SOLVING SECOND ORDER CONE PROGRAMMING VIA A REDUCED AUGMENTED SYSTEM APPROACH

ZHI CAI * AND KIM-CHUAN TOH †

To the memory of Jos Sturm

Abstract. The standard Schur complement equation based implementation of interior-point methods for second order cone programming may encounter stability problems in the computation of search directions, and as a consequence, accurate approximate optimal solutions are sometimes not attainable. Based on the eigenvalue decomposition of the $(1,1)$ block of the augmented equation, a reduced augmented equation approach is proposed to ameliorate the stability problems. Numerical experiments show that the new approach can achieve more accurate approximate optimal solutions than the Schur complement equation based approach.

Key words. second order cone programming, augmented equation, Nesterov-Todd direction, stability

AMS subject classifications. 90C20, 90C22, 90C51, 65K05

1. Introduction . A second order cone programming (SOCP) problem is a linear optimization problem over a cross product of second order convex cones. A wide range of problems can be formulated as SOCP problems; they include linear programming (LP) problems, convex quadratically constrained quadratic programming problems, filter design problems [5, 20], and problems arising from limit analysis of collapses of solid bodies [6]. An extensive list of applications problems that can be formulated as SOCPs can be found in [14]. For a comprehensive introduction to SOCP, we refer the reader to the paper by Alizadeh and Goldfarb [1].

SOCP itself is a subclass of semidefinite programming (SDP). In theory, SOCP problems can be solved as SDP problems. However, it is far more efficient computationally to solve SOCP problems directly. A few interior-point methods (IPMs) have been developed to solve SOCPs directly [3, 21, 25]. But these IPMs sometimes fail to deliver solutions with satisfactory accuracy. The main objective of this paper is to propose a method that can solve an SOCP to high accuracy, but with comparable or moderately higher cost than the standard IPMs employing the Schur complement equation (SCE) approach. We note that global polynomial convergence results for IPMs for SOCP can be found in [15] and the references therein.

Given a column vector x_i, we will write it as $x_i = [x_i^0\,;\,\bar{x}_i]$ with x_i^0 being the first component and $\bar{x}_i$ consisting of the remaining components. Given square matrices P, Q, the notation $[P\,;\,Q]$ means that Q is appended to the last row of P; and $\mathrm{diag}(P,Q)$ denotes the block diagonal matrix with P, Q as its diagonal blocks. Throughout this paper, $\|\cdot\|$ denotes the matrix 2-norm or vector 2-norm, unless otherwise specified. For a given matrix M, we let $\lambda_{\max}(M)$ and $\lambda_{\min}(M)$ be the largest and smallest eigenvalues of M in magnitude, respectively. The condition number of a matrix M (not necessarily square) is the number $\kappa(M) = \sigma_{\max}(M)/\sigma_{\min}(M)$, where $\sigma_{\max}(M)$ and $\sigma_{\min}(M)$ are the largest and smallest singular values of M, respectively. For a matrix M^μ which depends on a positive parameter μ, the notation

*High Performance Computing for Engineered Systems (HPCES), Singapore-MIT Alliance, 4 Engineering Drive 3, Singapore 117576. (smap0035@nus.edu.sg).

†Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543, Singapore; Singapore-MIT Alliance, 4 Engineering Drive 3, Singapore 117576. (mattohkc@math.nus.edu.sg).

$\|M^\mu\| = O(\mu)$ ($\|M^\mu\| = \Omega(\mu)$) means that there is a positive constant c such that $\|M^\mu\| \le c\mu$ ($\|M^\mu\| \ge c\mu$) as $\mu \downarrow 0$; and $\|M^\mu\| = \Theta(\mu)$ means that there are positive constants c_1, c_2 such that $c_1\mu \le \|M^\mu\| \le c_2\mu$ as $\mu \downarrow 0$. More generally, for a function K^μ depending on a positive parameter μ, the notation $\|M^\mu\| = \Theta(1)K^\mu$ means that there are positive constants c_1, c_2 such that $c_1K^\mu \le \|M^\mu\| \le c_2K^\mu$ for all μ sufficiently small.

Consider the following standard primal and dual SOCP problems:

$$\begin{array}{ll} \text{(P)} & \min\{\sum_{i=1}^N c_i^T x_i \,:\, \sum_{i=1}^N A_i x_i \,=\, b,\ x_i \succeq 0,\ i = 1, \ldots, N\} \\ \text{(D)} & \max\{b^T y \,:\, A_i^T y + z_i \,=\, c_i,\ z_i \succeq 0,\ i = 1, \ldots, N\}, \end{array} \tag{1.1}$$

where $A_i \in \mathbb{R}^{m\times n_i}$, $c_i, x_i, z_i \in \mathbb{R}^{n_i}$, $i = 1, \ldots, N$, and $y \in \mathbb{R}^m$. The constraint $x_i \succeq 0$ is a second order cone constraint defined by $x_i^0 \ge \|\bar{x}_i\|$. In particular, if the cone dimension n_i is 1, then the constraint is simply the standard non-negativity constraint $x_i \ge 0$, and such a variable is called a linear variable. For convenience, we define

$$A \,=\, [A_1 \;\; A_2 \;\; \cdots \;\; A_N], \quad c \,=\, [c_1 \,;\, c_2 \,;\, \cdots \,;\, c_N],$$

$$x \,=\, [x_1 \,;\, x_2 \,;\, \cdots \,;\, x_N], \quad z \,=\, [z_1 \,;\, z_2 \,;\, \cdots \,;\, z_N], \quad n \,=\, \sum_{i=1}^N n_i.$$

The notation $x \succeq 0$ ($x \succ 0$) means that each x_i is in (the interior of) the ith second order cone.

In this paper, we will assume that A has full row rank, and that (P) and (D) in (1.1) are strictly feasible. Under these assumptions, the solutions to the perturbed KKT conditions of (1.1) form a path (known as the central path) in the interior of the primal-dual feasible region. At each iteration of an IPM, the Newton equation associated with the perturbed KKT conditions needs to be solved. By performing block eliminations, one can either solve a system of linear equations of size $m + n$ or one of size m. These linear systems are known as the augmented equation and the Schur complement equation (SCE), respectively. The SCE has the obvious advantage of being smaller in size as well as being symmetric positive definite. Currently, most implementations of IPMs [3, 21, 25] are based on solving the SCE. However, as we shall see in Section 3, the SCE can be severely ill-conditioned when the barrier parameter is close to 0. This typically causes numerical difficulties and imposes a limit on how accurately one can solve an SOCP problem.

In the case of LP, the ill-conditioning of the augmented equation was analyzed by Wright [28, 29]. Under certain assumptions including nondegeneracy, the computed search direction from the augmented equation is shown to be sufficiently accurate for the IPM to converge to high accuracy. The structure of the ill-conditioning of the SCE arising from LP was analyzed in [13]. A stabilization method based on performing Gaussian elimination with a certain pivoting order was also proposed to transform the SCE into a better conditioned linear system of equations.

In nonlinear conic programming, however, the ill-conditioning of the augmented equation and SCE are much more complicated than that in LP. The potential numerical difficulties posed by the ill-conditioned SCE in SOCP were recognized by developers of solvers for SOCP such as [3, 4], [23], and [25]. It was also recognized by Goldfarb and Scheinberg [9] and that motivated them to propose and analyze a

product-form Cholesky factorization for the Schur complement matrix. Subsequently, Sturm [23] implemented the product-form Cholesky factorization [9] in his very popular code SeDuMi to solve the SCE arising at each iteration of a homogeneous self-dual (HSD) IPM. SeDuMi also employed sophisticated techniques to minimize numerical cancelations when computing the SCE and its factorization [23]. These sophisticated techniques typically greatly improve the stability of the SCE approach. However, for certain extreme cases, they do not entirely ameliorate the numerical difficulties caused by the inherently ill-conditioned SCE; see Section 4.

The IPM code SeDuMi differs from standard infeasible interior-point methods in that it solves the homogeneous self-dual embedding model. A natural question to ask is whether SeDuMi's unusually good performance arises from the inherent structure of the HSD model itself or from the sophisticated numerical techniques it uses in solving the SCE (or both). For a certain class of SOCPs with no strictly feasible primal/dual points, we show numerically in Section 4 that SeDuMi's superior performance can be explained by the structure of the HSD model itself. For some SOCPs with strictly feasible points, we shall also see in Section 4 that the sophisticated numerical techniques sometimes may offer only limited improvement in the attainable accuracy when compared to simpler techniques used to solve the SCE.

Herein we propose a method to compute the search directions based on a reduced augmented equation (RAE). This RAE is derived by applying block row operations to the augmented equation, together with appropriate partitioning of the eigen-space of its (1,1) block. The RAE is generally much smaller in size compared to the original augmented equation. By their construction, RAE-based IPMs are computationally more expensive than SCE-based IPMs. Fortunately, numerical experiments show that if sparsity in the SOCP data is properly preserved when forming the RAE, it can generally be solved rather efficiently by a judicious choice of a symmetric indefinite system solver.

The RAE-based IPMs are superior to SCE-based IPMs in that the former can usually deliver approximate optimal solutions that are much more accurate than the latter before numerical difficulties are encountered. For example, for the `schedxxx` SOCP problems selected from the DIMACS library [17], our RAE-based IPMs are able to obtain accuracies of 10^{-9} or better, while the SCE-based IPMs (SDPT3 version 3.1 and SeDuMi) can only obtain accuracies of 10^{-3} or 10^{-4} in some cases.

The paper is organized as follows. In Section 2, we introduce the augmented and Schur complement equations. In Section 3, we analyse the conditioning and the growth in the norm of the Schur complement matrix. We also discuss how the latter affects the primal infeasibility as the interior-point iterates approach optimality. In Section 4, we present numerical results obtained from two different SCE-based primal-dual IPMs. In Section 5, we derive the RAE. The conditioning of the reduced augmented matrix is analysed in Section 6. In Section 7, we discuss major computational issues for efficiently solving the RAE. Numerical results for an RAE-based IPM are presented in Section 8. We conclude the paper in Section 9.

2. The augmented and Schur complement equations. In this section, we present the linear systems that need to be solved to compute the search direction at each IPM iteration.

For x_i in a second order cone, we define

$$\boldsymbol{aw}(x_i) \;=\; \begin{bmatrix} x_i^0 & \bar{x}_i^T \\ \bar{x}_i & x_i^0 I \end{bmatrix}, \qquad \gamma(x_i) \;=\; \sqrt{(x_i^0)^2 - \|\bar{x}_i\|^2}. \tag{2.1}$$

For a given barrier parameter ν, the perturbed KKT conditions of (1.1) in matrix form are:

$$Ax \;=\; b, \quad A^T y + z \;=\; c, \quad \boldsymbol{aw}(x)\,\boldsymbol{aw}(z)e^0 \;=\; \nu e^0, \tag{2.2}$$

where $e^0 = [e_1\,;\, e_2\,;\, \cdots\,;\, e_N]$, with e_i being the first unit vector in $\mathbb{R}^{n_i}$. The matrix $\boldsymbol{aw}(x) = \mathrm{diag}(\boldsymbol{aw}(x_1), \cdots, \boldsymbol{aw}(x_N))$ is a block diagonal matrix with $\boldsymbol{aw}(x_1), \cdots, \boldsymbol{aw}(x_N)$ as its diagonal blocks. The matrix $\boldsymbol{aw}(z)$ is defined similarly.

