Solving the pooling problem with LMI relaxations

Lennart Frimannslund[1], Mohamed El Ghami[1],
Mohammed Alfaki[1] and Dag Haugland[1]

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

Abstract

We consider the standard pooling problem with a single quality parameter, which is a polynomial global optimization problem occurring among other places in the oil industry. In this paper, we show that if the feasible set has a nonempty interior, the problem can be solved by a hierarchy of semidefinite relaxations in which the resulting sequences of their optimal values converge to the global optimum. For a fixed relaxation order, this technique provides tight lower bounds for the global objective function value. Based on the experiments, for low order relaxations, the lower bound provided by this method matches the true global optimum in several instances.

Keywords Pooling Problem · Linear matrix inequality · Semidefinite programming · Polynomial optimization · Global optimization

1 Introduction

Consider the problem of transporting oil from producers to consumers, through a pipeline network. Suppose we have two sources of oil, for instance two offshore platforms. Suppose in addition that the oil from both sources contains a contaminant which cannot be above a certain level in order for the oil to be usable.

If there are no purification nodes in the network, the only way to control the level of contaminant (or *quality*) of the oil that reaches the terminals is to blend the oil from the different sources either at the terminals or within the network. One can imagine the oil being blended in a big vat, or pool, which gives rise to the name *pooling problem*. The pooling problem in itself is not inherently linked to oil, such problems can also occur with gas, chemicals, beverages or even food production – anywhere, when two or more source ingredients with a notion of quality can be blended in a network.

For oil, the contaminant can be e.g. sulfur, for natural gas, its contaminant can be CO_2, H_2S, or other components. In other words, there can be more than one quality attribute. In this work, however, we will focus on the situation where there is only one such contaminant.

The pooling problem has been studied for many years, see e.g. (Adhya et al., 1999; Haugland, 2010; Misener and Floudas, 2009) and the references therein for a comprehensive treatment. To the best of our knowledge, all existing global optimization methods for this problem (Adhya et al., 1999; Ben-Tal et al., 1994; Foulds et al., 1992; Quesada and Grossmann, 1995; Sahinidis and Tawarmalani, 2005; Visweswaran and Floudas, 1990) employ branch-and-bound based techniques for searching the feasible domain. In this work, we propose an alternative solution method that based on linear matrix inequality (LMI) relaxations proposed by Lasserre (2001a).

The paper is organized as follows. In Section 2, we introduce the problem under study through a popular instance, and give a general formulation for it. Section 3 describes the LMI relaxations for quadratic polynomial optimization problems. In Section 4, we show how we can apply this technique to the pooling problem. Finally, we present numerical experiments in Section 5, and conclude in Section 6.

2 The standard pooling problem with a single quality

2.1 Haverly's first instance

A frequently studied problem instance was constructed by Haverly (1978), and is henceforth referred to as Haverly1. This instance can be visualized as in Figure 1. As the figure shows, there are three sources on the left, which can provide oil with various levels of sulfur ("S") contamination, at different prices. On the right, there are two terminals, each with an upper bound on the amount of oil needed, which quality is acceptable, and the price they will pay. This is a pooling problem instance because of the structure around node 4.

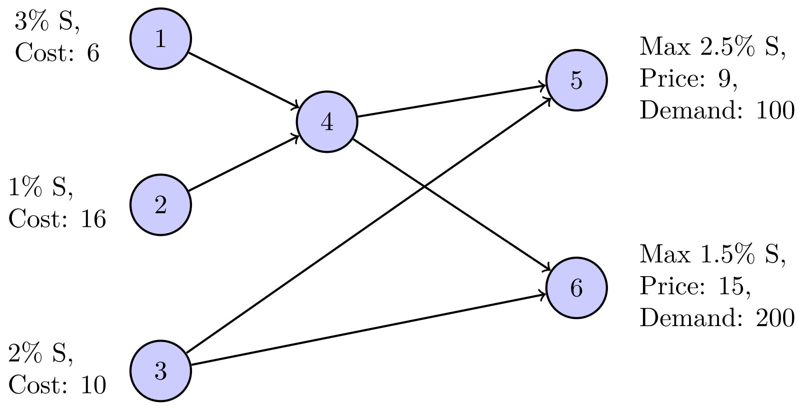

Figure 1: The Haverly1 pooling problem instance.

