Effective RLT Tightening in Continuous Bilinear Programs

LEO LIBERTI

Dipartimento di Elettronica e Informazione
Politecnico di Milano

(liberti@elet.polimi.it)

20 March 2003

Abstract

In this paper we give a theoretical analysis of the Reformulation-Linearization Technique (RLT) of H. Sherali when applied to continuous bilinear problems with linear equality constraints. From the analysis we derive a method that identifies "good" subsets of RLT constraints to add to the problem relaxation in order to make it as tight as possible.

1 Introduction

The Reformulation-Linearization Technique (RLT), first described by H. Sherali [SA92, She98, SA86, SW01, SA99, SSA00, She02], is a very efficient method for computing tight convex relaxations of various types of nonconvex problems. This is extremely useful in Branch-and-Bound algorithms where a tight lower bound on the objective function value is required at each step, which is usually computed by locally solving a convex relaxation of the original problem.

The basic idea at the heart of the RLT is to form new constraints by multiplying together bound factors $(x_i - x_i^L)$, $(x_i^U - x_i)$, where $x^L \leq x \leq x^U$ are the problem variables and their ranges, and constraint factors $g_i(x) - b_i$ where $g(x) = b$ (or $g(x) \leq b$) are linear problem constraints. We shall call all such new constraints *RLT constraints*. The main limitation of this approach is the computational explosion deriving from multiplying *all* such bounds together. This is especially true in view of the fact that the set of all new constraints deriving from all factor products usually contains a lot of redundant and inactive constraints. Many "limiting devices" have been suggested in the works cited above, some of which are precise and some of which are heuristic.

In this paper we shall present such a limiting device for the RLT applied to continuous bilinear problems with linear equality constraints having the following form:

$$\left.\begin{array}{rcl} \min_x \quad x^T Q x & + & c^T x \\ Ax & = & b \\ x^L \leq & x & \leq x^U. \end{array}\right\} \tag{1}$$

where $x \in \mathbb{R}^n$ are the problem variables, $A = (a_{ij})$ is an $m \times n$ matrix having rank $m \leq n$, $Q = (q_{ij})$ is an $n \times n$ matrix (that we can assume upper-triangular by commutativity of multiplication), $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $x^L, x^U \in \mathbb{R}$ are the variable bounds.

The method presented herein tries to find the minimal number of RLT constraints that give the best linearization benefit. This method is derived from a theoretical discussion of the RLT applied to linearly constrained continuous bilinear problems that will highlight the reason why RLT's convex relaxations of those problems are so tight.

In section 2 we shall give a short resumé of the RLT applied to continuous bilinear problems with linear equality constraints. Section 3 contains the main theoretical result of this paper. In section 4 we shall explain how the derived method works. Lastly, we shall draw the conclusions.

2 The RLT on continuous bilinear problems

In order to form a linear convex relaxation of problem (1), the RLT applied to bilinear problems considers the following sets [SA92]:

- the bound factor set $B_F = \{x_i - x_i^L \mid i \leq n\} \cup \{x_i^U - x_i \mid i \leq n\}$;
- the constraint factor set $C_F = \{\sum_{j=1}^n a_{ij}x_j - b_i \mid i \leq m\}$.

Note that for each $\beta \in B_F$ the constraint $\beta \geq 0$ is a valid problem constraint, and so is $\gamma = 0$ for all $\gamma \in C_F$. The RLT procedure for forming the convex relaxation consists in creating new linear valid constraints (reformulation step) by multiplying together bound factors and constraint factors as follows:

1. for all $\beta_1, \beta_2 \in B_F$, $\beta_1\beta_2 \geq 0$ is a valid constraint (generation via bound factors);
2. for all $\beta \in B_F$ and for all $\gamma \in C_F$, $\beta\gamma = 0$ is a valid constraint (mixed generation);
3. for all $\gamma_1, \gamma_2 \in C_F$, $\gamma_1\gamma_2 = 0$ is a valid constraint (generation via constraint factors).

Having created all these new constraints, we define new variables $w_j^i = x_i x_j$ for all i, j between 1 and n, and we substitute them whenever a bilinear product appears in problem (1) or in the newly generated constraints (linearization step). We then obtain a very tight linear relaxation of the original bilinear problem.

