A symmetric reduction of the NT direction

Florian Jarre\textsuperscript{a}, Chantal Hergenroeder\textsuperscript{a}

Nov. 24. 2013

\textsuperscript{a} Mathematisches Institut, Heinrich-Heine-Universität Düsseldorf, D-40225 Düsseldorf, Germany

Abstract

A stable symmetrization of the linear systems arising in interior-point methods for solving linear programs is introduced. A comparison of the condition numbers of the resulting interior-point linear systems with other commonly used approaches indicates that the new approach may be best suitable for an iterative solution. It is shown that there is a natural generalization of this symmetrization to the NT search direction for solving semidefinite programs. The generalization includes a novel pivoting strategy to minimize the norm of the right and side and heavily relies on the symmetry properties of the NT direction. As a byproduct, an approach to stabilize the systems of Schur-complement based interior-point solvers is derived. The search directions generated by iterative solvers typically have fairly low relative accuracy. Nevertheless, in some preliminary numerical examples, a suitably adapted interior point approach results in a rather small number of outer iterations.

Key words: Linear program; Semidefinite program; Condition number; Quasi- minimal residual iteration; Interior-point algorithm.

1. Introduction

This paper is concerned with large scale semidefinite programs that do not allow a direct factorization of the matrices associated with the linear systems arising in interior-point methods. Thus, the solutions of these systems will be approximated iteratively. Prior to applying an iterative solver to these systems, the systems are reformulated to equivalent systems with suitable numerical properties.

As explained below in more detail, the following aspects of suitable properties – in the order of decreasing importance – are considered:

1. Avoid \textit{systematic} cancellation errors in the transformations of the right hand side.
2. Generate linear systems with \textit{bounded} condition numbers.
3. Preserve the \textit{sparsity} structure of the constraints.
4. Generate \textit{symmetric} systems.
5. Generate \textit{positive definite} systems.
We will frequently refer to the above points 1. – 5. in the remainder of this paper.

In Point 1. we distinguish between ill-conditioning due to the problem data and systematic ill-conditioning introduced, for example, by pivoting with elements of small absolute value.

Point 4. is motivated by two observations: Symmetric systems can be solved, for example, by the QMR algorithm, see [7]. Symmetry not only reduces the amount of work per iteration by a factor of one half; also the number of iterations generally decreases compared to non-symmetric (non-normal) systems with the same condition number.

There has been growing interest in the solution of large scale semidefinite programs in the recent past, and in particular the paper [22] presents a very efficient numerical implementation. The paper [9] addresses the stable solution of interior-point systems for linear programs and [4] is concerned with a matrix-free approach for semidefinite programs based on a Gauss-Newton step. In this paper we also consider semidefinite programs, but remain with the standard symmetrization operators, in particular, with the Nesterov-Todd symmetrization, [15]. As detailed below, the ideas of this paper are related to a stable approach for solving reduced KKT systems arising in interior-point methods for linear programs in [2]. Further work addressing the stability of the direct solution of the reduced KKT systems is described in [13, 12] while the recent work [16] applies iterative methods to the full system.

In Section 2, the above concepts are studied for the linear systems arising in interior-point methods for solving linear programs. A new system is defined that satisfies the first four points above. A summary comparing the properties of the commonly used systems and the new system motivates a natural generalization of the new system for semidefinite programs in Section 3. The generalization heavily relies on the symmetry properties of the NT direction and can be modified to define an approach for reducing the cancellation errors of the standard systems in interior point methods. Section 4 closes with some preliminary numerical examples for semidefinite programs.

2. Linear programs

To motivate our approach for semidefinite programs we first consider the simpler case of linear programs in the following standard form

\[
\begin{align*}
\minimize_{x \in \mathbb{R}^n} & \quad c^T x \\
\text{subject to} & \quad Ax = b \quad \text{and} \quad x \geq 0,
\end{align*}
\]

where \( A \) is a given real \( m \times n \) matrix, and \( b \) and \( c \) are given real vectors of length \( m \) and \( n \). Further, let \( e := (1, \ldots, 1)^T \) be the all-ones-vector in \( \mathbb{R}^n \) and \( I \) be the identity matrix (with dimension given by the context). The points \( x, s \in \mathbb{R}^n, y \in \mathbb{R}^{m} \) on the central path of problem (1) are defined by the equations:

\[
\begin{align*}
Ax &= b, \\
A^T y + s &= c, \\
X s &= \mu e,
\end{align*}
\]

where \( x > 0, s > 0 \) and

\[
X := \text{Diag}(x_1, x_2, \ldots, x_n)
\]
is the diagonal matrix associated with the vector \( x = [x_1, x_2, \cdots, x_n]^T \in \mathbb{R}^n \) and \( \mu > 0 \) is a scalar parameter. Given an approximate solution \( x > 0, s > 0, y \in \mathbb{R}^m \) to (2) let
\[
p := b - Ax, \quad q := c - A^Ty - s, \quad \text{and} \quad r := \mu e - Xs.
\]
(3)

The value of \( \mu \) associated with the current point \( x, s \) is given by \( x^T s / n \), and the “new target value” is defined as \( \mu^+ := \sigma x^T s / n \) with \( \sigma \in (0, 1) \). Newton’s method for solving (2) then yields the system of linear equations
\[
\begin{bmatrix}
0 & A & 0 \\
A^T & 0 & I \\
0 & S & X
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\Delta x \\
\Delta s
\end{bmatrix}
= \begin{bmatrix}
p \\
q \\
r
\end{bmatrix}.
\]
(4)

This system is referred to as “Standard System”, (Std.S).

Below, we compare this system to three alternative systems, the stable reduction (11), the normal equations (13), and a (new) scaled reduction (14). While all systems return the exact same solution in the absence of rounding errors, we identify significant differences among all five approaches with respect to cancellation errors and conditioning of the system matrix.

We start with a brief repetition of a symmetric reduction presented in [5]: To reduce the dimension of system (4), one may eliminate some variables by using the diagonal structure of some of the blocks of the system matrix in (4); specifically, one may choose the matrix \( I \) in the second block row of (4) or the matrix \( X \) in the third block row to eliminate \( \Delta s \), or one may choose the matrix \( S \) in the third block row to eliminate \( \Delta x \). In the former two cases, the elimination of \( \Delta s \) (combined with a diagonal rescaling of the resulting system) leads to
\[
\begin{bmatrix}
0 & A \\
A^T & -SX^{-1}
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\Delta x
\end{bmatrix}
= \begin{bmatrix}
p \\
q - X^{-1}r
\end{bmatrix},
\]
(5)

while elimination of \( \Delta x \) leads to
\[
\begin{bmatrix}
0 & A \\
A^T & -SX^{-1}
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
u
\end{bmatrix}
= \begin{bmatrix}
p - AS^{-1}r \\
q
\end{bmatrix}.
\]
(6)

Here, system (6) is written with a symmetric matrix by defining \( u := -XS^{-1} \Delta s \).

As some of the components of \( x \) and of \( s \) are tiny in the final stage of an interior-point algorithm, the approaches (5) and (6) both lead to a systematic cancellation error in the right hand side – in violation of Point 1. in Section 1. As observed in [5], this violation implies strong limitations for iterative schemes for (5) or (6) in the final stage of the interior-point algorithm.

However, as proposed in [5], combining both approaches, (5) and (6) can be used to obtain a stable reduction of the standard \( 3 \times 3 \)-block system (4) to a linear system with a \( 2 \times 2 \)-block matrix. The basis of this stable reduction is a partition of the vectors \( x \) and \( s \) into two parts \( x_1, x_2 \) and \( s_1, s_2 \), respectively, such that \( x_1 \geq s_1 \) and \( x_2 < s_2 \) is satisfied.