For reasons of computational efficiency that we will explain later, in most IPM implementations for SOCP, a block diagonal scaling matrix is usually applied to the last equation in (2.2). Here, we apply the Nesterov-Todd (NT) scaling matrix [25] to produce the following equation:

$$\boldsymbol{aw}(Fx)\,\boldsymbol{aw}(F^{-1}z)e^0 = \nu e^0, \tag{2.3}$$

where $F = \mathrm{diag}(F_1, \cdots, F_N)$ is chosen such that $Fx \;=\; F^{-1}z \;=:\; v$. For details on the conditions that F must satisfy and on other scaling matrices, we refer the reader to [15]. Let

$$f_i \;=\; \begin{bmatrix} f_i^0 \\ \bar{f}_i \end{bmatrix} \;:=\; \frac{1}{\sqrt{2\Big(\gamma(x_i)\gamma(z_i) + x_i^T z_i\Big)}} \begin{bmatrix} \dfrac{1}{\omega_i} z_i^0 + \omega_i x_i^0 \\ \dfrac{1}{\omega_i}\bar{z}_i - \omega_i \bar{x}_i \end{bmatrix},$$

where $\omega_i = \sqrt{\gamma(z_i)/\gamma(x_i)}$. (Note that $\gamma(f_i) = 1$.) The precise form of F_i is given by

$$F_i \;=\; \omega_i \begin{bmatrix} f_i^0 & \bar{f}_i^T \\ \bar{f}_i & I + \frac{\bar{f}_i \bar{f}_i^T}{1+f_i^0} \end{bmatrix}. \tag{2.4}$$

Let $\mu = x^T z/N$ be the normalized complementarity gap. The Newton equation associated with the perturbed KKT conditions (2.2) with NT scaling is given by

$$A\Delta x \;=\; r_p, \quad A^T\Delta y + \Delta z \;=\; r_d, \quad VF\Delta x + VF^{-1}\Delta z \;=\; r_c, \tag{2.5}$$

where $V = \boldsymbol{aw}(v)$, $r_p = b - Ax$, $r_d = c - z - A^T y$, $r_c = \sigma\mu e^0 - Vv$. Note that we have chosen ν to be $\nu = \sigma\mu$ for some parameter $\sigma \in (0, 1)$.

The solution $(\Delta x, \Delta y, \Delta z)$ of the Newton equation (2.5) is referred to as the search direction. At each IPM iteration, solving (2.5) for the search direction is computationally the most expensive step. Observe that by eliminating Δz, the Newton equation (2.5) reduces to the so-called augmented equation:

$$\begin{bmatrix} -F^2 & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \;=\; \begin{bmatrix} r_x \\ r_p \end{bmatrix}, \tag{2.6}$$

where $r_x = r_d - FV^{-1}r_c$. The augmented equation can further be reduced in size by eliminating Δx in (2.6) to produce the SCE:

$$\underbrace{AF^{-2}A^T}_{M}\Delta y \;=\; r_y \;:=\; r_p + AF^{-2}r_x \;=\; r_p + AF^{-2}r_d - AF^{-1}V^{-1}r_c. \tag{2.7}$$

The coefficient matrix M in (2.7) is known as the Schur complement matrix. It is symmetric positive definite if $x, z \succ 0$. The search direction corresponding to (2.5)

always exists as long as $x, z \succ 0$. Note that if the scaling matrix F is not applied to the last equation in (2.2), the corresponding Schur complement matrix would be $A\,\boldsymbol{aw}(z)^{-1}\boldsymbol{aw}(x)\,A^T$, which is a nonsymmetric matrix. This nonsymmetric coefficient matrix is not guaranteed to be nonsingular even when $x, z \succ 0$. Moreover, computing its sparse LU factorization is usually much more expensive than computing the sparse Cholesky factorization of M.

In their simplest form, most current implementations of IPMs compute the search direction $(\Delta x, \Delta y, \Delta z)$ based on the SCE (2.7) via the following procedure.

Simplified SCE approach:

(i) Compute the Schur complement matrix M and the vector r_y;

(ii) Compute the Cholesky or sparse Cholesky factor of M;

(iii) Compute Δy by solving 2 triangular linear systems involving the Cholesky factor;

(iv) Compute Δz via $\Delta z = r_d - A^T\Delta y$; and Δx via $\Delta x = F^{-2}(A^T\Delta y - r_x)$.

We should note that various heuristics to improve the numerical stability of the simplified SCE approach are usually incorporated in the actual implementations. We will describe in Section 4 variants of the above approach implemented in two publicly available SOCP solvers, SDPT3, version 3.1 [26] and SeDuMi, version 1.05 [22].

The SCE is preferred because it is usually a much smaller system compared to the augmented or Newton equations. Furthermore, the Schur complement matrix has the highly desirable property of being symmetric positive definite. (In contrast, the coefficient matrix in (2.6) is symmetric indefinite while that of (2.5) is nonsymmetric.) Consequently, the SCE can be solved very efficiently via Cholesky or sparse Cholesky factorization of M. We should mention that there are highly efficient and machine optimized sparse Cholesky codes readily available in the public domain, the prime example being the sparse Cholesky codes of Ng and Peyton [16]. Comparatively, the state-of-the-art LDL^T factorization codes (an example being the MA47 codes of Duff and Reid [18]) for a sparse symmetric indefinite matrix available in the public domain are less advanced.

3. Conditioning of M and the deterioration of primal infeasibility. Despite the advantages of the SCE approach described in the last section, the SCE is however, generally severely ill-conditioned when the iterates (x, y, z) approach optimality, and this typically causes numerical difficulties. The most common numerical difficulty one may encounter in practice is that the Schur complement matrix M is numerically indefinite, although in exact arithmetic M is positive definite. Furthermore, the computed solution Δy from (2.7) may also be very inaccurate in that the residual norm $\|r_y - M\Delta y\|$ is much larger than the machine epsilon, and this typically causes the IPM to stall.

In this section, we will analyze the relationship between the norm $\|M\|$, the residual norm $\|r_y - M\Delta y\|$ of the computed solution Δy, and the primal infeasibility $\|r_p\|$, as the interior-point iterates approach optimality.

3.1. Eigenvalue decomposition of F^2. To analyze the norm $\|M\|$ and the conditioning of M, we need to know the eigenvalue decomposition of F^2. Recall that $F = \mathrm{diag}(F_1, \cdots, F_N)$. Thus it suffices to find the eigenvalue decomposition of F_i^2, where F_i is given in (2.4). By noting that for cones of dimensions $n_i \geq 2$, F_i^2 can be written as $F_i^2 = \omega_i^2\left(I + 2(f_i f_i^T - e_i e_i^T)\right)$, the eigenvalue decomposition of F_i^2 can readily be found. (The case where $n_i = 1$ is easy, and $F_i^2 = z_i/x_i$.) Without going

through the algebraic details, the eigenvalue decomposition of F_i^2 is given by

$$F_i^2 = Q_i \Lambda_i Q_i^T, \qquad Q_i \;=\; \begin{bmatrix} -\frac{1}{\sqrt{2}} & +\frac{1}{\sqrt{2}} & 0 & \cdots & 0 \\ \frac{1}{\sqrt{2}}\, g_i & \frac{1}{\sqrt{2}}\, g_i & q_i^3 & \cdots & q_i^{n_i} \end{bmatrix}, \tag{3.1}$$

with $\Lambda_i = \omega_i^2 \,\mathrm{diag}\Big((f_i^0 - \|\bar{f}_i\|)^2,\, (f_i^0 + \|\bar{f}_i\|)^2,\, 1, \cdots, 1\Big)$, and

$$g_i \;:=\; [g_i^0\,;\, \bar{g}_i] \;=\; \bar{f}_i / \|\bar{f}_i\| \;\in\; I\!R^{n_i - 1}. \tag{3.2}$$

Notice that since $\gamma(f_i) = 1$, the first eigenvalue is the smallest and the second is the largest. The set $\{q_i^3, \cdots, q_i^{n_i}\}$ is an orthonormal basis of the subspace $\{u \in I\!R^{n_i-1} : u^T g_i = 0\}$. To construct such an orthonormal basis, one may first construct the $(n_i - 1) \times (n_i - 1)$ Householder matrix H_i [10] associated with the vector g_i, then the last $n_i - 2$ columns of H_i is such an orthonormal basis. The precise form of H_i will be given later in Section 6.

3.2. Analysis of $\|M\|$ and the conditioning of M. Recall that M is dependent on the normalized complementarity gap μ. Here we analyze how fast the norm $\|M\|$ and the condition number of M will grow when $\mu \downarrow 0$, i.e., when the interior-point iterates approach an optimal solution (x^*, y^*, z^*). To simplify the analysis, we will assume that strict complementarity holds at the optimal solution. Unless otherwise stated, we assume that $n_i \geq 2$ in this subsection.

Strict complementarity [2] implies that for each pair of the optimal primal and dual solutions, x_i^* and z_i^*, we have $\gamma(x_i^*) + \|z_i^*\|$ and $\gamma(z_i^*) + \|x_i^*\|$ both positive. In other words, (a) either $\gamma(x_i^*) = 0$ or $z_i^* = 0$, but not both; and (b) either $\gamma(z_i^*) = 0$ or $x_i^* = 0$, but not both. Under the strict complementarity assumption, we have the following three types of eigenvalue structures (following the classification in [9]) for F_i^2 when $\boldsymbol{aw}(x_i)\boldsymbol{aw}(z_i)e_i = \mu e_i$ and μ is small. Note that $x_i^T z_i = \mu$.

- Type 1 solution: $x_i^* \succ 0$, $z_i^* = 0$. In this case, $\gamma(x_i) = \Theta(1)$, $\gamma(z_i) = \Theta(\mu)$, and $\omega_i = \Theta(\sqrt{\mu})$. Also, $f_i^0, \|\bar{f}_i\| = \Theta(1)$, implying that all the eigenvalues of F_i^2 are $\Theta(\mu)$.
- Type 2 solution: $x_i^* = 0$, $z_i^* \succ 0$. In this case, $\gamma(x_i) = \Theta(\mu)$, $\gamma(z_i) = \Theta(1)$, and $\omega_i = \Theta(1/\sqrt{\mu})$. Also, $f_i^0, \|\bar{f}_i\| = \Theta(1)$, implying that all the eigenvalues of F_i^2 are $\Theta(1/\mu)$.
- Type 3 solution: $\gamma(x_i^*) = 0$, $\gamma(z_i^*) = 0$, $x_i^*, z_i^* \neq 0$. In this case, $\gamma(x_i), \gamma(z_i) = \Theta(\sqrt{\mu})$, and $\omega_i = \Theta(1)$. This implies that $f_i^0, \|\bar{f}_i\| = \Theta(1/\sqrt{\mu})$. Thus the largest eigenvalue of F_i^2 is $\Theta(1/\mu)$ and by the fact that $\gamma(f_i) = 1$, the smallest eigenvalue of F_i^2 is $\Theta(\mu)$. The rest of the eigenvalues are $\Theta(1)$.