2.1 Example
Suppose the original problem contains the bilinear term $x_j x_k$ and that we need to find a concave/convex enclosure for the corresponding bilinear surface $w_j^k = x_j x_k$. By using the RLT, we form the following inequalities:

$$\begin{aligned} (x_j - x_j^L)(x_k - x_k^L) &\geq 0 \\ (x_j - x_j^L)(x_k^U - x_k) &\geq 0 \\ (x_j^U - x_j)(x_k - x_k^L) &\geq 0 \\ (x_j^U - x_j)(x_k^U - x_k) &\geq 0, \end{aligned}$$

which, on substituting $x_j x_k$ with w_j^k, imply the following linear enclosure for the bilinear surface:

$$\begin{aligned} w_j^k &\geq x_j^L x_k + x_j x_k^L - x_j^L x_k^L \\ w_j^k &\leq x_j^L x_k + x_j x_k^U - x_j^L x_k^U \\ w_j^k &\leq x_j^U x_k + x_j x_k^L - x_j^U x_k^L \\ w_j^k &\geq x_j^U x_k + x_j x_k^U - x_j^U x_k^U. \end{aligned}$$

The latter inequalities are also known as McCormick's convex envelope for bilinear terms [McC76].

2.2 Example
Let $\sum_{j=1}^n a_{ij}x_j = b_i$ be the i-th constraint of a bilinear problem which for all $j \leq n$ includes the products $x_j x_k$ for some $k \leq n$. Then $(x_k - x_k^L)(\sum_{j=1}^n a_{ij}x_j - b_i) = 0$ is a valid RLT constraint. Notice that $-x_k^L(\sum_{j=1}^n a_{ij}x_j - b_i)$ is just a scalar multiple of the original constraint, so we can limit our attention to $x_k(\sum_{j=1}^n a_{ij}x_j - b_i) = 0$. After the linearization step we obtain $\sum_{j=1}^n a_{ij}w_j^k = b_i x_k$ where $w_j^k = x_j x_k$ for all $j, k \leq n$. Thus an RLT constraint derived from a bound factor and a constraint factor is in fact a linear relationship between the original problem variables and the new "linearization" variables w_j^k.

Usually the set of all new constraints is redundant, in the sense that some of the constraints are linear combinations of other constraints, whereas other constraints are simply inactive. In this sense the precise RLT creation method, applied blindly, is not always practical because of the very high number of new constraints.

3 Exact reformulation of bilinear terms with linear equations

Lower bounds on bilinear problems calculated with the RLT are generally very tight, producing linear relaxations which are very close to the original bilinear problem. This would not be justified if one only took into account the bound factor products. These are also known as McCormick convex envelope for bilinear terms [McC76], and their geometry is such that the gap between the original bilinear surface and the enclosure formed by its convex/concave relaxations is usually very wide [AKF83, AMF95]. Furthermore, it can easily be shown that both the products of bound by constraint factors and the constraint factor products are in fact linear combinations of the bound factor products and the products of original problem variables x_k (for $k \leq n$) by original linear constraints $\sum_{j=1}^n a_{ij}x_j = b_i$ (for $i \leq n$). Thus, the excellent performance of the RLT should be attributed to RLT constraints obtained by multiplying original problem variables by linear equality constraints. Namely, given the i-th equality constraint $\sum_{j=1}^n a_{ij}x_j = b_i$ and the multiplier variable x_k, the derived RLT constraint is $x_k \sum_{j=1}^n a_{ij}x_j - x_k b_i = 0$; on substituting $w_j^k = x_j x_k$ for each $j, k \leq n$, we obtain $\sum_{j=1}^n a_{ij}w_j^k - x_k b_i = 0$. In this section we shall explain why these RLT constraints tighten the relaxation so effectively.

