The Gauss-Newton Direction in Semidefinite Programming

Serge Kruk [*] Masakazu Muramatsu [†] Franz Rendl [‡]
Robert J. Vanderbei [§] Henry Wolkowicz [¶]

May 14, 1998
University of Waterloo
Department of Combinatorics and Optimization
Waterloo, Ontario N2L 3G1, Canada
PRELIMINARY Research Report CORR 96-??

Key words: Semidefinite Programming, Newton Direction, Stability, Symmetrization.

Abstract

Primal-dual interior-point methods have proven to be very successful for both linear programming and, more recently, for semidefinite programming (SDP) problems. Many of the techniques that have been so successful for linear programming (LP) have been extended to SDP. In fact, interior point methods are currently the only successful techniques for SDP.

[*]Research supported by Natural Sciences Engineering Research Council Canada. E-mail sgkruk@barrow.uwaterloo.ca

[†]Department of Mechanical Engineering, Sophia University, 7-1 Kioi-cho, Chiyoda-ku, Tokyo 102 Japan

[‡]Technische Universität Graz, Institut für Mathematik, Kopernikusgasse 24, A-8010 Graz, Austria

[§]EMS Program Director, ACE-42 E-Quad, Princeton University, Princeton NJ 08544, Tel: 609-258-0876, Fax: 609-258-3796, E-mail rvdb@princeton.edu, http://www.princeton.edu/~rvdb/

[¶]Research supported by Natural Sciences Engineering Research Council Canada. E-mail hwolkowi@orion.math.uwaterloo.ca, http://orion.math.uwaterloo.ca~hwolkowi

[0]This report is available by anonymous ftp at orion.uwaterloo.ca, in directory pub/henry/reports and also with URL http://orion.uwaterloo.ca/~hwolkowi/henry/reports/newtondir.ps.gz

We present a new paradigm for deriving these methods: 1) using the optimality conditions from the dual log-barrier problem, we obtain primal feasibility, dual feasibility, and perturbed complementary slackness equations; 2) the perturbed complementary slackness condition is quite nonlinear, so we modify this condition to obtain a bilinear condition, i.e. a condition that is less nonlinear; 3) we now find a search direction by applying the Gauss-Newton method to the least squares problem for these optimality conditions.

In the case of LP, the Gauss-Newton direction for the least squares problem is equivalent to the Newton direction applied to solving the modified (square) nonlinear system of optimality conditions. Though this paradigm does not directly provide a new search direction for linear programming, it does provide a new approach for convergence proofs as well as motivation for step lengths larger than one.

However, there is one major difference between LP and SDP that raises several interesting questions. That difference is the form of the perturbed complementarity condition used in the optimality conditions. In LP this condition is modified to be $ZX - \mu I = 0$. The primal matrix X and the dual slack matrix Z are diagonal in LP but may only be symmetric in SDP; this results in ZX not being symmetric in general, i.e. the optimality conditions in the SDP case end up as an overdetermined system of nonlinear equations.

There have been various approaches which modify the complementarity condition so that the linearization of the optimality conditions are "square", i.e. map between the same spaces. These approaches can have several drawbacks, e.g. numerical instability near the optimum and lack of positive definiteness after symmetrization. Our least squares approach requires no symmetrization and does not suffer from these drawbacks. We concentrate on solving the "non-square," or overdetermined, system in the best way possible. In particular, we use Gauss-Newton type methods. This leads to numerically stable as well as excellent search directions which lead to the central path. Though the numerical efficient calculation of the Gauss-Newton direction is still an open question, we present a preliminary "top down" elimination approach that is competitive timewise and empirical evidence suggests that it is often more robust than other directions currently in use.

Contents

1 INTRODUCTION

Primal-dual interior-point (p-d i-p) methods have proven to be very successful for both linear programming (LP) and, more recently, for semidefinite programming (SDP) problems. Many of the techniques that have been so successful for linear programming have been extended to SDP. In fact, interior-point methods currently appear to be the only successful techniques for SDP.

The derivation of interior-point algorithms for SDP has followed closely upon the lessons learned from the very successful applications in LP. This has led to specific choices for the form of the perturbed optimality conditions to which Newton's method is applied. In this paper, we present a new paradigm for deriving these methods both for LP and for SDP.

1. Using the optimality conditions from the dual log-barrier problem, we obtain the system of three equations: primal feasibility; dual feasibility; and perturbed complementary slackness, see Section 5.4, program (5.21) and equations (5.23), (5.24), and (5.25).

2. The linear system for the Newton direction (N-direction) for the log-barrier problem is ill-conditioned but can be solved in a stable way by using an augmented system, i.e. this direction itself is *not ill-conditioned* and can be used successully. However, the perturbed complementary slackness condition (5.25) is quite nonlinear, so we modify this condition to obtain a bilinear condition, i.e. a condition that is less nonlinear, since Newton type methods converge much faster on systems that are less nonlinear.

3. We now find the Gauss-Newton search direction (GN-direction) to the least squares problem for the resulting equations.

This paradigm is very similar to what is currently done, e.g. see [13]. In fact, item 1 is unchanged. However, the motivation for the modifications of the optimality conditions in item 2 is usually from the ill-conditioning. Here we emphasize that this ill-conditioning can be avoided, see Section 5, but the true motivation should be to obtain a more linear system. The main difference with our paradigm is item 3, where we use a GN direction on the least squares problem rather than a Newton direction on modified optimality conditions. This GN approach gives rise to two new stable and efficient directions.

The p-d i-p approach has proven to be the superior choice of interior-point methods for LP. The current favourite is the Mehrotra predictor-corrector approach. This can be viewed as a simplified, composite, damped Newton method on the optimality conditions, with a backtracking linesearch, see [13]. In addition, it is shown in [13] that the method cannot be viewed as a Newton method on the log-barrier problem but can be explained as a modification of the optimality conditions that removes the ill-conditioning that arises in log-barrier Newton methods. However, as discussed in the above paradigm, the ill-conditioning is a *red herring*. The modification reduces the nonlinearity of the complementarity condition and thus speeds up Newton type methods.

1.1 Outline

In this paper we introduce and study the properties of search directions for primal-dual interior-point methods for semidefinite programming, where these directions arise from a Gauss-Newton method applied to minimizing the norm of the (overdetermined) system of optimality conditions, rather than a Newton method applied directly to modified optimality conditions. In addition, we compare several directions to see which direction results in fewer iterations in the p-d i-p framework. We conclude that the Gauss-Newton method is competitive and often more robust.

Several sections (title preceeded by *) are of secondary nature and can be skipped.

We complete this section with the background on SDP, see Section 1.2. Then in Section 2 we define the GN direction. Section 2.1.1 contains some theoretical results on the GN direction, e.g. results on full rank of the linearized systems. In Section 2.2 we present another approach that is based on truncation and also results in full rank linearized systems.

The solution techniques are discussed in Section 3. Section 3.1 contains the details of the elimination approach used in our implementation of the GN direction. In addition, we show that the linear system for the search direction is full rank at each iteration as well as in the limit. (This is not true for the various directions in use at this time.) Then, in Section 3.2 we study the adjoints and generalized inverses of the operators in the optimality conditions. In particular, we show that we can perform elimination on GN similar to what is done in LP.

The numerical tests appear in Section 4. These tests compare various search directions. We see that our new direction is very competitive in number of iterations, cpu time, and robustness. In particular, it is superior when we work with certain badly scaled problems.

In Section 5 we recall how ill-conditioning arises from log-barrier problems in NLP and how this can be avoided using an augmented system. In Section 5.4 we present the paradigm for SDP and derive the optimality conditions. In this last part of the paper we deal with justifying the ill-conditioning claims for our paradigm. This may be well known to many readers and can be skipped.

We conclude with several remarks and open questions in Section 6.

1.2 Background

We now include the notation and background information on SDP. Comprehensive uptodate notation is available on WWW with URL:
http://orion.uwaterloo.ca/˜hwolkowi/henry/software/psd_tool.d/sdpnotation.d/notation.ps.

Let $\mathcal{S}_n$ denote the space of $n \times n$ symmetric matrices equipped with the trace inner product, $\langle A, B\rangle = \operatorname{trace} AB$; and let $A \succeq 0$ (resp. $A \succ 0$) denote positive semidefiniteness (resp. positive definiteness), $A \succeq B$ denotes $A - B \succeq 0$, i.e. $\mathcal{S}_n$ is equipped with the Löwner partial order. We let $\mathcal{P}$ denote the cone of symmetric positive semidefinite matrices; $\mathcal{M}_n$ denotes the set of $n \times n$ real matrices.

SDP is an extension of linear programming, denoted LP, where nonnegative variables are replaced by semidefinite matrices. Our (primal) semidefinite programming problem is

$$
(PSDP) \qquad \begin{array}{rll} \mu^* := & \max & \operatorname{trace} CX \\ & \text{subject to} & \mathcal{A}X = b \\ & & X \succeq 0, \end{array}
$$

where the linear operator $\mathcal{A} : \mathcal{S}_n \to \Re^m$ is defined as

$$
\mathcal{A}X = (\operatorname{trace} A_i X),
$$

and A_i, $i = 1, \dots m$, are given symmetric matrices. To avoid trivialities, we assume that μ^* is finite valued. Thus the problem has a bounded optimal value and a feasible solution. We also assume that the matrices $\{A_i : i = 1, \dots, m\}$ are linearly independent.

This problem has recently generated a lot of interest. One reason is that there are many diverse applications: in discrete optimization see e.g. [3, 20, 44]; in engineering see e.g. [10, 60]; for matrix completions see e.g. [24, 1]. Another reason for the interest is that SDPs can be solved efficiently using interior-point methods. More applications and evidence of the current high level of activity can be found in the recent theses: [2, 43, 22, 39, 25, 65, 28, 26, 54]; and in the recent books and notes [63, 9, 36, 61, 56, 48, 38].

SDP has many similarities with LP. One important point, mentioned above, is that SDP problems can be solved very efficiently using primal-dual interior-point (p-d i-p) methods. These methods are based on applying Newton's method to the Karush-Kuhn-Tucker (KKT) optimality conditions: dual feasibility; primal feasibility; and complementary slackness; see e.g. (5.14,5.15,5.16) below. Many of the algorithms from LP can be extended to SDP and retain many of their nice properties. (See e.g. the books on interior-point methods: by Nesterov and Nemirovsky [36]; by Wright [61]; by Ye [62]; and edited by Terlaky [56].)

However, there are several subtle differences with LP.

- • One interesting difference is that duality gaps can exist for SDP in the absence of strictly feasible solutions (Slater's constraint qualification, CQ). This means that strong duality can fail, i.e. the dual may not be attained and/or there is a nonzero duality gap between the primal and dual optimal values. However, feasible interior-point methods usually assume strictly feasible solutions, so we will not dwell on that here. (See e.g. [42] for a discussion on strong duality and [40] for the generic properties.)

- • Another difference is that strict complementarity may fail at the optimum, i.e. there may be no optimal primal-dual pair for which $X + Z$ is full rank.

(Here Z is the dual slack variable.) (See e.g. [5, 50] for a discussion on strict complementarity and [40] for the generic properties.)

→ • Finally, the difference that we focus on is that the perturbed complementary slackness equations, that are essential for p-d i-p methods, involve possibly nondiagonal matrices and can be written in many different ways.

