CEJOR manuscript No.
(will be inserted by the editor)

The practical behavior of the homogeneous self–dual formulations in interior point methods

Csaba Mészáros *

Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest e-mail: meszaros.csaba@sztaki.mta.hu

February 2013

Abstract Interior point methods proved to be efficient and robust tools for solving large–scale optimization problems. The standard infeasible–start implementations scope very well with wide variety of problem classes, their only serious drawback is that they detect primal or dual infeasibility by divergence and not by convergence. As an alternative, approaches based on skew–symmetric and self–dual reformulations were proposed. In our computational study we overview the implementation of interior point methods on the homogeneous self–dual formulation of optimization problems and investigate the effect of the increased dimension from numerical and computational aspects.

Key words Interior point methods – convex quadratically constrained quadratic programming – homogeneous self–dual embedding

1 Introduction

Infeasible–start interior point methods were proposed and studied by [10, 4, 11, 26] and became very popular in implementations [14, 1]. The main disadvantage of this approach are the necessary regularity conditions, for example the existence of a feasible primal and dual interior solution. As these conditions are often not practical in real–life, several techniques were developed to handle such degenerate cases [12, 16, 32].

As an alternative, Ye, Todd and Mizuno [37] proposed a skew–symmetric and self–dual reformulation for LP problems that overcomes this drawback. The approach was also generalized for monotone complementarity problems [3, 2]. Nesterov, Todd, Ye [28] proposed a different self–dual embedding. The

* Supported in part by Hungarian Research Fund OTKA K-77420.

investigations of Mizuno and Todd [27] showed the equivalence of the central paths in these approaches but pointed out that the linearizations, therefore the practical behavior of the two approaches are different. Jansen, Terlaky and Roos [9] presented a similar self–dual model in a symmetric form. Xu, Hung and Ye [35,36] considered a homogeneous self–dual feasibility (HLF) model, which was already studied by Goldman and Tucker [6,29] and proved by numerical experiments that a long–step path following algorithm can solve the HLF model efficiently [33,34].

In our study we will use the homogeneous feasibility model presented in [35] and review its implementation in the path–following interior point method and compare the practical behavior on feasible and infeasible linear and quadratically constrained quadratic programming problems (QCQPs) to the implementation of the infeasible–start version. In section 2 we will describe the homogeneous self–dual (HSD) formulation and its implementation in the framework of our BPMPD package [17]. In Section 3 we will compare the behavior if the infeasible–start and the homogeneous self–dual implementations and investigate their behavior in different situations. In Section 4 we will summarize our findings.

2 Interior point methods and the homogeneous self–dual form

In this section we will consider the primal and dual linear programming problem in the form of

$$\begin{array}{cc} \min c^T x & \max\ b^T y \\ Ax = b & A^T y + z = c \\ x \geq 0 & z \geq 0 \end{array} \tag{1}$$

where $A \in R^{m \times n},\ c, x \in R^n,\ y, b \in R^m$. Following the success of the OB1 code [13], the primal–dual interior points methods (IPMs) gained popularity in practice. Theoretically, the starting point of a primal–dual IPM should be feasible and from the close neighborhood of the central path. Since such requirement is not practical, heuristics are used in the implementations that work usually well, but may fail to generate efficient starting points in certain situations. It was also observed that the detection of primal or dual infeasibility often results in difficulties for these IPM implementations because infeasibility is detected by divergence and not by convergence.

The remedy for these drawbacks was the main motivation for the self–dual formulations, which provide a feasible, well–centered starting point for the algorithm, allow to prove infeasibility or compute an optimal solution for the original problem by a log–barrier method with good complexity.