For all $j, k \leq n$ let $w_j^k = x_j x_k$. It is easy to show that the cardinality of $\{w_j^k \mid j, k \leq n\}$ is $t = \frac{1}{2}n(n+1)$. Consider the feasibile region $F = \{x \in \mathbb{R}^n \mid Ax = b, x^L \leq x \leq x^U\}$ of the original problem (1), and let $C \subseteq \mathbb{R}^{n+t}$ be the superset (or "lift") of F defined as

$$C = \{(w, x) \mid x \in F \wedge \forall\, j, k \leq n\ (w_j^k = x_j x_k)\}.$$

Notice that, because of the bilinear relationships $w_j^k = x_j x_k$, to each F there corresponds exactly one C. Furthermore, if we reformulate the original problem (1) so that we substitute all bilinearities in the objective function with the corresponding w variables, we obtain an equivalent bilinear problem in $\mathbb{R}^{n+t}$ whose feasible region is C. Now consider the linear system of all RLT constraints that are products of problem variables and linear equality constraints:

$$\forall k \leq n\ (Aw^k - x_k b = 0) \tag{2}$$

where $w^k = (w_1^k, \ldots, w_n^k)$ and $x_k b = (x_k b_1, \ldots, x_k b_m)$. On substituting $b = Ax$ we obtain $Aw^k - x_k Ax = 0$, i.e. $A(w^k - x_k x) = 0$. We define variables $z_j^k = w_j^k - x_j x_k$ for all $j, k \leq n$ and we express (2) as

$$\forall k \leq n\ (Az^k = 0) \tag{3}$$

where $z^k = (z_1^k, \ldots, z_n^k)$ for all $k \leq n$. We call system (3) the *companion system*. It is easy to show that provided $Ax = b$, (2) $\Leftrightarrow$ (3). Let I_A be a set of index pairs (j, k) such that $\{z_j^k \mid (j, k) \in I_A\}$ is a set of basic variables of system (3). Lastly, consider $R \subseteq \mathbb{R}^{n+t}$ such that

$$R = \{(w, x) \mid x \in F \wedge \forall k \leq n\ (Aw^k - x_k b = 0) \wedge \forall (j, k) \notin I_A\ (w_j^k = x_j x_k)\}.$$

We are now in the position to prove the main theorem of this paper, i.e. that $C = R$. This means that R is a precise reformulation of the feasible region of the original problem (1) which involves less bilinear terms.

3.1 Theorem
$C = R$.

Proof. The fact that $C \subseteq R$ is obvious: since x is such that $Ax = b$ then it is also true that for all $k \leq n$, $x_k(Ax) = x_k b$, which immediately implies system (2). Furthermore, restricting the number of bilinear relations between w and x variables ensures that $C \subseteq R$. We shall now prove that $R \subseteq C$. The relations $\forall (j, k) \notin I_A (w_j^k = x_j x_k)$ imply that all the nonbasic variables of the companion system (3) are zero. Thus, the only possible solution for the basic variables of the companion system z_j^k (such that $(j, k) \in I_A$) is the zero solution, i.e. $\forall (j, k) \in I_A\ (w_j^k = x_j x_k)$. This immediately implies that all the bilinear relations defining C are satisfied, thus establishing the result. $\square$

3.2 Corollary

Let r be the rank of the companion system (3). Then the reformulation R of the original feasible region C has $t-r$ bilinear terms, where $t = \frac{1}{2}n(n+1)$.

Proof. Any set of basic variables of (3) has cardinality r, and the number of variables z_j^k of system (3) is $\frac{1}{2}n(n+1)$; thus, by theorem (3.1), the number of bilinear terms required to define R is $t-r$. □

The geometric meaning of theorem (3.1) is that under the given conditions, the bilinearly constrained feasible region C can be reformulated precisely to the region R which has more variables, more linear constraints, but less bilinear terms. In terms of the problem at hand, i.e. the construction of the convex relaxation of the original bilinear problem, this in turn means that a smaller number of bilinear terms are actually being relaxed by their McCormick (also known as RLT "bound factor products") relaxations than would be required when not using the RLT constraints; hence the gap between the original problem and its convex relaxation is greatly reduced.

4 The derived method

From theorem (3.1) we can quickly derive a practical method which would seem to guarantee the tightest possible (linear) convex relaxation of a bilinear problem in form (1): find a set of basic variables of system (3), reformulate C to R and then form the convex relaxation of R by substituting all remaining bilinear surfaces $w_j^k = x_j x_k$ (corresponding to the nonbasic variables of (3)) with their McCormick convex relaxations.