For example, if Z is the slack matrix in the dual (5.20) then:

1. the following comes from the optimality conditions for the log-barrier problem of the dual SDP problem (5.21)
$$X - \mu Z^{-1} = 0;$$
2. a nonlinear transformation reduces the nonlinearity and yields
$$ZX - \mu I = 0;$$
3. equivalent forms for the above two equations are
$$Z - \mu X^{-1} = 0,$$
$$XZ - \mu I = 0;$$
4. a general symmetrization operator yields, see e.g. [32],
$$H_P(M) := \frac{1}{2}\left(PMP^{-1} + (PMP^{-1})^t\right) \tag{1.1}$$
where $M = XZ$ and P is any nonsingular matrix;
   (a) the A.H.O. direction [6] corresponds to choosing $P = I$; this is equivalent to symmetrizing ZX, i.e. replacing it by $(ZX + XZ)/2$;
   (b) the H.K.M. direction, ([23], [27], [30]), corresponds to choosing $P = Z^{1/2}$; this is equivalent to symmetrizing ΔX after solving the linear system with ΔX treated as a general matrix;
   (c) the NT direction, [59], corresponds to $P = G^{-1}$, where G is the unique matrix such that $D := G^t Z G = G^{-1} X G^{-t}$ is a diagonal matrix.

(See [6, 57, 48] for comparisons of various Newton directions.)

For SDP, there is an additional complication that arises from the various forms of the complementarity conditions, i.e. the most popular form of these conditions for LP is $ZX - \mu I = 0$. In the case of SDP, as mentioned above, this form does not map between the same spaces since ZX is not symmetric unless Z, X commute. (It is symmetric on the central path, i.e. when $ZX = \mu I$.) Moreover, the derivative (linearization) of the optimality conditions may not be a one-one map since strict complementarity may fail. The above equivalences present several of the approaches that "symmetrize" the complementarity conditions in order to get a "square" system so that they can apply Newton's method for the search direction, see e.g. [45, 23, 30, 27, 32, 31, 6, 64, 59, 52]. In fact, to our knowledge all of the current SDP p-d i-p algorithms perform some sort of symmetrization. (See e.g. the software packages [58, 18, 11, 4].)

2 THE ZX FORM of COMPLEMENTARITY

We now study the most popular form of the complementarity conditions i.e.

$$ZX - \mu I = 0. \tag{2.1}$$

This is the one that is used in LP. Though many other linearizations appear to be based on different forms, e.g. the symmetrization $XZ + ZX - 2\mu I = 0$ studied in [5, 6], these are all still trying to solve the above form. Thus, the main question is how to solve this form efficiently.

We now look at (2.1) as an operator mapping the symmetric matrix space $\mathcal{S}_n \times \mathcal{S}_n$ to the larger space of matrices $\mathcal{S}_n \times \mathcal{M}_n$. With this in mind, we see that we have an overdetermined system of nonlinear equations. The standard approach to solving such a system is to solve an equivalent least squares problem, see (2.3), i.e. no symmetrization is needed. We now explore this possibility and see that there are many advantages for this point of view.

Equivalently, we are solving the equation

$$\Phi_Z F_\mu(s) = \Phi_Z F_\mu(X, y, Z) = \begin{pmatrix} Z + C - \mathcal{A}^* y \\ b - \mathcal{A}(X) \\ ZX - \mu I \end{pmatrix} =: \begin{pmatrix} F_d \\ F_p \\ F_c \end{pmatrix} = 0, \tag{2.2}$$

where F_μ is defined in (5.18), $\Phi_Z = (I \ I \ Z)$, and $\cdot^*$ denotes adjoint operator. Thus we are solving a modified, less nonlinear, system. (Such modifications can result in improved speed of convergence, see e.g. [7, 19, 41]. In fact, such a modification is the basis behind the success of algorithms for trust region subproblems, see e.g. [41, 53, 46, 33].)

2.1 The GN-Direction

The standard approach to solving a system of nonlinear equations is to apply the Gauss-Newton method, i.e. we minimize the sum of squares of the equations and use an approximation to the Newton search direction, the GN direction.

Under very mild assumptions, e.g. full column rank of the derivative $(\Phi_Z F_\mu)'$, (which holds here, Theorem 2.2) there are several advantages to using a Gauss-Newton method for solving nonlinear equations, see e.g. [12, 14, 16, 15].

1. If the optimal value of the least squares problem is 0, then we get quadratic convergence asymptotically.
   In our case, if we assume that the duality gap is 0 and that both primal and dual SDP are attained, then there is a solution for the optimality conditions which means that the least squares problem does indeed have a (attained) minimum value of 0 for each $\mu > 0$.

2. The GN direction always exists.
   Thus there are no problems with existence as there are for the other search directions used in the literature, see e.g. [31].

3. The direction is always a descent direction for the least squares problem.
   Thus the least squares problem can be used as a merit function for monitoring sufficient decrease, though it is not a true potential function, see [35, Page 5].

4. The direction is scale invariant.

With this motivation, we now look at the nonlinear least squares problem

$$R^*_\mu = \min_s f_\mu(s) := \frac{1}{2}||\Phi_Z F_\mu(s)||_2^2. \tag{2.3}$$

The optimal value R^*_μ is called the residual.

Remark 2.1 *The GN direction is formed by dropping the second order terms of the Hessian of f_μ from the Newton equation $\nabla^2 f_\mu(s) d_N = -\nabla f_\mu(s)$. Recall that $F_c = F_c(X,Z) = ZX - \mu I$. The second order terms are: $\langle F_c(Z,X), F''_c(Z,X)(\cdot,\cdot)\rangle$. If we denote the changes in Z, X as l, h respectively, then*

$$(Z+l)(X+h) - \mu I = ZX - \mu I + Zl + lX + lh.$$

The curvature in this direction is therefore:

$$\operatorname{trace}(ZX - \mu I)lh.$$

Therefore, the second order terms that are dropped from the Newton direction to get the GN direction are rather simple, i.e. similar to the LP case

$$H := \frac{1}{2}\begin{bmatrix} 0 & (ZX-\mu I) \\ (ZX-\mu I) & 0 \end{bmatrix}.$$

This matrix is indefinite and can cause problems if included in the approximate Hessian. Note that the residual $R^*_\mu = 0$ in our case, which implies that the matrix of second derivatives is 0 at the optimum in (2.3). Thus the error is very small near the central path.

Finding the GN-direction is therefore equivalent to solving the normal equations

$$((\Phi_Z F_\mu)')^* (\Phi_Z F_\mu)'(\Delta s) = -\left((\Phi_Z F_\mu)'\right)^* (\Phi_Z F_\mu), \tag{2.4}$$

which is equivalent to finding the least squares solution of the overdetermined system

$$(\Phi_Z F_\mu)'(\Delta s) = -\left(\Phi_Z F_\mu\right). \tag{2.5}$$

One popular method for solving the above system is to solve (2.5) using a (sparse, e.g. [8, 55]) QR decomposition. While the N-direction for the least squares problem is solved from the system

$$\left(((\Phi_Z F_\mu)')^* (\Phi_Z F_\mu)' + H\right)(\Delta s) = -\left((\Phi_Z F_\mu)'\right)^* (\Phi_Z F_\mu) \tag{2.6}$$

which cannot exploit the normal equations. Moreover, the addition of the operator H introduces negative eigenvalues and so the N-direction may not point to the minimum.

The overdetermined linear system we are solving is:

$$\begin{bmatrix} 0 & -\mathcal{A}^* & I \\ -\mathcal{A} & 0 & 0 \\ Z\cdot & 0 & \cdot X \end{bmatrix} \begin{pmatrix} \Delta X \\ \Delta y \\ \Delta Z \end{pmatrix} = \begin{pmatrix} -F_d \\ -F_p \\ -F_c \end{pmatrix}. \tag{2.7}$$

The above system is very sparse and most of the terms do not change at each iteration. We have to exploit this fact as is done in linear programming where block elimination is used.

2.1.1 Theoretical Properties

We have assumed that the operator $\mathcal{A}$ is onto, i.e. the matrices A_k are linearly independent or, equivalently, $\mathcal{A}^*$ is one-one or full column rank. We now see that we get existence of the GN-direction without any other assumptions. However, to guarantee full rank of the Jacobian at the optimum, we do need to assume that the primal and dual optimum are both unique. This is equivalent to primal and dual nondegeneracy as defined and used in [5, 6] and shown to be generic, see also [51, 40].

We first prove the well known property of uniqueness of points on the central path.

Theorem 2.1 *Suppose that (X, y, Z) lie on the central path, the solution of (2.3), with log-barrier parameter $\mu > 0$. Then 0 is the unique solution of the system (2.7).*

Proof. Suppose that $(\Delta X, \Delta y, \Delta Z)$ solves (2.7). The central path properties yield

$$\Delta Z = \mathcal{A}^*(\Delta y),\ \mathcal{A}\Delta X = 0,\ Z\Delta X + \mathcal{A}^*(\Delta y)X = 0,\ Z^{-1} = \frac{1}{\mu}X.$$

Therefore

$$\begin{array}{rcl} 0 & = & \mathcal{A}\Delta X \\ & = & -\mathcal{A}(Z^{-1}\mathcal{A}^*(\Delta y)X) \\ & = & -\frac{1}{\mu}\mathcal{A}(X\mathcal{A}^*(\Delta y)X) \\ & = & -\left(\frac{1}{\mu}\sum_i (\Delta y)_i \operatorname{trace}\left((A_kX)(A_iX)\right)\right) \\ & = & -\left(\frac{1}{\mu}\sum_i (\Delta y)_i \operatorname{trace}\left((X^{\frac{1}{2}}A_kX^{\frac{1}{2}})(X^{\frac{1}{2}}A_iX^{\frac{1}{2}})\right)\right). \end{array}$$

The $m \times m$ matrix

$$\left[\operatorname{trace}(X^{\frac{1}{2}}A_kX^{\frac{1}{2}})(X^{\frac{1}{2}}A_iX^{\frac{1}{2}})\right]$$

is a Gram matrix and is nonsingular since the matrices A_k, and so also $X^{\frac{1}{2}}A_kX^{\frac{1}{2}}$, are linearly independent. This implies that $\Delta y = 0$.

□

We can also prove that the linear system is full rank off of the central path, which implies that the GN-direction always exists.

Theorem 2.2 *Suppose that both X and Z are positive definite. Then the linear operator in system (2.7) is full rank.*

Proof. Suppose that $(\Delta X, \Delta y, \Delta Z)$ solves (2.7) with right hand side 0. Then $\Delta Z = \mathcal{A}^*\Delta y$. Therefore, $\mathcal{A}\Delta X = 0$ and

$$Z\Delta X X^{-1} = S := -\mathcal{A}^*\Delta y \perp \Delta X. \tag{2.8}$$

This means that the quadratic form

$$0 = \operatorname{trace} S\Delta X = \operatorname{trace} Z\Delta X X^{-1}\Delta X = ||Z^{\frac{1}{2}}\Delta X X^{-\frac{1}{2}}||_F^2, \tag{2.9}$$

i.e. necessarily $\Delta X = 0$. This means that both Δy and ΔZ are 0 as well.

□

Under strict complementarity and nondegeneracy, we can also prove that the linear system at optimality is full rank.

Theorem 2.3 *Suppose that (X, y, Z) are a unique optimal primal and dual solution (equivalently primal and dual nondegenerate) of PSDP. In addition, suppose that strict complementarity holds, i.e. $X + Z \succ 0$. Then 0 is the unique solution of the system (2.7) at $\mu = 0$.*

Proof. Suppose that $(\Delta X, \Delta y, \Delta Z)$ solves (2.7). Then $\Delta X, \Delta Z$ are symmetric and so $(\Delta X, \Delta y, \Delta Z)$ also solves the symmetrized system

$$Z\Delta X + \Delta Z X + \Delta X Z + X\Delta Z = 0.$$

The result in [6, Theorem 1] implies that the Jacobian corresponding to the symmetrized system is nonsingular and so necessarily $(\Delta X, \Delta y, \Delta Z) = 0$.

