

SOME GENERALIZATIONS OF THE CRISS-CROSS METHOD FOR QUADRATIC PROGRAMMING

Emil Klafszky
Mathematical Institut,
Technical University, Miskolc
Miskolc-Egyetemváros
H-3515. Hungary

Tamás Terlaky¹
Dept. of Op. Res.
Eötvös University
Budapest, Muzeum krt. 6-8.
H-1088. Hungary

Abbreviated title: Criss-cross methods for quadratic programming.

Key words: criss-cross method, least index resolution, pivoting, quadratic programming, simplex method.

Some generalizations of the criss-cross method for quadratic programming

Abstract:

Three generalizations of the criss-cross method for quadratic programming are presented here. Tucker's, Cottle's and Dantzig's principal pivoting methods are specialized as diagonal and exchange pivots for the linear complementarity problem obtained from a convex quadratic program.

A finite criss-cross method, based on least-index resolution, is constructed for solving the LCP. In proving finiteness, orthogonality properties of pivot tableaus and positive semidefiniteness of quadratic matrices are used.

In the last section some special cases and two further variants of the quadratic criss-cross method are discussed. If the matrix of the LCP has full rank, then a surprisingly simple algorithm follows, which coincides with Murty's 'Bard type schema' in the P matrix case.

1 Introduction

Quadratic programming (QP) and linear complementarity problems (LCP) are examined and applied in many areas of operations research. The pioneers of operations research - Dantzig [8], [9], Cottle [7], [8], Beale [2], Lemke [13], Wolfe [29] - examined this subject and gave efficient algorithms, mainly based on the simplex method [9]. In some of these methods QP was reformulated as an LCP in order to apply the simplex method with special pivot selection rules. Some of these methods are based on Tucker's [26] principal pivot transformation. Later, several other variants were discovered [10], [18], [22], [27] and the relation between these methods was examined [3]. To review this extended literature is not the aim of the present paper.

The combinatorial behavior of the LCP was examined by Parsons [17]. It is well known that combinatorial type problems (e.g. cycling) may occur if LCP is solved by simplex type methods. This problem and the solution by using lexicography was mentioned first by Cottle and Dantzig [8]. The new results of linear programming (LP) [4], [5], namely least index resolution and orthogonality were adopted for LCP by Chang and Cottle [6]. They used the minimal index rule in Keller's [10] algorithm and proved finiteness in this way.

An important - but not so strongly related - area concerning our algorithms is the theory of oriented matroids. Bland [5] gave a combinatorial abstraction of LP giving a new insight into the theory of simplex type methods. The combinatorial equivalence of LP , the oriented matroid programming problem was solved by different pivot rules [5], [21], [24], [25]. Morris and Todd [14], [24], [25] generalized the theory of LCP for oriented matroids. Oriented matroid quadratic programming problem was formulated and solved by a generalized Lemke algorithm.

The idea of criss-cross type methods was introduced by Zionts [30], [31]. Here Terlaky's [20], [21] (see also Wang [28] finite criss-cross method is generalized for QP . As in the case of LP , our algorithms are based on the sign structure of pivot tableaus and on the least index selection. In fact three algorithms are presented, each of them is a possible generalization of the finite LP criss-cross method. Tucker's [26] principal pivots will be used in special cases as diagonal and exchange pivots (using Keller's terminology).

Since our algorithms are based only on the sign structure of pivot tableaus and on the least index selection, these algorithms can be generalized for oriented matroids. This generalization is presented in [12]. The aim of this paper is to present a new, theoretically interesting algorithm. This algorithm is not more efficient (at least not for general QP problems) than the other QP algorithms (like the other anticycling pivot rules such as the lexicographic rule, Bland's rule), it is efficient just in some special cases because it coincides with Murty's Bard type schema in the positive definite case. Further it has several consequences in combinatorics, especially in the theory of oriented matroids.

In the second paragraph basic properties of QP and the associated LCP are summarized. Most of these results are well known, some of them is easily derived. The third paragraph

contains the main result, a generalized criss-cross method for QP and the proof of its finiteness. Finally in the fourth chapter two modifications and some special cases are presented.