Let D be the diagonal matrix consisting of the eigenvalues of F^2 sorted in ascending order. Then we have $F^2 = QDQ^T$, where the columns of Q are the sorted eigenvectors of F^2. Let D be partitioned into $D = \mathrm{diag}(D_1, D_{2a}, D_{2b})$ such that $\mathrm{diag}(D_1)$ consists of all the small eigenvalues of F^2 of order $\Theta(\mu)$, and $\mathrm{diag}(D_{2a})$, $\mathrm{diag}(D_{2b})$ consist of the remaining eigenvalues of order $\Theta(1)$ and $\Theta(1/\mu)$, respectively. Note that the three groups of eigenvalues need not be all present. We also partition the matrix Q as $Q = [Q^{(1)}\ ,\ Q^{(2a)}\ ,\ Q^{(2b)}]$. Then $\widetilde{A} := AQ$ is partitioned as $\widetilde{A} = [\widetilde{A}_1\ ,\ \widetilde{A}_{2a}\ ,\ \widetilde{A}_{2b}] = [AQ^{(1)}\ ,\ AQ^{(2a)}\ ,\ AQ^{(2b)}]$. With the above partitions, we can express M as

$$M \;=\; \underbrace{\widetilde{A}_1 D_1^{-1} \widetilde{A}_1^T}_{M_1} + \underbrace{\widetilde{A}_{2a} D_{2a}^{-1} \widetilde{A}_{2a}^T}_{M_{2a}} + \underbrace{\widetilde{A}_{2b} D_{2b}^{-1} \widetilde{A}_{2b}^T}_{M_{2b}}. \tag{3.3}$$

LEMMA 3.1. *(a) For the matrices M_1, M_{2a} and M_{2b} in (3.3), we have*

$$\|M_1\| = \Theta(1/\mu)\|\widetilde{A}_1\|^2, \quad \|M_{2a}\| = \Theta(\|\widetilde{A}_{2a}\|^2), \quad \|M_{2b}\| = \Theta(\mu)\|\widetilde{A}_{2b}\|^2.$$

(b) Suppose there are Type 1 or Type 3 solutions so that $\widetilde{A}_1$ is not a null matrix. Then the following statments hold. [i] $\|M\| = \Theta(1/\mu)\|\widetilde{A}_1\|^2$. [ii] If $\widetilde{A}_1$ does not have full row rank, then $\|M^{-1}\| = \Big(O(1)\|\widetilde{A}_{2a}\|^2 + O(\mu)\|\widetilde{A}_{2b}\|^2\Big)^{-1} = \Omega(1)\Big(\|\widetilde{A}_{2a}\|^2 + \|\widetilde{A}_{2b}\|^2\Big)^{-1}$.

Proof. (a) We shall only prove the first result since the other two can be proved similarly. By the definition of D_1, there are positive constants c_1, c_2 such that $(c_1/\mu)I \preceq D_1^{-1} \preceq (c_2/\mu)I$. Thus $(c_1/\mu)\widetilde{A}_1\widetilde{A}_1^T \preceq M_1 \preceq (c_2/\mu)\widetilde{A}_1\widetilde{A}_1^T$. This implies that $(c_1/\mu)\|\widetilde{A}_1\widetilde{A}_1^T\| \le \|M_1\| \le (c_2/\mu)\|\widetilde{A}_1\widetilde{A}_1^T\|$, and the required result follows by noting that $\|\widetilde{A}_1\widetilde{A}_1^T\| = \|\widetilde{A}_1\|^2$.

(b)[i] From (3.3), it is clear that $\|M\| = \Theta(1/\mu)\|\widetilde{A}_1\|^2$. (b)[ii] If $\widetilde{A}_1$ does not have full row rank, then the null space $\mathcal{N}(\widetilde{A}_1^T)$ is non-trivial. Let U be a matrix whose columns form an orthonormal basis of $\mathcal{N}(\widetilde{A}_1^T)$ and $W := U^T(M_{2a} + M_{2b})U$. Since $\widetilde{A}_1^T U = 0$, we have $U^T M U = U^T(M_{2a} + M_{2b})U = W$. By the Courant-Fischer Theorem, it is clear that $\lambda_{\min}(M) \le \lambda_{\min}(W)$. Thus $\|M^{-1}\| = 1/\lambda_{\min}(M) \ge 1/\lambda_{\min}(W) = \|W^{-1}\| \ge 1/\|W\|$. Since $\|W\| \le \|M_{2a}\| + \|M_{2b}\| = O(1)\|\widetilde{A}_{2a}\|^2 + O(\mu)\|\widetilde{A}_{2b}\|^2 = O(1)(\|\widetilde{A}_{2a}\|^2 + \|\widetilde{A}_{2b}\|^2)$, the required result follows. □

REMARK 3.1. *(a) Lemma 3.1 implies that the growth in $\|M\|$ is caused by F^2 having small eigenvalues of order $\Theta(\mu)$.*

(b) If $\widetilde{A}_1$ is present and does not have full row rank, then $\kappa(M) = \Omega(1/\mu)\|\widetilde{A}_1\|^2\Big(\|\widetilde{A}_{2a}\|^2 + \|\widetilde{A}_{2b}\|^2\Big)^{-1}$. On the other hand, if $\widetilde{A}_1$ has full row rank (which implies that the number of eigenvalues of F^2 of order $\Theta(\mu)$ is at least m), then $\kappa(M) = \Theta(1)\kappa(\widetilde{A}_1)^2$.

(c) If there are only Type 2 solutions (thus $x^* = 0$ and $z \succ 0$), then $M = AQDQ^TA^T$ with $D = \Theta(\mu)$. In this case, we have $\kappa(M) = \Theta(1)\kappa(A)^2$.

Based on the results in [2], we have the following theorem concerning the rank of $\widetilde{A}_1$ and $\widetilde{A}_{2a}$. We refer the reader to [2] for the definitions of primal and dual degeneracies.

THEOREM 3.2. *Suppose that (x^*, y^*, z^*) satisfies strict complementarity. If the primal optimal solution x^* is primal nondegenerate, then $\Big[\widetilde{A}_1\ ,\ \widetilde{A}_{2a}\Big]$ has full row rank when μ is small. If the dual optimal solution (y^*, z^*) is dual nondegenerate, then $\widetilde{A}_1$ has full column rank when μ is small.*

Proof. The result follows from Theorems 20 and 21 in [1]. □

REMARK 3.2. *We should emphasize that while Theorem 3.2 said that $[\widetilde{A}_1,\ \widetilde{A}_{2a}]$ has full row rank when the optimal solution is strictly complementary and primal and dual nondegenerate, the matrix $\widetilde{A}_1$, however, does not necessarily have full row rank under the same condition. In the event that $\widetilde{A}_1$ is present and does not have full row rank, Remark 3.1 (b) said that M is ill-conditioned with $\kappa(M) = \Omega(1/\mu)$. Thus even if the optimal solution is strict complementary and primal and dual nondegenerate, M does not necessarily have bounded condition number when $\mu \downarrow 0$. In contrast, for a LP problem (for which all the cones have dimensions $n_i = 1$ and $\widetilde{A}_{2a}$ is absent), primal and dual nondegeneracy ensure that $\widetilde{A}_1$ has full column and row rank, and as a result, M has bounded condition number when $\mu \downarrow 0$.*

3.3. Analysis of the deterioration of primal infeasibility. Although Cholesky factorization is stable for any symmetric positive definite matrix, the conditioning of the matrix may still affect the accuracy of the computed solution of the SCE. It is a common phenomenon that for SOCP, the accuracy of the computed search direction deteriorates as μ decreases due to an increasingly ill-conditioned M. As a result of this loss of accuracy in the computed solution, the primal infeasibility $\|r_p\|$ typically increases or stagnates when the IPM iterates approach optimality.

With the analysis of $\|M\|$ given in the last subsection, we will now analyze why the primal infeasibility may deteriorate or stagnate as interior-point iterations progress.

LEMMA 3.3. *Suppose at the kth iteration, the residual vector in solving the SCE (2.7) is* $\xi = r_y - M\Delta y$. *Assuming that* Δx *is computed exactly via the equation* $\Delta x = F^{-2}(A^T\Delta y - r_x)$, *then the primal infeasibility for the next iterate* $x^+ = x + \alpha\Delta x$, $\alpha \in [0,1]$, *is given by*

$$r_p^+ := b - Ax^+ \;=\; (1-\alpha)r_p \;+\; \alpha\xi.$$

Proof. We have $r_p^+ = (1-\alpha)r_p + \alpha(r_p - A\Delta x)$. Now $A\Delta x = AF^{-2}(A^T\Delta y - r_x) = M\Delta y - r_y + r_p$, thus $r_p - A\Delta x = r_y - M\Delta y = \xi$ and the lemma is proved. ∎

REMARK 3.3. *(a) In Lemma 3.3, we assume for simplicity that the component direction* Δx *is computed exactly. In finite precision arithmetic, errors will be introduced in the computation of* Δx *and that will also worsen the primal infeasibility* r_p^+ *of the next iterate besides* $\|\xi\|$.

(b) Observe that if the SCE is solved exactly, i.e., $\xi = 0$, *then* $\|r_p^+\| = (1-\alpha)\|r_p\|$, *and the primal infeasibility should decrease monotonically.*

Lemma 3.3 implies that if the SCE is not solved to sufficient accuracy, then the inaccurate residual vector ξ may worsen the primal infeasibility of the next iterate. By standard perturbation error analysis, the worst-case residual norm of $\|\xi\|$ can be shown to be proportional to $\|M\|\|\Delta y\|$ times the machine epsilon u. The precise statement is given in the next lemma.

LEMMA 3.4. *Let* u *be the machine epsilon. Given a symmetric positive definite matrix* $B \in \mathbb{R}^{n\times n}$ *with* $(n+1)^2u \le 1/3$, *if Cholesky factorization is applied to* B *to solve the linear system* $Bx = b$ *to produce a computed solution* $\widehat{x}$, *then* $(B+\Delta B)\widehat{x} = b$, *for some* ΔB *with* $\|\Delta B\|$ *satisfying the following inequality:* $\|\Delta B\widehat{x}\| \le 3(n+1)^2u\|B\|\|\widehat{x}\|$. *Thus*

$$\|b - B\widehat{x}\| \;=\; \|\Delta B\widehat{x}\| \;=\; O(n^2)u\|B\|\|\widehat{x}\|.$$

Proof. The lemma follows straightforwardly from Theorem 10.3 and 10.4, and their extensions in [12]. ∎

REMARK 3.4. *Lemma 3.4 implies that if* $\|B\|\|\hat{x}\|$ *is large, then in the worst case scenario, the residual norm* $\|b - B\widehat{x}\|$ *is expected to be proportionately large.*

By Lemma 3.3 and the application of Lemma 3.4 to the SCE, we expect in the worst case the primal infeasibility $\|r_p\|$ to grow to some extent that is proportional to $\|M\|\|\Delta y\|u$. We end this section by presenting a numerical example to illustrate the relation between $\|r_p\|$ and $\|M\|\|\Delta y\|u$ in the last few iterations of an SCE-based IPM when solving the SOCP problems `rand200_800_1` and `sched_50_50_orig` (described in Section 4).

The IPM we use is the primal-dual path-following method with Mehrotra predictor-corrector implemented in the MATLAB software SDPT3, version 3.1 [26]. But we

should mention that to be consistent with the analysis presented in this section, the search directions are computed based on the simplified SCE approach presented in Section 2, not the more sophisticated variant implemented in SDPT3.

Table 3.1 shows the norms $\|M\|$, $\|M^{-1}\|$, $\|r_y - M\Delta y\|$ when solving the SCE (2.7). For this problem, $\|M\|$ and (hence $\kappa(M)$) grows like $\Theta(1/\mu)$ because its optimal solutions x_i^*, z_i^* are all of Type 3. The fifth and sixth columns in the table show that the residual norm in solving the SCE and $\|r_p\|$ deteriorate as $\|M\|$ increases. This is consistent with the conclusions of Lemmas 3.3 and 3.4. The last column further shows that $\|r_p\|$ increases proportionately to $\|M\|\|\Delta y\|u$, where the machine epsilon u is approximately 2.2×10^{-16}.