We can provide a self-contained proof using the approach above in Theorem 2.2, i.e. if $w = \Delta X$ and $s = \mathcal{A}^*\Delta y$, then $Zw + sX = 0 = ZX$, and $Z + X \succ 0$. Since Z and X commute, we can assume that they are diagonal. The strict complementary slackness and uniqueness implies that both s and w are 0.

□

Another important property is invariance.

Theorem 2.4 *The GN direction, the least-square solution to (2.5) used to solve program (2.3), is invariant under affine transformation of the variables $r = Hs + h$ where H is non-singular.*

Proof. The GN step in s-space, from current point s_c, is given by

$$s_+ = s_c - \left((\Phi_Z F_\mu(s_c))'\right)^\dagger \left(\Phi_Z F_\mu(s_c)\right).$$

Now, under the affine scaling $r = Hs + h$ we obtain that $\Phi_Z F_\mu(s) = \Phi_Z F_\mu(H^{-1}(r - h)) =: G(r)$. Using the full rank property in Theorem 2.3, we see that the GN step in the r-space, from the current point $r_c = Hs_c + h$, will be

$$
\begin{aligned}
r_+ &= r_c - (G(r_c)')^\dagger G(r_c) && (2.10)\\
&= Hs_c + h - \left((\Phi_Z F_\mu(s_c))' H^{-1}\right)^\dagger (\Phi_Z F_\mu(s_c)) && (2.11)\\
&= Hs_c + h - H\left((\Phi_Z F_\mu(s_c))'\right)^\dagger (\Phi_Z F_\mu(s_c)) && (2.12)\\
&= Hs_+ + h. && (2.13)
\end{aligned}
$$

The GN step is therefore invariant under such transformations of the space.

□

Perhaps these properties explain the behavior of the GN step for the badly scaled problems described in Section 4.

2.2 *The Truncated Form of Complementarity

We can find redundant equations in (2.2) and solve a smaller system. We can remove the lower triangular part from the nonlinear equation, i.e. we get the equivalent equation

$$\mathcal{T}(ZX - \mu I) = 0, \tag{2.14}$$

where the linear operator $\mathcal{T} : \mathcal{M}_n \to \mathcal{S}_n$, by ignoring the strictly lower triangular part of the matrix, i.e. the i, j components are

$$(\mathcal{T}(W))_{ij} = \begin{cases} W_{ij} & \text{if } i \le j \\ W_{ji} & \text{if } i > j \end{cases}.$$

(Note $\mathcal{T}(M) = PMP^{-1}$ implies $P^{-1}\mathcal{T}(M)P = M$ which shows that this does not fall under the general symmetrization framework (1.1).)

The resulting system is now square, i.e. the domain and range space are the same. Note that $\mathcal{T}$ is the composition of two linear maps, i.e. first apply the simple coordinate (orthogonal) linear projection onto the subspace of upper triangular matrices in the space of $n \times n$ matrices; then apply the one-one mapping to the space of symmetric matrices formed by duplicating the upper triangular part in the lower triangular part. We have $\mathcal{T}(\mathcal{T}(W)) = \mathcal{T}(W)$, i.e. $\mathcal{T}$ is idempotent (a projection). In fact, if we identify the upper triangular matrices with $\mathcal{S}_n$, then we can consider $\mathcal{T}$ as orthogonal as well.

The optimality conditions become

$$\begin{pmatrix} Z + C - \mathcal{A}^* y \\ b - \mathcal{A}(X) \\ \mathcal{T}(ZX - \mu I) \end{pmatrix} = 0. \tag{2.15}$$

This does not introduce new nonlinearities but does guarantee that we map between the same spaces. In fact, we can prove that we do not lose information in the optimality conditions when we only consider the upper triangular parts.

Lemma 2.1 *Suppose that $Z \succ 0$. Then the linear operator $\mathcal{T}_Z(\cdot) := \mathcal{T}(Z\cdot)$ is a one-one mapping on $\mathcal{S}_n$. In addition, if $\mathcal{T}_Z(X) = \mu I$, for some $X = X^t$, then $ZX = \mu I$.*

Proof. Suppose that $\mathcal{T}_Z(W) = 0$, for some $W = W^t$, i.e. $W \in \Re^{(n^2+n)/2}$. Then the eigenvalues of (the strictly lower triangular matrix) ZW are all 0. But the symmetric matrix $Z^{\frac{1}{2}} W Z^{\frac{1}{2}}$ is similar to ZW and so must have all 0 eigenvalues as well, i.e. this symmetric matrix must be the 0 matrix. Therefore, W must be the 0 matrix, i.e. the linear operator $\mathcal{T}_Z$ has trivial null space and so is one-one on $\mathcal{S}_n$.

Similarly, $\mathcal{T}_Z(X) = \mu I$ implies that $Z^{\frac{1}{2}} X Z^{\frac{1}{2}}$ has all eigenvalues equal to μ, i.e. $Z^{\frac{1}{2}} X Z^{\frac{1}{2}} = \mu I$ and so $ZX = \mu I$ as well.

□

The Newton-direction for (2.15) is obtained from solving the system

$$\begin{aligned} \Delta Z - \mathcal{A}^*(\Delta y) &= -F_d & (2.16) \\ -\mathcal{A}(\Delta X) &= -F_p & (2.17) \\ \mathcal{T}(Z\Delta X + \Delta Z X) &= -\mathcal{T}(Z F_{Z,X}). & (2.18) \end{aligned}$$

Equivalently,

$$\begin{bmatrix} 0 & -\mathcal{A}^* & I \\ -\mathcal{A} & 0 & 0 \\ \mathcal{T}_Z(\cdot) & 0 & \mathcal{T}(\cdot X) \end{bmatrix} \begin{pmatrix} \Delta X \\ \Delta y \\ \Delta Z \end{pmatrix} = \begin{pmatrix} -F_d \\ -F_p \\ -\mathcal{T}_Z(X) + \mu I \end{pmatrix}. \tag{2.19}$$

We can first solve (2.16) for an obviously symmetric ΔZ (in terms of Δy),

$$\Delta Z = -F_d + \mathcal{A}^*(\Delta y), \tag{2.20}$$

and then substitute this expression into (2.18) to get the symmetric ΔX

$$\Delta X = \mu \mathcal{T}_Z^{-1}(\mathcal{T}(I)) - \mathcal{T}_Z^{-1}(\mathcal{T}_Z(X)) + \mathcal{T}_Z^{-1}(\mathcal{T}(F_d X)) - \mathcal{T}_Z^{-1}\left[\mathcal{T}(\mathcal{A}^*(\Delta y) X)\right]$$

or more simply to

$$\Delta X = \mu Z^{-1} - X + \mathcal{T}_Z^{-1}\left[\mathcal{T}(F_d X)\right] - \mathcal{T}_Z^{-1}\left[\mathcal{T}\left(\mathcal{A}^*(\Delta y)X\right)\right]. \tag{2.21}$$

We remark that this is very similar to the expression obtained in [23, equation (4.16)]. The only difference is the addition of the operator that performs the symmetrization of the upper triangular parts. This avoids the nonsymmetry of ΔX; moreover, it does this in a stable fashion using the simple orthogonal linear projection. Unlike the symmetrization of ΔX done later on in [23], the solution we get is indeed the solution of the linearization of the optimality conditions, though without the lower triangular part.

We can now substitute for ΔX in (2.17). We get the following equation for Δy.

$$\mathcal{O}(\Delta y) = v_1, \tag{2.22}$$

where $\mathcal{O}$ is the linear operator defined by

$$\mathcal{O}(\cdot) := \mathcal{A}\left(\mathcal{T}_Z^{-1}\mathcal{T}(\mathcal{A}^*(\cdot)X)\right)$$

and v_1 is the vector

$$v_1 := \mu\mathcal{A}(Z^{-1}) - b + \mathcal{A}(\mathcal{T}_Z^{-1}\mathcal{T}(F_d X)). \tag{2.23}$$

Again this is similar to the equations in [23] except for the addition of the linear operator $\mathcal{T}$.

To solve the final system it is equivalent to find $W = W^t$ and Δy so that $\mathcal{A}(W) = v_1$ and $\mathcal{T}(\mathcal{A}^*(\Delta y)X) = \mathcal{T}(ZW)$. This is a square system with size $n + n(n+1)/2$. One then has to solve a $n(n+1)/2$ size system when doing the back substitution for ΔX.

Though we do not include numerical tests with this direction, the tests that we have done show that this direction is very robust and does not suffer from ill-conditioning near the optimum.

3 LINEARIZED LEAST SQUARES

The efficient calculation of our GN direction is still an open problem. This has to involve a clever implementation of a QR algorithm. Therefore, we present a constrained GN, denoted RGN, direction that is easier to calculate. The RGN direction still maintains the nice properties of the GN direction.

3.1 Basic Elimination

In order to reduce the execution time of the least-square solver, we can algebraically eliminate some variables using block elimination. The order we eliminate is different than that used in LP or other SDP codes, where ΔX is eliminated first from the linearized complementarity condition. One advantage that we have is that we never invert Z and so do not appear to get numerical problems when we are very close to the optimum.

It is simplest to first force dual feasibility and eliminate ΔZ from (2.7), which, we recall, is of size $(t(n) + m + n^2) \times (t(n) + m + t(n))$. We use

$$\Delta Z = \mathcal{A}^*(\Delta y) - F_d,$$

to obtain the $(m + n^2) \times (t(n) + m)$ reduced *dual feasible* system

$$\begin{bmatrix} -\mathcal{A} & 0 \\ Z\cdot & \mathcal{A}^* \cdot X \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} -F_p \\ F_d X - F_c \end{bmatrix}. \tag{3.1}$$

The upper part of this equation is still sparse and constant and can therefore be treated separately once for all iterations. Moreover, since the backsolve for ΔZ is a sum of matrices it is inexpensive and exploits any sparsity in $\mathcal{A}^*$.

A smaller system, everything else being equal, is faster to solve than a large one. In the absence of specific information about the structure of the problem, we pursue this idea further since it is possible to eliminate more variables and obtain an even smaller system. For a specific problem with known sparsity structure, maxcut for example, it is possible to eliminate variables while maintaining sparsity. In the general case we discuss here, the major bottleneck being the size of the system, we aim at a reduction in size with little concern for sparsity.

We now consider the primal constraint operator $\mathcal{A}$, which we write as

$$\mathcal{A} = [\mathcal{A}_B \mathcal{A}_N],$$

where the $\mathcal{A}_B$ is a subset of size $m \times m$ of $\mathcal{A}$ that is easily invertible. (We can think of $\mathcal{A}$ as an $m \times t(n)$ matrix operating on a vector $\Delta x = vec(\Delta X)$.) We now divide ΔX and ΔZ in the same manner to obtain

$$\Delta X_B = \mathcal{A}_B^{-1} F_p - \mathcal{A}_B^{-1} \mathcal{A}_N \Delta X_N, \tag{3.2}$$

which we substitute back into the dual feasible system (3.1), to obtain the $(n^2 \times t(n))$ system

$$\begin{bmatrix} -Z_B \mathcal{A}_B^{-1} \mathcal{A}_N + Z_N & \mathcal{A}^* \cdot X \end{bmatrix} \begin{bmatrix} \Delta X_N \\ \Delta y \end{bmatrix} = F_d X - F_c - Z_B \mathcal{A}_B^{-1} F_p. \tag{3.3}$$

Albeit more costly to setup and backsolve than (3.1), this system, for large dense problems is faster to solve because the bottleneck is the least square solve.