Here the oil coming in from nodes 1 and 2 is blended, and the sulfur content of the oil exiting node 4 is a weighted average of the sulfur content of the oil entering it. Letting w_4 be the relative sulfur content of the oil exiting node 4, and x_{ij} be the flow from node i to node j, we have:

$$w_4 = \frac{2x_{14} + x_{24}}{x_{14} + x_{24}},$$

which we can also write

$$w_4(x_{14} + x_{24}) = 2x_{14} + x_{24}.$$

This is a bilinear constraint, which in general makes the problem nonconvex and consequently hard to solve. If we wish to minimize the cost, the entire Haverly1 instance can be formulated as:

$$
\begin{aligned}
\min_{x,w} \;& 6x_{14} + 16x_{24} + 10(x_{35} + x_{36}) - 9(x_{35} + x_{45}) - 15(x_{36} + x_{46}), \\
\text{s.t.} \;& x_{35} + x_{45} \leq 100, \\
& x_{36} + x_{46} \leq 200, \\
& x_{14} + x_{24} - x_{45} - x_{46} = 0, \\
& 3x_{14} + x_{24} - w_4(x_{45} + x_{46}) = 0, \\
& 2x_{35} + w_4 x_{45} - 2.5(x_{35} + x_{45}) \leq 0, \\
& 2x_{36} + w_4 x_{46} - 1.5(x_{36} + x_{46}) \leq 0, \\
& x_{14}, x_{24}, x_{35}, x_{36}, x_{45}, x_{46} \geq 0, \qquad (1) \\
& w_4 \geq 0. \qquad (2)
\end{aligned}
$$

The globally optimal solution is:

$$
\begin{bmatrix} x_{14} \\ x_{24} \\ x_{35} \\ x_{36} \\ x_{45} \\ x_{46} \\ w_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 100 \\ 0 \\ 100 \\ 0 \\ 100 \\ 1 \end{bmatrix},
$$

which corresponds to the objective function value -400. Minimizing the cost is not the only possible objective function, for instance one might want to maximize the flow to the customers, which would have given the objective function

$$
\max_{x} \; x_{35} + x_{36} + x_{45} + x_{46},
$$

with the same constraints.

Representing the problem with arc flows as the variables and a linear objective function as we have done is called the P-formulation of the problem. The problem can also be written in other ways, for instance with the so-called Q-formulation

(Ben-Tal et al., 1994), which has a nonlinear objective function. We restrict ourselves to the P-formulation in this work.

2.2 General formulation

In general, we define our problem as follows. Consider a directed graph $G = (N, A)$ with the node set N consisting of sources S, pools I and terminals T. Let the arc set A be such that the graph is connected, and such that all arcs link either a source with a pool, a source with a terminal, or a pool with a terminal.

With each node $i \in N$, we define the node *capacity* b_i. At each source or terminal node $i \in S \cup T$ there is defined a quality q_i and a constant c_i. If $s \in S$, q_s is referred to as the *quality parameter* and c_s is the *cost* at that source. If $t \in T$, q_t is referred to as the *quality bound* and c_t is the *price* of selling at this terminal.

Choosing minimization of the total cost as the objective, we can write the problem as:

$$\min_{x,w} \sum_{s \in S} \sum_{j \in N:(s,j) \in A} c_s x_{sj} - \sum_{t \in T} \sum_{i \in N:(i,t) \in A} c_t x_{it},$$

$$\text{s.t.} \quad \sum_{j \in N:(s,j) \in A} x_{sj} \leq b_s, \qquad s \in S, \tag{3}$$

$$\sum_{j \in N:(j,i) \in A} x_{ji} \leq b_i, \qquad i \in I \cup T, \tag{4}$$

$$\sum_{s \in S:(s,i) \in A} x_{si} - \sum_{t \in T:(i,t) \in A} x_{it} = 0, \qquad i \in I, \tag{5}$$