Following Ye, Todd, Mizuno [37] let us introduce the new variables $\tau \geq 0,\quad \kappa \geq 0$ and θ and set a starting point $\left(x^0, z^0, \tau^0, \kappa^0,\ y^0,\ \theta^0\right)$ where $x^0, z^0, \tau^0, \kappa^0 > 0$ and $\theta^0 = 1$. Let us define

$$\begin{aligned} b^0 &= b\tau^0 - Ax^0, \\ c^0 &= c\tau^0 - A^T y^0 - s^0, \\ g^0 &= b^T y^0 - c^T x^0 - \kappa^0, \\ h^0 &= (z^0)^T x^0 + \tau^0 \kappa^0, \end{aligned}$$

and consider the skew–symmetric and self–dual system:

$$\begin{array}{rrrrrrr} \min & & & h^0\theta & & & \\ & Ax & -b\tau & +b^0\theta & & & = 0, \\ -A^Ty & & +c\tau & +c^0\theta & -z & & = 0, \\ b^Ty & -c^Tx & & -g^0\theta & & -\kappa & = 0, \\ -(b^0)^Ty & +(c^0)^Tx & +g^0\tau & & & & = -h^0, \\ & x \geq 0, & \tau \geq 0, & & z \geq 0, \kappa \geq 0. & & \end{array} \tag{2}$$

Ye, Todd, Mizuno [37] proved that (2) has always an optimal solution whose objective value is 0, and if $(x^*, z^*, \tau^*, \kappa^*,\ y^*,\ \theta^*)$ is a strictly complementary solution then either

$\tau^* > 0$ then x^*/τ^* and $(y^*/\tau^*, z^*/\tau^*)$ is optimal for (1), or
$\kappa^* > 0$ then (1) is infeasible.

It is easy to see that in (2)

$$x^T z + \tau\kappa = \theta h^0$$

thus the complementarity gap is linear with θ. Let

$$F = \{(x, y, z, \tau, \kappa, \theta) : \text{feasible for (2)}\}$$

then the central path may be defined as

$$P = \{(x, y, z, \tau, \kappa, \theta) \in F,\ Xz = \mu e,\ \tau\kappa = \mu\},$$

where e is the vector of ones, for some $\mu > 0$. Let us define

$$\mu^0 = \frac{\left(x^0\right)^T z^0 + \tau^0\kappa^0}{n+1},$$

then $\theta = \mu/\mu^0$, thus $\Delta\theta = -(1-\gamma)\theta$ where

$$\mu = \gamma \frac{x^T z + \tau\kappa}{n+1}.$$

In this way we can simplify (2) to a homogeneous feasibility model [35, 36] by expressing θ with μ. The resulting system can be written as

$$\begin{array}{rrrrr} Ax & & -b\tau & & = 0, \\ & -A^Ty \quad -z & +c\tau & & = 0, \\ -c^Tx & b^Ty & & -\kappa & = 0, \\ x \geq 0, & & z \geq 0, \tau \geq 0, \kappa \geq 0. & & \end{array} \tag{3}$$

This system has been already studied by Goldman and Tucker [7] and investigated with interior point methods by Xu, Hung and Ye [35,36,33,34].

In our further investigations we will consider the convex quadratically constrained quadratic programming problem (QCQP) as

$$\begin{array}{c} \min \ \frac{1}{2}x^TQx + c^Tx, \\ \frac{1}{2}x^TQ_ix + a_i^Tx \geq b_i \ \text{ for } i = 1, ..., m, \\ x \geq 0. \end{array} \tag{4}$$

where $x, a_i, c \in R^n,\ b_i \in R,\ Q, Q_i \in R^{n\times n}$, furthermore Q and $-Q_i$ for $i = 1...m$ are symmetric positive semidefinite, which conditions define a smooth convex optimization problem. The dual problem in the Wolfe-sense [31] can be formulated as:

$$\begin{array}{rl} \max & b^Ty - \frac{1}{2}\left(x^TQx - \sum_{i=1}^m y_ix^TQ_ix\right) \\ A^Ty + z = & c + Qx - \sum_{i=1}^m y_iQ_ix, \\ y, z \geq & 0. \end{array} \tag{5}$$

Using the ideas in [3,2] we derive the basic HSD algorithm for the QCQP problem (4– 5) as follows: let us introduce the variables $\tau, \kappa \geq 0$ and define the homogeneous system as

$$\begin{array}{c} \left(\frac{1}{2}\frac{x^TQ_ix}{\tau} + a_ix - s_i - \tau b_i\right)_i = 0 \text{ for } i = 1, ...m, \\ \tau c + Qx - A^Ty - \sum_{i=1}^m \frac{y_iQ_ix}{\tau} - z = 0, \\ -c^Tx - \frac{x^TQx}{\tau} + b^Ty + \sum_{i=1}^m \frac{y_ix^TQ_ix}{2\tau^2} - \kappa = 0. \end{array} \tag{6}$$

