusion and future work

5.1 Conclusion

In the last several decades, many inexact and exact solution methods to approach the pooling problem have been suggested in the literature. Although inexact solution methods are intended for large problem instances, most of these are under the influence of starting points, and hence can easily be trapped at weak solutions. On the other hand, most exact solution methods are based on branch-and-bound algorithms requiring strong relaxations in order to converge fast. The thesis has contributed to both modeling and algorithmic aspects by developing strong formulations and efficient solution methods for the pooling problem.

By constructing a polynomial reduction from the MIVS problem, we have proved formally that the pooling problem is strongly \mathcal{NP} -hard. We have also showed that the problem remains strongly \mathcal{NP} -hard when there is only one pool and no direct arcs from sources to terminals. Concerning the development of strong formulation for the standard pooling problem, we have derived new formulations based on terminal proportion variables. In the strongest model, referred to as the STP-formulation, we have combined source and terminal proportion variables. This formulation is proved to be stronger than competing formulations from the literature. Along with this model, we have suggested a new branching strategy that exploits the redundancy in the STP-formulation. The strength of the formulation and effectiveness of the branching strategy are demonstrated experimentally.

For the generalized pooling problem, we have proposed a multi-commodity flow formulation. The proposed model is an extension of the PQ-formulation for the standard version of the problem. We have proved that our multi-commodity flow formulation has stronger relaxation than state-of-the-art formulations from the literature. Experiments also confirm that ours performs better.

Three solution methods have been suggested in the thesis. A procedure based on a sequence of LMI relaxations has been proposed to solve the pooling problem with a single quality parameter. The experiments indicate that small instances of this problem can be solved with LMI relaxation of order 2 or 3. However, solving such relaxations implies a large computational effort.

We have also developed a mixed integer programming model, serving as an approximation to the pooling problem, by discretizing the proportion variables. Hence, bilinear constraints are transformed into linear ones, at the computational cost represented by the introduction of binary variables. Computational experiments on a set of large-scale instances show that a discrete model is superior to its continuous ancestor.

In the last approach, we have developed a construction heuristic for the pooling problem. It considers a sequence of subgraphs, each of which contains a single terminal, and an associated bilinear program for optimizing the flow to the terminal. The optimal solution serves as a feasible augmentation of the total flow accumulated so far. Experimental results indicate that, in large instances, our heuristic algorithm outperforms multi-start local optimization techniques provided by commercially available software.

5.2 Future work

Regarding the method developed in Paper [D](#), a topic for future research is to develop an adaptive discretization rule. Computations can be saved if the number of discretization points can be kept small, while gradually focusing the search on solution sets of decreasing size. This can also be integrated in a branch-and-bound algorithm to provide tighter upper bounds on the global optimum.

In Paper [E](#), instead of iterating the construction heuristic on the set of terminals, we will investigate a similar heuristic that considers the set of sources.

Extending the idea for other extensions of the pooling problem is also a possible direction for future work.

In the models for pipeline transportation of natural gas, we have extended the traditional flow models by adding constraints that track the flow quality throughout the system. Additional type of constraints that seem to be of particular importance when allocating flow in natural gas transportation networks is the interrelation between pipeline flow and pressure. That is, the inlet pressure of a pipeline is a function of upstream flow, and the outlet pressure is given by the inlet pressure and the pipeline flow. Increasing the flow implies a reduction of the outlet pressure, which in its turn reduces the flow capacity of downstream links. Inlet and outlet pressure variables must be introduced for each pipeline, and the pipeline flow must be modeled as a (nonlinear) function of these. In the ideal case, this is accomplished by use of e.g. the Weymouth equation (Osiađacz, 1987), which states that the square of the flow is proportional to the difference of the squares of inlet and outlet pressures. That takes the following form:

$$f_{ij}^2 = W_{ij} (p_i^2 - p_j^2), \quad (i, j) \in A.$$

where f_{ij} is the flow through pipeline $(i, j) \in A$, W_{ij} is a constant which depends on pipeline physical properties, and p_i is the pressure at node $i \in N$. Under certain conditions, such relations can be formulated in terms of convex constraints. However, when inhomogeneous flow streams are pooled, this is no longer realistic, and the capacity constraints become nonconvex.