Concerning notations, Balinski's and Tucker's [1] notations will be used as in our previous papers [11], [12], [20], [21], [22], [23]. Matrices are denoted by capital letters (A, B, P, \dots), vectors by small Latin letters and components of vectors and matrices by the corresponding Greek letters [e.g. $z = (\zeta_1, \dots, \zeta_n)$, $A = (\alpha_{ij})_{i=1}^m \quad j=1}^n$]. Index sets are denoted by I and J with proper sub (super) scripts. Symbols $+$, $-$, \oplus , \ominus , $\mathbf{0}$ denote in figures that the corresponding element is positive, negative, nonnegative, nonpositive and zero respectively. In figures a $*$ denotes that no information regarding that element is available. We will use the notation $x^2 = x^T x$ as well.

QP is presented here in symmetric form as it can be found in [25], [27]. This symmetric form is the base of the combinatorial abstraction of QP . We have chosen this form to emphasize the combinatorial feature of this algorithm. Most of the results presented in this section are well known, so they are presented without proofs.

Let $A : m \times n$; $B : m \times k$; $C : l \times m$ be arbitrary matrices and $c, x \in R^n$; $b, y \in R^m$; $z \in R^k$; $w \in R^l$ be vectors.

The quadratic programming problem

Find vectors x and z such that

$$cx + \frac{1}{2}(Cx)^2 + \frac{1}{2}z^2 \text{ is minimal}$$

under the assumptions

$$\begin{aligned} Ax + Bz &\geq b \\ x &\geq 0 \end{aligned}$$

The dual problem

Find vectors such y and w such that

$$yb - \frac{1}{2}(yB)^2 - \frac{1}{2}w^2 \text{ is maximal}$$

under the assumptions

$$\begin{aligned} yA - wC &\leq c \\ y &\geq 0. \end{aligned}$$

It is well known [7], [8], [23], [25], [27] that the above given QP is equivalent to the following LCP .

$$\begin{aligned} -Py - Ax + \bar{y} &= -b \\ A^T y - Qx + \bar{x} &= c \end{aligned} \tag{1}$$

$$y \geq 0, x \geq 0, \bar{y} \geq 0, \bar{x} \geq 0 \tag{2}$$

$$x\bar{x} = 0, y\bar{y} = 0 \tag{3}$$

where $P = BB^T$; $Q = C^T C$; $\bar{y} = Py + Ax - b$ and $\bar{x} = c + Qx - A^T y$. It is easy to see that matrices P and Q are positive semidefinite in this case. Assumptions (1) are called **equality conditions**, assumptions (2) are called **nonnegativity conditions** and assumptions (3) are called **complementarity conditions**. Index (variable) pairs x_i, \bar{x}_i and y_i, \bar{y}_i will be referred as complementary indices (variables).

Nonnegativity and complementarity conditions are preserved in Lemke's [13] and Wolfe's [29] methods. These algorithms stop if equality conditions are fulfilled. In Keller's [10], Cottle-Dantzig's [8] and Van de Panne-Whinston's [27] methods equality and complementarity conditions are satisfied while nonnegativity is partially preserved. In our quadratic criss-cross methods, as in the latter methods equality and complementarity conditions are satisfied but nonnegativity is not preserved. Denote $M = \begin{bmatrix} -P & -A \\ A^T & -Q \end{bmatrix}$ and $d = \begin{pmatrix} -b \\ c \end{pmatrix}$. Matrix M is called **bisymmetric** which means it has two diagonal blocks $(-P)$ and $(-Q)$ which are symmetric and negative semidefinite and the remaining matrix $\begin{bmatrix} 0 & -A \\ A^T & 0 \end{bmatrix}$ is skew symmetric. So M is a negative semidefinite matrix as well. A base of the equality system (1) is called complementary iff one and only one of the complementary indices is in the base.

If a set of basic vectors is replaced with its complementary set, then this basic transformation is called a **principal pivot transformation**. This terminology was introduced by Tucker [26]. Tucker proved that bisymmetric property of the nonbasic part of the basic tableau is preserved during principal pivot transformations. Following Keller, Van de Panne and Whinston a 1x1 principal pivot is called **diagonal pivot** (a basic vector is replaced with its complementary pair) and a 2x2 principal pivot is called **exchange pivot** (two basic vectors are replaced with their complementary pairs). If in case of exchange pivots at least one of the corresponding diagonal elements is nonzero, then an exchange pivot can be interpreted as two subsequent diagonal pivots. The semidefinite, symmetric diagonal blocks are defined by the basic vectors associated with the $u = (x, \bar{y})$ and $v = (\bar{x}, y)$ variables.