Corollary 3.1 *Theorems 2.2 and 2.3 hold with system (2.7) replaced by system (3.3).*

Proof. The proofs follow the same pattern as in the above mentioned theorems. For example, at each iteration we have

$$\mathcal{A}(\Delta X) = 0, \quad -\mathcal{A}^*(\Delta y) + \Delta Z = 0$$

and

$$Z\Delta X + \Delta Z X = 0.$$

Therefore we can apply (2.8) and (2.9).

□

3.2 *Elimination using Pseudo-Inverses and Special Structure

Another approach is to do elimination in the same vein as is done in LP but preserving the least squares properties. First note that $x^* := A^\dagger b$ is the (best) least squares solution of a system of linear equations $Ax = b$, where best refers to minimum norm among all least squares solutions and $A^\dagger$ is the Moore-Penrose inverse of A. Therefore, for a system of linear equations $Ax - By = c$, the vectors x^*, y^* are (best) least squares solutions if $x^* = A^\dagger(By^* + c)$. Therefore, we can eliminate x and solve the equivalent least squares problem with objective

$$||(I - AA^\dagger)(By + c)||_2^2.$$

Note that $(I - AA^\dagger)$ is the orthogonal projection onto the null space of A^*.

This motivates our need for the generalized inverses of the various operators and corresponding projections in the Gauss-Newton equation. Though we have not tested out these various approaches, we include the calculations in the hope that they will prove useful in future tests. Let

$$\mathcal{Z}(\cdot) := Z\cdot$$

and

$$\mathcal{X}(\cdot) := \cdot X.$$

Then the adjoint operators are

$$\mathcal{Z}^*(M) = \frac{1}{2}\left(M^t Z + ZM\right)$$

and

$$\mathcal{X}^*(M) = \frac{1}{2}\left(XM^t + MX\right).$$

For $M \in \mathcal{M}_n, S \in \mathcal{S}_n$, this can be verified by

$$\langle M, ZS\rangle = \text{trace}\, M^t ZS = \frac{1}{2}\text{trace}\left(S(M^t Z + ZM)\right) = \langle \mathcal{Z}^*(M), S\rangle,$$

$$\langle M, SX\rangle = \text{trace}\, M^t SX = \frac{1}{2}\text{trace}\left(S(XM^t + MX)\right) = \langle \mathcal{X}^*(M), S\rangle.$$

Then

$$\mathcal{Z}^*\mathcal{Z}(S) = \frac{1}{2}\left(SZ^2 + Z^2 S\right)$$

and

$$\mathcal{X}^*\mathcal{X}(S) = \frac{1}{2}\left(X^2 S + SX^2\right).$$

Since $X, Z \succ 0$, both $\mathcal{Z}(S)$ and $\mathcal{X}(M)$ are full rank (into operators). Therefore, both $\mathcal{Z}^*\mathcal{Z}(S)$ and $\mathcal{X}^*\mathcal{X}(S)$ are nonsingular operators.

And the Moore-Penrose inverses are surprisingly simple, i.e.

$$\mathcal{Z}^\dagger(M) = \frac{1}{2}\left(M^t Z^{-1} + Z^{-1}M\right)$$

and

$$\mathcal{X}^\dagger(M) = \frac{1}{2}\left(X^{-1}M^t + MX^{-1}\right)$$

This can be verified from

$$\mathcal{Z}\mathcal{Z}^\dagger\mathcal{Z}(S) = \mathcal{Z}\left(\frac{1}{2}\left((ZS)^t Z^{-1} + Z^{-1}(ZS)\right)\right), \mathcal{Z}^\dagger\mathcal{Z} = I,$$

$$\mathcal{Z}^\dagger\mathcal{Z}\mathcal{Z}^\dagger(M) = \mathcal{Z}^\dagger\left(\frac{1}{2}Z\left(M^t Z^{-1} + Z^{-1}M\right)\right), \quad \mathcal{Z}^\dagger(M) = \mathcal{Z}^\dagger(ZM^t Z^{-1}),$$

and

$$\mathcal{X}\mathcal{X}^\dagger\mathcal{X}(S) = \mathcal{X}\left(\frac{1}{2}\left(X^{-1}(SX)^t + (SX)X^{-1}\right)\right), \quad \mathcal{X}^\dagger\mathcal{X} = I,$$

$$\mathcal{X}^\dagger\mathcal{X}\mathcal{X}^\dagger(M) = \mathcal{X}^\dagger\left(\frac{1}{2}\left(X^{-1}M^t + MX^{-1}\right)X\right), \quad \mathcal{X}^\dagger(M) = \mathcal{X}^\dagger(X^{-1}M^t X).$$

In addition, $\mathcal{X}^\dagger\mathcal{X}$ and $\mathcal{Z}^\dagger\mathcal{Z}$ are both the trivial identity projection, i.e. $\mathcal{X}, \mathcal{Z}$ are both homomorphisms into $\mathcal{M}_n$ or into operators. And, it is easy to verify that both $\mathcal{X}\mathcal{X}^\dagger$

and $\mathcal{Z}\mathcal{Z}^\dagger$ are both orthogonal projections, i.e. they are self-adoint idempotents. For example, $\left\langle \mathcal{X}\mathcal{X}^\dagger(N), M \right\rangle = \left\langle \mathcal{X}\mathcal{X}^\dagger(M), N \right\rangle$. In fact,

$$\mathcal{X}\mathcal{X}^\dagger(M) = \frac{1}{2}\left(X^{-1}M^tX + M\right), \tag{3.4}$$

$$\mathcal{Z}\mathcal{Z}^\dagger(M) = \frac{1}{2}\left(ZM^tZ^{-1} + M\right). \tag{3.5}$$

If we rewrite the overdetermined system (2.7) as

$$\mathcal{K}_1\Delta X + \mathcal{K}_2\Delta y + \mathcal{K}_3\Delta Z = r,$$

then we can eliminate the variables in any order. Different orders will result in less fill-in. It would be preferable to end up in a system with just Δy. However, eliminating Δy first appears to result in little fill-in. Note that due to the least squares approach, we have to eliminate by column and we cannot eliminate rows.

We begin by preprocessing to make $\mathcal{A}$ orthogonal, i.e. $\mathcal{A}\mathcal{A}^* = I$. Now let

$$\Delta y = \mathcal{K}_2^\dagger\left(r - \mathcal{K}_1\Delta X - \mathcal{K}_3\Delta Z\right), \tag{3.6}$$

Note that, since $\mathcal{A}$ is orthogonal,

$$\mathcal{K}_2^\dagger = \begin{pmatrix} -\mathcal{A}^* & 0 & 0 \end{pmatrix}. \tag{3.7}$$

We now have an equivalent least squares problem in the variables $\Delta X, \Delta Z$ only

$$\min_{\Delta X, \Delta Z} \frac{1}{2}||\mathcal{P}_2\left(r - \mathcal{K}_1\Delta X - \mathcal{K}_3\Delta Z\right)||_2^2, \tag{3.8}$$

where the projection

$$\begin{aligned} \mathcal{P}_2 &:= \left(I - \mathcal{K}_3\mathcal{K}_3^\dagger\right) \\ &= I - \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}. \end{aligned}$$

Thus we get

$$\min_{\Delta X, \Delta Z} \frac{1}{2}\left\|\begin{pmatrix} -F_p \\ -F_c \end{pmatrix} - \begin{pmatrix} -\mathcal{A} \\ \mathcal{Z} \end{pmatrix}\Delta X - \begin{pmatrix} 0 \\ \mathcal{X} \end{pmatrix}\Delta Z\right\|_2^2 \tag{3.9}$$

We now proceed to eliminate ΔZ, since $\mathcal{K}_3$ is simpler that K_1.

$$\Delta Z = \mathcal{Q}_3^\dagger \left(\begin{pmatrix} -F_p \\ -F_c \end{pmatrix} - \begin{pmatrix} -\mathcal{A} \\ \mathcal{Z} \end{pmatrix} \Delta X \right), \tag{3.10}$$

where

$$\mathcal{Q}_3^\dagger = \begin{pmatrix} 0 \\ \mathcal{X}^\dagger \end{pmatrix}. \tag{3.11}$$

We now have an equivalent least squares problem in ΔX only

$$\min_X \frac{1}{2} \left\| \begin{pmatrix} -F_p \\ -\left(I - \mathcal{X}\mathcal{X}^\dagger\right) F_c \end{pmatrix} - \begin{pmatrix} -\mathcal{A} \\ \left(I - \mathcal{X}\mathcal{X}^\dagger\right) \mathcal{Z} \end{pmatrix} \Delta X \right\|_2^2. \tag{3.12}$$

From (3.4), we get the simple expression.

$$\left(I - \mathcal{X}\mathcal{X}^\dagger\right) \mathcal{Z}(S) = \left(I - \mathcal{X}\mathcal{X}^\dagger\right)(ZS) = \frac{1}{2}(X^{-1}(SZ)X + ZS).$$

Alternatively, we could eliminate ΔZ first and then ΔX. We use

$$\Delta Z = \mathcal{K}_3^\dagger \left(r - \mathcal{K}_1 \Delta X - \mathcal{K}_2 \Delta y\right), \tag{3.13}$$

where

$$\mathcal{K}_3^\dagger = \frac{1}{2} \begin{pmatrix} I & 0 & \mathcal{X}^\dagger \end{pmatrix}, \tag{3.14}$$

since $\mathcal{X}$ is an into operator. We now have an equivalent least squares problem in the variables $\Delta X, \Delta y$ only.

$$\min \frac{1}{2} || \mathcal{P}_3 \left(r - \mathcal{K}_1 \Delta X - \mathcal{K}_2 \Delta y\right) ||_2^2, \tag{3.15}$$

where the projection

$$\begin{aligned} \mathcal{P}_3 &:= \left(\mathcal{K}_3 \mathcal{K}_3^\dagger - I\right) \\ &= \begin{bmatrix} -\frac{1}{2} I & 0 & \frac{1}{2}\mathcal{X}^\dagger \\ 0 & -I & 0 \\ -\frac{1}{2}\mathcal{X} & 0 & \frac{1}{2}\frac{1}{2}\mathcal{X}\mathcal{X}^\dagger - I \end{bmatrix}. \end{aligned}$$

Note that

$$\begin{aligned} \mathcal{P}_3 \mathcal{K}_1 &= \begin{bmatrix} \frac{1}{2}\mathcal{X}^\dagger \mathcal{Z} \\ -\mathcal{A} \\ \left(\frac{1}{2}\mathcal{X}\mathcal{X}^\dagger - I\right) \mathcal{Z} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{4}\left(X^{-1}(Z\cdot)^t + (Z\cdot)X^{-1}\right) \\ -\mathcal{A} \\ \frac{1}{4}\left(X^{-1}(Z\cdot)^t X + (Z\cdot)\right) - \mathcal{Z} \end{bmatrix}; \end{aligned}$$

while

$$\mathcal{P}_3 r = \begin{pmatrix} \frac{1}{2}F_d - \frac{1}{2}\mathcal{X}^\dagger F_c \\ F_p \\ -\frac{1}{2}\mathcal{X}F_d - (\frac{1}{2}\mathcal{X}\mathcal{X}^\dagger - I)F_c \end{pmatrix}$$

and

$$\mathcal{P}_3\mathcal{K}_2 = \begin{pmatrix} \frac{1}{2}\mathcal{A}^* \\ 0 \\ -\frac{1}{2}\mathcal{X}\mathcal{A}^* \end{pmatrix}.$$

We now eliminate

$$\Delta X = (\mathcal{P}_3\mathcal{K}_1)^\dagger \left(\mathcal{P}_3(r - \mathcal{K}_2\Delta y)\right), \tag{3.16}$$

We set

$$\mathcal{P}_1 := \left((\mathcal{P}_3\mathcal{K}_1)(\mathcal{P}_3\mathcal{K}_1)^\dagger - I\right)\mathcal{P}_3.$$

The final system in just Δy is

$$\min_{\Delta y} ||\mathcal{P}_1\left(r - \mathcal{K}_2\Delta y\right)||_2^2. \tag{3.17}$$

4 NUMERICAL TESTS

4.1 Comparisons Between Directions

We now present some numerical results comparing the constrained GN directions to the well-known Alizadeh-Haeberly-Overton [6] (AHO), Helmberg-Rendl-Vanderbei-Wolkowicz [23]/ Kojima-Shindoh-Hara [27]/ Monteiro [30] (HKM) and Nesterov-Todd [37] (NT) directions. All tests were performed in Matlab using the SDPT3 code [58] from Todd, Toh, Tutuncu. We modified the mainline of SDPT3 to allow a fourth option, the constrained GN direction. The logic relating to step length and the updating of the centering parameters were untouched to allow a reasonable comparaison of the relative value of the GN directions.