When two or more pipelines meet at a junction node (pool), flow streams of unequal pressure are leveled to the smallest pressure value. In order to model this, a binary variable is required for each valve that can be either open or closed. Consequently, an otherwise continuous flow model is transferred into a discrete model, and thereby becomes much more difficult to solve. In fact, taking the above mentioned constraints into account, the pipeline transportation model of natural gas becomes a mixed integer nonlinear program (MINLP). In the suggested future work, we will investigate how the flow problems mentioned above can be formulated and solved. We will in particular study how solution approaches based on models with proportion variables can be applied.

Bibliography

- Adhya, N., Tawarmalani, M., and Sahinidis, N.V. (1999). A Lagrangian approach to the pooling problem. *Industrial & Engineering Chemistry Research*, **38** (5), 1956–1972.
- Al-Khayyal, F.A., and Falk, J.E. (1983). Jointly constrained biconvex programming. *Mathematics of Operations Research*, **8** (2), 273–286.
- Alfaki, M., and Haugland, D. (2010). Strong formulations for the pooling problem. In S. Cafieri, B. G.-Tóth, E. Hendrix, L. Liberti and F. Messine (Eds.), *Proceedings of the Toulouse Global Optimization Workshop* (pp. 15–18).
- Alfaki, M., and Haugland, D. (2011a). Comparison of discrete and continuous models for the pooling problem. In A. Caprara and S. Kontogiannis (Eds.), *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (Vol. 20, pp. 112–121). OpenAccess Series in Informatics (OASICs). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.
- Alfaki, M., and Haugland, D. (2011b). Computing feasible solutions to the pooling problem. *Submitted to: Annals of Operations Research*. First review in progress.
- Alfaki, M., and Haugland, D. (2012a). A multi-commodity flow formulation for the generalized pooling problem. *Journal of Global Optimization*. doi:[10.1007/s10898-012-9890-7](https://doi.org/10.1007/s10898-012-9890-7)
- Alfaki, M., and Haugland, D. (2012b). Strong formulations for the pooling problem. *Journal of Global Optimization*. doi:[10.1007/s10898-012-9875-6](https://doi.org/10.1007/s10898-012-9875-6)
- Almutairi, H., and Elhedhli, S. (2009). A new Lagrangian approach to the pooling problem. *Journal of Global Optimization*, **45** (2), 237–257.

- Amos, F., Rönnqvist, M., and Gill, G. (1997). Modelling the pooling problem at the New Zealand Refining Company. *Journal of the Operational Research Society*, **48** (8), 767–778.
- Androukakis, I.P., Visweswaran, V., and Floudas, C.A. (1996). Distributed decomposition-based approaches. In C. Floudas and P. Pardalos (Eds.), *State of the Art in Global Optimization: Computational Methods and Applications* (pp. 285–301). Kluwer Academic, Dordrecht.
- Armagan, E. (2009). *Decomposition Algorithms for Global Solution of Deterministic and Stochastic Pooling Problems in Natural Gas Value Chains*. (Master’s thesis, Massachusetts Institute of Technology, Cambridge, USA).
- Audet, C., Brimberg, J., Hansen, P., Le Digabel, S., and Mladenović, N. (2004). Pooling problem: Alternate formulations and solution methods. *Management science*, **50** (6), 761–776.
- Baker, T.E., and Lasdon, L.S. (1985). Successive linear programming at Exxon. *Management Science*, **31** (3), 264–274.
- Ben-Tal, A., Eiger, G., and Gershovitz, V. (1994). Global minimization by reducing the duality gap. *Mathematical Programming*, **63** (2), 193–212.
- Benders, J.F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, **4** (1), 238–252.
- Boyd, S.P., and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge, UK: Cambridge University Press.
- Dantzig, G.B. (1949). Programming of interdependent activities: II mathematical model. *Econometrica, Journal of the Econometric Society*, 200–211.
- Faria, D.C., and Bagajewicz, M.J. (2008). A new approach for the design of multi-component water/wastewater networks. *Computer Aided Chemical Engineering*, **25**, 43–48.