Figure 3.1 illustrates the phenomenon graphically for the SOCP problems `rand200_800_1` and `sched_50_50_orig`. The curves plotted correspond to the relative duality gap (`relgap`), and the relative primal and dual infeasibility (`p-inf` and `d-inf`), defined by

$$\texttt{relgap} \;=\; \frac{|c^Tx - b^Ty|}{1 + (|c^Tx| + |b^Ty|)/2}, \quad \texttt{p-inf} \;=\; \frac{\|r_p\|}{1+\|b\|}, \quad \texttt{d-inf} \;=\; \frac{\|r_d\|}{1+\|c\|}. \tag{3.4}$$

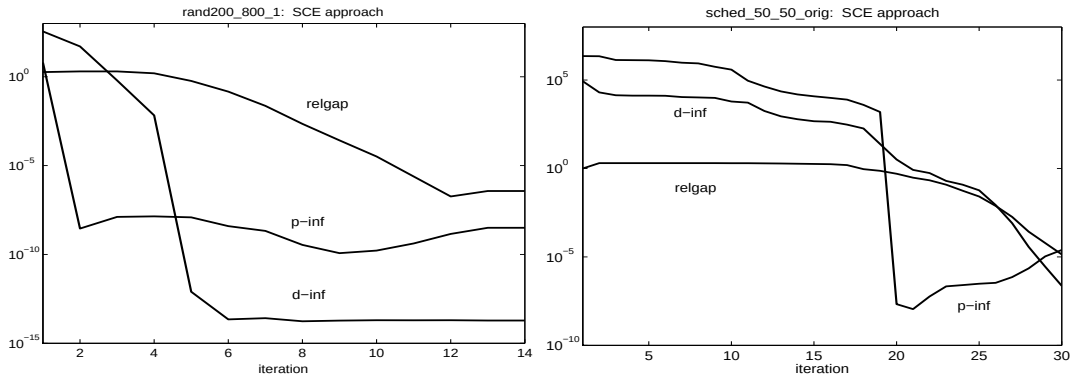

FIG. 3.1. *Convergence history of the SOCPs problems* `rand200_800_1` *and* `sched_50_50_orig` *when solved by the SCE-based IPM in SDPT3, version 3.1. Notice that the relative primal infeasibility* `p-inf` *deteriorates as interior-point iterates approach optimality, while* `relgap` *may stagnate.*

4. Computational results of two SCE-based IPMs on solving some SOCP problems. Here we present numerical results for the SCE-based IPMs implemented in the public domain solvers, SDPT3, version 3.1 [26], and SeDuMi, version 1.05 [22]. In this paper, all the numerical results are obtained in MATLAB 6.5 from a Pentium IV 2.4GHz PC with 1G RAM running a Linux operating system.

Before we analyze the performance of the SCE-based IPMs implemented in SDPT3 and SeDuMi, we must describe the methods employed to solve the SCE in both solvers. The IPM in SDPT3 is an infeasible path-following method that attempts to solve the central path equation based on (2.2), even if this path does not exist. It solves the resulting SCE at each IPM iteration as follows. First it computes the Cholesky or sparse Cholesky factor of the Schur complement matrix M. Then the computed Cholesky factor is used to construct a preconditioner within a preconditioned symmetric quasi-minimal residual (PSQMR) Krylov subspace iterative solver employed to solve the SCE for Δy. The computations of Δz and Δx are the same as in the simplified SCE approach presented in Section 2.

TABLE 3.1
The norm of the Schur complement matrix and $\|r_p\|$ associated with the last few IPM iterations for solving the SOCP problems `rand_200_800_1` *and* `sched_50_50_orig`.

Iter	$\|M\|$	$\|M^{-1}\|$	$\mu := x^T z/N$	$\|\Delta y\|$	$\|r_y - M\Delta y\|$	$\|r_p\|$	$\frac{\|r_p\|}{\|M\|\|\Delta y\|u}$
`rand_200_800_1`							
9	1.8e+13	4.9e+02	9.2e-07	2.0e-02	8.7e-06	7.1e-06	8.9e-02
10	2.9e+14	2.3e+02	1.2e-07	2.7e-03	2.5e-05	1.8e-05	1.0e-01
11	3.8e+15	4.0e+01	1.1e-08	5.1e-04	4.9e-05	6.1e-05	1.4e-01
12	1.9e+17	6.8e+00	1.2e-09	9.4e-05	2.2e-04	1.4e-04	3.5e-02
13	1.2e+18	3.8e+01	1.8e-10	2.5e-03	3.7e-02	9.0e-04	1.4e-03
`sched_50_50_orig`							
25	5.0e+08	1.5e+04	1.5e-01	3.3e+02	7.5e-09	1.1e-06	3.0e-02
26	3.0e+09	2.1e+04	4.3e-02	2.2e+02	2.1e-07	1.4e-05	9.8e-02
27	2.7e+10	1.7e+04	9.9e-03	4.8e+01	1.9e-07	1.5e-05	5.4e-02
28	4.9e+11	1.9e+04	1.4e-03	1.0e+01	1.5e-06	1.0e-05	9.3e-03
29	5.3e+12	2.1e+04	3.3e-04	2.1e+00	1.2e-05	2.9e-04	1.2e-01

SeDuMi is a very well implemented SCE-based public domain solver for both SOCP and SDP. The IPM in SeDuMi is not based on the central path for the original primal and dual problems (1.1), but that of the homogeneous self-dual (HSD) model of Ye, Todd, and Mizuno [30]. The HSD model has the nice theoretical property that a strictly feasible primal and dual point always exists even if the original problems do not have one, and as a result the central path for the HSD model always exists, which is not necessarily true for the original problems in (1.1). As a consequence of this nice property, its solution set is always bounded. The same cannot be said for the original problems. For a problem that models an unrestricted variable by the difference of 2 nonnegative variables, the solution set for the original primal SOCP (P) is unbounded, and the feasible region of (D) has an empty interior, implying that the primal-dual central path does not exist. The HSD model, on the other hand, does not suffer from these defects. Thus the IPM in SeDuMi will not feel the effect of the unbounded solution set and nonexistence of the central path in the original problems in (1.1), but the effect of the unboundedness of the solution set on the infeasible path-following IPM in SDPT3 can be substantial and it often causes serious numerical difficulties.

The computation of the search direction in SeDuMi is based on the SCE associated with the HSD model. But it employs sophisticated numerical techniques to minimize numerical cancelations in its implementation of the SCE approach [23]. It computes the Schur complement matrix in the scaled space (called the v-space) framework, and transforms back and forth between quantities in the scaled and original spaces. It employs the sparse Cholesky codes adapted from Ng and Peyton [16] to compute the factorization. It also employs the product-form Cholesky factorization [9] to handle dense columns. If the computed Cholesky factor is deemed sufficiently stable, SeDuMi will proceed to compute Δy by solving two triangular linear systems involving the Cholesky factor; otherwise, it will solve the SCE by using the preconditioned conjugate gradient iterative method with a preconditioner constructed from the Cholesky factor. Note that the Cholesky factorization has been shown in [9] to produce stable triangular factors for the Schur complement matrix if the iterates are sufficiently close to the central path and strict complementarity holds at optimality. It is important to note,

however, that using a stable method to solve the SCE does not necessarily imply that the computed direction $(\Delta x, \Delta y, \Delta z)$ based on the SCE approach will produce a small residual norm with respect to the original linear system (2.5); see Theorem 3.2 of [11] for the case of SDP.

We tested the SCE-based IPMs in SDPT3 and SeDuMi on the following set of SOCP problems. The statistics for the test problems are shown in Table 4.1.

(a) The first set consists of 18 SOCPs in the DIMACS library collected by Pataki and Schmieta [17], available at `http://dimacs.rutgers.edu/Challenges/Seventh/Instances/`

(b) The second set consists of 10 SOCPs from the FIR Filter Optimization Toolbox of Scholnik and Coleman, available at `http://www.csee.umbc.edu/~dschol2/opt.html`

(c) The last set consists of 10 randomly generated SOCPs. These random problems `randxxx` are generated to be feasible and dominated by Type 3 solutions. For each problem, the constraint matrix A has the form $V_1 \Sigma V_2^T$, where V_1, V_2 are matrices whose columns are orthonormal, and Σ is a diagonal matrix with random diagonal elements drawn from the standard normal distribution, but a few of the diagonal elements are set to 10^5 to make A moderately ill-conditioned.

In our experiments, we stop the IPM iteration in SDPT3 when any of the following situations are encountered: (1) $\max(\texttt{relgap}, \texttt{p-inf}, \texttt{d-inf}) \leq 10^{-10}$; (2) incurable numerical difficulties (such as the Schur complement matrix being numerically indefinite) occur; (3) `p-inf` has deteriorated to the extent that `p-inf` $>$ `relgap`. SeDuMi also has a similar set of stopping conditions but based on the variables of the HSD model. In SeDuMi, the dual conic constraints are not strictly enforced, thus the measure `d-inf` for SeDuMi is defined to be `d-inf` $= \max(\|r_d\|, \|z^-\|)$, where $\|z^-\|$ measures how much the dual conic constraints are violated. We define

$$\phi \;:=\; \log_{10}(\max\{\texttt{relgap}, \texttt{p-inf}, \texttt{d-inf}\}). \tag{4.1}$$

Table 4.1 shows the numerical results for SDPT3 and SeDuMi on 36 SOCP problems. Observe that the accuracy exponent (ϕ) for many of the problems fall short of the target of -10. For the `sched-xxx` problems, the accuracy exponents attained are especially poor, only -3 or -4 in some cases. We should mention that the results shown in Table 4.1 are not isolated to just the IPMs implemented in SDPT3 or SeDuMi; similar results were also reported in the SCE-based IPM implemented by Andersen et al. [3]. For example, for the problem `sched_50_50_orig`, the IPM in [3] reported the values 0.9 and 0.002 for the maximum violation of certain primal bound constraints and the dual constraints, respectively.

From Table 4.1, we have thus seen the performance of SCE-based IPMs for two rather different implementations in SDPT3 and SeDuMi. It is worthwhile to analyze the performance of these implementations to isolate the factor contributing to the good performance in one implementation, but not the other. On the first 10 SOCP problems, `nbxxx`, `nqlxxx` and `qsspxxx` in the DIMACS library, SeDuMi performs much better than the IPM in SDPT3 in terms of accuracy. We hypothesize that SeDuMi is able to obtain accurate approximate optimal solutions for these test problems primarily because of nice theoretical properties (existence of a strictly feasible point, and boundedness of solution set) of the HSD model. These problems contain linear variables that are the results of modeling unrestricted variables as the difference of two nonnegative vectors. Consequently, the resulting primal SOCPs have unbounded solution sets and the feasible regions of the dual SOCPs have empty interior. It should

come as no surprise that the IPM in SDPT3 has trouble solving such a problem to high accuracy since the ill-conditioning in the Schur complement matrix is made worse by the growing norm of the primal linear variables as the iterates approach optimality. On the other hand, for the IPM in SeDuMi, the ill-conditioning of the Schur complement matrix is not amplified since the norm of the primal variables in the HSD model stays bounded.

To verify the above hypothesis, we solve the `nbxxx`, `nqlxxx` and `qsspxxx` problems again in SDPT3, but at each IPM iteration, we trim the growth in the primal linear variables, x^u_+, x^u_-, arising from unrestricted variables x_u using the following heuristic [26]:

$$x^u_+ \;:=\; x^u_+ - 0.8\min(x^u_+, x^u_-), \qquad x^u_- \;:=\; x^u_- - 0.8\min(x^u_+, x^u_-). \tag{4.2}$$

This modification does not change the original variable x^u but it slows down the growth of x^u_+, x^u_-. After these modified vectors have been obtained, we also modify the associated dual linear variables z^u_+, z^u_- as follows if $\mu \le 10^{-4}$:

$$(z^u_+)_i \;:=\; \frac{0.5\mu}{\max(1,(x^u_+)_i)}, \qquad (z^u_-)_i \;:=\; \frac{0.5\mu}{\max(1,(x^u_-)_i)}. \tag{4.3}$$

Such a modification in z^u_+, z^u_- ensures that they approach 0 at the same rate as μ, and thus prevents the dual problem from attaining the equality constraints in (D) prematurely.

The results shown in Table 4.2 supported our hypothesis. Observe that with the heuristic in (4.2) and (4.3) to control the growth of $(x^u_+)_i/(z^u_+)_i$ and $(x^u_-)_i/(z^u_-)_i$, the IPM in SDPT3 can also achieve accurate approximate solutions, just as the IPM based on the HSD model in SeDuMi is able to achieve. It is surprising that such a simple heuristic to control the growth can result in such a dramatic improvement on the achievable accuracy, even though the problems (P) and (D) in (1.1) do not have a strictly feasible point and the corresponding central path does not exist.

