ORIGINAL PAPER

Exploiting vector space properties to strengthen the relaxation of bilinear programs arising in the global optimization of process networks

Juan P. Ruiz · Ignacio E. Grossmann

Abstract   In this paper we present a methodology for finding tight convex relaxations for a special set of quadratic constraints given by bilinear and linear terms that frequently arise in the optimization of process networks. The basic idea lies on exploiting the interaction between the vector spaces where the different set of variables are defined in order to generate cuts that will tighten the relaxation of traditional approaches. These cuts are not dominated by the McCormick convex envelopes and can be effectively used in conjunction with them. The performance of the method is tested in several case studies by implementing the resulting relaxation within a spatial branch and bound framework.

Keywords   Global optimization · Bilinear programs · Vector spaces · Tight formulations · Process networks

1 Introduction

The optimization of process networks is one of the most frequent problems that is addressed in process systems engineering [1]. In general, process networks are composed of a set of nodes (N) connected by a set of streams (S). Each stream in S is associated with a flow F and a set of properties J whose values P^j are relative to F. The flows and properties of the set of streams leaving a node (O_n) are related to the flow and properties of the set of streams entering the node (I_n) and the characteristics of the node itself. The most frequent subset of equations found in these networks is represented by the following set of quadratic and linear constraints:

J. P. Ruiz · I. E. Grossmann (✉)
Carnegie Mellon University, Pittsburgh, USA
e-mail: grossmann@cmu.edu

$$\sum_{i \in I_n} (F_{in} P_{in}^j) - F_{on} P_{on}^j = 0 \quad \forall n \in N, \forall j \in J \tag{1}$$

$$\sum_{i \in I_n} F_{in} - F_{on} = 0 \quad \forall n \in N \tag{2}$$

where F_{in} represents the flow of the entering stream $i \in I$ at a node $n \in N$ and P_{in}^j the value for the property j in that stream. Similarly, F_{on} and P_{on}^j are the characteristic values for the outlet stream. Without loss of generality we assume that only one stream leaves the node, namely, $|O_n| = 1$.

The constraints in (1) and (2) typically correspond to mass or energy balances. Although very simple in its representation, this set of constraints define a nonconvex region, and when embedded in the final formulation, requires global optimization techniques for solving the corresponding problem. Most of the practical solution methods to solve these problems rely on some variation of a spatial branch and bound framework [2] whose performace, in turn, heavily depends on generating tight relaxations.

In order to find a relaxation for (1) and (2), the most frequent technique is based on the work by McCormick [3] in which each bilinear term is replaced by its convex envelope [11]. Note that this approach is a particular case of the Reformulation-Linearization Technique (RLT) [9] in which cuts are constructed by multiplying constraints by appropriate variables and then linearizing the resulting bilinear terms. Efficient implementations of this approach for large scale problems were studied by Liberti and Pantelides [12]. Other techniques consider the convex envelope of the summation of the bilinear terms [4], the semidefinite relaxation of the whole set of bilinear terms [5] or the piecewise linear relaxations [13,14].

The major objective of this paper is to present a methodology for finding tighter relaxations by considering the interaction between the vector space defined by the property values with the one defined by the flow values.

This paper is organized as follows. In Sect. 2 we present a vectorial representation of the set of constraints (1) and (2), and introduce the concept of "minimal set" as the building blocks of these systems. In Sect. 3 we study the properties of the "minimal sets". In Sect. 4 we introduce the traditional relaxation for bilinear terms, and in Sects. 5 and 6 we describe our methodology by making extensive use of the properties presented in Sect. 3.

Finally, in Sect. 7 we present a set of numerical results.

2 Vectorial representation

Let us consider a node $n \in N$ in the network as represented in Fig. 1, with a set of inlet streams I_n and an outlet stream O_n.