$$\sum_{s \in S:(s,i) \in A} q_s x_{si} - w_i \sum_{t \in T:(i,t) \in A} x_{it} = 0, \qquad i \in I, \tag{6}$$

$$\sum_{s \in S:(s,t) \in A} q_s x_{sj} + \sum_{i \in I:(i,t) \in A} w_i x_{it} - q_t \sum_{j \in N:(j,t) \in A} x_{jt} \leq 0, \quad t \in T, \tag{7}$$

$$x_{ij} \geq 0, \qquad (i,j) \in A.$$

Inequalities (3) and (4) are the flow capacity constraints at all nodes, and (7) is the quality constraints at each terminal. Equation (5) is the flow conservation constraint for each pool, and Equation (6) is the corresponding quality balance constraint, stating that the amount of the contaminant entering a pool equals the amount leaving it.

As one can see the objective function is linear, and the constraints are either linear or bilinear. In other words, even though the feasible region is nonconvex, both the objective function and the constraints are polynomial, and in fact quadratic. We can therefore apply polynomial optimization techniques.

3 Polynomial optimization by LMI relaxations

We now give a brief outline of the technique of global optimization of (quadratic) polynomials by linear matrix inequality (LMI) relaxations, as presented by Lasserre (2001b). For a more thorough introduction of the case of polynomials of any degree, see (Lasserre, 2001a).

3.1 Moments and moment matrices

For a polynomial function f on $\mathbb{R}^n$, we wish to compute

$$f^* = \min_{x \in K} f(x), \tag{8}$$

where K is a subset of $\mathbb{R}^n$ defined by polynomial constraints

$$g_k(x) \geq 0, \;\; k = 1, 2, \ldots, m.$$

An equivalent problem is

$$\min_{\mu \in \mathcal{P}(K)} \int f(x)\mu(dx),$$

where we minimize over the space of finite Borel signed measures with their support in K. This is an infinite dimensional problem, so instead of determining the measure itself we try to determine its moments y defined as

$$y_\alpha = \int x^\alpha \mu(dx),$$

where α denotes a collection of n indices and x^α is a product of the components $x_1, x_2, \ldots, x_n$ to the power of the corresponding index in α. For example, if $n = 3$, then

$$y_{456} = \int x_1^4 x_2^5 x_3^6 \mu(dx).$$

For moments corresponding to a probability distribution, the moment matrix of order i is a positive semidefinite matrix $M_i(y)$ containing all moments up to order $2i$, which satisfies

$$p^T M_i(y) p = \int [p(x)]^2 \mu_y(dx),$$

where p is a vector of the coefficients of a polynomial, ordered so that they correspond to the entries of the moment matrix. Here, μ_y is the (not necessarily unique) probability distribution corresponding to the moments in $M_i(y)$. If $n = 2$ and $i = 2$ then

$$M_2(y) = \left[\begin{array}{c|cc|ccc} 1 & y_{10} & y_{01} & y_{20} & y_{11} & y_{02} \\ \hline y_{10} & y_{20} & y_{11} & y_{30} & y_{21} & y_{12} \\ y_{01} & y_{11} & y_{02} & y_{21} & y_{12} & y_{03} \\ \hline y_{20} & y_{30} & y_{21} & y_{40} & y_{31} & y_{22} \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & y_{13} \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} \end{array}\right].$$

In this case, p contains the coefficients of the base polynomials

$$1, x_1, x_2, x_1^2, x_1 x_2, x_2^2,$$

in this order. Similarly, there exist moment matrices relating to the constraints. For each constraint $g_k(x) \geq 0,\ k = 1, 2, \ldots, m$, there exists a matrix $M_i(g_k y)$, so that

$$p^T M_i(g_k y) p = \int g_k(x) [p(x)]^2 \mu_y(dx).$$

3.2 LMI Relaxations

These two matrix types are the main ingredients in the following hierarchy of convex LMI relaxations for quadratic problems:

$$\mathbb{Q}_i = \begin{cases} \min_{y} \sum_{\alpha} (g_0)_\alpha y_\alpha, & \\ \text{s.t.} \quad M_i(y) \succeq 0, & \\ M_{i-1}(g_k y) \succeq 0, & k = 1, 2 \ldots, m, \end{cases} \tag{9}$$