On other problems such as `schedxxx`, `firxxx`, and `randxxx`, the performance of SDPT3 and SeDuMi is quite comparable in terms of accuracy attained, although SeDuMi is generally more accurate on the `schedxxx` problems, while SDPT3 performs somewhat better on the `randxxx` problems. On the `firxxx` problems, SDPT3 seems to be more robust whereas SeDuMi runs into numerical difficulties quite early when solving `firL1Linfalph` and `firL2L1alph`.

The empirical evidence of Table 4.1 shows that even though sophisticated numerical techniques used to solve the SCE in SeDuMi can help to achieve better accuracy, sometimes these techniques give limited improvement over simpler techniques employed in SDPT3. On SOCP problems where the two solvers have vastly different performance in terms of accuracy, the difference can be attributed to the inherent IPM models used in the solvers rather than the numerical techniques employed to solve the SCE. The conclusion we may draw here is that the SCE is generally inherently ill-conditioned, and if our wish is to compute the search direction of (2.5) to higher accuracy, a new approach other than the SCE is necessary.

TABLE 4.1

Accuracy attained by 2 SCE-based IPMs for solving SOCP problems. The timings reported are in seconds. A number of the form "1.7-4" means 1.7×10^{-4}*. An entry of the form "793x3" in the "SOC" column means that there are 793 3-dimensional second order cones. The numbers under the "LIN" column are the number of linear variables.*

				SDPT3					SeDuMi			
problem	m	SOC	LIN	ϕ	Time	relgap	p-inf	d-inf	ϕ	Time	relgap	p-inf	d-inf
nb	123	793×3	4	-3.5	6.5	2.2-4	3.3-4	8.3 -9	-11.3	12.8	6.5-13	4.8-12	2.8-15
nb-L1	915	793×3	797	-4.9	13.0	7.1-7	1.4-5	9.6-11	-12.2	14.7	6.2-13	1.2-14	1.5-14
nb-L2	123	1×1677 ; 838×3	4	-5.5	10.9	5.1-8	3.1-6	8.9-12	-9.3	33.9	5.4-10	3.1-12	9.7-12
nb-L2-bessel	123	1×123 ; 838×3	4	-6.4	6.2	3.1-7	4.3-7	8.8-11	-10.5	20.1	3.3-11	8.0-14	1.7-13
nql30	3680	900×3	3602	-4.8	5.1	1.7-5	2.6-6	3.6-13	-10.2	2.5	6.8-11	3.4-11	3.4-11
nql60	14560	3600×3	14402	-6.5	21.1	3.4-7	1.3-7	2.8-12	-10.0	11.8	1.0-10	1.1-11	1.1-11
nql180	130080	32400×3	129602	-5.3	278.3	4.9-6	8.5-7	5.7-12	-9.2	229.8	5.8-10	1.9-11	1.9-11
qssp30	3691	1891×4	2	-8.7	4.5	2.0 -9	2.3-10	2.2-14	-11.1	4.5	7.1-13	4.8-12	7.5-12
qssp60	14581	7381×4	2	-7.9	24.1	1.4-8	1.7 -9	3.4-15	-10.6	26.5	3.3-12	1.7-11	2.7-11
qssp180	130141	65341×4	2	-7.6	493.1	2.8-8	4.8-10	1.7-14	-11.2	665.9	7.0-12	1.2-12	1.8-12
sched-50-50-o	2527	1×2474 ; 1×3	2502	-4.5	5.0	1.4-5	3.2-5	2.3-7	-7.0	6.2	1.0-12	1.0-7	4.0-14
sched-100-50-o	4844	1×4741 ; 1×3	5002	-3.7	11.7	1.8-4	4.2-6	9.7 -9	-6.0	14.4	2.9-13	1.0-6	1.4-12
sched-100-100-o	8338	1×8235 ; 1×3	10002	-2.8	20.8	1.2-3	1.6-3	1.5-5	-3.3	31.0	6.6-11	4.6-4	4.1-11
sched-200-100-o	18087	1×17884 ; 1×3	20002	-3.8	77.8	2.6-5	1.7-4	3.5-8	-3.9	66.4	4.8-12	1.2-4	2.3-11
sched-50-50-s	2526	1×2475	2502	-7.2	5.0	6.0-8	8.5 -9	4.5-15	-8.2	7.7	1.0-13	7.0 -9	1.1-14
sched-100-50-s	4843	1×4742	5002	-7.7	11.7	1.0-8	2.1-8	7.2-14	-8.9	21.3	1.1-11	1.3 -9	1.2-11
sched-100-100-s	8337	1×8236	10002	-6.2	21.2	5.5-8	7.0-7	2.8-14	-7.1	34.9	3.2-12	7.8-8	5.3-15
sched-200-100-s	18086	1×17885	20002	-6.5	61.3	3.1-7	3.2-7	2.2-13	-7.8	115.8	1.2-12	1.7-8	3.8-14
firL1Linfalph	3074	5844×3		-10.0	235.6	7.3-11	1.1-10	0.8-15	-4.7	286.8	5.2-7	1.8-5	0.0-16
firL1Linfeps	7088	4644×3	1	-9.9	255.5	1.1-10	1.2-10	6.8-16	-10.4	106.5	3.4-13	4.3-11	1.2-14
firL1	6223	5922×3		-10.1	612.6	3.0-11	7.3-11	1.0-15	-9.0	598.4	1.8-11	1.0 -9	8.1-12
firL2a	1002	1×1003		-10.3	35.1	5.0-11	7.2-16	0.8-16	-12.6	21.3	7.8-15	2.7-13	4.5-15
firL2L1alph	5868	1×3845 ; 1922×3	1	-10.1	117.6	8.0-11	1.2-11	6.2-16	-3.3	197.6	6.2-7	5.1-4	4.4-11
firL2L1eps	4124	1×203 ; 3922×3		-10.4	181.3	3.6-11	2.1-11	0.9-15	-9.3	198.9	1.3-11	4.8-10	1.1-11
firL2Linfalph	203	1×203 ; 2942×3		-10.0	127.7	7.8-11	9.3-11	7.4-16	-9.5	237.4	4.0-12	3.5-10	8.1-14
firL2Linfeps	6086	1×5885 ; 2942×3		-10.1	369.7	7.1-11	3.8-11	6.7-16	-9.1	262.2	8.1-10	5.7-10	0.0-16
firL2	102	1×103		-11.3	0.3	5.2-12	4.6-16	1.5-16	-13.1	0.1	1.3-15	7.3-14	3.3-15
firLinf	402	3962×3		-8.9	465.4	1.2 -9	1.1 -9	1.0-15	-9.3	936.5	6.6-13	5.4-10	2.4-13
rand200-300-1	200	20×15		-7.2	2.9	5.6-8	3.0 -9	6.4-15	-6.4	7.6	3.3-7	4.3-7	0.0-16
rand200-300-2	200	20×15		-6.3	3.0	5.6-7	1.4-8	5.6-14	-5.0	12.8	7.8-6	9.0-6	0.0-16
rand200-800-1	200	20×40		-6.1	5.5	8.0-7	1.6 -9	2.0-14	-5.0	25.4	1.0-5	1.1-6	0.0-16
rand200-800-2	200	20×40		-4.7	6.0	1.9-5	1.0-8	6.8-14	-5.8	56.2	8.8-7	1.6-6	0.0-16
rand400-800-1	400	40×20		-6.5	19.0	2.9-7	1.8-8	8.7-12	-5.1	29.0	7.1-6	2.7-6	0.0-16
rand400-800-2	400	40×20		-5.6	17.7	2.6-6	8.2 -9	1.3 -9	-4.5	56.6	3.5-5	2.2-5	0.0-16
rand700-1e3-1	700	70×15		-7.1	74.4	8.1-8	1.8-8	3.0-14	-5.7	142.6	1.6-6	1.9-6	0.0-16
rand700-1e3-2	700	70×15		-5.3	80.4	5.0-6	1.1-7	6.6-14	-4.6	199.4	1.4-5	2.6-5	0.0-16
rand1000-2e3	1000	100×20		-5.7	230.4	1.9-6	1.3-8	9.4-10	-5.0	600.0	7.4-6	9.8-6	0.0-16
rand1500-3e3	1500	150×20		-7.0	812.3	1.2-8	1.0-7	8.7-14	-7.0	2119.6	9.7-8	8.8-8	0.0-16

Table 4.2: Performance of the SCE-based IPM in SDPT3 in solving SOCP problems with linear variables coming from unrestricted variables. The heuristics in (4.2) and (4.3) are applied at each IPM iteration.

		SDPT3					SeDuMi			
problem	ϕ	Time	p-inf	d-inf	relgap	ϕ	Time	p-inf	d-inf	relgap
nb-u	-10.2	14.2	6.4-11	1.1-13	5.2-16	-11.1	13.6	6.5-13	8.4-12	0.0-16
nb-L1-u	-10.0	28.2	9.9-11	1.1-11	2.2-16	-12.2	15.1	6.1-13	1.0-14	1.0-14
nb-L2-u	-10.2	16.9	5.8-11	1.6-11	6.6-16	-9.3	33.8	5.4-10	3.1-12	6.5-12
nb-L2-bessel-u	-10.2	12.9	6.7-11	3.3-11	3.3-16	-10.5	20.6	3.3-11	7.9-14	1.7-13
nql30-u	-10.1	7.1	8.7-11	2.4-12	8.0-13	-10.2	3.5	6.8-11	3.4-11	2.8-11
nql60-u	-10.4	29.9	4.4-11	2.0-11	2.8-13	-10.0	12.0	1.0-10	1.1-11	8.9-12
nql180-u	-9.7	455.7	2.1-10	1.2-11	8.4-14	-9.2	263.8	5.8-10	1.9-11	1.1-11
qssp30-u	-10.0	4.3	7.4-11	9.2-11	2.7-15	-11.3	4.0	7.1-13	4.8-12	5.2-12
qssp60-u	-8.8	21.7	1.4 -9	1.8 -9	4.0-14	-10.8	26.5	3.3-12	1.7-11	1.7-11
qssp180-u	-9.0	560.9	1.1 -9	6.7-10	9.5-15	-11.2	694.2	7.0-12	1.2-12	9.9-13

5. Reduced augmented equation. In this section, we present a new approach to compute the search direction via a potentially better-conditioned linear system of equations. Based on the new approach, the accuracy of the computed search direction is expected to be better than that computed from the SCE when μ is small. In this new approach, we assume that the iterate (x, y, z) is sufficiently close to the central path so that the eigenvalues of F^2 separate into three distinct groups as described in Section 3.2.

In this approach, we start with the augmented equation in (2.6). By using the eigenvalue decomposition, $F^2 = QDQ^T$ presented in Section 3.1, where $Q = \mathrm{diag}(Q_1, \ldots, Q_N)$ and $D = \mathrm{diag}(\Lambda_1, \ldots, \Lambda_N)$. We can diagonalize the (1,1) block and rewrite the augmented equation (2.6) as follows:

$$\begin{bmatrix} -D & \widetilde{A}^T \\ \widetilde{A} & 0 \end{bmatrix} \begin{bmatrix} \Delta\widetilde{x} \\ \Delta y \end{bmatrix} = \begin{bmatrix} \widetilde{r} \\ r_p \end{bmatrix}, \tag{5.1}$$

where

$$\widetilde{A} = AQ, \quad \Delta\widetilde{x} = Q^T\Delta x, \quad \widetilde{r} = Q^T r_x. \tag{5.2}$$

The augmented equation (5.1) has dimension $m + n$, which is usually much larger than m, the dimension of the SCE. We can try to reduce its size while overcoming some of the undesirable features of the SCE such as the growth of $\|M\|$ when $\mu \downarrow 0$.