For simplicity the notation $u = (x, \bar{y}) = (\mu_1, \dots, \mu_N)$ and $v = (\bar{x}, y) = (\nu_1, \dots, \nu_N)$ will be used, where $N = n + m$. So μ_i and ν_i are complementary pairs for all $i = 1, \dots, N$.

Using these notations the following lemma is proved. This is a simple generalization of the simple fact, that in case of LP ($\min cx, \text{s.t. } Ax \geq b, x \geq 0$), the difference of two

Figure 1.

Denote for all $i \in J_B$

$$t^{(i)} = (\tau_j^{(i)})_{j=1}^n = \begin{cases} \tau_{ij} & \text{if } j \in \overline{J}_B, \\ 1 & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

and denote for all $k \in \overline{J}_B$

$$t_{(k)} = (\tau_{(k)j})_{j=1}^n = \begin{cases} \tau_{kj} & \text{if } j \in J_B, \\ -1 & \text{if } j = k, \\ 0 & \text{otherwise.} \end{cases}$$

It is well-known that $t^{(i)} - t_{(k)}$, for all $i \in J_B$ and $k \in \overline{J}_B$ even if the two bases B and B' are different [4], [20]. For better understanding let us consider the following simple numerical example, two basic tableaus, that can be transformed into each other by a single pivot.

$$\begin{array}{ccccc} & t_1 & t_2 & t_3 & t_4 & t_5 \\ t_5 & \boxed{3} & \boxed{1} & \boxed{3} & \boxed{0} & \boxed{1} \\ t_4 & \boxed{5} & \boxed{2} & \boxed{4} & \boxed{1} & \boxed{0} \end{array} \qquad \begin{array}{ccccc} & t_1 & t_2 & t_3 & t_4 & t_5 \\ t_2 & \boxed{3} & \boxed{1} & \boxed{3} & \boxed{0} & \boxed{1} \\ t_4 & \boxed{-1} & \boxed{0} & \boxed{-2} & \boxed{1} & \boxed{-2} \end{array}$$

It is easy to check for example, that from the first tableau $t_{(3)} = (0, 0, -1, 4, 3)$ and from the second tableau $t^{(4)} = (-1, 0, -2, 1, -2)$, and these vectors are orthogonal ($t_{(3)}t^{(4)} = 0$).

We will use this result, the so called **orthogonality property** of basic tableaus, for the matrix $\overline{T} = [d, -M, E]$.

2 A Finite criss-cross method for quadratic programming

Now we give a new finite algorithm for QP , which is a generalization of the finite criss-cross method of LP .

Let a QP be given and consider the corresponding LCP as it is given by (1), (2) and (3). Choose $(\overline{x}, \overline{y})$ and their coefficient matrix (unit matrix) as the initial basic solution. Denote $u = (x, \overline{y})$ and $v = (\overline{x}, y)$ and $N = n + m$.

Algorithm I. (a generalized criss-cross method)

0. **Initialization:** Let a starting, complementary but not necessarily feasible base be given. (One and only one of the complementary variables μ_i, ν_i is in the base.) Let us order the variables (and so their indices) as follows:

$$(\mu_1, \nu_1, \mu_2, \nu_2, \dots, \mu_i, \nu_i, \dots, \mu_n, \nu_n).$$

- k. **A general step:** A complementary, bisymmetric, negative semi-definite basic tableau is given.

$$\text{Let } r = \min\{i \mid \mu_i < 0 \text{ or } \nu_i < 0, i = 1, \dots, N\}.$$

- (a) If there is no such r , then stop, the current solution solves the *LCP* and so we have an optimal solution for *QP*.
- (b) If we have an r (without loss of generality we may assume that $\mu_r < 0$) and if $\tau_{rr} < 0$ (the coefficient of μ_r in the basic representation of ν_r is negative), then make a **diagonal pivot**, μ_r is replaced by ν_r in the base. Let $k \leftarrow k + 1$.
- (c) If $\tau_{rr} = 0$ let $s = \min\{j \mid \tau_{rj} < 0 \text{ } j=1, \dots, N\}$.
- (c1) If there is no s then stop, *LCP* is inconsistent, *QP* or *QD* has no solution.
- (c2) If we have an index s , then do an **exchange pivot**, basic vectors μ_r and ν_s are replaced by nonbasic vectors μ_s and ν_r . Let $k \leftarrow k + 1$.