Figures (1) to (3) illustrate the decrease in both the duality gap and in the infeasibility on a typical random semidefinite program with different scaling on the vector b. The problem has size $(n = 10, m = 15)$ and the algorithm used a predictor-corrector approach on all directions. The GN directions, seem numerically very stable, and reduce the duality gap as much as the AHO direction in about as many iterations, typically one more. For the constrained GN direction, we eliminated dy and part of dX to reduce the system size to $(t(n) \times n^2)$ by forcing the primal and dual feasibility equations. That is, we solved (3.3). On a random SDP, as is the case

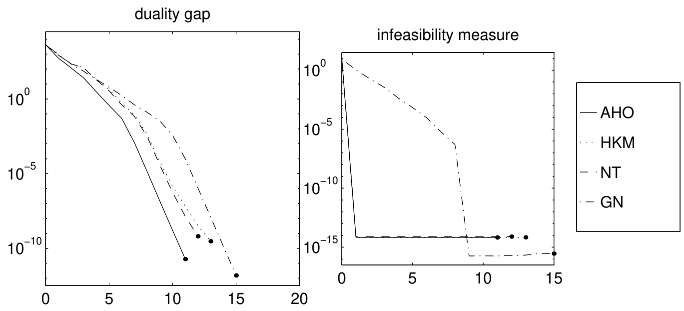

Figure 1: Random Semidefinite Program

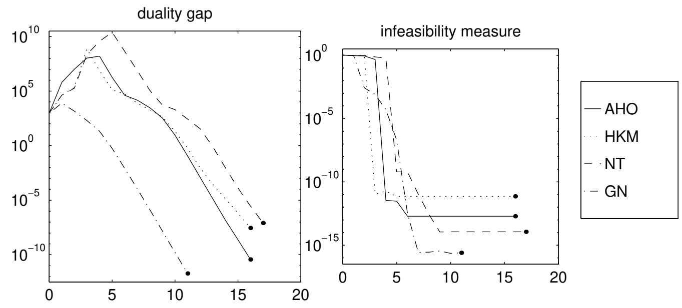

Figure 2: $b \leftarrow b * 100$

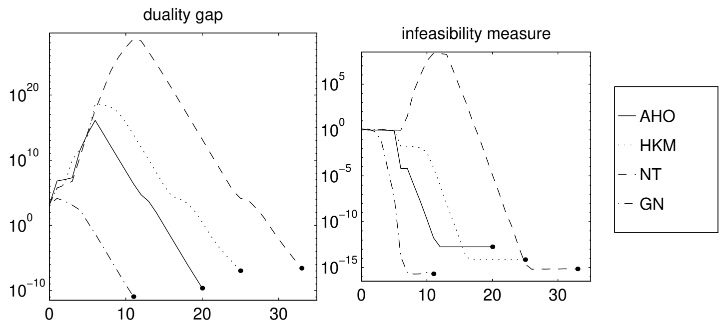

Figure 3: $b \leftarrow b * 500$

in this instance, with unknown and dense structure, this elimination will perform well.

For a really badly conditioned problem (b=b*500), all the methods (AHO,HKM,NT) start by increasing the infeasibility while the GN methods always reduces the gap monotonically.

AHO				GN			
Iter	P feas	D feas	Gap	Iter	P feas	D feas	Gap
9	1.2e-14	1.7e-15	4.4e-14	10	2.4e-16	1.8e-15	4.0e-15
9	8.0e-16	2.6e-16	5.8e-14	10	1.2e-16	2.6e-16	4.6e-15
9	1.8e-15	3.7e-16	1.1e-13	10	3.3e-16	5.3e-16	1.0e-14
9	1.6e-15	3.4e-16	7.4e-14	10	5.3e-16	3.9e-16	7.0e-15
9	2.8e-15	5.8e-16	4.5e-14	10	1.4e-16	4.8e-16	2.4e-15
9	1.0e-15	2.8e-16	8.0e-14	10	2.2e-16	1.9e-16	7.3e-15

Table 1: Random problems with $n = 7, m = 28$

Table (1) compares the number of iterations and the accuracy of the solutions of the AHO and GN directions on random problems. The columns represent, in order, the iteration count, primal feasibility ($||b - \mathcal{A}(X)||/||b||$), dual feassibility ($||\mathcal{A}^*(y) + Z - C||/||C||$, and the duality gap ($\langle X, Z\rangle$). The additional digit of accuracy for the GN case is most likely due to the presence of a Z^{-1} in the computation of the AHO direction.

5 *NLP ILL-CONDITIONING

It is well-known that the Hessian from log-barrier functions becomes ill-conditioned as the barrier parameter approaches 0. In this section we look at various techniques of correcting this ill-conditioning. This provides the motivation for the paradigm introduced above. The reader who is interested mainly in the different search directions can skip this section.

5.1 Optimality Conditions

Consider the convex, set constrained minimization problem

$$(Unc) \qquad \begin{array}{ll} \min & f(x) \\ \text{subject to} & x \in \Omega, \end{array}$$

where f is differentiable, $f : X \rightarrow \Re$ and Ω is a convex set in an inner-product space X. Then, see e.g. [47], the point $x^* \in \Omega$ solves (Unc) if and only if $\nabla f(x^*) \in (\Omega - x^*)^+$. This extension of stationarity (of Fermat's theorem) is often called the Pshenichnyi condition. The proof is elementary; it follows from looking at directional derivatives of f. The special case when Ω is a cone is particularly interesting.

Theorem 5.1 *Suppose that Ω is a closed convex cone. Then x^* solves (Unc) if and only if*

$$\begin{array}{cc} \nabla f(x^*) \in \Omega^+ & \textit{dual feasibility} \\ x^* \in \Omega & \textit{primal feasibility} \\ \langle \nabla f(x^*), x^* \rangle = 0 & \textit{complementary slackness} \end{array}$$

In the case that $\Omega = \Re^n_+$, the nonnegative orthant, we get $\langle \nabla f(x^*), x^* \rangle = 0$ if and only if $x_i \left(\nabla f(x^*)\right)_i = 0, \ \forall i$.

In the case that $\Omega = \mathcal{P}$, the cone of positive semidefinite matrices, we get $\langle \nabla f(x^*), x^* \rangle = 0$ if and only if $\nabla f(x^*) x^* = 0$, i.e. the product of the matrices is the 0 matrix.

However, these special cases do not hold in general. The purpose of our use of a general cone and a general inner-product space X is to emphasize this fact. We conclude from this that the optimality conditions for a general program involve an inner-product for the complementary slackness conditions and not a product of variables.

The advantage of transforming the inner-product to the more complicated product, is to have the powerful central path property, see Section 5.4.

5.2 Overcoming Ill-Conditioning

We first present the log-barrier approach for a simple nonlinear programming problem. We illustrate the problem of ill-conditioning of the Hessian of the log-barrier problem; we then present the approach used in [21] to overcome the ill-conditioning.

Let $f, g : \Re^n \to \Re$ and be twice continuously differentiable functions. (In our case, f is the linear part of the log-barrier problem, i.e. it is the Lagrangian of (PSDP); while $g(x) = \det(x)$, is the determinant. In the LP case, $g(x)$ is the product $\prod_i x_i$.) We consider the simple nonlinear program

$$(NLP) \qquad \begin{array}{ll} \min & f(x) \\ \text{subject to} & g(x) \geq 0, \end{array}$$

Under an appropriate constraint qualification, the first order necessary optimality conditions yield

$$\begin{array}{rcl} \nabla f(x) - \lambda \nabla g(x) & = & 0 \\ g(x) & \geq & 0 \\ \lambda & \geq & 0 \\ \lambda g(x) & = & 0, \end{array} \tag{5.1}$$

for an appropriate Lagrange (scalar) multiplier λ. The log-barrier problem, for fixed $\mu > 0$, consists in minimizing the log-barrier function

$$B_\mu(x) = f(x) - \mu \log g(x) \tag{5.2}$$

over the region where $g(x) > 0$. The stationarity condition for minimizing (5.2) yields the first order optimality condition

$$0 = \nabla B_\mu(x_\mu) = \nabla f(x_\mu) - \frac{\mu}{g(x_\mu)} \nabla g(x_\mu), \;\; g(x_\mu) > 0. \tag{5.3}$$

Under mild assumptions we can assume that

$$x_\mu \to x^*, \;\; g(x_\mu) \to 0.$$

Therefore, we see that

$$\frac{\mu}{g(x_\mu)} \to \lambda^*, \tag{5.4}$$

an optimal Lagrange multiplier.

The Hessian is

$$\nabla^2 B_\mu(x) = \left(\nabla^2 f(x) - \frac{\mu}{g(x)} \nabla^2 g(x)\right) + \left[\left(\frac{\mu}{g(x)}\right) \frac{1}{g(x)} \nabla g(x) \nabla g(x)^t\right]. \tag{5.5}$$

We have bracketed the Hessian into two parts. The first ( ) part converges to a fixed matrix, by (5.4). However, the second [ ] part is a rank one matrix which dominates the Hessian since $g(x)$ converges to 0. Therefore, (unless ∇g is nonsingular or the optimum satisfies $g(x^*) > 0$) the Hessian becomes increasingly ill-conditioned as μ approaches 0.

We now increase the size of the Hessian by 1 in order to remove the ill-conditioning. We introduce the new variable $\Delta v \in \Re$

$$\Delta v = -\frac{\mu}{g^2(x)} \nabla g(x)^t \Delta x. \tag{5.6}$$

The Newton equation, for the N-direction $d = (\Delta x, \Delta v)^t$, becomes

$$\left[\begin{array}{cc} \nabla^2 f(x_\mu) - \frac{\mu}{g(x_\mu)} \nabla^2 g(x_\mu) & \nabla g(x_\mu) \\ \nabla g(x_\mu)^t & \frac{g(x_\mu)^2}{\mu} \end{array} \right] \left(\begin{array}{c} \Delta x \\ \Delta v \end{array} \right) = \left(\begin{array}{c} -\nabla f(x_\mu) + \frac{\mu}{g(x_\mu)} \nabla g(x_\mu) \\ 0 \end{array} \right). \tag{5.7}$$

As $\mu \to 0$ the Hessian becomes

$$\left[\begin{array}{cc} \nabla^2 f(x^*) - \lambda^* \nabla^2 g(x^*) & \nabla g(x^*) \\ \nabla g(x^*) & 0 \end{array} \right]. \tag{5.8}$$

This matrix is nonsingular under the standard second order sufficient optimality conditions, i.e. that the Hessian of the Lagrangian is positive definite on the tangent space; equivalently, $\nabla^2 f(x^*) - \lambda^* \nabla^2 g(x^*)$ is positive definite on the orthogonal complement of $\nabla g(x^*)$. Thus we see that the ill-conditioning is easily corrected, i.e. the problem itself is not ill-posed; just the standard Newton equation is ill-posed.