Let the diagonal matrix D be partitioned into two parts as $D = \mathrm{diag}(D_1, D_2)$ with $\mathrm{diag}(D_1)$ consisting of the small eigenvalues of F^2 of order $\Theta(\mu)$ and $\mathrm{diag}(D_2)$ consisting of the remaining eigenvalues of order $\Theta(1)$ or $\Theta(1/\mu)$. We partition the eigenvector matrix Q accordingly as $Q = [Q^{(1)}\ ,\ Q^{(2)}]$. Then $\widetilde{A}$ is partitioned as $\widetilde{A} = [\widetilde{A}_1\ ,\ \widetilde{A}_2] = [AQ^{(1)}\ ,\ AQ^{(2)}]$ and $\widetilde{r} = [\widetilde{r}_1\,;\,\widetilde{r}_2] = [(Q^{(1)})^T r_x\,;\,(Q^{(2)})^T r_x]$. Similarly, $\Delta\widetilde{x}$ is partitioned as $\Delta\widetilde{x} = [\Delta\widetilde{x}_1\,;\,\Delta\widetilde{x}_2] = [(Q^{(1)})^T\Delta x\,;\,(Q^{(2)})^T\Delta x]$.

By substituting the above partitions into (5.1), and eliminating $\Delta\widetilde{x}_2$, it is easy to show that solving the system (5.1) is equivalent to solving the following:

$$\begin{bmatrix} \widetilde{A}_2 D_2^{-1}\widetilde{A}_2^T & \widetilde{A}_1 \\ \widetilde{A}_1^T & -D_1 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta\widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} r_p + \widetilde{A}_2 D_2^{-1}\widetilde{r}_2 \\ \widetilde{r}_1 \end{bmatrix}, \tag{5.3}$$

$$\Delta\widetilde{x}_2 \;=\; D_2^{-1}(\widetilde{A}_2^T\Delta y - \widetilde{r}_2) \;=\; D_2^{-1}(Q^{(2)})^T(A^T\Delta y - r_x). \tag{5.4}$$

By its construction, the coefficient matrix in (5.3) does not have large elements when $\mu \downarrow 0$. But its (1,1) block is generally singular or nearly singular, especially when μ is close to 0. Since a singular (1,1) block is not conducive for symmetric indefinite factorization of the matrix or the construction of preconditioners for the matrix, we will construct an equivalent system with a (1,1) block that is less likely to be singular. Let E_1 be a given positive definite diagonal matrix with the same dimension as D_1. Throughout this paper, we take $E_1 = I$. Let $S_1 = E_1 + D_1$. By adding $\widetilde{A}_1 S_1^{-1}$ times the second block equation in (5.3) to the first block equation, we get $\widetilde{A}\mathrm{diag}(S^{-1}, D_2^{-1})\widetilde{A}^T\Delta y + \widetilde{A}_1 S_1^{-1}E_1\Delta\widetilde{x}_1 \;=\; r_p + \widetilde{A}\mathrm{diag}(S^{-1}, D_2^{-1})\widetilde{r}$. This, together with the second block equation in (5.3) but scaled by $S_1^{-1/2}$, we get the following equivalent system:

$$\underbrace{\begin{bmatrix} \widetilde{M} & \widetilde{A}_1 S_1^{-1/2} \\ S_1^{-1/2}\widetilde{A}_1^T & -D_1E_1^{-1} \end{bmatrix}}_{\mathcal{B}} \begin{bmatrix} \Delta y \\ S_1^{-1/2}E_1\Delta\widetilde{x}_1 \end{bmatrix} = \begin{bmatrix} q \\ S_1^{-1/2}\widetilde{r}_1 \end{bmatrix}, \tag{5.5}$$

where

$$\widetilde{M} \;=\; \widetilde{A}\,\mathrm{diag}(S_1^{-1}, D_2^{-1})\widetilde{A}^T, \quad q \;=\; r_p + \widetilde{A}\mathrm{diag}(S_1^{-1}, D_2^{-1})\widetilde{r}. \tag{5.6}$$

We call the system in (5.5) the **reduced augmented equation** (RAE). Note that once Δy and $\Delta\widetilde{x}_1$ are computed from (5.5) and $\Delta\widetilde{x}_2$ is computed from (5.4), Δx can be recovered through the equation $\Delta x = Q[\Delta\widetilde{x}_1\,;\,\Delta\widetilde{x}_2]$.

REMARK 5.1. *(a) If the matrix D_1 is null, then the RAE (5.5) is reduced to the SCE (2.7).*

(b) $\mathcal{B}$ is a quasi-definite matrix [8, 27]. Such a matrix has the nice property that any symmetric reordering $\Pi\mathcal{B}\Pi^T$ has a "Cholesky factorization" $L\Lambda L^T$ where Λ is diagonal with both positive and negative diagonal elements.

Observe that the $(1,1)$ block, $\widetilde{M}$, in (5.5) has the same structure as the Schur complement matrix $M = \widetilde{A}\,\mathrm{diag}(D_1^{-1}, D_2^{-1})\widetilde{A}^T$. But for $\widetilde{M}$, $\|\mathrm{diag}(S_1^{-1}, D_2^{-1})\| = O(1)$, whereas for M, $\|\mathrm{diag}(D_1^{-1}, D_2^{-1})\| = O(1/\mu)$. Because of this difference, the reduced augmented matrix $\mathcal{B}$ has bounded norm as $\mu \downarrow 0$, but $\|M\|$ is generally unbounded. Under certain conditions, $\mathcal{B}$ can be shown to have a condition number that is bounded independent of the normalized complementarity gap μ. The precise statements are given in the following theorems.

THEOREM 5.1. *Suppose in (5.5) we use a partition such that* $\mathrm{diag}(D_1)$ *consist of all the eigenvalues of F^2 of order $\Theta(\mu)$. If the optimal solution of (1.1) satisfies strict complementarity, then $\|\mathcal{B}\|$ satisfies the following inequality: $\|\mathcal{B}\| = O(1)\,\|A\|^2$. Thus $\|\mathcal{B}\|$ is bounded independent of μ (as $\mu \downarrow 0$).*

Proof. It is easy to see that

$$\|\mathcal{B}\| \le \sqrt{2}\,\max(\,\|\widetilde{M}\| + \|\widetilde{A}_1 S_1^{-1/2}\|,\, \|S_1^{-1/2}\widetilde{A}_1^T\| + \|D_1E_1^{-1}\|\,).$$

Under the assumption that the optimal solution of (1.1) satisfies strict complementarity, then as $\mu \downarrow 0$, $\|D_1\| \downarrow 0$, and $\|D_2^{-1}\| = O(1)$, so it is possible to find a constant (independent of μ) $\tau \ge 1$ such that: $\max(\,\|S_1^{-1}\|, \|D_2^{-1}\|, \|D_1E_1^{-1}\|\,) \le \tau$.

Now $\|\widetilde{M}\| \le \|\widetilde{A}\| \max(\|S_1^{-1}\|, \|D_2^{-1}\|)\|\widetilde{A}\| \le \tau\|\widetilde{A}\|^2$ and $\|S_1^{-1/2}\widetilde{A}_1^T\| = \|\widetilde{A}_1 S_1^{-1/2}\| \le \tau\|\widetilde{A}_1\|$, thus we have

$$\|\mathcal{B}\| \le \tau\sqrt{2}\max(\ \|\widetilde{A}\|^2 + \|\widetilde{A}_1\|,\ \|\widetilde{A}_1\| + 1\ ) \le \tau\sqrt{2}(\|A\| + 1)^2.$$

From here, the required result follows. ∎

LEMMA 5.2. *The reduced augmented matrix $\mathcal{B}$ in (5.5) satisfies the following inequality:*

$$\|\mathcal{B}^{-1}\| \le 2\sqrt{2}\max(\|\widetilde{M}^{-1}\|\,,\ \|W^{-1}\|),$$

where $W = B_1^T\widetilde{M}^{-1}B_1 + D_1E_1^{-1}$ with $B_1 = \widetilde{A}_1 S_1^{-1/2}$.

Proof. From [19, p. 389], it can be deduced that

$$\mathcal{B}^{-1} = \begin{bmatrix} \widetilde{M}^{-1/2}(I-P)\widetilde{M}^{-1/2} & \widetilde{M}^{-1}B_1W^{-1} \\ W^{-1}B_1^T\widetilde{M}^{-1} & -W^{-1} \end{bmatrix},$$

where $P = \widetilde{M}^{-1/2}B_1W^{-1}B_1^T\widetilde{M}^{-1/2}$. Note that P satisfies the condition $0 \preceq P \preceq I$, i.e., P and $I - P$ are positive semidefinite. By the definition of W, we have $0 \preceq W^{-1/2}B_1^T\widetilde{M}^{-1}B_1W^{-1/2} \preceq I$, and thus $\|\widetilde{M}^{-1/2}B_1W^{-1/2}\| \le 1$. This implies that

$$\|\widetilde{M}^{-1}B_1W^{-1}\| \le \|\widetilde{M}^{-1/2}\|\ \|\widetilde{M}^{-1/2}B_1W^{-1/2}\|\ \|W^{-1/2}\| \le \max\Big(\|\widetilde{M}^{-1}\|\,,\ \|W^{-1}\|\Big).$$

It is easy to see that

$$\|\mathcal{B}^{-1}\| \le \sqrt{2}\max\Big(\|\widetilde{M}^{-1/2}(I-P)\widetilde{M}^{-1/2}\| + \|\widetilde{M}^{-1}B_1W^{-1}\|\ ,\ \|W^{-1}B_1^T\widetilde{M}^{-1}\| + \|W^{-1}\|\Big).$$

From here, the required result follows. ∎

THEOREM 5.3. *Suppose in (5.5) we use a partition such that* $\mathrm{diag}(D_1)$ *consist of all the eigenvalues of F^2 of order $\Theta(\mu)$. If the optimal solution of (1.1) satisfies strict complementarity and the primal and dual nondegeneracy conditions defined in [2], then the condition number of the coefficient matrix in (5.5) is bounded independent of μ (as $\mu \downarrow 0$).*

Proof. Let D_2 be further partitioned into $D_2 = \mathrm{diag}(D_{2a}, D_{2b})$ where $\mathrm{diag}(D_{2a})$ and $\mathrm{diag}(D_{2b})$ consist of eigenvalues of F^2 of order $\Theta(1)$ and $\Theta(1/\mu)$, respectively. Let $Q^{(2)}$ and $\widetilde{A}_2$ be partitioned accordingly as $Q^{(2)} = [Q^{(2a)},\ Q^{(2b)}]$ and $\widetilde{A}_2 = [AQ^{(2a)},\ AQ^{(2b)}] =: [\widetilde{A}_{2a},\ \widetilde{A}_{2b}]$. By Theorems 20 and 21 in [1], dual nondegeneracy implies that $\widetilde{A}_1 = AQ^{(1)}$ has full column rank and primal nondegeneracy implies that $[\widetilde{A}_1,\ \widetilde{A}_{2a}]$ has full row rank. Since $\|\widetilde{M} - [\widetilde{A}_1,\ \widetilde{A}_{2a}]\mathrm{diag}(S_1^{-1},\ D_{2a}^{-1})[\widetilde{A}_1\,,\ \widetilde{A}_{2a}]^T\| = O(\mu)$, thus $\sigma_{\min}(\widetilde{M})$ is bounded away from 0 even when $\mu \downarrow 0$. This, together with the fact that $\widetilde{A}_1 S_1^{-1/2}$ has full column rank, implies that the matrix $W := S_1^{-1/2}\widetilde{A}_1^T\widetilde{M}^{-1}\widetilde{A}_1S_1^{-1/2} + D_1E_1^{-1}$ has $\sigma_{\min}(W)$ bounded away from 0 even when $\mu \downarrow 0$. By Lemma 5.2, $\|\mathcal{B}^{-1}\|$ is bounded independent of μ. By Theorem 5.1, $\|\mathcal{B}\|$ is also bounded independent of μ, and the required result follows. ∎

6. Reduced augmented equation and primal infeasibility. Let $[\xi\,;\,\eta]$ be the residual vector for the computed solution of (5.5).