The \( \Delta s_1 \)-part is then eliminated via (5), and the \( \Delta x_2 \)-part is eliminated via (6): Without loss of generality, we assume that \( x_1 \) and \( s_1 \) are the leading entries of \( x \) and \( s \), i.e.,
\[
x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \text{and} \quad s = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}.
\]
(7)

\(^1\) Here, subscripts, as in \( x_1 \) and \( s_2 \), refer to this partition of the vectors \( x \) and \( s \), and they do not denote single components of the vectors \( x \) and \( s \).
Conforming with (7), we write the constraint matrix $A$ as follows:

$$A = [A_1 \ A_2].$$

(8)

Similarly, (7) induces the partition

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \quad \text{and} \quad r = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}$$

(9)

of the residual vectors $q$ and $r$ defined in (3). By $X_1$, $X_2$, $S_1$, and $S_2$ we denote the diagonal matrices associated with the vectors $x_1$, $x_2$, $s_1$, and $s_2$, respectively. Setting

$$\tilde{q} := \begin{bmatrix} q_1 - X_1^{-1}r_1 \\ q_2 \end{bmatrix} \quad \text{and} \quad \tilde{p} := p - A_2 S_2^{-1} r_2,$$

(10)

we define the “Stable Reduction” $(\text{Stb.R})$

$$\begin{bmatrix} 0 \\ A^T \\ -S X^{-1} \end{bmatrix} \begin{bmatrix} \Delta y \\ u \end{bmatrix} = \begin{bmatrix} \tilde{p} \\ \tilde{q} \end{bmatrix}.$$ 

(11)

Note that only the inverses of the large components (of $x_1$ and of $s_2$) are used in (9). Here, the new variable $u$ is also partitioned into $u_1, u_2$ conforming with (7). Using (7)–(9), and (10), together with

$$\Delta x_1 = u_1$$

$$\Delta x_2 = S_2^{-1} r_2 + u_2,$$

$$\Delta s = q - A^T \Delta y,$$

(12)

one readily verifies that the reduced $2 \times 2$-block system (11) is indeed equivalent to the standard system (4). Since also the computation of $\Delta x, \Delta s$ in (12) only involves the inverses of the large components of $s$, this reduction may be considered “stable” – justifying the name $(\text{Stb.R})$.

**Note:** When $x, s$ are close to the central path, the vector $r$ is close to a multiple of the all-ones-vector, saying all of its entries are of about the same magnitude. Thus, if the partition (7) was modified such that $x_1$ contains some small components of $x$ and $s_2$ contains some small components of $s$ then the norm of right hand side $\tilde{p}$, $\tilde{q}$ in (10) would increase. This observation allows a “dual” approach for selecting the partition $x_1, x_2$, namely such that the norm of the right hand side in (10) is minimized. This is the approach that will be taken when generalizing the partition (7) to semidefinite programs below.

A further reduction to the Schur complement leads to the “Normal Equations” $(\text{Nrm.E})$

$$AXS^{-1} A^T \Delta y = p + AS^{-1}(Xq - r).$$

(13)

The matrix in (13) satisfies Point 5. in Section 1, but in the final stage of an interior-point algorithm, system (13) will violate Point 1.

While the stable reduction does not introduce any systematic cancellation error in the right hand side, it will be shown in the next section that it does systematically increase the condition number of the $2 \times 2$-block KKT matrix. This systematic deterioration of the condition number of
the $2 \times 2$-system matrix was also observed in [16], and therefore, the authors in [16] revert to the $3 \times 3$-block matrix, adding a suitable regularization in order to apply an iterative scheme. Below, we use a seemingly unstable approach to remain with the $2 \times 2$-block KKT matrix and scale this matrix by a diagonal matrix that is singular in the limit as $\mu \rightarrow 0$ (and $Xs \approx \mu e$). We will then analyze how to apply a back solve such that this nearly singular scaling does not introduce any systematic cancellation error. To simplify the further derivations we denote the diagonal matrices

$$D_x := \begin{bmatrix} I & 0 \\ 0 & S_2^{-1/2}x_2^{1/2} \end{bmatrix}, \quad D_s := \begin{bmatrix} X_1^{-1}S_1 & 0 \\ 0 & I \end{bmatrix}.$$ 

Then, consider the system obtained by rescaling (11) from left and right with $D_x$,

$$\begin{bmatrix} 0 & AD_x \\ D_xA^T & D_s \end{bmatrix}\begin{bmatrix} \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} A \begin{bmatrix} 0 \\ S_2^{-1}r_2 \end{bmatrix} - p \\ D_xq - \begin{bmatrix} X_1^{-1}r_1 \\ 0 \end{bmatrix} \end{bmatrix}.$$ (14)

Given the solution of (14), we define $\Delta x$ and $\Delta s$ via

$$\Delta x = \begin{bmatrix} 0 \\ S_2^{-1}r_2 \end{bmatrix} - D_x\Delta z, \quad \Delta s = q - A^T\Delta y.$$ (15)

**Proposition 2.1** The solution of (14), (15) coincides with the solution of (4).

**Proof:** The last equation of (15) is equivalent to the second equation of (4). Solving (15) for $\Delta z$ and for the term “$A^T\Delta y$” and inserting this into the second block row of (14) we obtain

$$D_x(q - \Delta s) + D_sD_x^{-1}\left(\begin{bmatrix} 0 \\ S_2^{-1}r_2 \end{bmatrix} - \Delta x\right) = D_xq - \begin{bmatrix} X_1^{-1}r_1 \\ 0 \end{bmatrix}.$$ 

The term $D_xq$ on both sides cancels, and left multiplying this with $XD_x^{-1}$ yields

$$-X\Delta s + \begin{bmatrix} 0 \\ r_2 \end{bmatrix} - S\Delta x = -\begin{bmatrix} r_1 \\ 0 \end{bmatrix}$$

which is equivalent to the third block row of (4). Finally, the first block row of (14) yields

$$A\left(\begin{bmatrix} 0 \\ S_2^{-1}r_2 \end{bmatrix} - \Delta x\right) = A\begin{bmatrix} 0 \\ S_2^{-1}r_2 \end{bmatrix} - p$$

which is equivalent to the first block row of (4). #

System (14) was also considered by Gill et al. in [8]. Note that rescaling the matrix in (11) with $D_x$ may be problematic because $D_x$ is singular in the limit when $x_2 \rightarrow 0$. However, there is no systematic magnification of the right hand side in (14) (as long the iterates $x$ and $s$ remain bounded); only the entries of $q_2$ are multiplied by tiny numbers in the right hand side of (14). Moreover, as $x_2 \rightarrow 0$, System (2) converges to a system that is “logically block-triangular” in the sense that the unknowns $\Delta x, \Delta y, \Delta s_1$ can be determined first independently of $q_2$, and then, based on $\Delta y$ and $q_2$, the correction $\Delta s_2$ can be determined in a stable fashion. For small $x_2 > 0$ while the information of $q_2$ is systematically canceled in (14), the back substitution (15) recovers $q_2$ in a stable fashion. Thus, the special structure of (2) does allow rescaling with a matrix $D_x$ that is singular in the limit without introducing systematic cancellation.
System (14) will be denoted by “scaled reduction” (Scl.R). A further reduction of (14) is possible by taking the Schur complement with respect to the $I$-block of the matrix $D_s$. A system with the same structure as the one obtained by such a further reduction was considered in [2]. While the derivations in [2] are quite different from the ones in this paper, they also result in a very stable approach for solving linear programs. However, such reduction may destroy the sparsity structure of the constraint matrix, and in particular so for semidefinite programs considered next; we therefore do not pursue it in this paper.

Next, we present a brief comparison of the systems considered so far.

### 2.1 Comparison for LP

We compare the Standard System (4) (Std.S), the Stable Reduction (11) (Stb.R), and the Normal Equations (13) (Nrm.E) for some random problems.

Random problems are known to be unreliable when aiming to establish the efficiency of a given method. Here, the random problems are used here to establish inefficiency of certain competing approaches. The efficiency of the remaining approach needs to be established by other means.

To detect possible systematic ill-conditioning of the above four systems, we generated some nondegenerate random linear instances. Systems that show ill-conditioning even for such simple examples, are unlikely to be of practical interest for iterative solvers.

In the next section we will consider the case where a Cholesky factor $L$ of $AA^T$ is available at moderate cost, and where the matrix $A$ can be premultiplied by the inverse of the Cholesky factor (resulting in a matrix with orthonormal rows). Table 1. below is based on 100 nondegenerate random examples of the same size using such orthogonalization of the constraint matrix $A$. (When choosing $A$ with independent entries form a standard normal distribution, all results get slightly worse, but the comparison of the approaches remains the same.)

For each example, the matrix $A \in \mathbb{R}^{100 \times 250}$ was chosen with (pseudo-) random orthonormal rows. This normalization is motivated by the extension to semidefinite programs in the next section, where a sparse Cholesky factor of $AA^T$ is available for many problems, but where nevertheless, the interior-point systems cannot be factorized directly. To further separate the effects of systematic ill-conditioning from the effects of ill-conditioning due to the nature of the underlying problem, the vectors $x, s$ were generated in a “fairly well-conditioned fashion”. More precisely, for a given value of $\mu > 0$, some nondegenerate nearly central points were generated as follows: $x > 0$ having $m$ small entries uniformly distributed in $[0, 1 - \mu, 1 + \mu]$ and $n - m$ larger entries in $[0, 1, 1]$, and $s > 0$ such that $\|Xs - \mu e\|_\infty \leq 0.5\mu$.