To justify the algorithm remark, that in case (c) where $\tau_{rr} = 0$ ($\tau_{rr} > 0$ is impossible since M is negative semidefinite), implies that the μ_r row of the associated diagonal block is zero, so s is in the skew symmetric part and so it belongs to a μ_s variable.

It is proved e.g. in [7], [10] that *LCP* is inconsistent if we have a nonnegative row in the basic tableau and so *QP* has no optimal solution. This is a simple consequence of the orthogonality property, it is independent of the special properties of the coefficient matrix M .

In case (a) we have solved *LCP* and so optimal solutions were obtained for both of the primal and dual *QP* problems. Since we use only diagonal and exchange pivots, bisymmetric and negative semidefinite property of basic tableaus are preserved during Algorithm I. Our algorithm goes through complementary bases of *LCP* and stops either at an optimal or at an inconsistent tableau, so to prove finiteness one have to prove only that cycling cannot occur (there are only finite number of different bases).

Theorem 1 *Algorithm I. is finite.*

Proof:

Let us suppose to the contrary that cycling occurs. Denote $J^* = \{i \mid \mu_i \text{ or } \nu_i \text{ entered the base through the cycle}\}$. It is obvious, if a variable is in J^* then so is its complementary

pair, since if a variable enters the base then its pair leaves. Furthermore if a variable enters the base through the cycle, then in some step it has to leave the base.

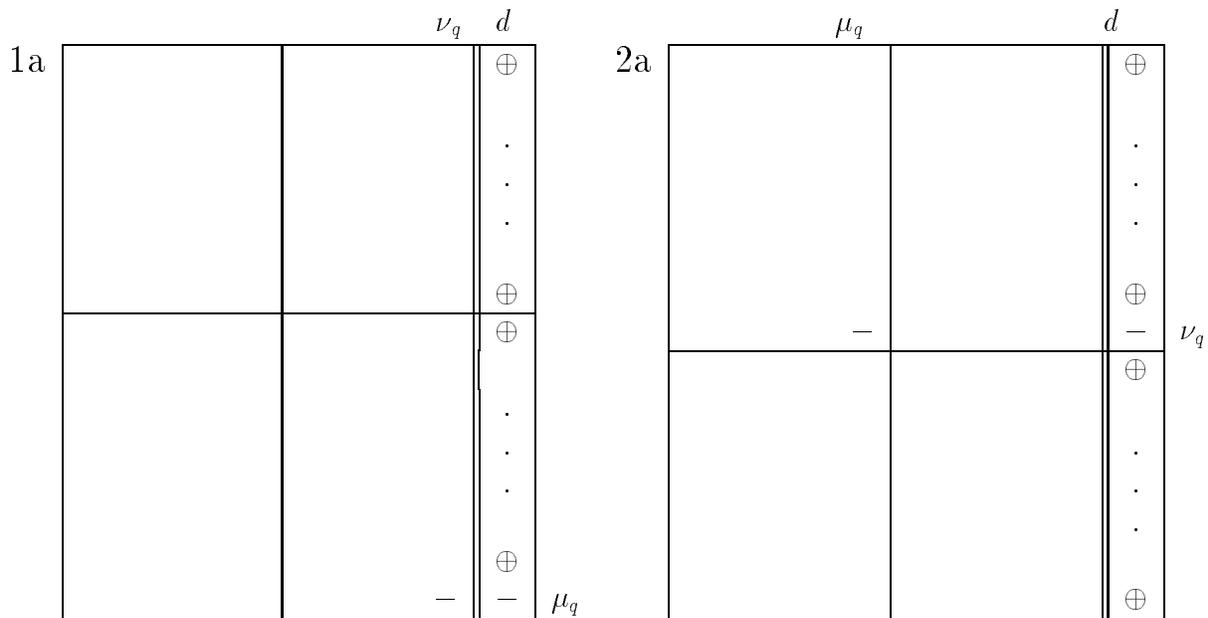
Let $q = \max\{i | i \in J^*\}$. According to the above remarks μ_q and ν_q entered (left) and left (entered) the base at some iteration through the cycle. Examine the situations when ν_q entered and when it left the base. We have the following situations to consider:

- | | |
|---|---|
| 1. ν_q enters the base | 2. ν_q leaves the base |
| 1.a. at a diagonal pivot | 2.a. at a diagonal pivot |
| 1.b. at an exchange pivot
(first order selection) | 2.b. at an exchange pivot
(first order selection) |
| 1.c. at an exchange pivot
(second order selection) | 2.c. at an exchange pivot
(second order selection) |

where "first order selection" means that index q was chosen as lowest infeasible index and "second order selection" means that index q was chosen in step (c) of the algorithm.