LEMMA 6.1. *Let u be the machine epsilon and l be the dimension of $\Delta\widetilde{x}_1$. Suppose $(l+m)u \le 1/2$ and we use Gaussian elimination with partial pivoting (GEPP) to solve (5.5) to get the computed solution $(\Delta y\,;\,\Delta\widetilde{x}_1)$, then the residual vector $[\xi\,;\,\eta]$ for the computed solution satisfies the following inequality:*

$$\|(\xi\,;\,\eta)\|_\infty \;\le 4(l+m)^3 u\rho\,\|\mathcal{B}\|_\infty \|(\Delta y\,;\,\Delta\widetilde{x}_1)\|_\infty$$

where ρ is the growth factor associated with GEPP.

Proof. This lemma follows from Theorem 9.5 in [12]. □

REMARK 6.1. *Theorem 5.1 stated that if strict complementarity holds at the optimal solution, then $\|\mathcal{B}\|_\infty$ will not grow as $\mu \downarrow 0$ in contrast to $\|M\|$, which usually grows proportionately to $\Theta(1/\mu)$. Now because the growth factor ρ for GEPP is usually $O(1)$, Lemma 6.1 implies that the residual norm $\|(\xi\,;\,\eta)\|_\infty$ will be maintained at some level proportional to $u\|A\|^2$ even when $\mu \downarrow 0$.*

Now we establish the relationship between the residual norm in solving (5.5) and the primal infeasibility associated with the search direction computed from the RAE approach. Suppose that in computing $\Delta\widetilde{x}_2$ from (5.4), a residual vector δ is introduced, i.e.,

$$\Delta\widetilde{x}_2 = D_2^{-1}(Q^{(2)})^T(A^T\Delta y - r) - \delta.$$

Then we have the following lemma for the primal infeasibility of the next iterate.

LEMMA 6.2. *Suppose Δx is computed from the RAE approach. Then the primal infeasibility $\|r_p^+\|$ for the next iterate $x^+ = x + \alpha\Delta x$, $\alpha \in [0,1]$, satisfies the following inequality:*

$$\|r_p^+\| \;\le\; (1-\alpha)\|r_p\| + \alpha\|\xi + \widetilde{A}_2\delta - \widetilde{A}_1 S_1^{-1/2}\eta\|.$$

Proof. The proof is quite routine and we omit it. □

REMARK 6.2. *From Lemma 6.2, we see that if the RAE returns a small residual norm, then the primal infeasibility of the next iterate would not be seriously worsened by the residual norm. From Theorem 5.1 and Lemma 6.1, we expect the residual norm $\|[\xi;\eta]\|$ to be small since the upper bound on $\|\mathcal{B}\|$ is independent of μ. Also, since by its construction, D_2^{-1} does not have large elements, $\|\delta\|$ is expected to be small as well.*

Figure 6.1 shows the convergence behavior of the IPM in SDPT3, but with search directions computed from the RAE (5.5) for the problems `ran200_800_1` and `sched_50_50_orig`. As can be seen from the relative primal infeasibility curves, the RAE approach is more stable than the SCE approach. It is worth noting that under the new approach, the solver is able to deliver 10 digits of accuracy, i.e. $\phi \le -10$. This is significantly better than the accuracy $\phi \approx -6$ attained by the SCE approach. Note that we use a partition such that eigenvalues of F^2 that are smaller than 10^{-3} are put in D_1.

In Table 7.1, we show the norms $\|\mathcal{B}\|$, $\|\mathcal{B}^{-1}\|$ and the residual norm in solving the RAE (5.5) for the last few IPM iterations in solving the problem2 `rand200_800_1` and `sched_50_50_orig`. Observe that $\|\mathcal{B}\|$ and $\kappa(\mathcal{B})$ do not grow when $\mu \downarrow 0$ in contrast to $\|M\|$ and $\kappa(M)$ in Table 3.1. The residual norm for the computed solution of (5.5) remains small throughout, and in accordance with Lemma 6.1, the residual norm is approximately equal to $u\|\mathcal{B}\|$ times the norm of the computed solution. By Lemma 6.2, the small residual norm in solving the RAE explains why the primal infeasibility computed from the RAE approach does not deteriorate as in the SCE approach.

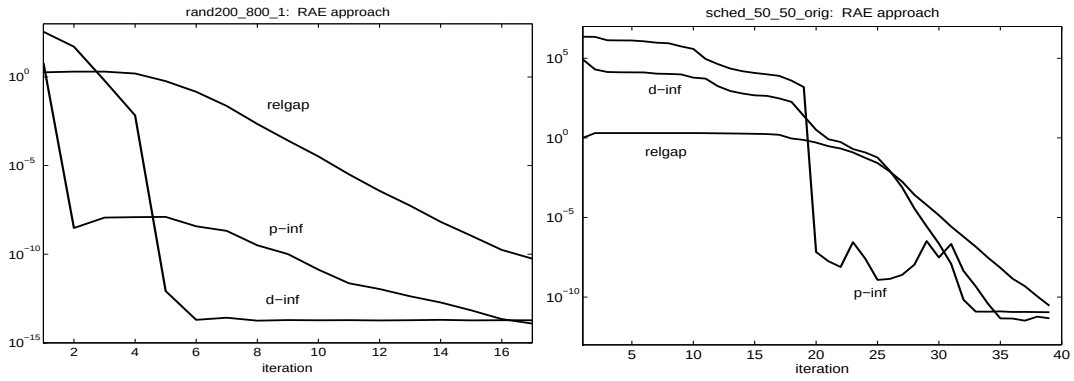

FIG. 6.1. *Same as Figure 3.1 but for the RAE approach in computing the search directions for the problems* `rand200_800_1` *and* `sched_50_50_orig`. *Notice that the primal infeasibility does not deteriorate when the iterates approach optimality. Both problems are primal and dual non-degenerate, and strict complementarity holds at optimality.*

7. Computational issues. The theoretical analysis in the last section indicates that the RAE approach is potentially more stable than the standard SCE approach, but the trade-off is that the former needs to solve a larger indefinite linear system. Thus, how to efficiently solve (5.5) is one of our major concerns in the implementation.

In forming the reduced augmented matrix $\mathcal{B}$, those operations involving Q (the eigenvector matrix of F^2) must be handled carefully by exploiting the structure of Q to avoid incurring significant storage and computational cost. Also, the sparsity of AA^T must be properly preserved when computing $\widetilde{M}$.

TABLE 7.1

Condition number of the reduced augmented matrix $\mathcal{B}$ associated with the last few IPM iterations for solving the SOCP problem `rand200_800_1` *and* `sched_50_50_orig`. *The maximum number of columns in $\widetilde{A}_1$ for the former problem is* 19, *and that for the latter is* 82.

Iter	$\|\mathcal{B}\|$	$\|\mathcal{B}^{-1}\|$	$x^T z/N$	$\|[\Delta y; \Delta\widetilde{x}_1]\|$	residual norm	$\|r_p\|$	$\frac{\|r_p\|}{\|\mathcal{B}\|\|[\Delta y; \Delta\widetilde{x}_1]\|u}$
`rand200_800_1`							
12	3.7e+11	2.7e+02	1.3e-09	4.6e-04	6.9e-09	1.8e-08	4.7e-01
13	3.4e+11	2.8e+02	1.9e-10	2.3e-04	4.3e-09	8.1e-09	4.7e-01
14	2.6e+11	3.7e+02	2.5e-11	1.0e-04	1.2e-09	2.9e-09	4.9e-01
15	2.3e+11	4.2e+02	4.0e-12	3.6e-05	3.4e-10	9.4e-10	5.1e-01
16	2.0e+11	5.2e+02	5.6e-13	1.3e-05	1.4e-10	4.8e-10	8.6e-01
`sched_50_50_orig`							
33	1.1e+08	3.9e+04	7.9e-07	1.4e-02	9.3e-13	1.2e-12	3.6e-03
34	6.2e+07	1.4e+04	1.6e-07	2.1e-03	3.9e-14	1.9e-11	6.8e-01
35	6.2e+07	2.7e+04	3.7e-08	1.7e-03	8.4e-15	1.2e-12	5.0e-02
36	6.2e+07	1.5e+04	7.5e-09	3.8e-04	2.8e-15	2.7e-11	5.1e+0
37	6.2e+07	2.2e+04	2.8e-09	2.2e-04	6.5e-16	2.5e-11	8.1e+0

7.1. Computations involving Q. The operations involving Q in assembling the RAE (5.5) are as follows:

• Computation of the (1,1) block $\widetilde{M} = AQ\mathrm{diag}(S_1^{-1}, D_2^{-1})Q^TA^T$;

• Computation of the (1,2) block $\widetilde{A}_1 = AQ^{(1)}$ and the right hand side vector $\widetilde{r} = Q^Tr_x$.

To carry out the above operations efficiently, we need to derive an explicit formula for Q to facilitate such calculations. Recall the eigenvector matrix Q_i (3.1) associated with the ith second order cone. To get an explicit description of Q_i, we need to construct the Householder matrix H_i explicitly. Without going into the algebraic details, the precise form of H_i is given as follows:

$$(H_i) = I - h_ih_i^T, \quad h_i := \begin{bmatrix} h_i^0 \\ \bar{h}_i \end{bmatrix} = \frac{1}{\tau_i}\begin{bmatrix} \tau_i^2\,\mathrm{sign}(g_i^0) \\ \bar{g}_i \end{bmatrix} \in \mathbb{R}^{n_i-1}, \quad \tau_i := \sqrt{1+|g_i^0|}.$$

With some algebraic manipulations, the eigenvector Q_i can be rewritten in the form given in the next lemma.

LEMMA 7.1. *Let $\beta_i = -\mathrm{sign}(h_i^0)/\sqrt{2}$. We have $Q_i = \mathrm{diag}(K_i, I) - u_iv_i^T$, where*

$$K_i = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \beta_i & \beta_i \end{bmatrix}, \quad u_i = \begin{bmatrix} 0 \\ h_i \end{bmatrix}, \quad v_i = \begin{bmatrix} \beta_ih_i^0 \\ \beta_ih_i^0 \\ \bar{h}_i \end{bmatrix}. \tag{7.2}$$

Proof. Note that by construction, the first column of H_i is given by $-\mathrm{sign}(g_i^0)g_i$. Let $\alpha = \frac{1}{\sqrt{2}} + \mathrm{sign}(g_i^0)$. From (3.1), we have

$$Q_i = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & \cdots & 0 \\ \frac{1}{\sqrt{2}}g_i & \alpha g_i & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & I - h_ih_i^T \end{bmatrix}$$

$$= \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & \ldots & 0 \\ -\frac{\mathrm{sign}(g_i^0)}{\sqrt{2}} & -\frac{\mathrm{sign}(g_i^0)}{\sqrt{2}} - 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} 0 \\ h_i \end{bmatrix}\begin{bmatrix} -\frac{\tau_i}{\sqrt{2}} \\ h_i^0 - \alpha\tau_i \\ \bar{h}_i \end{bmatrix}^T.$$

It is readily shown that $h_i^0 - \alpha\tau_i = -\tau_i/\sqrt{2} = \beta_ih_i^0$. Now it is easy to see that the required results hold. □

Observe that each Q_i is a rank-one perturbation of a highly sparse block diagonal matrix. Based on the above lemma, those operations listed at the beginning of this subsection, except the first one, can be computed straightforwardly. To compute the matrix $\widetilde{M}$, we have to further analyze the structure of the matrix $Q_i\mathrm{diag}(S_{1i}^{-1}, D_{2i}^{-1})Q_i^T$.