Let a starting point be $x^0, y^0, z^0, s^0, \tau^0, \kappa^0 > 0$ and define

$$\begin{array}{l} r_p = \left(\frac{1}{2}\frac{x^TQ_ix}{\tau} + a_ix - s_i - \tau b_i\right)_i, \\ r_d = c\tau + Qx - A^Ty - \sum_{i=1}^m \frac{y_iQ_ix}{\tau} - z, \\ r_g = -c^Tx - \frac{x^TQx}{\tau} + b^Ty + \sum_{i=1}^m \frac{y_ix^TQ_ix}{2\tau^2} - \kappa, \end{array}$$

and set $\mu, \eta > 0$ to perturb complementarity and feasibility as

$$\alpha_p = \eta r_p^0, \quad \alpha_d = \eta r_d^0, \quad \alpha_g = \eta r_g^0,$$
$$Xz = \mu e, \quad Sy = \mu e, \quad \tau\kappa = \mu.$$

Then the Newton–system for (6) can be formulated as

$$\begin{bmatrix} \frac{\sum Q_i x}{\tau} + A & -I & & & -b - \frac{\sum x Q_i x}{2\tau^2} \\ Q - \frac{\sum y_i Q_i}{\tau} & & -I & -A - \frac{\sum Q_i x}{\tau} & c + \frac{\sum y_i Q_i x}{\tau^2} \\ -c - \frac{2Qx}{\tau} + \frac{\sum y_i Q_i x}{\tau^2} & & b + \frac{\sum x Q_i x}{2\tau^2} & \frac{xQx}{\tau^2} - \frac{\sum y_i x Q_i x}{\tau^3} & -1 \\ Z & & X & & \\ & Y & & S & \\ & & & \kappa & \tau \end{bmatrix} \begin{bmatrix} d_x \\ d_s \\ d_z \\ d_y \\ d_\tau \\ d_\kappa \end{bmatrix} =$$
$$= \begin{bmatrix} \alpha_p \\ \alpha_d \\ \alpha_g \\ \mu e - Xz \\ \mu e - Sy \\ \mu - \tau\kappa \end{bmatrix}.$$

The above system can be simplified by the elimination of d_κ, d_s and d_z to

$$\begin{bmatrix} Q - \frac{\sum y_i Q_i}{\tau} + X^{-1}z & -\frac{\sum Q_i x}{\tau} - A^T & c_0 + \frac{\sum y_i Q_i x}{\tau^2} \\ \frac{\sum Q_i x}{\tau} + A & Y^{-1}s & -b - \frac{\sum x Q_i x}{2\tau^2} \\ -c_0 - \frac{2Q_0 x}{\tau} + \frac{\sum y_i Q_i x}{\tau^2} & b + \frac{\sum x Q_i x}{2\tau^2} & \frac{xQx}{\tau^2} - \frac{\sum y_i x Q_i x}{\tau^3} + \frac{\kappa}{\tau} \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_\tau \end{bmatrix} =$$
$$= \begin{bmatrix} \alpha_p + X^{-1}(\mu e - Xz) \\ \alpha_d + Y^{-1}(\mu e - Sy) \\ \alpha_g + \frac{\mu - \tau\kappa}{\tau} \end{bmatrix}. \quad (7)$$

By using the notations

$$Q(y,\tau) = Q - \frac{\sum y_i Q_i}{\tau},$$
$$A(x,\tau) = A + \frac{\sum Q_i x}{\tau} \text{ and}$$
$$M = \begin{bmatrix} -Q(y,\tau) - X^{-1}z & A(x,\tau)^T \\ A(x,\tau) & Y^{-1}s \end{bmatrix}$$

the reduced Newtons system can be written as

$$\begin{bmatrix} M & g \\ f & h \end{bmatrix} \begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} \gamma \\ \delta \end{bmatrix}, \quad (8)$$

where f, g, γ and δ are defined by (7). In each iteration we define and solve systems in the form of (8) and derive the components of the search direction by substitution.