Such a straightforward application of theorem (3.1), however, has a very serious drawback. By corollary (3.2), reformulation R must contain $s = t - r = \frac{1}{2}n(n+1) - r$ terms. Numerical experiments in this sense seem to point out the fact that $s > 0$ unless $m = n$, i.e. unless the feasible region of the original problem (1) is trivial, in which case $s = 0$. If the number of bilinear terms in the original problem (1) happens to be strictly less than s, as is often the case in sparse bilinear problems (i.e. problems where the matrix Q is sparse), then our reformulation R actually has more bilinear terms than the original problem, thus resulting in a very loose convex relaxation. Even if the number of bilinear terms in the original problem is greater than s, it might still happen that each set of nonbasic variables of the companion system involves bilinear terms which are not present in the original problem.

The way to circumvent this problem is to identify subsets of candidate multiplier variables and linear equation constraints so that the resulting RLT system of equations replaces (in the sense of theorem (3.1)) more bilinear terms than it needs to add during the linearization step. I.e. find a subset of equations $A'x = b'$ of the original system $Ax = b$, and a subset K of $\{1, \dots n\}$ such that the RLT subsystem

$$\forall k \in K \ (A'w^k - b'x_k = 0) \tag{4}$$

(where $w^k = (w_1^k, \dots w_n^k)$ and $w_j^k = x_k x_j$ for each $k \in K$ and $j \leq n$) eliminates more bilinear terms than the additional bilinear terms $w_j^k = x_k x_j$, not originally in the problem, needed to define the RLT subsystem. Let $H = K \cap \{1, \dots, n\}$. The size of the set of bilinear terms $L_R = \{x_k x_j \mid k \in K, j \leq n\}$ needed to define (4) is $|L_R| = \frac{1}{2}|H|(|H|+1) + |K|(n - |K|)$. Suppose that the set of bilinear terms in the original problem is L. Then the number of additional bilinear terms required by (4) is $\beta = |L_R| - |L \cap L_R|$. Thus, we require that the rank r' of the RLT companion subsystem

$$\forall k \in K \ (A'z^k = 0) \tag{5}$$

(where $z_j^k = w_j^k - x_k x_j$ for each $k \in K$, $j \leq n$) is strictly greater than β. An optimal search in this sense would be prohibitively expensive: for each subset of equations of $Ax = b$ and each subset $K \subseteq \{1, \dots, n\}$, it would involve finding the minimum rank of all the possible companion RLT subsystems.

In the algorithm that follows, we shall denote systems (2) and (3) in more compact forms as, respectively, $T(w,x)=0$ and $Bz=0$, where B depends on A, and T depends on A and b. It is easily shown that T has the structure $(B|\cdot)$. This fact emphasizes a 1-1 correspondence between the rows of $Bz=0$ and those of $T(w,x)=0$. In particular, any row operation on $Bz=0$ can be carried out on $T(w,x)=0$, and the resulting RLT system is valid in the original problem as it is just a linear combination of valid equations.

The algorithm below identifies and erases from $Bz=0$ a sufficient number of equations so that the corresponding RLT subsystem $T(w,x)=0$ will require the creation of less bilinear terms than it can replace. This algorithm may not find the globally optimal RLT subsystem; however, it is somewhat tuned to the typical shape of the matrix B, so that, paired with graph-theoretical based techniques such as that described in [LP03], it is able to find good locally optimal subsets of RLT constraints.

1. Put $Bz=0$ in row echelon form and delete any zero rows, so that we obtain a system with r rows (and having full rank r).
2. For all $i\le r$, let $R(i)$ be the i-th row of $Bz=0$:
    (a) denote with $\nu(i)$ the number of nonzero coefficients of $R(i)$,
    (b) assign to $R(i)$ a bonus $\psi(i)$ equal to the number of nonzero coefficients in $R(i)$ which correspond to bilinear terms that already exist in the problem;
    (c) assign to $R(i)$ a cost $\nu(i)-\psi(i)$.
3. Let J be the subset of column indices corresponding to bilinear terms that are not in the problem, and let I be the subset of row indices corresponding to rows with positive cost.
4. Let $j\in J$ so that the j-th column of $Bz=0$ has the maximum number of nonzero coefficients.
5. Let $\zeta(j)$ be the subset of row indices that have a nonzero coefficient in column j.
6. Decrease the cost of each row indexed by $\zeta(j)$ by 1.
7. Set $J\leftarrow J\backslash\{j\}$ and $I\leftarrow I\backslash\zeta(j)$.
8. If $|J|=0$, go to 9; otherwise, repeat from 4.
9. Generate the RLT equations corresponding to the equations in the companion system that have nonpositive cost.