The first column of Table 1 lists different values of $\mu$ and the remaining columns list the average of the condition numbers of the standard system, the stable reduction, and the normal equations. In brackets, also the maximum and the minimum values are listed.

<table>
<thead>
<tr>
<th>$\mu$</th>
<th>cond. (Std.S)</th>
<th>cond. (Stb.R)</th>
<th>cond. (Nrm.E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>170 (230/120)</td>
<td>1000 (1500/720)</td>
<td>11 (13/7.2)</td>
</tr>
<tr>
<td>1.0e-4</td>
<td>1.4e4 (3.9e5/2400)</td>
<td>3.5e8 (2.1e10/5.4e7)</td>
<td>6.1e5 (2.0e8/1.5e5)</td>
</tr>
<tr>
<td>1.0e-8</td>
<td>1.5e4 (2.8e6/2400)</td>
<td>3.0e12 (4.7e14/5.3e11)</td>
<td>6.9e5 (2.6e10/2.3e5)</td>
</tr>
</tbody>
</table>

When representing the block elimination from (Std.S) to (Stb.R) as a matrix decomposition, the condition number of the system matrix in (Stb.R) cannot be worse than the product of the
condition numbers of the matrices in its factorization – provided the condition is measured with a submultiplicative norm. Thus, it may seem counterintuitive that a stable reduction transforms a system with moderate condition number (1.5e4 for (Std.S) with $\mu = 10^{-8}$) into a system with a very bad condition number (3.0e12 for (Stb.R) with $\mu = 10^{-8}$). To elaborate on this intuition, denote the projections onto the first components and onto the remaining components by

$$P_1 = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad P_2 = I - P_1 = \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix},$$

again, conforming with the partition in (7). Then, (11) and (12) of (Stb.R) can be written as

$$\begin{bmatrix} 0 & A & 0 & 0 \\ A^T & -SX^{-1} & 0 & 0 \\ 0 & -I & I & 0 \\ A^T & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \Delta y \\ u \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} p - AP_2 S^{-1} r \\ q - P_1 X^{-1} r \\ P_2 S^{-1} r \\ q \end{bmatrix}.$$  

To relate this with (Std.S), we augment (Std.S) by including the variable $u = \Delta x - P_2 S^{-1} r$ and obtain (after reordering) an “augmented standard system” (A.Std.S),

$$\begin{bmatrix} 0 & 0 & A & 0 \\ 0 & 0 & S & X \\ 0 & -I & I & 0 \\ A^T & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \Delta y \\ u \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} p \\ r \\ P_2 S^{-1} r \\ q \end{bmatrix}.$$  

It then follows that the system matrices of (A.Std.S) and (Stb.R) are related via

$$\begin{bmatrix} I & 0 & -A & 0 \\ 0 & -X^{-1} & SX^{-1} & I \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} 0 & 0 & A & 0 \\ 0 & 0 & S & X \\ 0 & -I & I & 0 \\ A^T & 0 & 0 & I \end{bmatrix} \begin{bmatrix} 0 & A & 0 & 0 \\ A^T & -SX^{-1} & 0 & 0 \\ 0 & -I & I & 0 \\ A^T & 0 & 0 & I \end{bmatrix},$$

where the “transformation matrix” on the left may be ill-conditioned in spite of representing a stable reduction and stable back solve. The corresponding condition numbers of the problems from Table 1 are given as follows:

<table>
<thead>
<tr>
<th>cond. (Std.S)</th>
<th>cond. (A.Std.S)</th>
<th>cond. (Stb.R)</th>
<th>cond. (transform. matrix)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5e4 (2.8e6/2400)</td>
<td>1.5e4 (2.8e6/2400)</td>
<td>4.2e12 (6.7e14/7.5e11)</td>
<td>1.1e11 (1.7e11/6.0e10)</td>
</tr>
</tbody>
</table>

Above, the condition number of (Stb.R) refers to the 3×3 block system not just to the 2×2 block system of (11) listed in Table 1. As expected, (and as can be proved in a straightforward manner) introducing the artificial variable $u$ to (Std.S) (leading to (A.Std.S)) does not have significant influence on the condition numbers, but the transformation matrix (leading to (Stb.R)) is very poorly conditioned, in spite of the fact that the reduction to (11) and the back solve (12) given the solution of (11) both are stable. In particular, the second row of the transformation matrix contains very large elements.

Multiplying the second row of the transformation matrix (containing the large elements) with the right hand side of (A.Std.S) yields,

$$-X^{-1} r + SX^{-1} P_2 S^{-1} r + q = q - P_1 X^{-1} r,$$
i.e. a vector of small norm. The definition of the artificial variable \( u \) thus implies a special structure of the right hand side of \((\mathbf{A.\text{Std}.S})\), due to which the large components (namely \( X^{-1}r_2 \)) cancel, resulting in a stable reduction in spite of an ill-conditioned transformation matrix.

The above factorization may seem arbitrary as \((\text{Std.S})\) was augmented by an artificial variable \( u \). Instead of augmenting \((\text{Std.S})\) one may also eliminate \( u \) first from \((\text{Stb.R})\). The \( 3 \times 3 \) block system obtained then, however, does suffer from a systematic cancellation error in the right hand side – and thus does not contradict the intuition that an ill-conditioned transformation matrix would be associated with systematic cancellation.

In Table 2, we list the same problems as in Table 1, comparing the condition numbers of the standard system (as a reference), the scaled reduction, and of the optimal basis.

Table 2, Condition numbers for small nondegenerate random LPs (new systems):

<table>
<thead>
<tr>
<th>( \mu )</th>
<th>cond. ((\text{Std.S}))</th>
<th>cond. ((\text{Scl.R}))</th>
<th>cond. opt. basis</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>170 (230/120)</td>
<td>16 (20/12)</td>
<td>300 (4.5e4/60)</td>
</tr>
<tr>
<td>1.0e-4</td>
<td>1.4e4 (3.9e5/2400)</td>
<td>360 (2.2e4/63)</td>
<td>300 (4.5e4/60)</td>
</tr>
<tr>
<td>1.0e-8</td>
<td>1.5e4 (2.8e5/2400)</td>
<td>310 (4.6e4/61)</td>
<td>300 (4.5e4/60)</td>
</tr>
</tbody>
</table>

It appears that for small \( \mu > 0 \), the condition number of \((\text{Scl.R})\) is close to the one of the optimal basis. This observation can in fact be made precise:

**Theorem 1**

Assume that \((1)\) has a unique (primal-dual) optimal solution \( \bar{x}, \bar{y}, \bar{s} \) and let \( \bar{\kappa} \) be the 2-norm-condition number of the optimal basis of \((1)\). Assume further that the 2-norm of the optimal basis matrix and of its inverse are both at least 1.

For \( \mu > 0 \) let \( x_\mu, y_\mu, s_\mu \) be any (not necessarily feasible) approximate solution of \((1)\) with \( \| (x_\mu, y_\mu, s_\mu) - (\bar{x}, \bar{y}, \bar{s}) \|_2 \leq \mu \). Let \( \kappa(\mu) \) be the condition number of system \((\text{Scl.R})\) evaluated at \( x_\mu, s_\mu \). Then, \( \lim_{\mu \to 0} \kappa(\mu) = \bar{\kappa} \).

Moreover, the right hand side in \((14)\) is uniformly bounded for sufficiently small \( \mu \geq 0 \).

**Remark:**

Theorem 1 is valid also for points at the boundary of the primal-dual feasible set, where the condition number of the interior-point barrier systems is unbounded.

The assumption that the 2-norm of the optimal basis matrix and of its inverse are both at least 1 is not essential; if it is violated the result is still true with minor technical modifications.