It seems that we have to consider 3x3, that is nine cases, but it will be shown that this can be reduced to four cases.

The structure of the corresponding basic tableaus are shown on Figure 2, where the strictly negative and positive elements of the tableau indicate the actual pivot elements. The parts of the basic tableau that correspond to indices larger than q are suppressed on the figure, since no information is available for the sign structure for that parts, and they have no influence on the performance of the algorithm.



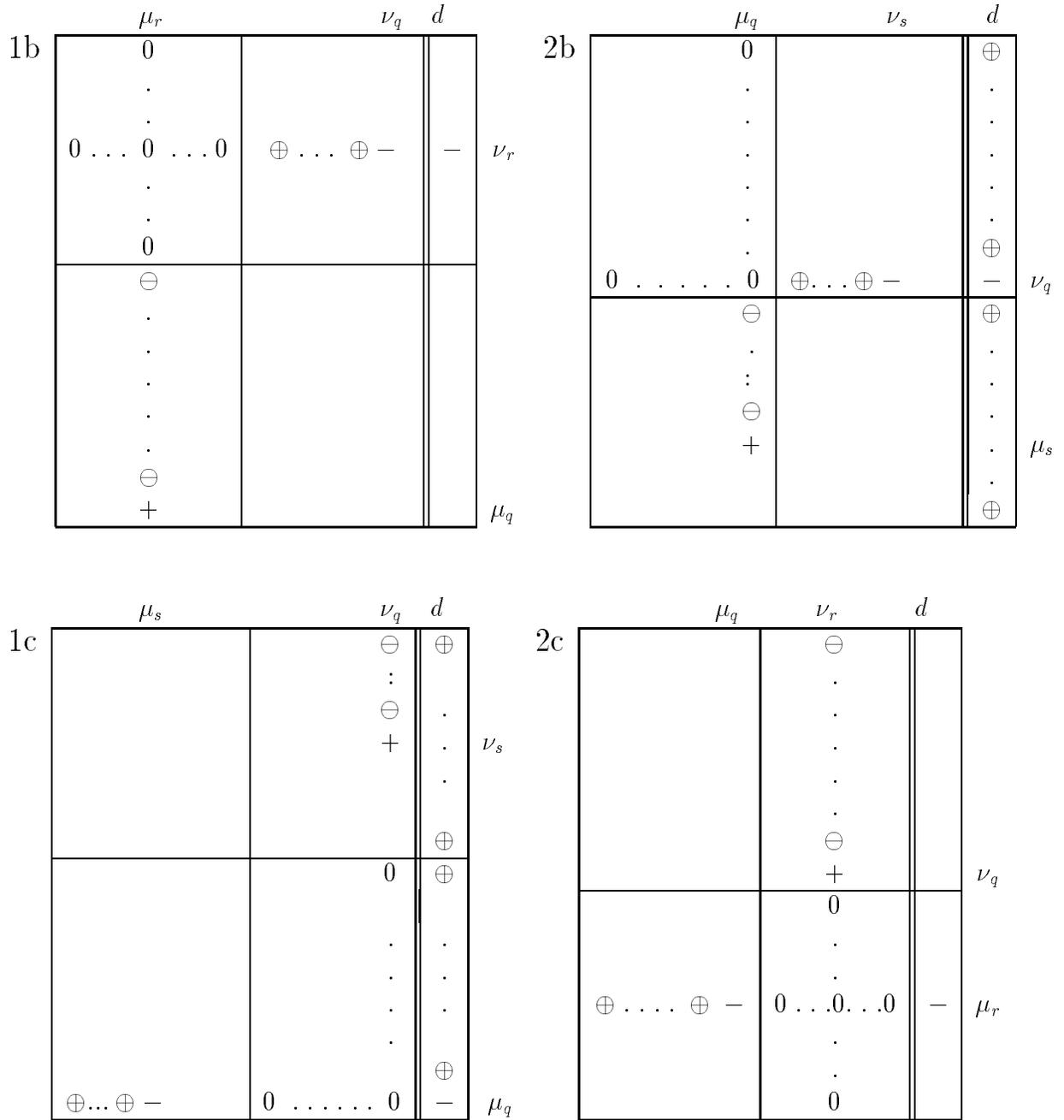


Figure 2.