where g_0 is a vector containing the coefficients of $f(x)$ ordered according to the order of all the moments $\{y_\alpha\}$ stacked in a vector. The moments y_α are the unknowns. For the different orders of relaxations and the original problem (8), we have (Lasserre, 2001b, Proposition 3.1),

$$\inf \mathbb{Q}_i \leq \inf \mathbb{Q}_{i+1} \leq f^*, \ i = 1, 2, \ldots$$

Under certain conditions we have that, for increasing i, the objective function value

$$\lim_{i \to \infty} \inf \mathbb{Q}_i \uparrow f^* \tag{10}$$

A sufficient condition for (10) to hold that can be applied to the pooling problem is that we can add a redundant constraint of the form

$$g_{m+1}(x) = M - \|x\|^2 \geq 0, \tag{11}$$

for some finite constant M. For the exact requirements, see (Lasserre, 2001a,b). Let $\mathbb{Q}_i^*$ denote the dual of (9). It can be written

$$\mathbb{Q}_i^* = \begin{cases} \max_{X, Z_k} -X_{11} - \sum_{k=1}^{m} (g_k)_0 (Z_k)_{11}, & \\ \text{s.t.} \quad X \bullet B_\alpha + \sum_{k=1}^{m} Z_k \bullet C_\alpha^k = (g_0)_\alpha, & \forall \alpha \neq 0, \\ X, Z_k \succeq 0, & k = 1, 2, \ldots, m. \end{cases} \tag{12}$$

Here, X_{11} and $(Z_k)_{11}$ denote the elements in position (1,1) of these matrices, and $(g_k)_0$ denotes the constant term in the expression $g_k(x) \geq 0$. The operator $\bullet$ denotes the inner product between symmetric matrices, that is, $A \bullet B = trace(AB)$. The matrices B_α and C_α are defined by

$$\sum_{\alpha} y_\alpha B_\alpha \quad = \quad M_i(y), \text{ and}$$

$$\sum_{\alpha} y_\alpha C_\alpha^k \quad = \quad M_{i-1}(g_k y),$$

where the sum is over each component in α, that is, corresponding to each unique member of $M_i(y)$. The y-component corresponding to the constant terms is chosen to be 1. Similarly, the expression $\forall \alpha \neq 0$ in the constraints of (12) means that there is one such constraint for each component of α not equal to zero.

3.3 Finite convergence

In many cases, there exists an integer i_0 such that

$$\max \mathbb{Q}_i^* = \inf \mathbb{Q}_i = f^* \quad \forall\, i \geq i_0, \tag{13}$$

and in some instances i_0 is relatively small, say 2 or 3. If K has a nonempty interior, then a necessary and sufficient condition for (13) to hold is that the polynomial $f(x) - f^*$ can be written as a sum of squares, particularly (see Equation 18.12 in Lasserre, 2001b):

$$f(x) - f^* = \sum_{j=1}^{r_0} [p_j(x)]^2 + \sum_{k=1}^{m} g_k(x) \left[\sum_{j=1}^{r_k} [p_{kj}(x)]^2 \right], \tag{14}$$

where the functions $p_j(x)$ and $p_{kj}(x)$ are polynomials, for all j and all (k, j)-pairs. The coefficients of these polynomials can be retrieved from the dual problem $\mathbb{Q}_i^*$. Specifically, if (14) holds and given the solution to $\mathbb{Q}_i^*$ (e.g. $X, Z_k,\ k = 1, \ldots, m$), then if we let p_j and p_{kj} be the vectors of coefficients for the corresponding polynomials, we have

$$\begin{aligned} \sum_{i=1}^{r_0} p_j p_j^T \quad &= \quad X, \text{ and} \\ \sum_{j=1}^{r_{kj}} p_{kj} p_{kj}^T \quad &= \quad Z_k,\ k = 1, 2, \ldots, m. \end{aligned}$$

If K does *not* have a nonempty interior then the theory of Lasserre (2001b) cannot give an if-and-only-if condition for finite convergence. Nevertheless, as pointed out in the introduction of Lasserre (2001a), from a numerical point of view this is not important, and numerical results are promising for this case as well (Lasserre, 2001b). Recent results regarding the representation of polynomials (Kojima and Muramatsu, 2009) may close this theoretical gap in the future.