**Proof:** Let \( \bar{x}, \bar{y}, \bar{s} \) be the (primal-dual) optimal solution of \((1)\). By uniqueness of the optimal solution, it follows \( x_\mu \to \bar{x} \) and \( s_\mu \to \bar{s} \) for \( \mu \to 0 \). Moreover, \( \bar{x} \) has exactly \( m \) nonzero components and \( \bar{s} \) is strictly complementary to \( \bar{x} \). For the system matrix in \((14)\), this implies

\[
\begin{bmatrix}
0 & AD_x \\
D_x A^T & D_s
\end{bmatrix} \to \begin{bmatrix}
0 & A_1 & 0 \\
A_1^T & 0 & 0 \\
0 & 0 & I
\end{bmatrix} = \begin{bmatrix}
0 & A_1^T \\
A_1^{-1} & 0 & 0 \\
0 & 0 & I
\end{bmatrix}^{-1}
\]

where \( A_1 \) is the optimal basis matrix of \((1)\). Using the definition

\[
\| A_1 \|_2 = \max_{x \neq 0} \frac{\| A_1 x \|_2}{\| x \|_2} = \sqrt{\lambda_{\max} A_1 A_1^T} = \sqrt{\lambda_{\max} A_1^T A_1} = \| A_1^T \|_2
\]
the claim follows when observing that
\[
\begin{pmatrix}
0 & A_1 & 0 \\
A_1^T & 0 & 0 \\
0 & 0 & I
\end{pmatrix}
\begin{pmatrix}
0 & A_1 & 0 \\
A_1^T & 0 & 0 \\
0 & 0 & I
\end{pmatrix}^T
= \begin{pmatrix}
A_1A_1^T & 0 & 0 \\
0 & A_1^TA_1 & 0 \\
0 & 0 & I
\end{pmatrix}
\]
and likewise for the inverse. Boundedness of the right hand side in (14) directly follows from the definition of \( r \) in (3).

Remark:
In general, without using any form of preconditioning, we cannot expect much better condition numbers for any system leading to the optimal solution of (1) – if there was a general approach with smaller condition numbers, this would allow stabilizing the solution of any (nonsymmetric) linear system.

The experiments of Table 1 and Table 2 were repeated for random degenerate LP-problems where 20% of the primal basic variables were zero, and likewise for the dual variables. Again, the condition numbers were evaluated near the central path.

Table 3, Condition numbers for small degenerate random LPs:

<table>
<thead>
<tr>
<th>( \mu )</th>
<th>\text{cond. (Std.S)}</th>
<th>\text{cond. (Stb.R)}</th>
<th>\text{cond. (Nrm.E)}</th>
<th>\text{cond. (Scl.R)}</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>160 (230/120)</td>
<td>1.0e3 (1.5e3/710)</td>
<td>11 (13/7.2)</td>
<td>16 (20/12)</td>
</tr>
<tr>
<td>1.0e-4</td>
<td>1.3e4 (1.9e4/7.2e3)</td>
<td>6.5e7 (1.4e8/2.9e7)</td>
<td>5.4e6 (1.3e7/2.5e6)</td>
<td>99 (220/48)</td>
</tr>
<tr>
<td>1.0e-8</td>
<td>1.3e6 (1.9e6/7.2e5)</td>
<td>6.5e11 (1.4e12/2.9e11)</td>
<td>5.4e9 (1.3e10/2.5e9)</td>
<td>99 (220/48)</td>
</tr>
</tbody>
</table>

For small numbers of \( \mu \), here, too, the scaled reduction compares very favorably to the standard system or the normal equations.

Since the Stable Reduction is highly ill-conditioned even for such simple examples, and the Normal Equations also suffer from a systematic cancellation error in the right hand side (not listed in Table 1 and 2), the Scaled Reduction will be pursued further in the context of semidefinite programs in the next section.

In Table 4, we briefly summarize Points 1. - 5. for the above systems. A “+”-sign in Table 4 indicates that the point is satisfied, “0” indicates that it is partially satisfied, a “-”-sign indicates that this point is not satisfied, and two “-”-signs indicate that this point is gravely violated.

Table 4, Linear Systems for LP:

<table>
<thead>
<tr>
<th>Point</th>
<th>(cancellation, rhs)</th>
<th>(condition numbers)</th>
<th>(sparsity structure)</th>
<th>(symmetry)</th>
<th>(positive definiteness)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>+</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>0</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>–</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>–</td>
<td>–</td>
<td>+</td>
<td>–</td>
<td></td>
</tr>
</tbody>
</table>

We stress that the judgment “0” in Point 2. based on Tables 1 and 2 is unreliable; for real world large scale LPs or far from the central path, the system (\text{Std.S}) needs more extensive testing...
regarding Point 2. The numerical experiments in Section 4. confirm that the scaled reduction is fairly efficient for larger values of \( \mu \), say \( \mu > 10^{-4} \) in a normalized framework, but it is not well suited for small values of \( \mu \).

3. Generalization to the NT direction

In this section we consider semidefinite programs in the standard form

\[
\text{minimize } C \cdot X \quad \text{subject to } A(X) = b \quad \text{and } \quad X \succeq 0,
\]

where \( S^n \) is the space of symmetric \( n \times n \)-matrices, \( A : S^n \to \mathbb{R}^m \) is a linear map, \( b \in \mathbb{R}^m \), and \( C \in S^n \) are given data. The trace inner product is denoted by \( C \cdot X = \text{tr}(CX) \) and \( A \) and its adjoint are represented as

\[
A(X) = \begin{bmatrix}
A^{(1)} \cdot X \\
\vdots \\
A^{(m)} \cdot X
\end{bmatrix}, \quad A^*(y) = \sum_{i=1}^m y_i A^{(i)}.
\]

3.1 Derivation of the “Standard System”

Let \( S_P \) denote the symmetrization operator introduced by Monteiro and Zhang, \cite{14},

\[
S_P(U) := \frac{1}{2}(PUP^{-1} + (PUP^{-1})^T)
\]

with some nonsingular matrix \( P \). In generalization of (2), the central path is then defined by \( X, S > 0, y \in \mathbb{R}^m \) with

\[
A(X) = b, \\
A^*(y) + S = C,
\]

\[
S_P(XS) = \mu I.
\]

Let \( X, S > 0, y \in \mathbb{R}^m \) be given, and set

\[
p := b - A(X), \quad Q := C - A^*(y) - S, \quad \text{and} \quad R := \mu^+ I - S_P(XS).
\]

As in (3), we assume that \( \mu := X \cdot S/n \), and \( \mu^+ \) is some “new target value”; \( \mu^+ = \sigma \mu \) with \( \sigma \in (0,1) \). Newton’s method for solving (17) then yields the system of linear equations

\[
\begin{bmatrix}
0 & \mathcal{A} & 0 \\
\mathcal{A}^* & 0 & I \\
0 & S_P(X) & S_P(XS)
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\Delta X \\
\Delta S
\end{bmatrix} =
\begin{bmatrix}
p \\
Q \\
R
\end{bmatrix}.
\]

Here, the last row stands for the equation

\[
S_P(\Delta XS) + S_P(X \Delta S) = R.
\]
System (19) corresponds to the “Standard System” (4) in Section 2. This system has over \(n^2\) unknowns. In some cases, e.g. for the SDP relaxation of the max-cut-problem [10], system (19) can be solved with order \(n^3\) operations. Typically, however, the computational effort for solving (19) is much higher, up to order \(n^6\) operations for some instances. In this paper we target such more involved instances and consider \(O(n^3)\) operations as comparatively cheap.

In the following we consider the NT-direction [15] where \(P\) is defined as follows. Let

\[
W := S^{-1/2}(S^{1/2}X^{1/2})^{1/2}S^{-1/2}
\]

be the metric geometric mean of \(X\) and \(S^{-1}\). It is easy to see that this implies

\[
W SW = X, \quad S = W^{-1}XW^{-1},
\]

\[
W^{1/2}SW^{1/2} = W^{-1/2}XW^{-1/2} =: E \quad (= E(X, S)),
\]

and when \(XS \approx \mu I\), we have

\[
E \approx \sqrt{\mu I}.
\]

Then set \(P := W^{-1/2}\).

If \(X, S\) are diagonal matrices, then \(P = S^{1/4}X^{-1/4}\), and if \(X, S\) are near the central path of a nondegenerate SDP, the eigenvalues of \(P\) range from \(O(\sqrt{\mu})\) to \(O(\sqrt{1/\mu})\). Defining scaled corrections

\[
\widetilde{\Delta}X := P\Delta XP, \quad \widetilde{\Delta}S := P^{-1}\Delta SP^{-1}
\]

it follows that

\[
\mathbf{S}_P(\Delta XS) = \frac{1}{2}(P\Delta XS P^{-1} + (P\Delta XS P^{-1})^T)
\]

\[
= \frac{1}{2}((\widetilde{\Delta}X P^{-1}SP^{-1} + (\widetilde{\Delta}X P^{-1}SP^{-1})^T)
\]

\[
= \frac{1}{2}((\widetilde{\Delta}XE + E\widetilde{\Delta}X)
\]

(21)

and likewise

\[
\mathbf{S}_P(X\Delta S) = \frac{1}{2}(\widetilde{\Delta}SE + E\widetilde{\Delta}S). \quad (22)
\]