The solution of (8) can be computed by the following steps:

$$\begin{aligned} a &= M^{-1}g, \\ d &= M^{-1}\gamma, \\ d_y &= \frac{\delta - d}{h - fa}, \\ d_x &= M^{-1}(\gamma - gd_y) = d - d_y a. \end{aligned}$$

Clearly, the additional work compared to the infeasible primal–dual method is one backsolve operation to compute $a = M^{-1}g$ in every iteration. It is to be noted, however, that the operations $a = M^{-1}g$ and $d = M^{-1}\gamma$ are independent, therefore their computation can be combined into one backsolve step with two vectors which can be performed significantly faster than two consecutive backsolves due to the reduced data flow.

2.1 Implementation and numerical results

We implemented the approach in our software called BPMPD [17]. In our experiments we compare the infeasible–start (IS) and homogeneous self–dual (HSD) approaches in this framework. Our implementation employs presolve [24] and sparse matrix ordering [21] to reduce the computational work and to improve the efficiency. After reordering the problem for sparsity, our implementation in every iteration computes a symmetric factorization

$$M = L\Lambda L^T,$$

where $L \in R^{(m+n)\times(m+n)}$ symmetric lower triangular and $\Lambda \in R^{(m+n)\times(m+n)}$ is diagonal. The quasidefinite property of M ensures that such decomposition always exists [30]. This is the computationally most costly operation and our implementation uses vectorization and parallel computation techniques to exploit the hardware features [20]. The decomposition is used in several backsolve steps to compute the predictor–corrector direction [15] and further corrector steps [8]. After the determination of the search directions, suitable steplengths are chosen to preserve the nonnegativity of the variables and to optimize a merit function that warrants the convergence.

In our experiments we used two different starting points. The first one, called "simple" sets the variables $x = z = e, \quad y = s = e$ and additionally $\tau = \kappa = 1$ for the HSD variant. The other starting point is our "standard" version which is described in [22] in details. The approach is based on the relaxed problem

$$\begin{aligned} \max \ & \frac{1}{2}\sum_{i=1}^{m} w_i \left(x^T Q_i x\right) - \sum_{j=1}^{n} x_j^2 \\ & A(0)x - s = b, \end{aligned} \tag{9}$$

where $w_i > 0$ and set accordingly to $\|Q_i\|$ and b_i for $i = 1, ..., m$. The optimal solution (x^*, s^*) of the above problem can be computed directly and (x^0, s^0) is derived by shifting (x^*, s^*) into the positive orthant, similarly as described in [1]. The starting point for the dual variables is set as: $y^0 = w,\ z^0 = c_0 + Q_0 x^0 - A(x^0)^T y^0 + \delta$ where $\delta > 0$ is chosen suitably to provide positiveness. For the HSD approach, τ^0 is set to 1 and κ^0 is computed as

$$\kappa^0 = \frac{\left(x^0\right)^T z^0 + \left(y^0\right)^T s^0}{n+m}.$$

First, we compare the performance on feasible problems with different starting points. We selected the largest problems form the NETLIB [5] feasible testcases and the results are summarized in Table 1. Figures given include the number of iterations to achieve 8-digits of accuracy by the two methods using the two starting points. Failures are marked by the "*" symbol.

Table 1 Efficiency on standard feasible LP problems

Problem	IS IPM		HSD IPM	
name	simple	standard	simple	standard
25fv47	341	15	15	14
80bau3b	45*	29	28	26
bnl2	22*	17	19	18
cycle	33*	20	19	17
d2q06c	400*	18	18	16
degen3	9	12	9	9
dfl001	203*	20	26	23
fit2p	72	17	16	22
fir2d	27	15	19	21
greenbeb	50*	22	25	30
grow22	41	13	15	15
nem	25*	28	28	28
pilot87	113	22	33	30
sctap3	16	13	10	10
ship12l	31	13	20	19
stocfor3	36*	18	21	18
truss	33	13	15	15
Average		17.94	19.76	19.47

The results show that the infeasible–start IPM fails to converge on several test cases with the simple starting point, in contrary to the HSD implementation, which performed nearly equally with both starting point choices. The results also indicate that the HSD implementation needs about 10% more iterations than the IS implementation when using the more efficient starting point choice.