As figure 2 shows, cases (1a) and (1c) and cases (2a) and (2b) contains the same useful information - the solution columns have the same sign structure. So the following cases remained to consider.

ν_q enters the base

(α) Cases (1a) and (1c) We have vector a (the solution column), which is orthogonal to the row space of matrix $\overline{T} = [d, -M, E]$ with the following sign structure

$$a = \begin{array}{c} d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ \boxed{- \mid \oplus \dots \oplus \mid - \mid 0 \mid * \dots * } \end{array}$$

Figure 3.

(β) Case (1b) We have a vector b_1 (the column of μ_r), orthogonal to the row space of matrix \overline{T} and we have a vector b_2 (the row of ν_r), which is in the row space of matrix \overline{T} with the following sign structure.

$$\begin{array}{c} d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ b_1 = \boxed{0 \mid \ominus \dots \ominus \mid + \mid 0 \mid * \dots * } \\ \\ d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ b_2 = \boxed{- \mid \oplus \dots \oplus \mid 0 \mid - \mid * \dots * } \end{array}$$

Figure 4.

ν_q leaves the base

(γ) Cases (2a) and (2b) We have a vector c (the solution column), which is orthogonal to the row space of matrix \overline{T} with the following sign structure.

$$c = \begin{array}{c} d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ \boxed{- \mid \oplus \dots \oplus \mid 0 \mid - \mid * \dots * } \end{array}$$

Figure 5.

(δ) Case (2c) We have a vector d_1 (the column of ν_r), which is orthogonal to the row space of matrix \overline{T} and we have a vector d_2 (the row of ν_r), which is in the row space of \overline{T} with the following sign properties.

$$\begin{array}{c} d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ d_1 = \boxed{0 \mid \ominus \dots \ominus \mid 0 \mid + \mid * \dots * } \\ \\ d \qquad i < q \quad \mu_q \quad \nu_q \qquad i > q \\ d_2 = \boxed{- \mid \oplus \dots \oplus \mid - \mid 0 \mid * \dots * } \end{array}$$

Figure 6.

So our problem is reduced to showing that cases $\alpha\gamma$; $\alpha\delta$; $\beta\gamma$; $\beta\delta$ cannot hold simultaneously. Before doing that, remark, that in any of the above mentioned four cases, indices greater than q can be omitted since the corresponding variables were always basic or nonbasic variables through the cycle, and so they have no effect to the algorithm.

Case $\alpha\gamma$: We have a vector (u, v) from vector a and vector (u', v') from vector c . So by Lemma 1 we have $(u-u')(v-v') \geq 0$, but in this case $(u-u')(v-v') = uv + u'v' - uv' - u'v = -(uv' + u'v) < 0$ since all the coordinates in the last term are nonnegative, and coordinates q give a zero and a positive product respectively. This is a contradiction, so this case is impossible.

Case $\alpha\delta$: In this case $a - d_2$, but their dot product is positive as it can easily be verified using the sign structure of these vectors.

Case $\beta\gamma$: In this case $b_1 - d_2$ or equivalently $b_2 - d_1$, but their dot product is negative.

Case $\beta\delta$: In this case $b_2 - c$, but their dot product is positive.

So all of the possible cases led to a contradiction, that is Algorithm I. is finite. \square

The proof shows that we have to consider essentially four different cases as it was in proving finiteness of the LP criss-cross method. In the next section it will be shown that QP criss-cross method is really a generalization of the finite LP criss-cross method, so these four cases are necessary to distinguish in proving finiteness.

3 Modifications and special cases

First we show how Algorithm I. and the proof of Theorem 1 simplifies in two important special cases. Finally modifying Algorithm I. two additional new, finite algorithms are constructed and their basic properties are summarized.

4.1. The case of definite matrices

Algorithm I. and the proof of its finiteness surprisingly simplifies if the quadratic matrices P and Q are positive definite matrices, and so M is negative definite matrix.

In this case the diagonal elements of M are never zero, so exchange pivots are never performed by Algorithm I. Only diagonal pivots occur. After initialization Algorithm I. simplifies to the following, extremely simple algorithm.

- Let $r = \min\{i \mid \mu_i < 0 \text{ or } \nu_i > 0 \text{ for } i = 1, \dots, N\}$.