Given a solution $\max \mathbb{Q}_i^*$, then the corresponding x-variables for the original problem can be extracted from $M_i(y)$ using the procedure by Henrion and Lasserre (2005) or be read from the first order moments directly. To verify global optimality, the obtained x must be feasible for the original problem, and attain the same objective function value.

If the original polynomial optimization problem has n variables and m constraints, then the i-th order LMI relaxation (9) has $O(n^{2i})$ variables and $m+1$ positive semidefiniteness constraints. The LMI can be cast as an SDP problem (Lasserre, 2008). Such problems can be solved in polynomial time using interior point methods. In other words, any polynomial optimization problem for which there exists an i_0 such that (13) holds in all instances of the problem, is solvable in polynomial time. It is however important to note that i_0 must be independent of the problem instance.

4 LMI relaxations applied to the pooling problem

For the solution framework in Section 3 to work, the set of feasible solutions must have a nonempty interior. This means we will have to eliminate all equality constraints somehow. We show how to do this by example, on the Haverly1 instance.

4.1 Preprocessing the Haverly1 instance

First, we eliminate the flow conservation equation, that is,

$$x_{14} + x_{24} - x_{45} - x_{46} = 0.$$

Let us perform the substitution

$$x_{14} \leftarrow (-x_{24} + x_{45} + x_{46}).$$

This gives the following formulation:

$$
\begin{aligned}
\min_{x,w} \quad & 10x_{24} + x_{35} - 5x_{36} - 3x_{45} - 9x_{46}, \\
\text{s.t.} \quad & x_{35} + x_{45} \leq 100, \\
& x_{36} + x_{46} \leq 200, \\
& -2x_{24} + 3x_{45} + 3x_{46} - w_4(x_{45} + x_{46}) = 0, \\
& 2x_{35} + w_4 x_{45} - 2.5(x_{35} + x_{45}) \leq 0, \\
& 2x_{36} + w_4 x_{46} - 1.5(x_{36} + x_{46}) \leq 0, \\
& -x_{24} + x_{45} + x_{46} \geq 0, \\
& x_{24}, x_{35}, x_{36}, x_{45}, x_{46} \geq 0, \\
& 1 \leq w_4 \leq 3.
\end{aligned}
$$

Note here that we have replaced the constraint $w_4 \geq 0$ in (2) with the tighter constraints $w_4 \leq 3$, and $w_4 \geq 1$. These bounds are easily identified, since the quality of the flow that leaves a pool is bounded by the qualities of the flows entering it. Note also that constraint (1), which includes the nonnegativity constraint $x_{14} \geq 0$ subject to substitution. We continue in the same fashion and use the remaining equality constraint to remove the variable x_{24}, by performing the substitution

$$x_{24} \leftarrow 1.5x_{45} + 1.5x_{46} - 0.5w_4(x_{45} + x_{46}).$$

This gives the formulation:

$$
\begin{aligned}
\min_{x,w} \quad & x_{35} - 5x_{36} + 12x_{45} + 6x_{46} - 5w_4(x_{45} + x_{46}), \\
\text{s.t.} \quad & x_{35} + x_{45} \leq 100, \\
& x_{36} + x_{46} \leq 200, \\
& 2x_{35} + w_4 x_{45} - 2.5(x_{35} + x_{45}) \leq 0, \\
& 2x_{36} + w_4 x_{46} - 1.5(x_{36} + x_{46}) \leq 0, \\
& x_{45} + x_{46} - (1.5x_{45} + 1.5x_{46} - 0.5w_4(x_{45} + x_{46})) \geq 0, \\
& 1.5x_{45} + 1.5x_{46} - 0.5w_4(x_{45} + x_{46}) \geq 0, \\
& x_{35}, x_{36}, x_{45}, x_{46} \geq 0,
\end{aligned}
$$

$$1 \leq w_4 \leq 3.$$

Note that the objective function is now nonlinear, and that the two last constraints correspond to the nonnegativity constraints on the two eliminated variables, $x_{14} \geq 0$ and $x_{24} \geq 0$. We can visualize this reduced formulation as in Figure 2.