- Fieldhouse, M. (1993). The pooling problem. In T. Ciriani and R. Leachman (Eds.), *Optimization in Industry: Mathematical Programming and Modeling Techniques in Practice* (pp. 223–230). New York, USA: John Wiley & Sons Ltd.
- Floudas, C.A. (2000). *Deterministic Global Optimization: Theory, Methods, and Applications*. Springer.
- Floudas, C.A., and Aggarwal, A. (1990). A decomposition strategy for global optimization search in the pooling problem. *Operations Research Journal On Computing*, **2** (3), 225–235.
- Foulds, L.R., Haugland, D., and Jörnsten, K. (1992). A bilinear approach to the pooling problem. *Optimization*, **24** (1), 165–180.
- Frimannslund, L., El Ghami, M., Alfaki, M., and Haugland, D. (2010). Solving the Pooling Problem with LMI Relaxations. In S. Cafieri, B. G.-Tóth, E. Hendrix, L. Liberti and F. Messine (Eds.), *Proceedings of the Toulouse Global Optimization Workshop* (pp. 51–54).
- Geoffrion, A.M. (1972). Generalized benders decomposition. *Journal of Optimization Theory and Applications*, **10** (4), 237–260.
- Gounaris, C.E., Misener, R., and Floudas, C.A. (2009). Computational comparison of piecewise-linear relaxation for pooling problems. *Industrial & Engineering Chemistry Research*, **48** (12), 5742–5766.
- Griffith, R.E., and Stewart, R.A. (1961). A nonlinear programming technique for the optimization of continuous processing systems. *Management Science*, **7**, 379–392.
- Guo, B., and Ghalambor, A. (2005). *Natural Gas Engineering Handbook*. Houston, USA: Gulf Publishing Company.
- Haverly, C.A. (1978). Studies of the behavior of recursion for the pooling problem. *ACM SIGMAP Bulletin*, **25**, 19–28.
- Haverly, C.A. (1979). Behavior of recursion model-more studies. *ACM SIGMAP Bulletin*, **26**, 22–28.

- Haverly, C.A. (1980). Recursion model behavior: More studies. *ACM SIGMAP Bulletin*, **28**, 39–41.
- Hendrix, E.M.T., and G.-Tóth, B. (2010). *Introduction to Nonlinear and Global Optimization*. Cambridge, UK: Springer.
- Horst, R., Pardalos, P.M., and Thoai, N.V. (2000). *Introduction to Global Optimization* (Second). Dordrecht, The Netherlands: Kluwer Academic Publishers.
- Hubbard, R. (2009). The role of gas processing in the natural-gas value chain. *Journal of Petroleum Technology*, **61** (8), 65–71.
- Kantorovich, L.V. (1940). A new method of solving some classes of extremal problems. *Doklady Academy of Sciences USSR*, **28**, 211–214.
- Karmarkar, N. (1984). A new polynomial time algorithm for linear programming. *Combinatorica*, **4**, 373–395.
- Lasdon, L.S., and Joffe, B. (1990). The relationship between distributive recursion and successive linear programming in refining production planning models. In *Proceedings of the NPRA Computer Conference, October 29–31, Seattle, Washington*.
- Lasdon, L.S., Waren, A.D., Sarkar, S., and Palacios, F. (1979). Solving the pooling problem using generalized reduced gradient and successive linear programming algorithms. *ACM Sigmap Bulletin*, **27**, 9–15.
- Lasserre, J.B. (2001a). Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, **11** (3), 796–817.
- Lasserre, J.B. (2001b). New positive semidefinite relaxations for nonconvex quadratic programs. In H. Hadjisavvas and P. Pardalos (Eds.), *Advances in Convex Analysis and Global Optimization: Honoring the Memory of C. Caratheodory (1873-1950)* (pp. 161–189). Kluwer Academic Publishers.
- Lasserre, J.B. (2010). *Moments, Positive Polynomials and Their Applications*. London, UK: Imperial College Press.