Fig. 1 Inlet–outlet stream representation for node n

$$F_{in} P_{in}^j \quad\quad\quad\quad F_{on} P_{on}^j$$
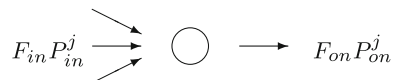
where $F_{in}P_{in}^j$ represents the flow of property j entering the node through stream in. As described in Sect. 1, the system is modeled by Eqs. (1) and (2). Without loss of generality we will consider the analysis for a single node and a single property, and consequently, we will not use indices n and j.

Defining the vectors in $R^{|I|+1}$, $v_F := (F_1, F_2, F_3, \ldots, F_{|I|}, F_o)$, $v_P := (P_1, P_2, P_3, \ldots, P_{|I|}, -P_o)$ and $v_E := (1, 1, 1, \ldots, 1, -1)$ we can rewrite (1) as:

$$v_P \cdot v_F = 0 \tag{3}$$

and (2) as:

$$v_E \cdot v_F = 0 \tag{4}$$

this leads to the following compact vectorial representation:

$$\begin{aligned} v_P \perp v_F \\ v_E \perp v_F \end{aligned} \tag{5}$$

We call this form a *3-Vector Representation*, which clearly exposes the interaction between the vector spaces of properties v_P, flows v_F and the unit vector v_E.

In order to facilitate the analysis, we propose a decomposition of the 3-Vector Representation by using *minimal sets*. We define the *minimal set* as the set composed by three elements (i.e. $|I| + 1 = 3$).

Lemma 1 *Any system of the form (5) can be decomposed as the intersection of $|I|-1$ 3-Vector Representation of minimal sets*

Proof Given the set of vectors in $R^{|I|+1}$, $v_F := (F_1, F_2, F_3, \ldots, F_{|I|}, F_o)$, $v_P := (P_1, P_2, P_3, \ldots, P_{|I|}, -P_o)$ and $v_E := (1, 1, 1, \ldots, 1, -1)$ we can find an equivalent representation of (5) by the application of the following recursive operation.

Generate the kth *3-Vector Representation of minimal set* by:

$$\begin{aligned} v_F^k &:= (F_{1k}, F_{k+1}, F_{1k+1}) \\ v_E^k &:= (1, 1, -1) \\ v_P^k &:= (P_{1k}, P_{k+1}, -P_{1k+1}) \end{aligned}$$

where F_{1k} and P_{1k} are new variables that are generated and F_{11} and P_{11} are equal to F_1 and P_1 respectively, and similarly $F_{1|I|}$ and $P_{1|I|}$ are equal to F_o and P_o, respectively. The recursive procedure stops when $k + 1 = |I|$ ☐

Figure 2 presents the illustration of Lemma 1 for $|I| = 4$.

Fig. 2 Decomposition in minimal sets $|I| = 4$
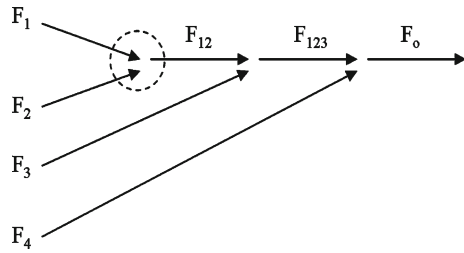
3 Vector space properties for the minimal set

In this section we study the properties of the space defined through a minimal set (i.e. $|I| + 1 = 3$).

Given a minimal set

$$v_P \perp v_F$$
$$v_E \perp v_F$$

(6)

we can establish the following property which relates the vector spaces of v_P and v_F with v_E.

Lemma 2 *The property vectors and flow vectors in a minimal set are related as follows:*

$$v_P \perp v_F \wedge v_E \perp v_F \Rightarrow v_P \times v_E \parallel v_F$$
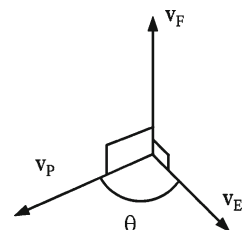
(7)

or equivalently:

$$v_E \cdot v_F = 0, v_P \cdot v_F = 0 \Rightarrow v_P \times v_E = \alpha v_F$$

(8)

where the operator $\times$ *represents the cross product in 3-D,* $\alpha = \frac{\|v_P\|\sqrt{3}\sin\theta}{\|v_F\|}$ *and* $0 \le \theta \le 2\pi$

Proof The proof trivially follows from the dot and cross product in 3-D [10].   □

Figure 3 is the illustration of Lemma 2.