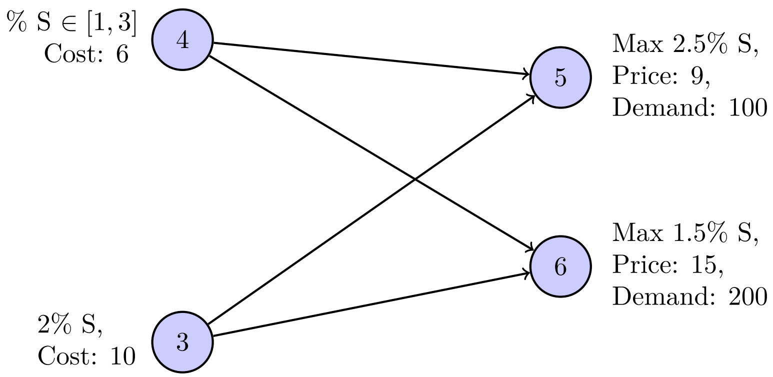

Figure 2: The Haverly1 pooling problem instance, with two variables substituted.

The feasible region has a nonempty interior, since all the inequalities hold strictly, e.g.

$$\begin{bmatrix} x_{35} \\ x_{36} \\ x_{45} \\ x_{46} \\ w_4 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 1 \\ 1 \\ 7/6 \end{bmatrix}.$$

Having eliminated the equality constraints, we can add a constraint of the form (11). In the current instance, it can be

$$(10^5 + 9) - x_{35}^2 - x_{36}^2 - x_{45}^2 - x_{46}^2 - w_4^2 \geq 0.$$

4.2 Preprocessing general instances

All pooling problem instances have a flow conservation and quality balance equation for each pool, but these can be eliminated by the method outlined in the example.

There may be additional equalities present stemming from the bounds on the flow variables, as well as the bounds on the quality entering the pools. In most cases these equalities can be eliminated by repeated application of one or more of the following preprocessing steps:

1. No flow possible because of quality constraints $\Rightarrow$ remove edge.
2. Disconnected node $\Rightarrow$ remove node
3. Quality or flow restricted to one value $\Rightarrow$ replace variable in question with a constant.

However, the feasible set for a formulation with no equality constraints is not always guaranteed to have a nonempty interior, but the converse is true. For example, if we have the network to the left in Figure 3, using the technique we have described, we can eliminate all equality constraints, but the feasible set still has empty interior.

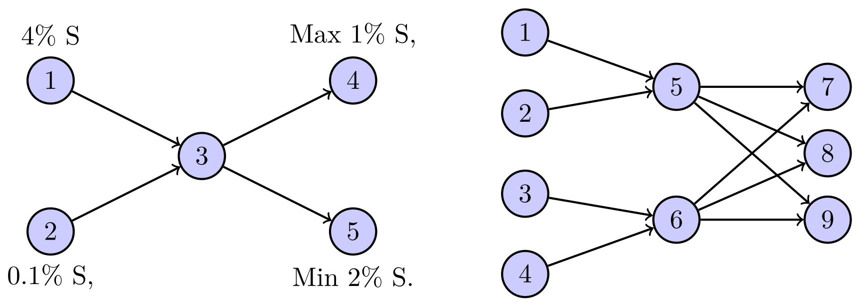

Figure 3: To the left, an instance where the feasible set has a nonempty interior. To the right the network used in the experiments.

5 Numerical experiments

We test the LMI relaxation technique using the software package Gloptipoly 3 (Henrion et al., 2009). First we try a few instances from the literature, listed in the first five rows of Table 2. All of these are defined in e.g. (Adhya et al., 1999).