In the following, the “scaling” operator \(\mathcal{D}_P : S_+ \to S_+\) and the “Lyapunov operator” \(\mathcal{L}_E : S_+ \to S_+\) defined via

\[
\mathcal{D}_P(\Delta X) := P\Delta XP \quad \text{and} \quad \mathcal{L}_E(\Delta X) := \frac{1}{2}(E\Delta X + \DeltaXE)
\]

will be used. Both operators, \(\mathcal{D}_P\) and \(\mathcal{L}_E\) are self-adjoint, and the inverse operations \((\mathcal{D}_P)^{-1} = \mathcal{D}_{P^{-1}}\) and \((\mathcal{L}_E)^{-1}\) can be carried out with \(O(n^3)\) arithmetic operations each (assuming that a numerical eigenvalue decomposition of the symmetric matrix \(E\) is computable with \(O(n^3)\) operations). Since \(\mathcal{D}_P\) takes over the role of the diagonal scaling in Section 2 and since it is easy to invert, we denote it by the letter \(\mathcal{D}\) thinking of it as some form of a “diagonal operator”. Evidently,

\[
\widetilde{\Delta}X = \mathcal{D}_P(\Delta X) \quad \text{and} \quad \widetilde{\Delta}S = \mathcal{D}_{P^{-1}}(\Delta S).
\]

Relations (21), (22) can now be written as

\[
\mathbf{S}_P(\Delta XS) = \mathcal{L}_E(\widetilde{\Delta}X) = \mathcal{L}_E\mathcal{D}_P(\Delta X), \quad \mathbf{S}_P(X\Delta S) = \mathcal{L}_E(\widetilde{\Delta}S) = \mathcal{L}_E\mathcal{D}_{P^{-1}}(\Delta S).
\]
Using this, system (19) can then be written as
\[
\begin{bmatrix}
0 & A & 0 \\
A^* & 0 & I \\
0 & \mathcal{L}_E \mathcal{D}_P & \mathcal{L}_E \mathcal{D}_{P-1}
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\Delta X \\
\Delta S
\end{bmatrix}
= \begin{bmatrix}
p \\
Q \\
R
\end{bmatrix}.
\tag{23}
\]

or
\[
\begin{bmatrix}
0 & A & 0 \\
A^* & 0 & I \\
0 & \sqrt{\mu} \mathcal{D}_P & \sqrt{\mu} \mathcal{D}_{P-1}
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\Delta X \\
\Delta S
\end{bmatrix}
= \begin{bmatrix}
p \\
Q \\
\sqrt{\mu} (L_E)^{-1} R
\end{bmatrix}.
\tag{24}
\]

Note that \( E \approx \sqrt{\mu} I \), so that the last row of (24) does not suffer from systematic cancellation. For nondegenerate SDP and near the central path, the matrices \( \sqrt{\mu} P \) and \( \sqrt{\mu} P^{-1} \) defining the scalings in the last row of (24) both have eigenvalues ranging from \( O(4) \) for nondegenerate SDP and near the central path, the matrices \( \sqrt{\mu} P \) and \( \sqrt{\mu} P^{-1} \) defining the scalings in the last row of (24) both have eigenvalues ranging from \( O(4) \) to \( O(1) \).

Let \( P = U\Lambda U^T \) be the eigenvalue decomposition of \( P \), and \( \mathcal{U} : \mathcal{S}^n \to \mathcal{S}^n \) be defined by \( \mathcal{U}(Z) = UZU^T \) with adjoint \( \mathcal{U}^*(Y) = U^{-1}(Y) = U^TYU \). Multiplying the second and third block column of (24) by \( \mathcal{U} \) and the second and third block row by \( \mathcal{U}^* \), it follows that (24) is equivalent to
\[
\begin{bmatrix}
0 & \mathcal{A} \mathcal{U} & 0 \\
\mathcal{A}^* \mathcal{U} & 0 & I \\
0 & \sqrt{\mu} \mathcal{D}_\Lambda & \sqrt{\mu} \mathcal{D}_{\Lambda^{-1}}
\end{bmatrix}
\begin{bmatrix}
\Delta y \\
\mathcal{U}^* \Delta X \\
\mathcal{U}^* \Delta S
\end{bmatrix}
= \begin{bmatrix}
p \\
\mathcal{U}^* Q \\
\mathcal{U}^* \sqrt{\mu} (L_E)^{-1} R
\end{bmatrix}.
\tag{25}
\]

Here, \( \mathcal{D}_\Lambda, \mathcal{D}_{\Lambda^{-1}} \) are “true” diagonal scalings in the sense that they are represented by a diagonal matrix when
\[
\hat{\Delta} X := \mathcal{U}^* \Delta X \quad \text{and} \quad \hat{\Delta} S := \mathcal{U}^* \Delta S
\]
are represented as vectors,
\[
\mathcal{D}_\Lambda(M) = \Lambda M \Lambda = M \circ (\text{diag}(\Lambda)(\text{diag}(\Lambda))^T)
\]
where \( \circ \) denotes the Hadamard (componentwise) product. Thus, system (25) has identical structure to system (4), (\textbf{Std.S}), with the exception that the partition of the diagonal operators into small and large components is more complicated, and \( \mathcal{A} \mathcal{U} \) typically is no longer sparse, even when \( \mathcal{A} \) is so.

We stress that the matrix representing \( \mathcal{A} \mathcal{U} \) or its adjoint will never be formed explicitly, iterative schemes as considered below only use that fact that multiplications with \( \mathcal{A} \mathcal{U} \) and its adjoint can be carried out. The reformulation (25) has been considered earlier, for example in [20]; the considerations below rely on (25) and aim at generalizing the system (\textbf{Scl.R}) to semidefinite programs.

We now turn to an observation exploited already in [18]: For many large scale semidefinite programs with sparse constraints \( \mathcal{A} \), the Cholesky factorization \( LL^T = \mathcal{A} \mathcal{A}^* \) (with suitable reordering) is sparse and can be computed cheaply, while for any commonly used symmetrization operator \( \mathcal{S}_P \) the system of normal equations never is sparse, and thus a direct factorization of (19) is out of reach.

In the sequel we assume that a Cholesky factor \( L \) of \( \mathcal{A} \mathcal{A}^* \) is available. (If \( L \) is not available we may choose instead any approximation of the Cholesky factor of \( \mathcal{A} \mathcal{A}^* \) or of \( \mathcal{A}(\mathcal{S}_P(\mathcal{S}_P(X \mathcal{A}^* . )S)^{-1}) \), or simply \( L = I \) and modify the subsequent derivations accordingly.)

Then, \( L^{-1} \mathcal{A} \) (and also \( L^{-1} \mathcal{A} \mathcal{U} \)) is part of an orthogonal matrix, and again, the matrix \( L^{-1} \mathcal{A} \mathcal{U} \) (and its transpose) will not be formed explicitly – as this matrix typically would be dense, only multiplications with \( \mathcal{U} \) and \( \mathcal{A} \) and a back solve with \( L \) will be needed at each iteration.
3.2 Derivation of the “Scaled Reduction”

Next, we aim at extending the partition \(x_1,x_2\) that was used for linear programs and that led to \((\text{Scl.R})\). Let \(D_M\) be the matrix with entries

\[
(D_M)_{i,j} = \frac{1}{\sqrt{\mu}} \min \left\{ (\lambda_i \lambda_j)^{1/2}, (\lambda_i \lambda_j)^{-1/2} \right\}
\]

and define the operator \(D_M\) via \(D_M(Z) = D_M \circ Z\). Likewise, define \(D_x\) and \(D_s\) as the matrices with entries

\[
(D_x)_{i,j} = \begin{cases} 1 & \text{if } \lambda_i \lambda_j \leq 1, \\ \lambda_i^{-1} \lambda_j^{-1} & \text{else}, \end{cases} \quad (D_s)_{i,j} = \begin{cases} \lambda_i^2 \lambda_j^2 & \text{if } \lambda_i \lambda_j \leq 1, \\ 1 & \text{else}, \end{cases}
\]

and the operators \(D_x,D_s\) via \(D_x(Z) = D_x \circ Z\) and \(D_s(Z) = D_s \circ Z\).