Fig. 3 Interaction between vector spaces in a minimal set

Although the space defined by the Eq. (1) is clearly nonconvex, the question that arises is whether the space defined by (1) and (2) is also nonconvex. The following property describes the convexity characteristics of the space defined by the minimal set.

Lemma 3 *The space defined by the 3-Vector Representation of a minimal set is non-convex*

Proof It suffices to show that there exists a point between two points belonging to the set that does not belong to the set.

Given the point in $R^{3|I|+3}$, $v^1_{PFE} = \left(v^1_P, v^1_F, v^1_E\right) = (1, 1, -1, 2, 1, 3, 1, 1, -1)$ in the set and $v^2_{PFE} = \left(v^2_P, v^2_F, v^2_E\right) = (3, 1, -2, 1, 1, 2, 1, 1, -1)$ also in the set. It is easy to show that the point which is a convex combination of both, namely, $v^{12}_{PFE} = 0.5v^1_{PFE} + 0.5v^2_{PFE} = \left(v^{12}_P, v^{12}_F, v^{12}_E\right) = (2, 1, -1.5, 1.5, 1, 2.5, 1, 1, -1)$ is not in the set. Note that $v^{12}_P \cdot v^{12}_F \neq 0$. □

4 Convex relaxation of the minimal set

The traditional approach for finding a relaxation for the nonconvex set defined by the 3-Vector Representation is by considering the set in algebraic form and using the McCormick convex envelopes for the bilinear terms [3].

Namely, the system given by (1) and (2) can be relaxed by:

$$\sum_{i=1}^{2} f_{Pi} - f_{Po} = 0 \tag{9}$$

$$\sum_{i=1}^{2} F_i - F_o = 0 \tag{10}$$

$$\begin{aligned}
f_{Pi} &\leq P_i F_i^{up} + P_i^{lo} F_i - P_i^{lo} F_i^{up} \\
f_{Pi} &\leq P_i F_i^{lo} + P_i^{up} F_i - P_i^{up} F_i^{lo} \\
f_{Pi} &\geq P_i F_i^{lo} + P_i^{lo} F_i - P_i^{lo} F_i^{lo} \\
f_{Pi} &\geq P_i F_i^{up} + P_i^{up} F_i - P_i^{up} F_i^{up}
\end{aligned} \quad i = 1, 2, o \tag{11}$$

This strategy, which is widely used, does not fully exploit the interaction of the different vectors in the flow and property space, but only the interactions between individual components of these vectors.

5 Cuts to strengthen the relaxation

The convex relaxation presented in the previous section does not consider the interaction between vectors defined in the different spaces. In this section we show that by considering this interaction we can generate constraints that will tighten the relaxation.

It is clear that by following the traditional approach, the orthogonality between v_F and v_P is not captured. In order to circumvent this problem, we propose to add the following constraint.

$$v_P \times v_E = \alpha v_F \tag{12}$$

where $\alpha = \frac{\|v_P\|\sqrt{3}\sin\theta}{\|v_F\|}$ and $0 \leq \theta \leq 2\pi$, which in algebraic form reads:

$$\begin{aligned} -P_2 + P_o &= \alpha F_1 \\ P_1 - P_o &= \alpha F_2 \\ P_1 - P_2 &= \alpha F_o \end{aligned} \tag{13}$$

Since these constraints are nonconvex we need to relax them previous to their use as cuts. We apply the McCormick envelopes to αF_1, αF_2 and αF_o in conjunction with Eq. (13), yielding:

For αF_1

$$\begin{aligned} -P_2 + P_o &\leq \alpha F_1^{up} + \alpha^{lo} F_1 - \alpha^{lo} F_1^{up} \\ -P_2 + P_o &\leq \alpha F_1^{lo} + \alpha^{up} F_1 - \alpha^{up} F_1^{lo} \\ -P_2 + P_o &\geq \alpha F_1^{lo} + \alpha^{lo} F_1 - \alpha^{lo} F_1^{lo} \\ -P_2 + P_o &\geq \alpha F_1^{up} + \alpha^{up} F_1 - \alpha^{up} F_1^{up} \end{aligned} \tag{14}$$

For αF_2

$$\begin{aligned} P_1 - P_o &\leq \alpha F_2^{up} + \alpha^{lo} F_2 - \alpha^{lo} F_2^{up} \\ P_1 - P_o &\leq \alpha F_2^{lo} + \alpha^{up} F_2 - \alpha^{up} F_2^{lo} \\ P_1 - P_o &\geq \alpha F_2^{lo} + \alpha^{lo} F_2 - \alpha^{lo} F_2^{lo} \\ P_1 - P_o &\geq \alpha F_2^{up} + \alpha^{up} F_2 - \alpha^{up} F_2^{up} \end{aligned} \tag{15}$$

For αF_o

$$\begin{aligned} P_1 - P_2 &\leq \alpha F_o^{up} + \alpha^{lo} F_o - \alpha^{lo} F_o^{up} \\ P_1 - P_2 &\leq \alpha F_o^{lo} + \alpha^{up} F_o - \alpha^{up} F_o^{lo} \\ P_1 - P_2 &\geq \alpha F_o^{lo} + \alpha^{lo} F_o - \alpha^{lo} F_o^{lo} \\ P_1 - P_2 &\geq \alpha F_o^{up} + \alpha^{up} F_o - \alpha^{up} F_o^{up} \end{aligned} \tag{16}$$

Remark If $F_o^{up} = F_1^{up} + F_2^{up}$ and $F_o^{lo} = F_1^{lo} + F_2^{lo}$ as it happens when F_o is an intermediate flow that arises from the decomposition of a node with more than two inlet streams, Eqs. (16) are redundant and should not be used in the relaxation.

It is clear that the strength of the proposed cuts heavily relies on the bounds of α. In the following section we describe how to obtain them. We also note that in this derivation we assume $\|v_F\|$ to be strictly positive.

6 Obtaining bounds for α

Note that in the previous section explicit bounds of α are necessary. A direct way to obtain them is by considering the norm of the vector resulting from a cross product. In other words, we can use $\alpha^{up} = \max \left(\frac{\|v_P\|\sqrt{3}\sin\theta}{\|v_F\|} \right) = \frac{\|v_P\|_{max}\sqrt{3}}{\|v_F\|_{min}} = \frac{\|v_{P^{up}}\|\sqrt{3}}{\|v_{F^{lo}}\|}$. Similarly, $\alpha^{up} = -\alpha^{lo}$

Although the bounds proposed are valid, these might not be tight. In order to tighten these bounds we can use Eqs. (13) to infer them.

$$\alpha^{up} = \min \left(\max \frac{-P_2 + P_o}{F_1}, \max \frac{P_1 - P_o}{F_2}, \max \frac{-P_2 + P_1}{F_o} \right)$$

Similarly,

$$\alpha^{lo} = \max \left(\min \frac{-P_2 + P_o}{F_1}, \min \frac{P_1 - P_o}{F_2}, \min \frac{-P_2 + P_1}{F_o} \right)$$