As Table 2 reflects, the first order relaxation only finds lower bounds on global optimum f^* of the original problem. For order 2 we have $\max \mathbb{Q}_2^* = f^*$ and Gloptipoly is able to extract the globally optimal solution x^* for all but the Foulds2 instance. Foulds2 has infinitely many globally optimal solutions corresponding the same value of the qualities at its two pools. One such solution is identifiable from the first-order moments in the moment matrix $M_2(y)$, but the software is not able to detect this, presumably for numerical reasons. It should, namely, be noted that all of the instances tested are extremely sensitive to scaling, and most can only be solved successfully with Gloptipoly with the variables scaled to their expected magnitude at the optimal solution. Since the problem is invariant to scaling this is a numerical, not a theoretical issue. Unfortunately we are not able to solve Foulds2 with relaxation order 3 due to lack of memory on the computer used for testing.

We also generated some instances, using the graph to the right in Figure 3. The different values for the parameters and constraints are found in Table 1.

The first instance is defined to have a nonempty interior and consistency between quality and price. The rest were randomly generated. As it turns out, instances B and D have empty interiors. For B, no flow can reach the second of third terminals because of their strict constraints on the quality. For D, no flow can reach the first and second terminals, for the same reason. We therefore construct two versions of each of these instances. Instance B and D denote the original instances, and B2 and D2 denote the same instances with the unreachable terminals removed. The results in these instances are reported in the bottom half of Table 2.

Table 1: Parameter settings for test instances.

	Sources			Terminals		
	Cost	Quality parameter	Flow capacity	Price	Quality bound	Flow capacity
A	$\begin{bmatrix} 10 \\ 4 \\ 5 \\ 2 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 3 \\ 2 \\ 4 \end{bmatrix}$		$\begin{bmatrix} 20 \\ 7 \\ 5 \end{bmatrix}$	$\leq \begin{bmatrix} 1.5 \\ 2.5 \\ 3.5 \end{bmatrix}$	$\leq \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix}$
B	$\begin{bmatrix} 8.5577 \\ 6.7080 \\ 5.2359 \\ 2.9882 \end{bmatrix}$	$\begin{bmatrix} 7.0397 \\ 3.8161 \\ 5.6768 \\ 8.8786 \end{bmatrix}$		$\begin{bmatrix} 0.8747 \\ 2.6073 \\ 0.2280 \end{bmatrix}$	$\leq \begin{bmatrix} 8.1544 \\ 0.0136 \\ 0.0309 \end{bmatrix}$	$\leq \begin{bmatrix} 84.2949 \\ 89.8799 \\ 93.9003 \end{bmatrix}$
C	$\begin{bmatrix} 6.4832 \\ 6.1467 \\ 4.6965 \\ 5.7778 \end{bmatrix}$	$\begin{bmatrix} 9.1131 \\ 3.7622 \\ 2.2876 \\ 4.2352 \end{bmatrix}$	$\leq \begin{bmatrix} 6.1825 \\ 23.1367 \\ 11.8486 \\ 9.8780 \end{bmatrix}$	$\begin{bmatrix} 3.0035 \\ 4.0003 \\ 5.1772 \end{bmatrix}$	$\leq \begin{bmatrix} 5.3464 \\ 3.8544 \\ 8.7345 \end{bmatrix}$	$\leq \begin{bmatrix} 27.3596 \\ 44.4566 \\ 62.7515 \end{bmatrix}$
D	$\begin{bmatrix} 8.1472 \\ 9.0579 \\ 1.2699 \\ 9.1338 \end{bmatrix}$	$\begin{bmatrix} 6.3236 \\ 0.9754 \\ 2.7850 \\ 5.4688 \end{bmatrix}$		$\begin{bmatrix} 8.0028 \\ 1.4189 \\ 4.2176 \end{bmatrix}$	$\geq \begin{bmatrix} 9.7059 \\ 9.5717 \\ 4.8538 \end{bmatrix}$	$\leq \begin{bmatrix} 95.7507 \\ 96.4889 \\ 15.7613 \end{bmatrix}$

Table 2: Results from five instances from the literature and the instances defined in Table 1.