- If there is no r , then stop, optimal solution has been found.
- If an index r were obtained then replace μ_r by ν_r (or ν_r by μ_r), do a **diagonal pivot**.

In this case LCP always has a solution. The proof of Theorem 1 simplifies, since only cases (1a) and (2a), that is only case $\alpha\gamma$ occurs, so there is only one case to consider.

This simple variant coincides with Murty's [16] 'Bard type schema' for the P matrix linear complementarity problem. It is easy to see, that our proof remains valid for that case. One has to recall only the 'sign nonreversibility property' of P matrices (see e.g. Murty [15]). Our proof is much more simple than Murty's proof.

4.2. Linear programming

In case of LP both of the quadratic matrices P and Q are identically zero matrices and so M gives an identically zero quadratic form M is skew symmetric in this case. Only exchange pivots occur in this case since diagonal elements vanish, so the size of diagonal blocks are fixed and so the basic tableau has the following structure.

0	$B^{-1}N$	x_B	$= B^{-1}b$
$-(B^{-1}N)^T$	0	z_N	$= c_N - c_B(B^{-1}N)^T$

Figure 7.

It is easy to see that in this case Algorithm I. gives Terlaky's [20] LP criss-cross method. In fact the original LP simplex tableau is doubled in this case.

4.3. Modifications

Note, that it is not necessary to distinguish and prefer diagonal element (diagonal pivot) as it is in Algorithm I. Instead we have the following, modified algorithm.

Algorithm II.

Initialization: The same as in Algorithm I.

k. **A general step.** Till (b) the same as in Algorithm I.

(b) - If we have an index r , then let $s = \min\{i \mid \tau_{ri} < 0 \text{ for all } i\}$. (All the negative coefficients of row r are examined).

- If we have no s , then (as in case (c1) of Algorithm I.), LCP is inconsistent.

(c1) - If s is in the symmetric diagonal block (belongs to ν_s), then do a **diagonal pivot** (μ_r is replaced by ν_r). Let $k \leftarrow k + 1$.

(c2) - If s out of the symmetric diagonal block (belongs to μ_s), then do an **exchange pivot** (μ_r and ν_s are replaced by ν_r and μ_s). Let $k \leftarrow k + 1$.

It is easy to see that this modified algorithm is finite again and the proof of finiteness holds without modification. In case of LP we get back again the LP criss-cross method as a special case, but Algorithm II. does not specialize for definite matrices.

Comparing Algorithm I. and Algorithm II. one can observe that diagonal pivots occur more frequently in Algorithm I. (always if it is possible). Algorithm I. can be modified in the other direction as well, if diagonal pivots are even more preferred.

Algorithm III.

Initialization: As in Algorithm I.

k. **A general step:** Till step (c2) the same as Algorithm I.

(c2) - If we have an index s and $s > r$ with $\tau_{ss} < 0$ (the diagonal element in column s is negative) then do a **diagonal pivot** (ν_s is replaced by μ_s). Let $k \leftarrow k + 1$.

(c3) - If we have an index s and $s < r$ or $\tau_{ss} = 0$ then do an **exchange pivot** (μ_r and ν_s are replaced by ν_r and μ_s). Let $k \leftarrow k + 1$.

Finiteness and the proof of finiteness holds the same way as in case of Algorithm I. In both of the special cases Algorithm III. specializes the same way as Algorithm I.

4 Summary

In this paper three possible generalizations of the LP criss-cross method are presented for QP . Finiteness is proved using orthogonality properties of basic tableaus In case of definite matrices a very simple algorithm, which turned to coincide Murty's Bard type schema, was derived.

We do not think that any of these algorithms are practically efficient (at least not for general QP problems), they are combinatorial type algorithms and interesting mainly from theoretical and combinatorial points of view. This is true in spite of the well known fact, that Murty's algorithm is considered as an efficient algorithm for the P-matrix linear complementarity problem.

Finally remark, that these algorithms are based only on sign structure of basic tableaus, proofs relieved on orthogonality properties, so these algorithms can be generalized to oriented matroids. This generalization is presented in [12].