There is an alternative way to overcome the ill-conditioning. This is to use the perturbed KKT conditions as done in [17] and, also, described in [13]. We use the substitution

$$\lambda = \frac{\mu}{g(x)}. \tag{5.9}$$

Note that this substitution is equivalent to (5.6), i.e. if we equate λ and v and differentiate we get

$$\frac{\Delta v}{\Delta x} = \frac{\partial \lambda}{\partial x} = \mu \frac{\partial}{\partial x} \frac{1}{g(x)} = -\frac{\mu}{g^2(x)} \nabla g(x).$$

Now rather than add the new variable in this way, the ill-conditioning is removed by adding the equivalent equation $\lambda g(x) = \mu$. This removes the ill-conditioning that arises from differentiation. Moreover, this is the form of the KKT conditions that is used in p-d i-p methods. In fact, in [13], this process of removing the ill-conditioning is used as the motivation for introducing this product form of the complementary slackness condition. Then (5.3) becomes (Note that if g is vector valued, then $\lambda_i g_i(x) = \mu$ is added.)

$$\begin{array}{rcl} \nabla f(x) - \lambda \nabla g(x) & = & 0 \\ \lambda g(x) & = & \mu \\ g(x) & > & 0 \end{array} \tag{5.10}$$

Thus we see that there are two ways to avoid the ill-conditioning of the log-barrier problem. The first technique increases the dimension of the problem by one and still finds the N-direction from the log-barrier problem. The second technique changes the optimality conditions (perturbs) and finds a new direction based solely on the optimality conditions and not on a log-barrier problem. In fact, the latter does not yield a symmetric system of equations for the N-direction. The second technique is the one that gives rise to the equations used in p-d i-p methods.

However, from the above we see that we can find the N-direction of the original log-barrier problem in a stable way without the transformation of the complementarity condition. If the log-barrier problem is motivating p-d i-p methods, then it is reasonable to expect that this direction is the one we want. However, there is another reason for transforming the complementarity, i.e. to make the optimality conditions more linear; in fact, they become bilinear. And, it is well known that Newton's method converges faster on more linear functions.

5.3 Linear Programming

We now look at the special case of linear programming.

$$\begin{array}{lll} (PLP) \qquad & \max & c^t x \\ & \text{subject to} & Ax = b \\ & & x \geq 0, \end{array}$$

For $x, z \geq 0$, the optimality conditions are:

$$F(s) = F(x,y,z) := \begin{pmatrix} z + c - A^t y \\ b - Ax \\ x^t z \end{pmatrix} = 0. \tag{5.11}$$

This is an underdetermined system of bilinear equations. However, if we use these equations, then we can end up with very small steps when trying to stay interior to the nonnegative orthant.

We will follow the p-d i-p approach discussed in [29].

We first introduce the associated *log-barrier problem* for the dual LP

$$
(DLP) \qquad \begin{array}{ll} \min & \operatorname{trace} b^t y \\ \text{subject to} & A^t y - z = c \\ & z \geq 0, \end{array}
$$

where A^t denotes the *transpose*. The *dual log-barrier problem* is:

$$
(DBP) \qquad \begin{array}{cc} \text{minimize} & b^t y - \mu \sum_i \log z_i \\ \text{subject to} & A^t y - z = c \\ & z > 0, \end{array} \tag{5.12}
$$

where μ is a positive real number called the *barrier parameter.*

For each $\mu > 0$, there is a corresponding Lagrangian:

$$
L_\mu(x, y, z) = \quad b^t y - \mu \textstyle\sum_i \log z_i + x^t \left(z + c - A^t y\right). \tag{5.13}
$$

The dual log-barrier problem is equivalent to $\max_x \min_{y,z>0} L_\mu(x,y,z)$. The first-order optimality conditions for the saddle point of this Lagrangian are obtained by differentiation:

$$
\begin{aligned}
\nabla_x L_\mu &= z + c - A^t y = 0 &(5.14)\\
\nabla_y L_\mu &= b - Ax = 0 &(5.15)\\
\nabla_z L_\mu &= (x_i - \mu \frac{1}{z_i}) = 0. &(5.16)
\end{aligned}
$$

These conditions are equivalent to finding the projected optimality conditions for the barrier problem. There exists a unique solution (x_μ, y_μ, z_μ) to these optimality conditions. The one-parameter family $\{(x_\mu, y_\mu, z_\mu) : 0 \leq \mu \leq \infty\}$ is called the *central trajectory.* Given a point (x, y, z) on the central trajectory it is easy to determine its associated μ value using (5.16):

$$
\mu = \frac{z^t x}{n}. \tag{5.17}
$$

We note that if the point is a feasible solution of the primal and the dual problem, then $z^t x$ is the gap between the primal and the dual objective value.

For simplicity of notation, we rewrite the optimality conditions as the function

$$
F_\mu(s) = F_\mu(x, y, z) := \begin{pmatrix} z + c - A^t y \\ b - Ax \\ (x_i - \mu / z_i) \end{pmatrix} =: \begin{pmatrix} F_d \\ F_p \\ F_{zx} \end{pmatrix}. \tag{5.18}
$$

The equations correspond to: dual feasibility, primal feasibility, and complementary slackness, respectively. The solution s^* to $F_\mu(s) = 0$ satisfies the Karush-Kuhn-Tucker conditions (5.14) to (5.16) and is the optimal solution to the barrier problem.

The function $F_\mu : \Re^n \times \Re^m \times \Re^n \to \Re^n \times \Re^m \times \Re^n$. The Hessian of the Lagrangian is:

$$\nabla^2 L_\mu(s) = \left[\begin{array}{cc|c} 0 & -A^t & I \\ -A & 0 & 0 \\ \hline I & 0 & \mu \mathrm{Diag}\,(\frac{1}{z_i^2}) \end{array}\right]. \tag{5.19}$$

Using the Schur complement determinant formula we see that (using $M := \mu\mathrm{Diag}\,(\frac{1}{z_i^2})$)

$$\begin{aligned} &\det\left(\nabla^2 L_\mu(s)\right) \\ &\quad = \det(M)\det\left(\begin{bmatrix} 0 & -A^t \\ -A & 0 \end{bmatrix} - \begin{bmatrix} I \\ 0 \end{bmatrix} M^{-1}[I\ 0]\right) \\ &\quad = \det(M)\det\begin{pmatrix} -M^{-1} & -A^t \\ -A & 0 \end{pmatrix} \\ &\quad = \det(M)\det(-M^{-1})\det(AMA^t) = \det(-AMA^t) \end{aligned}$$

The matrix M clearly becomes singular as μ approaches 0. Thus, as we saw above, the Hessian indeed does become ill-conditioned as μ approaches 0.

5.3.1 Stabilizing (Dual Approach)

It is possible to use a slight variation of the Gould trick to stabilize a dual LP log-barrier problem. An application of Newton's method to the stationarity condition of the log-barrier yields the system

$$[\mu A Z^{-2} A^t] d = -(b - \mu A Z^{-1} e),$$

where

$$z := A^t y - c \text{ and } Z := \mathrm{Diag}\,(z).$$

We can define multiplier approximations $x_i := \mu/z_i$. Strictly satisfied and "nearly"-active sets are defined by

$$\mathcal{I} := \{1, 2, 3, \ldots, m\},\ \mathcal{I}_S := \{i \in \mathcal{I} \mid \mu/z_i < \tau\},\ \mathcal{I}_A = \mathcal{I} \setminus \mathcal{I}_S.$$

We define the conditioning vector,

$$r := \mu Z_A^{-2} A_A^t d - x_A,$$

$$\text{where} \quad \begin{array}{llll} A_A & \text{are the columns of} & A & \text{indexed by} \quad \mathcal{I}_A \\ Z_A & \text{are the rows and columns of} & Z & \text{indexed by} \quad \mathcal{I}_A \\ x_A & \text{are the entries of} & x & \text{indexed by} \quad \mathcal{I}_A \end{array} .$$

With this notation, the new system (which still solves for the true N-direction) reads

$$\begin{bmatrix} \mu A_S Z_S^{-2} A_S^t & A_A \\ A_A^t & -Z_A^2/\mu \end{bmatrix} \begin{bmatrix} d \\ r \end{bmatrix} = - \begin{bmatrix} b - [Ax]_S \\ -z_A \end{bmatrix}.$$

To see that this is well-conditioned, consider that, as $\mu \to 0$, we have that $\mu A_S Z_S^{-2} A_S^t \to 0_{n\times n}$ and $z_A \to 0_{(n-m)\times 1}$. So that we can consider the matrix

$$F := \begin{bmatrix} 0_{n\times n} & A_A \\ A_A^t & 0 \end{bmatrix}.$$

Under the usual assumption that A is full rank , the matrix A_A will converge to a square matrix so that

$$\det(F) = \det(A_A A_A^t) \neq 0.$$

And we conclude that this is well-conditioned. This could be called a dual-only approach as the primal variables appear only as multipliers. A primal-dual approach is also possible as we now illustrate.

5.3.2 Stabilizing (Primal-Dual Approach)

The usual primal-dual interior-point approach is derived from the constrained dual barrier problem

$$\min \left\{ b^t y - \mu \sum_{\mathcal{I}} log z_i \mid A^t y - c - z = 0, z \geq 0 \right\}.$$

The Lagrangian reads

$$\mathcal{L}(x, y, z) = b^t y + \mu \sum_{\mathcal{I}} log z_i + x^t (A^t y - c - z),$$

and a solution of the optimality conditions can be attempted by Newton's method and yields the system

$$\begin{bmatrix} 0 & -A^t & I \\ -A & 0 & 0 \\ I & 0 & \mu Z^{-2} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} c + z - A^t y \\ b - Ax \\ x - \mu Z^{-1} e \end{bmatrix}.$$

The ill-conditioning is due to the third equation since some components of μZ^{-2}, that read $\mu \frac{1}{z_i^2} \to \infty$ as $z_i \to 0$. Therefore active dual constraints cause problem. To transform the system we first define index sets

$$\mathcal{I}_S := \{i \in \mathcal{I} \mid \mu/z_i < \tau\}, \ \ \mathcal{I}_A := \mathcal{I} \setminus \mathcal{I}_A,$$

and we split the offending equation in terms of strictly satisfied and nearly-active constraints as

$$I\Delta x + \mu I_S Z_S^{-2} I_S^t \Delta z + \mu I_A Z_A^{-2} I_A^t \Delta z = -I_S x_S - I_A x_A + \mu Z^{-1} e,$$

$$\text{where} \quad \begin{array}{lllll} A_A & \text{are the columns of} & A & \text{indexed by} & \mathcal{I}_A \\ I_A & \text{are the columns of} & I & \text{indexed by} & \mathcal{I}_A \\ Z_A & \text{are the rows and columns of} & Z & \text{indexed by} & \mathcal{I}_A \\ x_A & \text{are the entries of} & x & \text{indexed by} & \mathcal{I}_A \end{array} .$$

Similarly for A_S, I_S, Z_S, x_S. We also define the conditioning vector

$$r := \mu Z_A^{-2} I_A^t \Delta z + x_A,$$

constructed from the elements responsible for the ill-conditioning. The point here is that, writing Δz in terms of r, we will invert the Z_A^{-2}, producing Z_A^2, a matrix of elements going to zero instead of infinity. This leads to the Newton system

$$\begin{bmatrix} 0 & -A^t & I & 0 \\ -A & 0 & 0 & 0 \\ I & 0 & \mu I_S Z_S^{-2} I_S^t & I_A \\ 0 & 0 & I_A^t & -Z_A^2/\mu \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ r \end{bmatrix} = - \begin{bmatrix} c + z - A^t y \\ b - Ax \\ I_S x_S - \mu Z^{-1} e \\ Z_A^2 x_A/\mu \end{bmatrix}.$$

Informally, μZ_S^{-2} is bounded as it is composed of slack variables corresponding to strictly satisfied constraints and its complement, Z_A^2/μ does go to zero so that the last system is well-conditioned.