Clearly, we can calculate these bounds as a function of the lower and upper bounds of the flows and properties as follows:

$$\alpha^{up} = \min \left(\begin{array}{c} \max \left(\frac{-P_2^{lo}+P_o^{up}}{F_1^{up}}, \frac{-P_2^{lo}+P_o^{up}}{F_1^{lo}} \right), \\ \max \left(\frac{P_1^{up}-P_o^{lo}}{F_2^{up}}, \frac{P_1^{up}-P_o^{lo}}{F_2^{lo}} \right), \\ \max \left(\frac{-P_2^{lo}+P_1^{up}}{F_o^{up}}, \frac{-P_2^{lo}+P_1^{up}}{F_o^{lo}} \right) \end{array} \right)$$

and,

$$\alpha^{lo} = \max \left(\begin{array}{c} \min \left(\frac{-P_2^{up}+P_o^{lo}}{F_1^{up}}, \frac{-P_2^{up}+P_o^{lo}}{F_1^{lo}} \right), \\ \min \left(\frac{P_1^{lo}-P_o^{up}}{F_2^{up}}, \frac{P_1^{lo}-P_o^{up}}{F_2^{lo}} \right), \\ \min \left(\frac{-P_2^{up}+P_1^{lo}}{F_o^{up}}, \frac{-P_2^{up}+P_1^{lo}}{F_o^{lo}} \right) \end{array} \right)$$

The proposed cuts differ from the ones obtained by using the traditional approach [3], as it is proved in the following proposition.

Proposition 1 *The proposed cuts* (14), (15) *and* (16) *are not dominated by the McCormick convex envelopes* (11).

Proof It suffices to show that a point lying in the relaxed region using the McCormick envelopes is cut off when using the proposed approach. The following example demonstrates this.

Given the minimal set:

$$F_1 P_1 + F_2 P_2 - F_o P_o = 0 \tag{17}$$

$$F_1 + F_2 - F_o = 0 \tag{18}$$

where $0.5 \leq F_1 \leq 2, 1.5 \leq F_2 \leq 2.5, 2 \leq F_o \leq 4.5, 0.5 \leq P_1 \leq 1.5, 0 \leq P_2 \leq 2,$
$0.1 \leq P_o \leq 1.9$

At the point $F_1 = 0.5, F_2 = 2.3, F_o = 2.8, P_1 = 1.2, P_2 = 0.1, P_o$ varies in the interval [0.20,0.38] using the McCormick relaxation whereas P_o varies in the interval [0.26,0.31] when using the proposed relaxation. Note that the value obtained for P_o by solving the original nonconvex problem is $P_o = 0.30$ $\qquad\square$

7 Numerical performance

In this section we study the performance of the proposed relaxation with a problem found in the Process Systems Engineering area known as the *Data Reconciliation in Process Networks*. The optimization, control or operation of a process often relies on having measurements (e.g. flows, compositions) that can be used to adjust models. In general, the number of sensors in a plant is larger than the degrees of freedom of the models. However, the information they provide is often not accurate due to noise of measurements or failures in the sensors. In order to infer better values for the measurements, previous to their use for optimization or control, a data reconciliation is necessary [6].

The first example (Instance 1) consists of a mixer with three inputs and one output (see Fig. 4). Given the measurements of the inlet and outlet flows as well as their correspondent properties, the goal is to find a set of new values that minimize the errors while satisfying the mass and property balances.
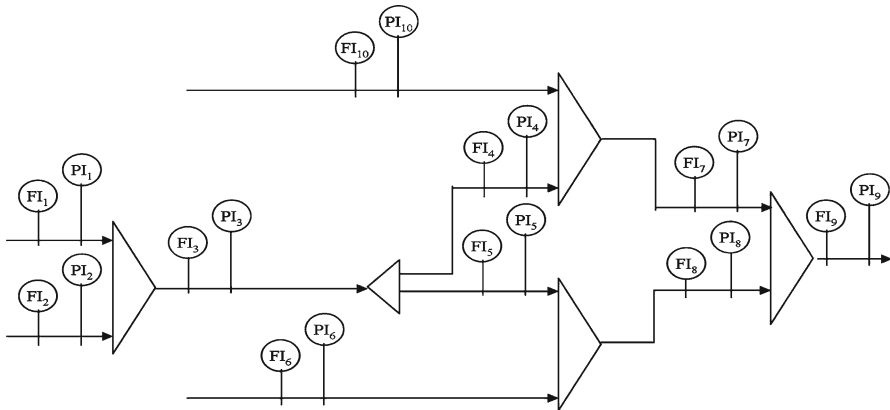
The model for this problem can be represented as follows:

$$\min Z = w_1 \left(F_1 - FI_1\right)^2 + w_2 \left(F_2 - FI_2\right)^2 + w_3 \left(F_3 - FI_3\right)^2 + w_4 \left(F_4 - FI_4\right)^2$$
$$w_5 \left(P_1 - PI_1\right)^2 + w_6 \left(P_2 - PI_2\right)^2 + w_7 \left(P_3 - PI_3\right)^2 + w_8 \left(P_4 - PI_4\right)^2$$
$$s.t. \quad F_1 P_1 + F_2 P_2 + F_3 P_3 = F_4 P_4$$
$$F_1 + F_2 + F_3 = F_4 \tag{19}$$
$$F_1^{lo} \leq F_1 \leq F_1^{up}, F_2^{lo} \leq F_2 \leq F_2^{up}, F_3^{lo} \leq F_3 \leq F_3^{up}, F_4^{lo} \leq F_4 \leq F_4^{up}$$
$$P_1^{lo} \leq P_1 \leq P_1^{up}, P_2^{lo} \leq P_2 \leq P_2^{up}, P_3^{lo} \leq P_3 \leq P_3^{up}, P_4^{lo} \leq P_4 \leq P_4^{up}$$

Fig. 4 Mixer with sensors
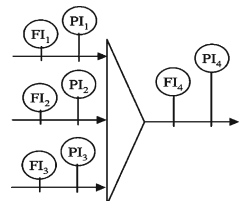
Fig. 5 Process network with sensors

where FI_1, FI_2, FI_3 and FI_4 correspond to measured values of the flows and PI_1, PI_2, PI_3, PI_4 correspond to the measured values of the property considered.

The second example consists of a network of mixers and splitters (see Fig. 5). Given the measurements of the inlet and outlet flows of each mixer as well as their correspondent properties, the goal is to find a set of new values that minimize the errors with the measurements while satisfying the mass and property balances. For this case we present three instances (Instances 2–4) with different sets of measurements.[1] The model for this problem is an extension of the model in (19).

The performance of the proposed method is evaluated by implementing the proposed relaxation in a branch and bound framework, and comparing the lower bound obtained at the root node, the number of nodes necessary to find the solution and the time required by the traditional McCormick relaxation. The particular implementation of the branch and bound method is described as follows. The first step consists of finding an optimal or suboptimal solution of the problem to obtain an upper bound. This is accomplished by solving the original NLP with a local optimizer such as GAMS/CONOPT2 [8]. By using the result obtained in the previous step, a bound contraction of each continuous variable is performed [7]. This is done by solving min/max subproblems in which the objective function is the value of the continuous variable to be contracted subject to the condition that the objective of the original problem is less than the upper bound. After the relaxed feasible region is contracted, a spatial branch and bound search is performed in which the relaxed feasible region is split recursively into subproblems that are eliminated when it is established that their descendents cannot hold a better solution than the one obtained so far. The splitting is performed according to a branching rule, and the way to decide when to eliminate the subproblems is by comparing the lower bound LB (i.e. the solution of the subproblem) with the upper bound UB (i.e. the lowest feasible solution in the original problem obtained so far, which can be accomplished by using the original NLP); if UB–LB

---

[1] Data for these instances is available upon request from the authors.

Table 1 Performance of the branch and bound using different relaxation strategies

Instance	GO	Traditional approach			Proposed approach		
		LB	Nodes	Time (s)	LB	Nodes	Time (s)
1	151.96	142.1	35	30	144.3	17	17
2	5.19	4.86	87	109	5.05	6	22
3	9.79	8.22	650	542	8.78	365	325
4	12.60	12.48	5	19	12.60	1	17

$\leq \epsilon$, where ϵ is a given tolerance, then the node (i.e. subproblem) is eliminated or pruned. The search terminates when no more nodes remain.