References

- [1] M.L. Balinski and A.W. Tucker, "Duality theory of linear programs: A constructive approach with applications", *SIAM Review* 11 (1969), 347-377.
- [2] E.M.L. Beale, "On quadratic programming", *Naval Research Logistics Quarterly*, 6 (1969), 227-244.
- [3] M.J. Best, "Equivalence of some quadratic programming algorithms", *Mathematical Programming* 30 (1984), 71-87.
- [4] R.G. Bland, "New finite pivoting rules for the simplex method", *Mathematics of Operations Research*, 2 (1977), 103-107.
- [5] R.G. Bland, "A combinatorial abstraction of linear programming", *Journal of Combinatorial Theory Ser. B.* 23 (1977), 33-57.
- [6] Y.Y. Chang and R.W. Cottle, "Least index resolution of degeneracy in quadratic programming", *Mathematical Programming*, 18 (1980), 127-137.
- [7] R.W. Cottle, "The principal pivoting method of quadratic programming", in *Mathematics of the Decision Sciences Part I*. Eds. G.B. Dantzig and A.F. Veinott, *Lectures in Applied Mathematics* 11.
- [8] R.W. Cottle and G.D. Dantzig, "Complementary pivot theory of mathematical programming", *Linear Algebra and Its Applications*, 1 (1968), 103-125.
- [9] G.B. Dantzig, *Linear Programming and Extensions*, Princeton University, N.J. 1963.
- [10] E. Keller, "The general quadratic optimization problem", *Mathematical Programming*, 5 (1973), 311-337.
- [11] E. Klafszky and T. Terlaky, "Variants of the "Hungarian Method" for solving linear programming problems", *Mathematicshe Operationsforschung und Statiscics, Series Optimization*, 20 (1989), 1, 79-91.
- [12] E. Klafszky and T. Terlaky, "Some generalizations of the criss-cross method for the linear complementarity problem of oriented matroids", *Combinatorica*, 9 (2), (1989) 189-198.
- [13] C.E. Lemke, "Bimatrix equilibrium points and mathematical programming", *Management Science*, 11 (1965), 681-689.
- [14] Jr. Morris and M.J. Todd, "Symmetry and positive definiteness in oriented matroids". Cornell University, School of Operations Research and Industrial Engineering, Ithaca, N.Y. Technical Report no. 631.

- [15] Murty, K.G., (1988), *Linear Complementarity, Linear and Nonlinear Programming*. Sigma Series in Applied Mathematics, Vol. 3. Heldermann Verlag, Berlin.
- [16] Murty, K.G., (1974), A Note on a Bard Type Scheme for Solving the Complementarity Problem. *Opsearch* 11, (2-3), 123-130.
- [17] T.D. Parsons, "A combinatorial approach to convex quadratic programming", Princeton University, Ph.D. Thesis, 1966.
- [18] K. Ritter, "A dual quadratic programming algorithm", University of Wisconsin-Madison, Mathematics Research Center, Technical Report No. 2773, 1984.
- [19] C. Roos, "On Terlaky path in the umbrella graph of a linear programming problem", Reports of the Department of Informatics, Delft University of Technology, No. 85-12.
- [20] T. Terlaky, "A convergent criss-cross method", *Optimization*, 16 (1985), 5, 683-690.
- [21] T. Terlaky, "A finite criss-cross method for oriented matroids", *Journal of Combinatorial Theory, Ser. B.*, 42 (1987), 319-327.
- [22] T. Terlaky, "A new algorithm for quadratic programming", *European Journal of Operations Research*, 32 (1987), 294-301.
- [23] T. Terlaky, "On lp programming", *European Journal of Operations Research*, 22 (1985), 70-101.
- [24] M.J. Todd, "Complementarity in oriented matroids", *SIAM J. Alg. Disc. Math.* 5 (1984), 467-485.
- [25] M.J. Todd, "Linear and quadratic programming in oriented matroids", *J. Combinatorial Theory Ser. B.* 39 (1985), 105-133.
- [26] A.W. Tucker, "Principal pivotal transforms of square matrices", *SIAM Review*, 5 (1963), 305.
- [27] C. Van de Panne and A. Whinston, "The symmetric formulation of the simplex method for quadratic programming", *Econometrica*, 37 (1969), 507-527.
- [28] Wang, Zh., (1987), A conformal elimination free algorithm for oriented matroid programming. *Chinese Annals of Mathematics* 8 B 1.
- [29] P. Wolfe, "The simplex method for quadratic programming", *Econometrica*, 27 (1959), 382-398.
- [30] S. Zionts, "The criss-cross method for solving linear programming problems", *Management Science* 15, (1969), 426-445.
- [31] S. Zionts, "Some empirical test of the criss-cross method", *Management Science*, 19 (1972), 406-410.