- Li, X., Tomasgard, A., and Barton, P.I. (2011a). Decomposition strategy for the stochastic pooling problem. *Journal of Global Optimization*, 1–26.
- Li, X., Armagan, E., Tomasgard, A., and Barton, P.I. (2011b). Stochastic pooling problem for natural gas production network design and operation under uncertainty. *AIChE Journal*, **57** (8), 2120–2135.
- Liberti, L., and Pantelides, C.C. (2006). An exact reformulation algorithm for large nonconvex NLPs involving bilinear terms. *Journal of Global Optimization*, **36** (2), 161–189.
- Main, R.A. (1993). Large recursion models: Practical aspects of recursion techniques. In T. Ciriani and R. Leachman (Eds.), *Optimization in Industry: Mathematical Programming and Modeling Techniques in Practice* (pp. 241–249). New York, USA: John Wiley & Sons Ltd.
- McCormick, G.P. (1976). Computability of global solutions to factorable nonconvex programs: part I - convex underestimating problems. *Mathematical Programming*, **10** (1), 147–175.
- Meyer, C.A., and Floudas, C.A. (2006). Global optimization of a combinatorially complex generalized pooling problem. *AIChE Journal*, **52** (3), 1027–1037.
- Misener, R., and Floudas, C.A. (2009). Advances for the pooling problem: Modeling, global optimization, and computational studies. *Applied and Computational Mathematics*, **8** (1), 3–22.
- Misener, R., and Floudas, C.A. (2010). Global optimization of large-scale generalized pooling problems: Quadratically constrained MINLP models. *Industrial & Engineering Chemistry Research*, **49** (11), 5424–5438.
- Mokhatab, S., Poe, W.A., and Speight, J.G. (2006). *Handbook of Natural Gas Transmission and Processing*. Gulf Professional Publishing.
- Nesterov, Y., and Nemirovskii, A. (1994). *Interior-point Polynomial Algorithms in Convex Programming*. SIAM Studies in Applied Mathematics. Philadelphia, USA: Society for Industrial and Applied Mathematics (SIAM).

- Nocedal, J., and Wright, S.J. (2000). *Numerical Optimization*. Springer Verlag.
- Osiadacz, A. (1987). *Simulation and analysis of gas networks*. Houston, USA: Gulf Publishing Company.
- Palacios-Gomez, F., Lasdon, L.S., and Engquist, M. (1982). Nonlinear optimization by successive linear programming. *Management Science*, **28** (10), 1106–1120.
- Pham, V. (2007). *A Global Optimization Approach to Pooling Problems in Refineries*. (Master’s thesis, Department of Chemical Engineering, Texas A&M University, Texas, USA).
- Pham, V., Laird, C., and El-Halwagi, M. (2009). Convex hull discretization approach to the global optimization of pooling problems. *Industrial & Engineering Chemistry Research*, **48** (4), 1973–1979.
- Quesada, I., and Grossmann, I.E. (1995). Global optimization of bilinear process networks with multi-component flows. *Computers & Chemical Engineering*, **19** (12), 1219–1242.
- Rømo, F., Tomasgard, A., Hellemo, L., Fodstad, M., Eidesen, B.H., and Pedersen, B. (2009). Optimizing the Norwegian natural gas production and transport. *Interfaces*, **39** (1), 46–56.
- Sahinidis, N.V. (1996). BARON: A general purpose global optimization software package. *Journal of Global Optimization*, **8** (2), 201–205.
- Sahinidis, N.V., and Tawarmalani, M. (2005). Accelerating branch-and-bound through a modeling language construct for relaxation-specific constraints. *Journal of Global Optimization*, **32** (2), 259–280.
- Sarker, R.A., and Gunn, E.A. (1997). A simple SLP algorithm for solving a class of nonlinear programs. *European Journal of Operational Research*, **101** (1), 140–154.
- Sherali, H.D., and Alameddine, A. (1992). A new reformulation-linearization technique for bilinear programming problems. *Journal of Global Optimization*, **2**, 379–410.