Analogously to (14) the scaled reduction then takes the form

\[
\begin{bmatrix} 0 & L^{-1} \mathcal{A} \mathcal{U} D_x \\ D_s \mathcal{U}^* A^* \Lambda^{-1/2} \mathcal{U}^* & D_s \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta \mathcal{X} \end{bmatrix} = \begin{bmatrix} L^{-1} \mathcal{A} R_2 - L^{-1} p \\ D_s \mathcal{U}^* Q - \mathcal{D}_M \mathcal{U}^* R_1 \end{bmatrix},
\]

(27)

where \(R_1\) and \(R_2\) are any two symmetric \(n \times n\)-matrices such that

\[
\sqrt{\mu} D_{\Lambda} \mathcal{U}^* R_2 + \sqrt{\mu} D_{\Lambda^{-1/2}} \mathcal{U}^* R_1 = \mathcal{U}^* \sqrt{\mu} (\mathcal{L}_E)^{-1} R.
\]

(28)

Given the solution of this system, we can retrieve

\[
\Delta y = L^{-T} \Delta y, \quad \Delta \mathcal{X} = \mathcal{U}^* R_2 - D_x \Delta \mathcal{Z}, \quad \text{and} \quad \Delta \mathcal{S} = \mathcal{U}^* Q - \mathcal{U}^* A^* \Delta y.
\]

(29)

Proposition 3.1 The solution of (27), (28), (29) also solves (24) and (25), (26).

Proof: Observe that the last relation of (29) coincides with the second block row of (25). Using the second relation in (29), and premultiplying the first block row of (27) with \(L\) we obtain

\[
\mathcal{A} \mathcal{U} D_x (D_x^{-1} (\mathcal{U}^* R_2 - \Delta \mathcal{X}))) = \mathcal{A} R_2 - p
\]

which in turn is equivalent to the first block row of (25). Finally, premultiplying the second row of (27) with \(D_x^{-1}\) and using (29) one obtains

\[
\mathcal{U}^* A^* \Delta y + D_s D_x^{-2} (\mathcal{U}^* R_2 - \Delta \mathcal{X}) = \mathcal{U}^* Q + D_s^2 \mathcal{U}^* R_2 - \mathcal{D}_M \mathcal{U}^* R_1.
\]

Observing that \(D_x^{-1} D_M = \frac{1}{\sqrt{\mu}} \mathcal{D}_\Lambda^{1/2}\) and using (28) to eliminate \(\mathcal{U}^* R_1\), this equation is equivalent to

\[
\mathcal{U}^* A^* \Delta y + D_s D_x^{-2} (\mathcal{U}^* R_2 - \Delta \mathcal{X}) = \mathcal{U}^* Q + D_s^2 \mathcal{U}^* R_2 - \mathcal{D}_M \mathcal{U}^* (\mathcal{L}_E)^{-1} R.
\]

Using the relation \(D_s D_x^{-2} = \mathcal{D}_\Lambda^2\) and the last relation in (29), one arrives at

\[
-\mathcal{D}_\Lambda^2 \Delta \mathcal{X} = \Delta \mathcal{S} - \mathcal{D}_M \mathcal{U}^* (\mathcal{L}_E)^{-1} R
\]

which (when multiplied from left by \(\sqrt{\mu} \mathcal{D}_\Lambda^{-1}\)) is equivalent to the last relation of (25). #

Block Pivoting: We recall that pivoting with tiny diagonal elements such as in (5) or (6) results
in an increase of the norm of the right hand side. Next, we aim at limiting the increase of the right hand side of (27) by using a $2 \times 2$ block pivot rule that exploits the freedom given in (28). More precisely, as $L^{-1}A$ is part of an orthogonal matrix, we aim at choosing $R_2, R_1$ by solving

$$\minimize_{R_1, R_2} \| R_2 \|_F^2 + \| D_M R_1 \|_F^2 \quad \text{s.t.} \quad \sqrt{\mu} D_\Lambda R_2 + \sqrt{\mu} D_{\Lambda^{-1/2}} R_1 = U^* \tilde{R}$$

(30)

where $\tilde{R}_1 \equiv U^* R_1$, $\tilde{R}_2 \equiv U^* R_2$, and $\tilde{R} := \sqrt{\mu}(L_E)^{-1} R$. This is a problem separable into $2 \times 2$ subproblems; the $i, j$-th entries of $\tilde{R}_1, \tilde{R}_2$ are given by

$$(\tilde{R}_2)_{i,j} = (U^* \tilde{R})_{i,j} \frac{\lambda_i \lambda_j \min\{\pm \lambda_i \lambda_j\}}{\mu^{1/2}(\lambda_i^2 \lambda_j^2 \min\{\pm \lambda_i \lambda_j\} + \lambda_i^{-1} \lambda_j^{-1})},$$

$$(\tilde{R}_1)_{i,j} = (U^* \tilde{R})_{i,j} \frac{\mu^{1/4} \lambda_i^{-1/2} \lambda_j^{-1/2}}{\lambda_i^2 \lambda_j^2 \min\{\pm \lambda_i \lambda_j\} + \lambda_i^{-1} \lambda_j^{-1}}.$$  

(31)

**Discussion:** The scaled reduction $(\text{Scl.R})$ of Section 2. corresponds to a feasible solution of (30). As indicated in the note after relation (12), the optimal solution (31) is a slight improvement of this partition when considering linear programs. For semidefinite programs, the situation is slightly different; the diagonal entries of the diagonal scaling operator behave as $\lambda_i^2 \rightarrow 0$ for some $i$, or $\lambda_j^2 \rightarrow \infty$ for some $j$, or as $\lambda_i \lambda_j \rightarrow \text{const}$ for some $i, j$. Thus, the partition into complementary small and large components is no longer applicable. The solution (31) is a natural generalization of this partition. However, when $\lambda_i \lambda_j \approx 1$, the components of $\tilde{R}_1, \tilde{R}_2$ in (31) are larger by a factor of almost $\mu^{-1/2}$ than the corresponding components of $U^* \tilde{R}$. For such cases, the pivoting for arriving at (27) does introduce a certain amount of systematic cancellation, though not as extreme as for (5) (6).

The numerical experiments in Section 4. indicate that the norm of the right hand side is reduced by several orders of magnitude with this form of pivoting, and thus, the accuracy needed for computing approximate solutions of the linear systems in an interior-point algorithm must be much higher without using (31).

In closing we note that $L$ is the Cholesky factor of $AD_*^2 A^*$ and $D_* x$ has exactly $m$ large components while the remaining ones are tiny, then the condition number of the matrix in (27) is close to one. It remains a challenging open problem to define a sparse approximation to such a Cholesky factor.

**Remark, AHO direction:** The final iteration of a numerical implementation might be based on the AHO-direction: To this end, the eigenvalue decomposition of the matrix $X - S$ can be chosen as a common eigenbasis for $X$ and $S$, i.e. $X, S$ are projected onto this eigenbasis before computing the AHO step. (The projection of $X$ onto some given unitary eigenbasis $U$ is simply given by $U \text{Diag(diag}(U^T X U))U^T$ where $\text{Diag}(M)$ denotes the vector of diagonal entries of $M$ and $\text{Diag}(x)$ is the diagonal matrix with diagonal entries from $x$.) Due to the common eigenbasis, the AHO step can be written in a symmetric form resembling (27).

More precisely, when $X$ and $S$ have a common eigenbasis, then this eigenbasis also is an eigenbasis of the matrix $P$ (which is not needed for the AHO direction but which illustrates the similarities to the NT direction.) Let the eigenvalue decompositions of $X$ and $S$ and the scaling matrices associated with the diagonal matrices $\Gamma$ and $\Sigma$ be given by

$$X = U \Gamma U^T, \quad S = U \Sigma U^T \quad \text{and} \quad (D_\Gamma)_{i,j} := \gamma_i \gamma_j, \quad (D_\Sigma)_{i,j} := \sigma_i \sigma_j.$$
Further, let $D_M$ be the matrix with entries

$$(D_M)_{i,j} = \frac{1}{\sqrt{\mu}} \min \left\{ \left( \gamma_i \gamma_j \right)^{-1/2}, \left( \sigma_i \sigma_j \right)^{-1/2} \right\}$$

and define $D_x := D_M \circ D_{F^{1/2}}$ and $D_s := D_M \circ D_{\Sigma}$. Then, system (27) remains nearly the same as the one of the NT direction.

Note that the matrix $D_M$ is not defined for infeasible points close to a nondegenerate optimal solution: If $\gamma_i \leq 0$ for some $i$ and $\sigma_j \leq 0$ for some other $j$, then both terms in the right hand side of (32) are undefined.

This is in contrast to Theorem 1 for linear programs, and in contrast to the properties of the AHO direction which does have a stable extension sufficiently close to the optimal solution.

In the final stage of the algorithm the outer iterations based on the AHO direction may converge faster than iterations based on the NT direction, see e.g. [1]. However, the AHO approach presented here requires a projection onto a common eigenbasis which may deteriorate convergence. Moreover, the critical point is the convergence of the (inner) QMR iterations. As the structure of the linear system of AHO is identical to (27) there is no reason to assume that the symmetric QMR algorithm will perform very differently for the AHO direction.