In the implementation of this work we have chosen to branch on the property variables by choosing the one belonging to the bilinear term that most violates the feasible region in the original problem (i.e. the violation to the feasible region is computed by taking the difference between the bilinear term and the associated relaxed variable). To generate the subproblems when branching on the continuous variables, we split their domain by using the bisection method. To choose the node to branch next, we follow the "Best First" heuristic, which consists in considering the subproblem with lowest lower bound LB.

As it is shown in Table 1, when using the proposed relaxation in conjunction with the McCormick envelopes, the lower bound obtained at the root node is closer to the global optimum (GO) and the number of nodes needed to find the solution is smaller compared to the traditional approach when only the McCormick envelopes are applied to the model in (19). These results provide a clear indication of tightening. Also, it is important to note that the increase in the size of the formulation by the addition of the proposed cuts, which can be estimated as $8(n - 1) + 4$ new constraints and $3(n - 1) - 2$ new variables per node with n inlet streams, is compensated by the tightening they produce. This can be seen by the significant decrease in the computational time required. All problems were solved using a Pentium(R) CPU 3.40 GHz and 1 GB of RAM, considering an $\epsilon = 10^{-2}$ and the beginning incumbent equal to the global optimum.

8 Conclusions

In this paper, we have presented a method for finding tight convex relaxations for a special set of quadratic constraints given by bilinear and linear terms that commonly arise in the optimization of process networks. The basic idea lies on exploiting the interaction between the vector spaces where the different set of variables are defined to tighten the relaxation of traditional approaches. In order to do so, we first described the system in vectorial form exposing the interaction of the different vector spaces. Second, we defined and characterized the elementary building blocks of the system given by the "minimal sets". This has allowed us to understand this interaction and generate from its properties cuts that tighten the original relaxation. Finally, we have

assessed the performance of the proposed method by testing it in four test problems within a spatial branch and bound framework.

References

1. Quesada, I., Grossmann, I.E.: Global optimization of bilinear process networks with multicomponent flows. Comput. Chem. Eng. **19**(12), 1219–1242 (1995)
2. Horst, R., Tuy, H.: Global Optimization Deterministic Approaches, 3rd edn. Springer-Verlag, Berlin (1996)
3. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs. Part I: convex underestimating problems. Math. Program. **10**, 146–175 (1976)
4. Tawarmalani, M., Sahinidis, N.: Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming. Kluwer Academic Publishers, Dordrecht (2002)
5. Anstreicher, K.M.: Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. J. Glob. Optim. **43**(2–3), 471–484 (2009)
6. Narasimhan, S.: Data Reconciliation & Gross Error Detection: An Intelligent Use of Process Data. Gulf Professional Publishing, Houston (2001)
7. Zamora, J.M., Grossmann, I.E.: A branch and bound algorithm for problems with concave univariate, bilinear and linear fractional terms. J. Glob. Optim. **14**(3), 217–249 (1999)
8. Drud, S.A.: CONOPT—a large-scale GRG code. ORSA J. Comput. **6**, 207–216 (1992)
9. Sherali, H.D., Alameddine, A.: A new reformulation linearization technique for bilinear programming problems. J. Glob. Optim. **2**, 379–410 (1992)
10. Kreiszig, E.: Advanced Engineering Mathematics. Wiley, New York (2001)
11. Al-Khayyal, F.A., Falk, J.E.: Jointly constrained biconvex programming. Math. Oper. Res. **8**(2), 273–286 (1983)
12. Liberti, L., Pantelides, C.C.: An exact reformulation algorithm for large nonconvex NLPs involving bilinear terms. J. Glob. Optim. **36**(2), 161–189 (2006)
13. Gounaris, C.E., Misener, R., Floudas, C.A.: Computational comparison of piecewise-linear relaxations for pooling problems. Ind. Eng. Chem. Res. **48**(12), 5742–5766 (2009)
14. Wicaksono, D.S., Karimi, I.A.: Piecewise MILP under- and overestimators for global optimization of bilinear programs. AICHE J. **54**, 991–1008 (2008)