In the second experiments we compared the performance on infeasible cases. Here we selected the infeasible test set of the NETLIB repository. It is

to be noted, that in most cases our presolve already detects the infeasibility in the problems. We included in Table 2 only the test cases which required the execution of the interior point algorithm.

Table 2 Efficiency on standard infeasible LP problems

Problem name	IS IPM	HSD IPM	τ
box1	3	5	10^{-9}
cplex2	34	34	10^{-3}
ex72a	4	4	10^{-9}
ex73a	4	4	10^{-9}
forest6	7	7	10^{-11}
klein1	19	9	10^{-6}
klein2	10	6	10^{-5}
klein3	23	9	10^{-7}
mondou2	4	7	10^{-8}
pang	15	23	10^{-9}
qual	16	12	10^{-8}
refinery	8	9	10^{-10}
vol1	13	11	10^{-10}

Figures given include the number of iterations and the final value of τ in case of the HSD algorithm. The infeasible status was correctly identified by both algorithms in all cases, and in average, the effort is roughly equal. Let us note that on these problems the minimal infeasibility is considerably large while the problem sizes are rather moderate, this is why the infeasible–start algorithm had no trouble to detect the infeasibility. Our experiences on real–life problems showed that infeasible problems, which are at the "boundary" of the feasibility, i.e. on which the norm of the minimal infeasibility is small (e.g. $< 10^{-6}$), often presents a challenge for the infeasible–start implementation.

Table 3 presents test results on feasible QCQP testcases from [25]. The results indicate the infeasible–start IPM is very sensitive to the choice of the starting point on QCQP problems.

In our next experiment we created such a testcase. Our QCQP testcase has 10 variables, 4 linear and one quadratic constraint. The quadratic constraint was defined as

$$a_1 x + \frac{1}{2} x^T Q_1 x \leq z + \epsilon \quad \text{where} \quad z = \min \; a_1 x + \frac{1}{2} x^T Q_1 x$$

i.e. with $\epsilon \geq 0$ the problem is feasible, otherwise it is infeasible. Table 4 collects the results with different values of ϵ.

The experiments show that both versions are equally robust to solve the feasible cases with $\epsilon > 0$. The results also show that the HSD implementation behaves significantly better if the problem is infeasible and the infeasibility is very small.

Table 3 Efficiency on standard feasible QCQP problems

Problem	IS IPM		HSD IPM	
name	simple	standard	simple	standard
boyd1	87	23	17	15
boyd2	380*	89	73	75
cont-201	23	12	11	11
cont-300	152*	13	11	10
cvxqp1-l	128	12	8	12
cvxqp2-l	400*	12	8	8
cvxqp3-l	400*	15	7	7
stadat1	78	23	42	40
stadat2	39*	18	12	14
stadat3	57*	20	13	13
laser	56	13	8	7
exdata	112	19	11	10
Average		22.42	19.76	18.50

Table 4 Behaviour on the boundary of feasibility on a QCQP problem

ϵ	IS IPM	HSD
10^{-4}	8	8
10^{-6}	9	12
10^{-8}	10	16
10^{-9}	fail	fail
0	12	12
-10^{-4}	15	12
-10^{-6}	fail	16
-10^{-8}	fail	32
-10^{-10}	fail	42

In [18] it was pointed out that badly scaled optimal solutions may present numerical challenges for the interior point methods due to the increasing ill–conditioning of the underlying Newton system. This situation was further discussed in [19, 23]. Most common causes are problems with unbounded optimal faces and the presence of large bounds. To compare the behavior on such case, we created a QCQP testcase with 4 linear and 1 quadratic constraints, 9 nonnegative and one free variable, whose optimal solution is (1,...,1). Then we replaced the free variable in the model as $M \geq x_f \geq -M$, where $M \geq 1$. Table 5 collects the number of iterations depending on the value of M. During the execution we turned off our dynamic bound relaxation technique [17] that was developed to help in similar situations.