- Simmons, D., Horlings, H., Cronshaw, I., and Others. (2006). *Natural Gas Market Review 2006: Towards a Global Gas Market*. International Energy Agency.
- Simon, J.D., and Azma, H.M. (1983). Exxon experience with large scale linear and nonlinear programming applications. *Computers & Chemical Engineering*, **7** (5), 605–614.
- Takama, N., Kuriyama, T., Shiroko, K., and Umeda, T. (1980). Optimal water allocation in a petroleum refinery. *Computers & Chemical Engineering*, **4** (4), 251–258.
- Tawarmalani, M., and Sahinidis, N.V. (2002). *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*. Dordrecht, The Netherlands: Kluwer Academic Publishers.
- Tomasgard, A., Rømo, F., Fodstad, M., and Midthun, K. (2007). Optimization models for the natural gas value chain. In G. Hasle, K. Lie and E. Quak (Eds.), *Geometric Modelling, Numerical Simulation, and Optimization: Applied Mathematics at SINTEF* (pp. 521–558). Springer.
- Visweswaran, V., and Floudas, C.A. (1990). A global optimization algorithm (GOP) for certain classes of nonconvex NLPs–II. Application of theory and test problems. *Computers & chemical engineering*, **14** (12), 1419–1434.
- Visweswaran, V., and Floudas, C.A. (1993). New properties and computational improvement of the GOP algorithm for problems with quadratic objective functions and constraints. *Journal of Global Optimization*, **3** (4), 439–462.
- Wardzinski, J., Foss, M.M., and Delano, F. (2004). Interstate natural gas–quality specifications and interchangeability. *Center for Energy Economics, Bureau of Economic Geology, University of Texas at Austin, December*.
- White, D.L., and Trierwiler, L.D. (1980). Distributive recursion at SoCal. *ACM SIGMAP Bulletin*, **28**, 22–38.

Bibliography

- Wicaksono, D.S., and Karimi, I.A. (2008). Piecewise MILP under- and over-estimators for global optimization of bilinear programs. *AIChE Journal*, **54** (4), 991–1008.
- Wolsey, L.A. (1998). *Integer Programming*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons Ltd.
- Zhang, J.H., Kim, N.H., and Lasdon, L. (1985). An improved successive linear programming algorithm. *Management Science*, **31** (10), 1312–1331.

Part II

Scientific contributions

Paper A

Strong formulations for the pooling problem²

Mohammed Alfaki¹ and Dag Haugland¹

¹Department of Informatics, University of Bergen,
P.O. Box 7803, N-5020 Bergen, Norway.
mohammeda@ii.uib.no and dag@ii.uib.no

²In: *Journal of Global Optimization*, doi: [10.1007/s10898-012-9875-6](https://doi.org/10.1007/s10898-012-9875-6), 2012.

Paper B

B

A multi-commodity flow formulation for the generalized pooling problem²

Mohammed Alfaki¹ and Dag Haugland¹

¹Department of Informatics, University of Bergen,
P.O. Box 7803, N-5020 Bergen, Norway.
mohammeda@ii.uib.no and dag@ii.uib.no

²In: *Journal of Global Optimization*, doi: [10.1007/s10898-012-9890-7](https://doi.org/10.1007/s10898-012-9890-7), 2012.

Paper C

Solving the pooling problem with LMI relaxations²

Lennart Frimannslund¹, Mohamed El Ghami¹,
Mohammed Alfaki¹ and Dag Haugland¹

¹Department of Informatics, University of Bergen,
P.O. Box 7803, N-5020 Bergen, Norway.
{lennart,mohamed,mohammeda,dag}@ii.uib.no

²A short version of this paper is published in: S. Cafieri, B. G.-Tóth, E. Hendrix, L. Liberti and F. Messine (Eds.), *Proceedings of the Toulouse Global Optimization Workshop* (pp. 51–54), 2010.

Paper D

Comparison of discrete and continuous models for the pooling problem²

Mohammed Alfaki¹ and Dag Haugland¹

¹Department of Informatics, University of Bergen,
P.O. Box 7803, N-5020 Bergen, Norway.
mohammeda@ii.uib.no and dag@ii.uib.no

²In: A. Caprara and S. Kontogiannis (Eds.), *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (Vol. 20, pp. 112–121). OpenAccess Series in Informatics (OASICS). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, doi: [10.4230/OASICS.ATMOS.2011.112](https://doi.org/10.4230/OASICS.ATMOS.2011.112), 2011.

Paper E

Computing feasible solutions to the pooling problem²

Mohammed Alfaki¹ and Dag Haugland¹

¹Department of Informatics, University of Bergen,
P.O. Box 7803, N-5020 Bergen, Norway.
mohammeda@ii.uib.no and dag@ii.uib.no

²Submitted to: *Annals of Operations Research*, 2011.