5 Conclusion

In this paper we discussed a theoretical aspect of the RLT applied to continuous bilinear problems with linear equality constraints. We showed that RLT constraints can geometrically substitute bilinear products by exhibiting a precise reformulation of the feasible region which has more linear constraints and less bilinear products. We then gave an algorithm that identifies a set of RLT constraints that reduce the number of bilinear products in the original bilinear problem.

Although this paper does not include computational results for the implementation of the proposed methods, the implementation of another algorithm for the creation of RLT constraints was carried out and tested, within the Branch-and-Bound global optimization algorithm proposed in [SP99]. The results point out the fact that RLT constraints slash the order of magnitude of the computational cost of a generic Branch-and-Bound algorithm by factors of 10, 100 or sometimes more. This fact is all the more evident on pooling and single-quality blending problems [ATS99], where a running time of 10,000 or 100,000 iterations is sometimes reduced to just 1 iteration. The performance on multi-quality problems, though less spectacular, is impressive nonetheless. Formulation-wise, the effect of RLT on blending problems is emphasized in [TS02].

References

[AKF83] F.A. Al-Khayyal and J.E. Falk. Jointly constrained biconvex programming. *Mathematics of Operations Research*, 8(2):273–286, 1983.

[AMF95] I. P. Androulakis, C. D. Maranas, and C. A. Floudas. alpha bb: A global optimization method for general constrained nonconvex problems. *Journal of Global Optimization*, 7(4):337–363, December 1995.

[ATS99] N. Adhya, M. Tawarmalani, and N.V. Sahinidis. A lagrangian approach to the pooling problem. *Industrial and Engineering Chemistry Research*, 38:1956–1972, 1999.

[LP03] L. Liberti and C.C. Pantelides. An exact reformulation algorithm for nonconvex nlps involving bilinear terms. *Internal report, CPSE, Imperial College, London, UK*, 2003.

[McC76] G.P. McCormick. Computability of global solutions to factorable nonconvex programs: Part i — convex underestimating problems. *Mathematical Programming*, 10:146–175, 1976.

[SA86] H.D. Sherali and W.P. Adams. A tight linearization and an algorithm for 0-1 quadratic programming problems. *Management Science*, 32(10):1274–1290, 1986.

[SA92] H.D. Sherali and A. Alameddine. A new reformulation-linearization technique for bilinear programming problems. *Journal of Global Optimization*, 2:379–410, 1992.

[SA99] H.D. Sherali and W.P. Adams. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Kluwer Academic Publishers, Dodrecht, 1999.

[She98] H.D. Sherali. Global optimization of nonconvex polynomial programming problems having rational exponents. *Journal of Global Optimization*, 12:267–283, 1998.

[She02] H.D. Sherali. Tight relaxations for nonconvex optimization problems using the reformulation-linearization/convexification technique (rlt). 2:1–63, 2002.

[SP99] E.M.B. Smith and C.C. Pantelides. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex minlps. *Computers and Chemical Engineering*, 23:457–478, 1999.

[SSA00] H.D. Sherali, J.C. Smith, and W.P. Adams. Reduced first-level representations via the reformulation-linearization technique: Results, counterexamples, and computations. *Discrete Applied Mathematics*, 101:247–267, 2000.

[SW01] H.D. Sherali and H. Wang. Global optimization of nonconvex factorable programming problems. *Mathematical Programming*, A89:459–478, 2001.

[TS02] M. Tawarmalani and N.V. Sahinidis. Convexification and global optimization of the pooling problem. *Mathematical Programming (submitted)*, 2002.