The results show that the behavior of the two implementation is rather similar and both fail if M is sufficiently large.

Our last test examples were derived form 57 mixed integer QCQP test cases of varying sizes, ranging the number of constraints between 3 and 23562 and the number of variables between 5 and 31316. In total, 698 subproblems were extracted from the branch and bound trees which presented

Table 5 Effect of ill–conditioning

M	IS IPM	HSD
10^0	6	5
10^2	6	6
10^4	7	9
10^6	7	11
10^7	14	18
10^8	15	fail
10^9	fail	fail
10^{10}	fail	fail
10^{11}	fail	fail

difficulties for the embedded previous solver. We found that 297 of these problems were not successfully solved with our infeasible–start implementation either. Switching to the HSD implementation the unsuccessfully solved cases were reduced to 33, and we found that these needed only some tweaking in the default parameters.

2.2 Conclusion

Our experiments showed that the implementation of interior point methods for the homogeneous self–dual model requires very moderate additional work per iteration compared to the infeasible–start implementation. This additional work can be greatly reduced with a special backsolve operation by exploiting data independence in the required steps. The main disadvantage is the slightly increased iteration numbers on the feasible problems. We also observed that present implementations of the infeasible–start methods are robust to detect infeasibility if the inconsistency in the constraints is sufficiently large. On infeasible problems with little inconsistency the homogeneous self–dual model showed its advantages and proved to be very reliable. This was also observed when the solver was used in the branch and bound tree to solve relaxations of mixed integer problems, where the appearance of infeasible cases is more likely. The homogeneous self–dual model is also very robust to the choice of the starting point. Our experiments also showed that both versions face similar numerical problems with badly scaled solutions.

References

1. E.D. Andersen, J. Gondzio, C. Mészáros, and X. Xu. Implementation of interior point methods for large scale linear programs. In T. Terlaky, editor, *Interior Point Methods of Mathematical Programming*, pages 189–252. Kluwer Academic Publishers, 1996.
2. E.D. Andersen and Y. Ye. A computational study of the homogeneous algorithm for large–scale convex optimization. *Computational Optimization and Applications*, 10:243–289, 1998.

3. E.D. Andersen and Y. Ye. On a homogeneous algorithm for the monotone complementarity problem. *Mathematical Programming*, 84(2):375–399, 1999.
4. K.M. Anstreicher and J.-P. Vial. On the convergence of an infeasible primal-dual interior-point method for convex programming. *Optimization Methods and Software*, 3:285–316, 1994.
5. D.M. Gay. Electronic mail distribution of linear programming test problems. *COAL Newsletter*, 13:10–12, 1985.
6. A.J. Goldman and A.W. Tucker. Polyhedral convex cones. In H.W. Kuhn and A.W. Tucker, editors, *Linear Inequalities and Related Systems*, pages 19–40, Princeton, New Jersey, 1956. Princeton University Press.
7. A.J. Goldman and A.W. Tucker. Theory of linear programming. In H.W. Kuhn and A.W. Tucker, editors, *Linear Inequalities and Related Systems*, pages 53–97, Princeton, New Jersey, 1956. Princeton University Press.
8. J. Gondzio. Multiple centrality corrections in a primal-dual method for linear programming. *Computational Optimization and Applications*, 6:137–156, 1996.
9. B. Jansen, T. Terlaky, and C. Roos. The theory of linear programming: Skew symmetric self-dual problems and the central path. *Optimization*, 29:225–233, 1994.
10. M. Kojima, N. Megiddo, and S. Mizuno. A primal-dual infeasible-interior-point algorithm for linear programming. Technical report, 1991.
11. M. Kojima, N. Megiddo, and S. Mizuno. A primal-dual infeasible-interior-point algorithm for linear programming. *Mathematical Programming*, 61:263–280, 1993.
12. I.J. Lustig. Feasibility issues in primal-dual interior-point methods for linear programming. *Mathematical Programming*, 49:145–162, 1990.
13. I.J. Lustig, R.E. Marsten, and D.F. Shanno. On implementing Mehrotra's predictor-corrector interior-point method for linear programming. *SIAM J. on Optimization*, 2(3):435–449, 1992.
14. I.J. Lustig, R.E. Marsten, and D.F. Shanno. Interior point methods for linear programming: Computational state of the art. *ORSA J. on Computing*, 6(1):1–15, 1994.
15. S. Mehrotra. High order methods and their performance. Technical Report 90-16R1, Department of Industrial Engineering and Managment Sciences, Northwestern University, Evanston, USA., 1991.
16. C. Mészáros. On free variables in interior point methods. *Optimization Methods and Software*, 9:121–139, 1997.
17. C. Mészáros. The BPMPD interior-point solver for convex quadratic problems. *Optimization Methods and Software*, 11&12:431–449, 1999.
18. C. Mészáros. On the Cholesky factorization in interior point methods. *Computers & Mathematics with Applications*, 50:1157–1166, 2005.
19. C. Mészáros. On numerical issues of interior point methods. *SIAM J. on Matrix Analysis*, 30(1):223–235, 2008.
20. C. Mészáros. On the implementation of interior point methods for dual–core platforms. *Optimization Mathods and Software*, 25(3):449–456, 2010.
21. C. Mészáros. On sparse matrix orderings in interior point methods. Working paper, Computer and Automation Institute, Hungarian Academy of Sciences, Budapest, 2011.
22. C. Mészáros. Solving quadratically constrained convex optimization problems with an interior point method. *Optimization Methods and Software*, 26(3):421–429, 2011.