**Remark, numerical stability of small to medium size problems:** In closing this section we briefly point out a related approach that may be suitable for controlling rounding errors of small to medium size problems. We assume that initially, a Cholesky factor $L$ of the operator $AA^*$ is computed (for example for some form of preprocessing as in [11] or for a central starting point) and that $L$ is stored for future use. At some later iteration we consider the standard system (19) and split the unknown corrections $\Delta X, \Delta y, \Delta S$ as

$$\Delta X = dX^{(1)} + dX^{(2)}, \quad \Delta y = dy^{(1)} + dy^{(2)}, \quad \Delta S = dS^{(1)} + dS^{(2)},$$

where we fix

$$dX^{(1)} := A^* L^{-T} L^{-1} p, \quad dy^{(1)} := L^{-T} L^{-1} A Q, \quad dS^{(1)} := Q - A^* dy^{(1)}.$$ 

This split allows to transform a standard system (19) to an equivalent system for the unknowns $dX^{(2)}, dy^{(2)}, dS^{(2)}$ with right hand side

$$p^{(2)} = 0, \quad Q^{(2)} = 0, \quad R^{(2)} = R - S_P(dX^{(1)} S) - S_P(X dS^{(1)}).$$

By definition of $L$ it follows that

$$\|dX^{(1)}\|_F \leq \|\Delta X\|_F, \quad \|dS^{(1)}\|_F \leq \|\Delta S\|_F,$$

so that there is no systematic cancellation error with this transformation. We then consider the equivalent formulation (24) of (19) and eliminate $dS^{(2)}$ using the last block row. Rescaling with $\sqrt{\mu D_P^{-1}}$ leads to

$$\left[ \begin{array}{c} \frac{0}{\sqrt{\mu D_P^{-1}}} \\ \sqrt{\mu A D_P^{-1}} \frac{1}{I} \\ 0 \end{array} \right] \left[ \begin{array}{c} \frac{1}{\sqrt{\mu}} dy^{(2)} \\ \frac{1}{\sqrt{\mu}} dZ^{(2)} \end{array} \right] = \left[ \begin{array}{c} 0 \\ -L E^{-1} R^{(2)} \end{array} \right],$$

where $dZ^{(2)} = -D_P dX^{(2)}$. Here, as well, there is no cancellation due to the fact that $p^{(2)} = 0$ and $Q^{(2)} = 0$, however, even for the case of solving LPs, the system matrix on the left side may have an
unbounded condition number as $X, S$ approach optimality, and thus this system is less suitable for iterative approaches. When a Cholesky factor $\hat{L}$ of $AD^{-2}_P A^*$ is given, this system is equivalent to

$$
\begin{bmatrix}
0 \\
D^{-1}_P A^* \hat{L}^{-T} \\
\hat{L}^{-1} A D^{-1}_P \\
\end{bmatrix}
\begin{bmatrix}
\hat{d}_y^{(2)} \\
dZ^{(2)} \\
\end{bmatrix}
= \begin{bmatrix}
0 \\
-I \\
\end{bmatrix} \hat{L}^{-1} R^{(2)}
$$

(33)

with $\hat{d}_y^{(2)} = \hat{L}^T dy^{(2)}$. The system matrix of (33) has eigenvalues equal to 1 and $(1 \pm \sqrt{5})/2$. (Indeed, the columns $q^i$ of $D^{-1}_P A^* \hat{L}^{-T}$ are orthonormal, and the eigenvectors of the system matrix are $[0; v]$ where $v$ is in the null space of $\hat{L}^{-1} A D^{-1}_P$ and $[(-1 \pm \sqrt{5})/2 e^i; q^i]$ where $e^i$ is the $i$-th canonical unit vector in $R^m$.) Thus, the symmetric QMR algorithm for (33) will terminate after 3 steps — and without systematic cancellation errors. When $\hat{L}$ is an approximate Cholesky factor, system (33) can still be solved by the symmetric QMR algorithm and the back substitution $dy^{(2)} = \hat{L}^{-T} \hat{d}_y^{(2)}$ does not depend on how close the approximation $\hat{L} \hat{L}^T \approx AD^{-2}_P A^*$ is satisfied. System (33) thus offers a general principle to reduce cancellation errors in interior point methods that can also be applied in the final iterations of a standard SDP solver such as SDPT3 [21] or SEDUMI [19]. An illustration of this effect is addressed in Table 9 below.

4. Numerical examples for semidefinite programs

Compared to interior-point methods with infeasible starting points, self-dual interior-point approaches require one additional back solve at each iteration. When using iterative approaches for solving the linear systems, the extra back solve is numerically expensive. We therefore implemented an infeasible starting point primal-dual predictor-corrector interior-point algorithm. To reduce the effect of ill-conditioning, the step lengths and centering parameters in the algorithm were chosen such that the iterates remained in a rather small neighborhood of the infeasible central path. For many SDP problems a suitable permutation of the system of linear equations yields a sparse Cholesky factor of the normal equations. Based on this special situation, the violation of the primal-dual equality constraints can be reduced even when the search directions are computed with low accuracy.

Below we report the results of some simple numerical examples. The experiments indicate that in spite of moderate accuracy of the search directions, the algorithm works with a rather small number of outer iterations and retains iterates fairly close to the central path.

First, we list the results of the algorithm for a small dense random SDP problem with $X, S$ of dimension $50 \times 50$ and 230 linear constraints. The optimal solution was strictly complementary but not well-conditioned in the following sense: We compared the approximate solution $X^{alg}, y^{alg}, S^{alg}$ generated by the algorithm to the exact optimal solution $X^{opt}, y^{opt}, S^{opt}$ and obtain:

$$
\frac{\|A(X^{alg}) - b\|_2}{\|b\|_2} = 1.4 \cdot 10^{-9}, \quad \frac{X^{alg} \cdot S^{alg}}{\|X^{alg}\|_F \|S^{alg}\|_F} = 2.9 \cdot 10^{-8}, \quad \frac{\|X^{alg} - X^{opt}\|_F}{\|X^{alg}\|_F} = 0.008,
$$

$$
\frac{\|A^* y^{alg} + S^{alg} - C\|_F}{\|C\|_F} = 5.0 \cdot 10^{-13}, \quad \frac{\|S^{alg} - S^{opt}\|_F}{\|S^{alg}\|_F} = 1.1 \cdot 10^{-5}.
$$

The relative errors being significantly larger (by a factor of $10^5$) than the relative residuals, indicates that the optimal solution is not well conditioned.
The relative accuracy required in the predictor step and in the corrector step were both $10^{-1}$, and QMR was interrupted after at most 1000 steps, except for the last iteration, where 2000 steps were set as a limit to allow for a final extrapolation step. In spite of this limit on the accuracy of the interior-point system, it took only 16 interior-point iterations to obtain about 8 digits relative accuracy in the residuals.

In Table 5 we list an iteration log with the number of QMR steps in the corrector (which was always slightly higher than for the predictor), the centrality of the iterate given by $\text{cond}(E(X,S)) - 1$, the step length for the predictor, and the value of $\mu$.

Table 5, Iteration log-file, small dense SDP:

<table>
<thead>
<tr>
<th>iteration</th>
<th>corr. QMR-steps</th>
<th>centrality</th>
<th>step length</th>
<th>$-\log_{10} \mu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>.88</td>
<td>.70</td>
<td>.58</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>2.1</td>
<td>.78</td>
<td>1.3</td>
</tr>
<tr>
<td>5</td>
<td>103</td>
<td>2.6</td>
<td>.74</td>
<td>3.1</td>
</tr>
<tr>
<td>6</td>
<td>121</td>
<td>1.5</td>
<td>.68</td>
<td>3.7</td>
</tr>
<tr>
<td>10</td>
<td>495</td>
<td>3.0</td>
<td>.66</td>
<td>6.1</td>
</tr>
<tr>
<td>11</td>
<td>485</td>
<td>3.0</td>
<td>.63</td>
<td>6.6</td>
</tr>
<tr>
<td>15</td>
<td>1000</td>
<td>3.2</td>
<td>.41</td>
<td>8.5</td>
</tr>
<tr>
<td>16</td>
<td>2000</td>
<td>2.6</td>
<td>.59</td>
<td>9.0</td>
</tr>
</tbody>
</table>

The overall number of QMR Iterations was 15477.

In Table 6, we list the same values for a “comparable” linear program with 2500 unknowns and 230 linear constraints. While the the iteration count is almost twice as high the total number of QMR iterations did not change much.