Note that this system could radically change at each of the algorithm iterations since it depends on which constraints are deemed near-active at a particular step. Yet it does not rely on an accurate determination of active constraints since all constraints are kept; some are just algebraically transformed to reduce the condition number. This transformation may therefore be applied only after the algorithm has converged to a point where the original system has exhibited ill-conditioning.

We now show the conditioning of the transformed system. To simplify notation we denote

$$k := |\mathcal{I}_{\mathcal{A}}|, \ \text{and}\ l := |\mathcal{I}_{\mathcal{S}}|,$$

so that $n = k + l$. We will assume, without loss of generality, that the last k constraints are active. This can be achived by permutations of rows and columns which do not affect the conditioning. Consider the determinant of the left-hand side matrix,

$$F := \begin{bmatrix} O_{n\times n} & -A^t & I_{n\times n} & O_{n\times k} \\ -A & O_{m\times m} & O_{n\times n} & O_{m\times k} \\ I_{n\times n} & O_{n\times m} & \mu I_S Z_S^{-2} I_S^t & I_A \\ O_{k\times n} & O_{k\times m} & I_A^t & -Z_A^2/\mu \end{bmatrix}.$$

After a number straight-forward and tedious simplifications using Schur complements, we finally obtain,

$$\det(F) = -\det(A_S Z_S^{-2} A_S^t)\det(Z_A^2 + A_A^t \left[A_S Z_S^{-2} A_S^t\right]^{-1} A_A).$$

First we note that the determinant does not depend on μ. Also, Z_S is the diagonal matrix composed of slack variables. Therefore each component satisfies $0 < [Z_S]_i < \infty$ and the determinants of the last expression are non-zero and finite. We can conclude that the transformed system is well-conditoned as the numerical tests will attest.

We emphasize that the alternative way to avoid the ill-conditioning that arises in the differentiation, is to replace the complementarity condition with

$$ZX = 0,$$

where Z, X are the diagonal matrices with z, x on the diagonal, respectively. But, as Tapia et al.[13] demonstrated, the resulting system (or for that matter, any of the successful systems used in primal-dual interior-point) does not corresponds to a true Newton step.

Therefore the purpose for the ZX change is not just to avoid the ill-conditioning, (which is the motivation suggested in [13]) because, as we see, it can be avoided without using a completely different N-direction. The main purpose for this change is to increase the linearity of the system, since Newton's method speeds up on systems which are almost linear. (See e.g. the measure of nonlinearity in [12, σ in Theorem 10.2.1] and also [34, page 47] for a similar phenomenon for the trust region subproblem.)

5.3.3 Newton Versus Gauss-Newton

It is generally accepted that p-d i-p methods obtain their search direction by applying one step of Newton's method to the perturbed optimality conditions. In particular, see e.g. [13], this direction is not obtained from a Newton minimization method on some merit function, since the Jacobian is not symmetric. However, our paradigm points out that it is equivalent to a GN-direction obtained from the least squares minimization problem.

Though the direction from these two points of view are the same, one does get different insights. For example, one might now hope that Newton's method applied to the least squares problem might yield an improved direction. Most textbooks choose the GN-direction over the N-direction only because it is less expenive. In fact, they point out that the error could be large far from the solution and, therefore, a damped Gauss-Newton should be used.

However, the difference between the Gauss-Newton and Newton equations is easily calculated from 2.2. The term dropped in Gauss-Newton involves

$$H := (\Phi_Z F)''(\Phi_Z F) = \begin{bmatrix} 0 & ZX - \mu I \\ ZX - \mu I & 0 \end{bmatrix}$$

This symmetric matrix H has a 0 diagonal and therefore negative eigenvalues off the central path. Therefore, adding this on to the Gauss-Newton equation could cause the direction to point to a maximum or a saddle point.

Many questions arise: can we improve on the GN-direction by properly using the information in H; will a standard or scaled trust region approach mean that the N-direction from the least squares problem is better than the GN-direction; when does the Newton direction point to the minimum and when away from the minimum?

To improve over the GN-direction, one can apply an indefinite scaled trust region approach. Adding the indefinite trust region constraint

$$\langle H\Delta s, \Delta s\rangle = \delta,$$

for an appropriate δ results in the same direction as the GN-direction, since the H will cancel off. Therefore, we can do at least as well as the GN-direction by being clever enough. To improve on this, we need to either find a better scaling than H or adaptively choose δ to get closer to the boundary and improve the merit (least squares) function.

5.4 Primal-Dual Framework for SDP

In this section we will follow the p-d i-p approach developed in [23]. We follow the usual derivation of p-d i-p methods in Linear Programming and first introduce the associated *log-barrier problem* for the dual SDP

$$(DSDP) \qquad \begin{array}{ll} \min & \operatorname{trace} b^t y \\ \text{subject to} & \mathcal{A}^* y - Z = C \\ & Z \succeq 0, \end{array} \tag{5.20}$$

where $\mathcal{A}^*$ denotes the *adjoint operator*, $\mathcal{A}^* y = \sum y_i A_i$. The *dual barrier problem* is:

$$(DBP) \qquad \begin{array}{rc} \text{minimize} & b^t y - \mu \log\det Z \\ \text{subject to} & \mathcal{A}^* y - Z = C \\ & Z \succeq 0, \end{array} \tag{5.21}$$

where μ is a positive real number called the *barrier parameter.*

For each $\mu > 0$, there is a corresponding Lagrangian:

$$L_\mu(X, y, Z) = \quad b^t y - \mu \log\det Z + \operatorname{trace} X \left(Z + C - \mathcal{A}^* y\right). \tag{5.22}$$

With $X_\mu, Z_\mu \succ 0$, the first-order optimality conditions for the saddle point of this Lagrangian are obtained by differentiation:

$$\begin{aligned} \nabla_X L_\mu &= Z + C - \mathcal{A}^* y = 0 & (5.23) \\ \nabla_y L_\mu &= b - \mathcal{A}(X) = 0 & (5.24) \\ \nabla_Z L_\mu &= X - \mu Z^{-1} = 0. & (5.25) \end{aligned}$$

There exists a unique solution (X_μ, y_μ, Z_μ) to these optimality conditions. The one-parameter family $\{(X_\mu, y_\mu, Z_\mu) : 0 \leq \mu \leq \infty\}$ is called the *central trajectory* or the central path. Given a point (X, y, Z) on the central trajectory it is easy to determine its associated μ value using (5.25):

$$\mu = \frac{\operatorname{trace}(ZX)}{n}. \tag{5.26}$$

We note that if the point is a feasible solution of the primal and the dual problem, then $\operatorname{trace}(ZX)$ is the gap between the primal and the dual objective value.

For simplicity of notation, we rewrite the optimality conditions as the function

$$F_\mu(s) = F_\mu(X, y, Z) := \begin{pmatrix} Z + C - \mathcal{A}^* y \\ b - \mathcal{A}(X) \\ X - \mu Z^{-1} \end{pmatrix} =: \begin{pmatrix} F_d \\ F_p \\ F_{Z,X} \end{pmatrix}. \tag{5.27}$$

The equations correspond to: dual feasibility, primal feasibility, and complementary slackness, repectively. The solution s^* to $F_\mu(s) = 0$ satisfies the Karush-Kuhn-Tucker conditions (5.23) to (5.25) and is the optimal solution to the barrier problem.

5.4.1 The Log-Barrier N-Direction

The function $F_\mu : \mathcal{S}_n \times \Re^m \times \mathcal{S}_n \to \mathcal{S}_n \times \Re^m \times \mathcal{S}_n$ and is continuously differentiable on the open set $\mathcal{D} := \mathcal{S}_{n\succ} \times \Re^m \times \mathcal{S}_{n\succ}$, i.e. when restricted to positive definite matrices. Therefore, by a theorem of Sard [49], the image $F_\mu(C)$ of $C = \{x \in \mathcal{D} : F'_\mu \text{ is singular}\}$ has empty interior. Therefore, we would expect good behaviour for Newton type methods.

The Hessian of the Lagrangian is:

$$\nabla^2 L_\mu(s) = \left[\begin{array}{cc|c} 0 & -\mathcal{A}^* & I \\ -\mathcal{A} & 0 & 0 \\ \hline I & 0 & \mu Z^{-1} \cdot Z^{-1} \end{array}\right]. \tag{5.28}$$

Using the Schur complement determinant formula we see that (using $M := \mu Z^{-1} \cdot Z^{-1}$)

$$\begin{aligned} &\det\left(\nabla^2 L_\mu(s)\right) \\ &= \det(M)\det\left(\begin{bmatrix} 0 & -\mathcal{A}^* \\ -\mathcal{A} & 0 \end{bmatrix} - \begin{bmatrix} I \\ 0 \end{bmatrix} M^{-1} [I \ 0]\right) \\ &= \det(M)\det\begin{pmatrix} -M^{-1} & -\mathcal{A}^* \\ -\mathcal{A} & 0 \end{pmatrix} \\ &= \det(M)\det(M^{-1})\det(-\mathcal{A}M\mathcal{A}^*) = \det(-\mathcal{A}M\mathcal{A}^*) \end{aligned}$$

i.e. the nonsingularity of the Hessian depends essentially on the last block. The last block appears to converge to 0, since μ approaches 0. However, this is not the case, since the determinant of Z is hiding in the formula for the inverse. However, as mentioned above, this last block does become singular as shown in Section 5. To find a direction $\Delta s = (\Delta X, \Delta y, \Delta Z)$, we need to linearize the nonlinear complementary slackness condition. The Z^{-1} in $F_{Z,X}$ makes the equation highly nonlinear.

A similar approach to the LP case shows that one can find a stable way of finding this Newton direction.

6 CONCLUSION

In this paper we have studied new search directions for semidefinite programming. The search directions are all based on the GN direction for minimizing the norm of modified, perturbed, optimality conditions. This follows from a new paradigm based on the success in the LP case. Though the paradigm does not provide a new direction for LP, it does for SDP. In fact, the paradigm provides several new directions, and in addition can be used to derive existing directions and show existence where previously existence was a question.

The new approaches have several advantages: e.g. numerical stability, existence and descent generically, scale invariance, etc... And, it has proven to be competitive time wise with existing directions such as AHO and NT and often more robust.

In the future, we still have to study how to calculate the direction more efficiently, how to improve the Gauss-Newton direction using more of the Hessian, and how to exploit sparsity. In addition, many theoretical questions such as a proof of global convergence remain.

References

[1] A. ALFAKIH, A. KHANDANI, and H. WOLKOWICZ. An interior-point method for approximate distance matrix completions. Technical Report CORR Report 98-9, University of Waterloo, 1997. URL: ftp://orion.uwaterloo.ca/pub/henry/reports/distmat.ps.gz.

[2] F. ALIZADEH. *Combinatorial optimization with interior point methods and semidefinite matrices.* PhD thesis, University of Minnesota, 1991.

[3] F. ALIZADEH. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1995.

[4] F. ALIZADEH, J.-P. HAEBERLY, M. V. NAYAKKANKUPPAM, and M. L. OVERTON. Sdppack user's guide – version 0.8 beta. Technical Report TR1997–734, Courant Institute of Mathematical Sciences, NYU, New York, NY, March 1997.