23. C. Mészáros. Regularization techniques in interior point methods. *Journal of Computational and Applied Mathematics*, 236:3704–3709, 2012.
24. C. Mészáros and U.H. Suhl. Advanced preprocessing techniques for linear and quadratic programming. *OR Spectrum*, 25:575–595, 2004.
25. H.D. Mittelmann and P. Spellucci. Decision tree for optimization software. World Wide Web, http://plato.la.asu.edu/guide.html, 1998.
26. S. Mizuno. Polynomiality of infeasible-interior-point algorithms for linear programming. *Mathematical Programming*, 67(1), 1994.
27. Shinji Mizuno and Michael J. Todd. On two homogeneous self-dual approaches to linear programming and its extensions. *Mathematical Programming*, 89:517–534, 2001.
28. Y. Nesterov, M.J. Todd, and Y. Ye. Infeasible–start primal–dual methods and infeasibility detectors for nonlinear programming problems. *Mathematical Programming*, 84:227–267, 1999.
29. A.W. Tucker. Dual systems of homogeneous linear relations. In *Linear Inequalities and Related Systems*, pages 3–18. Princeton University Press, Princeton, NJ, 1956.
30. R.J. Vanderbei. Symmetric quasi-definite matrices. *SIAM J. on Optimization*, 5(1):100–113, February 1995.
31. P. Wolfe. A duality theorem for non-linear programming. *Quarterly of Applied Mathematics*, 19:239–244, 1961.
32. S.J. Wright. Modified Cholesky factorizations in interior-point algorithms for linear programming. *SIAM J. on Optimization*, 9(4):1159–1191, 1999.
33. X. Xu. An $\mathcal{O}(\sqrt{n}L)$-iteration large-step infeasible path-following algorithm for linear programming. Technical report, College of Business Administration, The University of Iowa, Iowa City, IA 52242, August 1994.
34. X. Xu. On the implementation of a homogeneous and self-dual linear programming algorithm. *Mathematical Programming*, 76(2):155–181, 1996.
35. X. Xu, P.-F. Hung, and Y. Ye. A simplified homogeneous and self-dual linear programming algorithm and its implementation. *Annals of Operations Research*, 62(1):151–171, 1996.
36. X. Xu and Y. Ye. A generalized homogeneous and self-dual algorithm for linear programming. *Oper. Res. Lett.*, 17:181–190, 1995.
37. Y. Ye, M.J. Todd, and S. Mizuno. An $O(\sqrt{n}L)$ - iteration homogeneous and self-dual linear programming algorithm. *Math. Oper. Res.*, 19:53–67, 1994.