Name	max $\mathbb{Q}_i^*$ for relaxation order			Notes
	1	2	3	
Haverly1	-600	-400	–	x^* found at order 2
Haverly2	-1200	-600	–	x^* found at order 2
Haverly3	-875	-750	–	x^* found at order 2
BenTal4	-600	-450	–	x^* found at order 2
Foulds2	-1200	-1100	–	Sol. in $M_2(y)$
A	-1925	-1553	-1541	x^* found at order 3
B	0	–	–	x^* found at order 1
B2	0	–	–	x^* found at order 1
C	-5.69	-5.69	–	x^* found at order 2
D	0	–	–	x^* found at order 1
D2	0	–	–	x^* found at order 1

Two things happen here that is different from the results of the experiments with the instances from the literature. One is that for instance A, the solution is not found with relaxation order 2, but that order 3 is needed. Secondly, for instances with zero as their solution the global optimum is found and verified for order 1.

Table 3: Results from experiments with max flow instances.

Name	max $\mathbb{Q}_i^*$ for relaxation order			Notes
	1	2	3	
Haverly1	300	300	–	x^* found at order 2
Haverly2	800	800	–	x^* found at order 2
Haverly3	300	300	–	x^* found at order 2
BenTal4	300	300	–	x^* found at order 2
Foulds2	600	600	–	x^* found at order 2
A	300	300	–	x^* found at order 2
B	181	84.30	84.29	x^* found at order 3
B2	84.29	–	–	x^* found at order 1
C	51.04	51.04	–	x^* found at order 2
D	22.89	15.76	15.76	x^* found at order 3
D2	15.76	15.76	–	x^* found at order 2

We also test the same instances, but with a max flow objective function.

These results are reported in Table 3. For the max flow objective function, the instances with an empty interior requires a higher relaxation order than for min cost, namely order 3. In the remaining instances relaxation order 1 always provides the correct objective function value, and in the B2 instance, this applies to the variables as well. In the rest of the instances, relaxation order 2 is needed in order to obtain the optimal variable values.

6 Conclusion

We use LMI relaxations for solving the standard pooling problem with a single quality. Based on our experiments, small instances of this problem can be solved with low LMI relaxation orders, provided that they can be formulated in such a way that they have a nonempty interior. However, solving such relaxation implies a large computational effort, which for large instances makes the method impractical to use. Current research on solution methods for sparse semidefinite optimization has good promise to ease this limitation.

Acknowledgements This research was sponsored by the Norwegian Research Council, Gassco, and Statoil under contract 175967/S30

References

Adhya, N., Tawarmalani, M., and Sahinidis, N.V. (1999). A Lagrangian approach to the pooling problem. *Industrial & Engineering Chemistry Research*, **38** (5), 1956–1972.

Ben-Tal, A., Eiger, G., and Gershovitz, V. (1994). Global minimization by reducing the duality gap. *Mathematical Programming*, **63** (2), 193–212.

Foulds, L.R., Haugland, D., and Jörnsten, K. (1992). A bilinear approach to the pooling problem. *Optimization*, **24** (1), 165–180.

Haugland, D. (2010). An overview of models and solution methods for pooling problems. In E. Bjørndal, M. Bjørndal, P. Pardalos and M. Rönnqvist. (Eds.), *Energy, Natural Resources and Environmental Economics* (pp. 459–469). Springer.

Haverly, C.A. (1978). Studies of the behavior of recursion for the pooling problem. *ACM SIGMAP Bulletin*, **25**, 19–28.

Henrion, D., and Lasserre, J.B. (2005). Detecting global optimality and extracting solutions in GloptiPoly. *Positive polynomials in control*, 293–310.

Henrion, D., Lasserre, J.B., and Löfberg, J. (2009). GloptiPoly 3: moments, optimization and semidefinite programming. *Optimization Methods & Software*, **24** (4-5), 761–779.

Kojima, M., and Muramatsu, M. (2009). A note on sparse SOS and SDP relaxations for polynomial optimization problems over symmetric cones. *Computational Optimization and Applications*, **42** (1), 31–41.

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. Paradalos (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. (2008). A semidefinite programming approach to the generalized problem of moments. *Mathematical Programming*, **112** (1), 65–92.

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.

Quesada, I., and Grossmann, I.E. (1995). Global optimization of bilinear process networks with multi-component flows. *Computers & Chemical Engineering*, **19** (12), 1219–1242.

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.

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.