Table 6, Iteration log-file, small dense LP:

<table>
<thead>
<tr>
<th>iteration</th>
<th>corr. QMR-steps</th>
<th>centrality</th>
<th>step length</th>
<th>$-\log_{10} \mu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>.48</td>
<td>.54</td>
<td>.43</td>
</tr>
<tr>
<td>5</td>
<td>9</td>
<td>4.4</td>
<td>.27</td>
<td>2.4</td>
</tr>
<tr>
<td>10</td>
<td>110</td>
<td>3.3</td>
<td>.60</td>
<td>4.2</td>
</tr>
<tr>
<td>15</td>
<td>125</td>
<td>3.9</td>
<td>.42</td>
<td>5.3</td>
</tr>
<tr>
<td>20</td>
<td>287</td>
<td>2.0</td>
<td>.39</td>
<td>6.5</td>
</tr>
<tr>
<td>25</td>
<td>699</td>
<td>2.1</td>
<td>.34</td>
<td>7.9</td>
</tr>
<tr>
<td>28</td>
<td>1205</td>
<td>1.4</td>
<td>.69</td>
<td>9.0</td>
</tr>
</tbody>
</table>

The overall number of QMR Iterations was 18700.

When defining the right hand side without the “pivoting” (31), the convergence of the algorithm deteriorated as long as the tolerance of the predictor and the corrector step were set to $10^{-1}$ or to $10^{-2}$; a minimum tolerance of about $10^{-3}$ was needed to obtain comparable convergence.

In Table 7 we report the times (on a standard Linux desktop from 2010) and the numbers of iterations for some sparse random semidefinite problems. The matrices representing the equality constraints are of size $n^2 \times m$ with 6 nonzero elements in each row. (In this preliminary implementation, the matrices are represented as vectors of size $n^2$ ignoring the symmetry.) The table further
lists the overall number of interior-point iterations, and of QMR iterations, the total time, and the norms of the final residuals divided by the norms of the right hand sides. The complementarity is measured by the cosine of the angle between the primal and dual solution.

Table 7, Sparse SDP:

<table>
<thead>
<tr>
<th>n</th>
<th>m</th>
<th>ipm it.</th>
<th>QMR it.</th>
<th>time</th>
<th>prim.res.</th>
<th>dual.res.</th>
<th>compl.</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>100</td>
<td>13</td>
<td>1802</td>
<td>0.68</td>
<td>1.7e-16</td>
<td>2.0e-15</td>
<td>1.7e-9</td>
</tr>
<tr>
<td>50</td>
<td>500</td>
<td>13</td>
<td>1538</td>
<td>1.4</td>
<td>1.9e-16</td>
<td>4.8e-16</td>
<td>5.1e-10</td>
</tr>
<tr>
<td>200</td>
<td>5000</td>
<td>13</td>
<td>1688</td>
<td>16</td>
<td>1.9e-16</td>
<td>3.3e-16</td>
<td>1.4e-7</td>
</tr>
<tr>
<td>500</td>
<td>20000</td>
<td>13</td>
<td>1836</td>
<td>130</td>
<td>1.8e-16</td>
<td>2.1e-16</td>
<td>7.9e-8</td>
</tr>
<tr>
<td>2000</td>
<td>100000</td>
<td>13</td>
<td>2328</td>
<td>6800</td>
<td>1.7e-16</td>
<td>1.4e-16</td>
<td>2.5e-7</td>
</tr>
</tbody>
</table>

As can be seen, the total number of QMR iterations was generally much less than for the small SDP example, possibly due to the fact that this class of SDPs tended to be less ill-conditioned in spite of its high dimension.

In Table 8 we report the same features as in Table 7, now, for some doubly nonnegative problems (i.e. problems where the matrix variable is required to be positive semidefinite and entry-wise nonnegative). Doubly nonnegative problems are computationally rather expensive to solve by interior-point methods using exact factorizations for the linear systems, even when the data matrix $A$ is sparse. It turned out that these instances also were hard to solve by the iterative approach of the present paper. The algorithm failed for the standard setting of computing the search directions only up to 10% accuracy used in the previous examples; moreover the number of QMR steps needed to obtain 10% accuracy was very high. The numbers below refer to an accuracy of 0.1% (with an even higher number of QMR steps).

Table 8, Sparse DNN problems:

<table>
<thead>
<tr>
<th>n</th>
<th>m</th>
<th>ipm it.</th>
<th>QMR it.</th>
<th>time</th>
<th>prim.res.</th>
<th>dual.res.</th>
<th>compl.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>20</td>
<td>24</td>
<td>5341</td>
<td>1.6</td>
<td>4.7e-13</td>
<td>5.5e-13</td>
<td>2.0e-6</td>
</tr>
<tr>
<td>30</td>
<td>100</td>
<td>36</td>
<td>54155</td>
<td>21</td>
<td>1.9e-11</td>
<td>1.8e-11</td>
<td>2.3e-6</td>
</tr>
<tr>
<td>100</td>
<td>300</td>
<td>54</td>
<td>225018</td>
<td>470</td>
<td>1.9e-12</td>
<td>1.7e-12</td>
<td>1.0e-4</td>
</tr>
<tr>
<td>300</td>
<td>1000</td>
<td>77</td>
<td>524937</td>
<td>10000</td>
<td>2.2e-16</td>
<td>3.9e-16</td>
<td>1.7e-5</td>
</tr>
</tbody>
</table>

The results of Table 8 indicate the limits of the current implementation; for several interior-point iterations very many QMR steps were needed to achieve the required accuracy. In particular, the solution times reported here are not competitive with SDPNAL [22]. Not only are the implementation techniques of SDPNAL much more efficient, but also a convincing scheme for preconditioning the interior-point linear systems (27) is still missing at this point. Without suitable preconditioner, our algorithm cannot be expected to work well for poorly conditioned problem instances.

In a final example, the accuracy of the search directions generated by the standard Schur-complement-Cholesky approach was compared with the accuracy of the search directions generated via QMR applied to (33). To this end 100 random instances of SDPs of size $50 \times 50$ were generated and the optimal solution was perturbed by a random positive semidefinite matrix of 2-norm $10^{-9}$. For the search directions computed at the perturbed point the primal and dual backwards error
was computed. The geometric mean of these errors is listed in Table 9. Almost identical numbers were obtained when perturbing a degenerate random optimal solution by a small positive definite matrix – for such degenerate instances the linearization error will be large and the quality of the search direction will deteriorate; here, only the backward error of the linear system is recorded.

Table 9, Accuracy of the search directions, QMR vs. Cholesky:

<table>
<thead>
<tr>
<th>QMR, prim. res. norm</th>
<th>QMR, dual res. norm</th>
<th>Chol, prim. res. norm</th>
<th>Chol, dual res. norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.6e-14</td>
<td>1.8e-13</td>
<td>1.1e-06</td>
<td>5.8e-10</td>
</tr>
</tbody>
</table>

In some implementations, the accuracy of the search direction generated by the Cholesky approach is improved by applying a preconditioned conjugate gradient scheme to the system with the preconditioner given by the Cholesky factor. In the above examples, such a preconditioner was tested as well, but it did not result in any significant improvement of the accuracy compared to the search direction obtained by the Cholesky approach; the corresponding results are omitted. Also, there are several ways to arrange the back substitution once the correction $\Delta y$ is given. One could have chosen a back substitution where the dual residual is zero, but that would have resulted in a further increase of either the primal residual or of the complementary residual. For shortness just one version is recorded here where the dual residual is not zero, but the complementarity residual is zero. The improvement of the accuracy suggested by the figures in this table could only be observed close to the optimal solution; when $S_P(XS) \succeq 10^{-5}I$ the results obtained by the Cholesky factor are comparable to the results obtained by QMR.

5. Concluding remarks

A new stable reformulation and symmetrization of linear interior-point systems for linear programs was derived. Based on a novel pivoting strategy for minimizing the norm of the right hand side, the symmetrization was extended to the NT direction for solving semidefinite programs.

Preliminary numerical results show that the approach of the present paper is able to handle large problem sizes. As the results for some doubly nonnegative problems indicate, the approach is limited, however, to well conditioned problems at this point.

Low accuracy solutions of the interior-point linear systems not only occur with iterative solvers but also with direct solvers applied to poorly conditioned systems. For such ill-conditioned problems, the (inexact) factorization of the normal equations may be used as a preconditioner applied to conjugate gradient iterations for the normal equations. This is the strategy currently implemented with SEDUMI. The same (inexact) factorization can also be lifted and used as a preconditioner for a modified reduction in the present paper which may have significantly lower cancellation errors. Thus, while the current implementation is not yet competitive with SDPNAL [22], there may be applications of the reductions presented in this paper also in some other context.

Acknowledgments

The authors are thankful for the helpful criticism of two anonymous referees. The first author also likes to Roland Freund for very helpful discussions.
References