[5] F. ALIZADEH, J-P.A. HAEBERLY, and M.L. OVERTON. Complementarity and nondegeneracy in semidefinite programming. *Mathematical Programming*, 77:111–128, 1997.

[6] F. ALIZADEH, J-P.A. HAEBERLY, and M.L. OVERTON. Primal-dual interior-point methods for semidefinite programming: convergence rates, stability and numerical results. *Siam J. Optimization*, page to appear in SIOPT, 1998.

[7] Adi Ben-Israel. Newton's method with modified functions. In *Recent developments in optimization theory and nonlinear analysis (Jerusalem, 1995)*, volume 204 of *Contemp. Math.*, pages 39–50. Amer. Math. Soc., Providence, RI, 1997.

[8] A. BJORCK. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, 1996.

[9] S. BOYD, V. BALAKRISHNAN, E. FERON, and L. El GHAOUI. Control system analysis and synthesis via linear matrix inequalities. *Proc. ACC*, pages 2147–2154, 1993.

[10] S. BOYD, L. El GHAOUI, E. FERON, and V. BALAKRISHNAN. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics.* SIAM, Philadelphia, PA, June 1994.

[11] N. BRIXIUS, F.A. POTRA, and R. SHENG. SDPHA: a matlab implementation of homogeneous interior-point algorithms for semidefinite programming. Technical report, University of Iowa, Iowa City, IA, 1998.

[12] J.E. DENNIS Jr. and R.B. SCHNABEL. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, NJ, 1983. Russian edition, Mir Publishing Office, Moscow, 1988, O. Burdakov, translator.

[13] A.S. EL-BAKRY, R.A. TAPIA, T. TSUCHIYA, and Y. ZHANG. On the formulation and theory of the Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications*, 89:507–541, 1996.

[14] M. ELHALLABI. An inexact hybrid algorithm for nonlinear systems of equations. Technical Report CRPC-TR94372, Department of Mathematical Sciences, Rice University, Houston, TX, 1994.

[15] M. ELHALLABI and R.A. TAPIA. An inexact trust-region feasible-point algorithm: globalization of Gauss-Newton method. Technical Report TR93-43, Department of Mathematical Sciences, Rice University, Houston, TX, 1993.

[16] M. ELHALLABI and R.A. TAPIA. An inexact trust-region feasible-point algorithm for nonlinear systems of equalities and inequalities. Technical Report CRPC-TR95543, Department of Mathematical Sciences, Rice University, Houston, TX, 1995.

[17] A.V. FIACCO and G.P. McCORMICK. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques.* John Wiley & Sons, New York, NY, 1968.

[18] K. FUJISAWA, M. KOJIMA, and K. NAKATA. SDPA semidefinite programming algorithm. Technical report, Dept. of Information Sciences, Tokyo Institute of Technology, Tokyo, Japan, 1995.

[19] J. GERLACH. Accelerated convergence in Newton's method. *SIAM Review*, 36:272–276, 1994.

[20] M. X. GOEMANS. Semidefinite programming in combinatorial optimization. *Mathematical Programming*, 79:143–162, 1997.

[21] N.I.M. GOULD. On the accurate determination of search directions for simple differentiable penalty functions. *IMA J. Num. Anal.*, 6:357–372, 1986.

[22] C. HELMBERG. *An interior point method for semidefinite programming and max-cut bounds.* PhD thesis, Graz University of Technology, Austria, 1994.

[23] C. HELMBERG, F. RENDL, R. J. VANDERBEI, and H. WOLKOWICZ. An interior point method for semidefinite programming. *SIAM Journal on Optimization*, pages 342–361, 1996.
URL: ftp://orion.uwaterloo.ca/pub/henry/reports/sdp.ps.gz.

[24] C. JOHNSON, B. KROSCHEL, and H. WOLKOWICZ. An interior-point method for approximate positive semidefinite completions. *Computational Optimization and Applications*, 9(2):175–190, 1998.

[25] S. KARISCH. *Nonlinear Approaches for Quadratic Assignment and Graph Partition Problems.* PhD thesis, University of Graz, Graz, Austria, 1995.

[26] E. DE KLERK. *Interior point methods for semidefinite programming.* PhD thesis, Delft University, 1997.

[27] M. KOJIMA, S. SHINDOH, and S. HARA. Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM J. Optim.*, 7(1):86–125, 1997.

[28] S. KRUK. Semidefinite programming applied to nonlinear programming. Master's thesis, University of Waterloo, 1996.
URL: http://orion.uwaterloo.ca/ hwolkowi/henry/software/sqqp.d/krukmsthesis.ps.gz.

[29] I. J. LUSTIG, R. E. MARSTEN, and D. F. SHANNO. Computational experience with a primal–dual interior point method for linear programming. *Linear Algebra and Its Applications*, 152:191–222, 1991.

[30] R.D.C. MONTEIRO. Primal-dual path-following algorithms for semidefinite programming. *SIAM J. Optim.*, 7(3):663–678, 1997.

[31] R.D.C. MONTEIRO and Y. ZANJACOMO. A note on the existence of the Alizadeh-Haeberly-Overton direction for semidefinite programming. *Mathematical Programming*, 78:393–401, 1997.
URL http://www.isye.gatech.edu/ monteiro/tech_reports/note.ps.

[32] R.D.C. MONTEIRO and Y. ZHANG. A unified analysis for a class of path-following primal-dual interior-point algorithms for semidefinite programming. Technical report, Georgia Tech, Atlanta, GA, 1996. ftp://pc5.math.umbc.edu/pub/mz_sdp.ps.gz or ftp://pc5.math.umbc.edu/pub/mz_sdp.dvi.gz.

[33] J.J. MORÉ and D.C. SORENSEN. Computing a trust region step. *SIAM J. Sci. Statist. Comput.*, 4:553–572, 1983.

[34] J.J. MORÉ and D.C. SORENSEN. Newton's method. In Gene H. Golub, editor, *Studies in Numerical Analysis*, volume 24 of *MAA Studies in Mathematics*. The Mathematical Association of America, 1984.

[35] L. NAZARETH. Computer solution of linear programs: non-simplex algorithms II. Technical report, Dept. of Pure and Applied Mathematics, Washington State University, Pullman, 1996.

[36] Y. E. NESTEROV and A. S. NEMIROVSKY. *Interior Point Polynomial Algorithms in Convex Programming : Theory and Algorithms*. SIAM Publications. SIAM, Philadelphia, USA, 1994.

[37] Y. E. NESTEROV and M. TODD. Primal-dual interior-point methods for self-scaled cones. Technical Report TR1125, School of OR and IE, Cornell University, Ithaca, NY, 1995.

[38] P. PARDALOS and H. WOLKOWICZ (Editors). *Topics in Semidefinite and Interior-Point Methods*, volume 18 of *The Fields Institute for Research in Mathematical Sciences, Communications Series*. American Mathematical Society, Providence, Rhode Island, 1998.

[39] G. PATAKI. *Cone programming and eigenvalue optimization: geometry and algorithms*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1996.

[40] G. PATAKI and L. TUNCEL. On the generic properties of convex optimization problems in conic form. Technical Report CORR 97-16, Department of Combinatorics and Optimization, Waterloo, Ont, 1997.

[41] M.J.D. POWELL. The use of band matrices for second derivative approximations in trust region algorithms. DAMTP Report 1997/NA12, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, England, 1997.

[42] M. RAMANA, L. TUNCEL, and H. WOLKOWICZ. Strong duality for semidefinite programming. *SIAM Journal on Optimization*, 7(3):641–662, 1997. URL:ftp://orion.uwaterloo.ca/pub/henry/reports/strongdual.ps.gz.

[43] M.V. RAMANA. *An algorithmic analysis of multiquadratic and semidefinite programming problems*. PhD thesis, Johns Hopkins University, Baltimore, Md, 1993.

[44] F. RENDL. Semidefinite programming and combinatorial optimization. Technical report, University of Graz, Graz, Austria, 1997. Notes for a lecture series given at the University of Pisa.

[45] F. RENDL, R. J. VANDERBEI, and H. WOLKOWICZ. Max-min eigenvalue problems, primal-dual interior point algorithms, and trust region subproblems. *Optimization Methods and Software*, 5:1–16, 1995.

[46] F. RENDL and H. WOLKOWICZ. A semidefinite framework for trust region subproblems with applications to large scale minimization. *Mathematical Programming*, 77(2):273–299, 1997.

[47] R.T. ROCKAFELLAR. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.

[48] R. SAIGAL, L. VANDENBERGHE, and H. WOLKOWICZ (Editors). *Handbook on: SEMIDEFINITE PROGRAMMING AND APPLICATIONS*. Kluwer Academic Publishers, Waterloo, Canada, 1998. in progress.

[49] A. SARD. The measure of the critical values of differential maps. *Bull. Amer. Math. Soc.*, pages 883–890, 1942.

[50] A. SHAPIRO. First and second order analysis of nonlinear semidefinite programs. *Mathematical Programming*, 77:301–320, 1997.

[51] A. SHAPIRO and M.K.H. FAN. On eigenvalue optimization. *SIAM J. Opt.*, 5:552–569, 1995.

[52] R. SHENG and F.A. POTRA. Nonsymmetric search directions for semidefinite programming. Technical report, University of Iowa, Iowa City, IA, 1997.

[53] D.C. SORENSEN. Newton's method with a model trust region modification. *SIAM J. Numer. Anal.*, 19:409–426, 1982.

[54] J. STURM. *Primal-dual interior point approach to semidefinite programming.* PhD thesis, Erasmus University Rotterdam, 1997.

[55] C. SUN. Dealing with dense rows in the solution of sparse linear least squares problems. Technical Report TR95-227, Department of Computer Science, Cornell University, Ithaca, NY, 1995.

[56] (Editor) T. TERLAKY. *Interior Point Methods of Mathematical Programming.* Kluwer Academic Publishers, Dordrecht; Boston, 1996.

[57] M. TODD. On search directions in interior-point methods programming. Technical Report TR1205, School of OR and IE, Cornell University, Ithaca, NY, 1997.

[58] M. TODD, K.C. TOH, and R.H. TUTUNCU. A MATLAB software package for semidefinite programming. Technical report, School of OR and IE, Cornell University, Ithaca, NY, 1996.

[59] M. TODD, K.C. TOH, and R.H. TUTUNCU. On the nesterov-todd direction in semidefinite programming. Technical Report TR1154, School of OR and IE, Cornell University, Ithaca, NY, 1996.

[60] L. VANDENBERGHE and S. BOYD. Semidefinite programming. *SIAM Review*, 38:49–95, 1996.

[61] S. WRIGHT. *Primal-Dual Interior-Point Methods.* SIAM, Philadelpia, Pa, 1996.

[62] Y. YE. *Interior-Point Algorithm: Theory and Practice.* University of Iowa, 1996.

[63] Y. YE. *Interior Point Algorithms: Theory and Analysis.* Wiley-Interscience series in Discrete Mathematics and Optimization. John Wiley & Sons, New York, 1997.

[64] Y. ZHANG. Basic equalities and inequalities in primal-dual interior-point methods for semidefinite programming. Technical report, Department of Mathematics and Statistics, University of Maryland Baltimore County, 1995.

[65] Q. ZHAO. *Semidefinite Programming for Assignment and Partitioning Problems*. PhD thesis, University of Waterloo, 1996.
URL: ftp://orion.uwaterloo.ca/pub/henry/software/qap.d/zhaophdthesis.ps.gz.

Note that a comprehensive bibliography is available over WWW with URL:
http://orion.uwaterloo.ca/~hwolkowi/henry/reports/psd.bib.gz