Let $G_i = \mathrm{diag}(S_{1i}^{-1}, D_{2i}^{-1})$ and $\Sigma_i = \mathrm{diag}(K_i, I)$, then $Q_i = \Sigma_i - u_iv_i^T$ and $Q_i\,G_i\,Q_i^T = \Sigma_i\,G_i\,\Sigma_i^T - \Sigma_iG_iv_iu_i^T - u_iv_i^TG_i\Sigma_i^T + u_iv_i^T\,G_i\,v_iu_i^T$. By setting $\rho_i = v_i^TG_iv_i$ and $\widetilde{v}_i = \Sigma_iG_iv_i/\sqrt{\rho_i}$, we have $Q_i\,G_i\,Q_i^T = \Sigma_iG_i\Sigma_i^T + l_il_i^T - \widetilde{v}_i\widetilde{v}_i^T$, where $l_i = \widetilde{v}_i - \sqrt{\rho_i}u_i$. Thus each component matrix $\widetilde{M}_i$ in $\widetilde{M} = \sum_{i=1}^N \widetilde{M}_i$ can be expressed as:

$$\widetilde{M}_i = A_iQ_iG_iQ_i^TA_i^T = A_i\,(\Sigma_iG_i\Sigma_i^T)\,A_i^T + (A_il_i)(A_il_i)^T - (A_i\widetilde{v}_i)(A_i\widetilde{v}_i)^T. \tag{7.3}$$

Since $\Sigma_i G_i \Sigma_i^T$ is a highly sparse block diagonal matrix, $\widetilde{M}_i$ is a symmetric rank-two perturbation to a sparse matrix if $A_i A_i^T$ is sparse. Hence, the computational complexity of $\widetilde{M}$ is only slightly more expensive than that for the Schur complement matrix M.

7.2. Handling dense columns. Let $\Sigma = \mathrm{diag}(\Sigma_1, \ldots, \Sigma_N)$, and

$$A_l = [A_1 l_1 , \ \ldots \ , \ A_N l_N], \quad A_v = [A_1 \widetilde{v}_1 , \ \ldots \ , \ A_N \widetilde{v}_N].$$

Then it is readily shown that

$$\widetilde{M} = A\,\Sigma \mathrm{diag}(S_1^{-1}, D_2^{-1})\Sigma^T A^T + A_l A_l^T - A_v A_v^T. \tag{7.4}$$

If AA^T is sparse, then the first matrix in (7.4) is sparse as well. For an SOCP problem where all the cones are low dimensional, typically the matrices A_l and A_v are also sparse. In that case, the RAE (5.5) may be solved directly. However, if high dimensional cones exist, then A_l and A_v invariably contain dense columns. Moreover, when A is sparse but has dense columns, AA^T will also be dense. In order to preserve the sparsity in $\widetilde{M}$, it is necessary to handle the dense columns separately when they exist.

Let P_1 be the dense columns in $A\Sigma\mathrm{diag}(S_1^{-1/2}, D_2^{-1/2})$ and A_l, and P_2 be the dense columns in A_v. Let $\widetilde{M}_s = \widetilde{M} - P_1 P_1^T + P_2 P_2^T$ be the "sparse part" of $\widetilde{M}$. It is well known that by introducing the following auxiliary variables, $t_1 = P_1^T \Delta y$, $t_2 = -P_2^T \Delta y$, the dense columns can be removed from $\widetilde{M}$; see [4]. The precise form of the RAE (5.5) with dense column handling is as follows:

$$\begin{bmatrix} \widetilde{M}_s & U \\ U^T & -C \end{bmatrix} \begin{bmatrix} \Delta y\,;\, S_1^{-1/2} E_1 \Delta \widetilde{x}_1\,;\, t_1\,;\, t_2 \end{bmatrix} = \begin{bmatrix} q\,;\, S_1^{-1/2}\widetilde{r}_1\,;\, 0\,;\, 0 \end{bmatrix}, \tag{7.5}$$

where q is defined in (5.6) and $U = \begin{bmatrix} \widetilde{A}_1 S_1^{-1/2}, & P_1, & P_2 \end{bmatrix}$, $C = \mathrm{diag}(D_1 E_1^{-1}, I_1, -I_2)$. Here I_1, I_2 are identity matrices.

7.3. Direct solvers for symmetric indefinite systems. Solving the sparse symmetric indefinite system (7.5) is one of the most expensive step at each IPM iteration. Thus, it is critical that the solver used must be as efficient as possible.

We consider two methods for solving (7.5). The first is the Schur complement method, which is also equivalent to the Sherman-Morrison-Woodbury formula. The second is the LDL^T factorization implemented in MA47 [18]. Each of these methods has its own advantages under different circumstances.

Schur complement method. This method is widely used for dense column handling in IPM implementations; see [4] and the references therein. It uses the sparse matrix $\widetilde{M}_s$ as the pivoting matrix to perform block eliminations in (7.5). It is readily shown that solving (7.5) is equivalent to solving the following systems:

$$\left(U^T \widetilde{M}_s^{-1} U + C\right) \begin{bmatrix} S_1^{-1/2} E_1 \Delta \widetilde{x}_1\,;\, t_1\,;\, t_2 \end{bmatrix} = U^T \widetilde{M}_s^{-1} q - \begin{bmatrix} S_1^{-1/2}\widetilde{r}_1\,;\, 0\,;\, 0 \end{bmatrix} \tag{7.6}$$

$$\Delta y = \widetilde{M}_s^{-1} q - \widetilde{M}_s^{-1} U \begin{bmatrix} S_1^{-1/2} E_1 \Delta \widetilde{x}_1\,;\, t_1\,;\, t_2 \end{bmatrix}.$$

Note that since $\widetilde{M}$ is symmetric positive definite, its "sparse part", $\widetilde{M}_s$, is typically also positive definite if the number of dense columns removed from $\widetilde{M}$ is small. If $\widetilde{M}_s$

is indeed positive definite, then (7.6) can be solved by Cholesky or sparse Cholesky factorization. As mentioned before, highly efficient and optimized Cholesky solvers are readily available in the public domain. Another advantage of the Schur complement method is that the symbolic factorization of $\widetilde{M}_s$ and the pivoting order of the Cholesky factorization needs only be computed once or twice during the initial phase of the IPM iteration and it can be re-used for subsequent IPM iterations even when the partition in D changes.

But the Schur complement method does have a major disadvantage in that the matrix $U^T\widetilde{M}_s^{-1}U+C$ is typically dense. This can lead to a huge computational burden when U has a large number of columns, say, more than a few hundreds. Furthermore, the Schur complement method is numerically less stable than a method that solves (7.5) directly.

Roughly speaking, the Schur complement method is best suited for problems with U having a small number of columns. When U has a large number of columns or when $\widetilde{M}_s$ is not positive definite, we have to solve (7.5) directly by the second method described below.

MA47. MA47 is a direct solver developed by Reid and Duff [18] for sparse symmetric indefinite systems. This is perhaps the only publicly available state-of-the-art direct solver for sparse symmetric indefinite systems. It appears not to be as efficient as the sparse Cholesky codes of Ng and Peyton [16].

The MA47 solver implements the multi-frontal sparse Gaussian elimination described in [7]. In the algorithm, the pivots used are not limited only to 1×1 diagonal pivots but also 2×2 block diagonal pivots. The solver performs a pre-factorization phase (called symbolic factorization) on the coefficient matrix to determine a pivoting order so as to minimize fill-ins. In the actual factorization process, this pivoting order may be modified to obtain better numerical stability. Note that in sparse Cholesky factorization, the pivoting order is not modified after the symbolic factorization phase. Because significant overhead may be incurred when the pivoting order is modified in the factorization process, running MA47 is sometimes much more expensive than the sparse Cholesky routine of Ng and Peyton on matrices with the same dimensions and sparsity patterns.

The advantage of using MA47 to solve (7.5) is that it does not introduce a fully dense matrix in the solution process. Thus it is more suitable for SOCP problems with U having a relatively large number of columns.

However, the MA47 method does have a disadvantage in that the symbolic factorization of the reduced augmented matrix needs to be re-computed whenever the partition in D changes.

7.4. Partitioning Strategy. As shown in Section 6, the RAE approach for computing the search directions has the potential to overcome certain numerical instabilities encountered in the SCE approach. The RAE was derived from the augmented equation (2.6) by modifying the part of the coefficient matrix involving the small eigenvalues of F^2. Here we will describe the partition we use in $D = \mathrm{diag}(D_1, D_2)$.

The choice of D_1 is dictated by the need to strike a balance between our desire to compute more accurate search directions and to minimize the size of the RAE to be solved. For computational efficiency, it is better to have as few columns in the matrix U (7.5) as possible, thus suggesting that the threshold for labelling an eigenvalue as "small" should be low. But to obtain better accuracy, it is beneficial to partition eigenvalues that are smaller than, say 10^{-3}, into D_1 to improve the conditioning of the reduced augmented matrix.

With due consideration in balancing the two issues just mentioned, we adopt a hybrid strategy in computing the search direction at each IPM iteration. If $\kappa(F^2) \geq 10^6$, put the eigenvalues of F^2 that are smaller than 10^{-3} in D_1, and the rest in D_2; Otherwise, put all the eigenvalues of F^2 in D_2.

Some of our test problems also contain linear blocks (i.e., cones with dimensions $n_i = 1$). In this case, $F_i^2 = z_i/x_i$ is a scalar, and we put F_i^2 in D_1 if it is smaller than 10^{-3}, otherwise, we put it in D_2.

As noted in Remark 3.1, when $\widetilde{A}_1$ has full row rank (for which a necessary condition is that the number of small eigenvalues put into D_1 is at least m), the Schur complement matrix M is not highly ill-conditioned, and it is not necessary to use the RAE approach to compute the search directions. When such a situation occurs, we use the SCE approach.

8. Numerical experiments. The RAE (5.5) or (7.5) is more expensive to solve than the SCE (2.7) because it is larger in size. As we have discussed in the last section, we can try to minimize the additional computational cost by a judicious choice of the solver used. If the number of columns in U is small, then using the Schur complement method to solve (7.5) should not be much more expensive than solving the SCE. We adopt the following heuristic rule to select the solver used to solve (7.5). If the number of columns in U is less than 200, we use the Schur complement method; otherwise, we use the MA47 method.

The RAE approach is implemented in MATLAB based on the IPM in SDPT3, version 3.1; see [25]. But the search direction at each iteration is computed based on the RAE (7.5). We use the same stopping criteria mentioned in Section 4. Again, the numerical results are obtained from a Pentium IV 2.4GHz PC with 1G RAM.

We consider the same SOCP problems in Section 4. But in order to focus on the comparison between the SCE and RAE approaches without the complication of unbounded primal solution sets, we exclude the `nbxxx`, `nqlxxx`, and `qsspxxx` problems from the numerical experiments in this section. Our major concern in the experiments are efficiency and accuracy. We measure efficiency by the total CPU time taken; while accuracy is again measured by accuracy exponent defined in (4.1).

The numerical results for the RAE-based IPM is presented in Table 9.1. In the table, $T_{\rm iter}$ denotes the average CPU time taken per iteration. For the RAE-based IPM, the number of IPM iterations taken for each problem is given under the column "iter". The number in each bracket gives the number of iterations using the RAE approach. The total CPU time taken to solve each problem is given under the column "Time". The number in each bracket gives the CPU time taken by the iterations using the RAE approach.

The numerical results in Table 4.1 show that the SCE-based IPMs may not deliver approximate optimal solutions with small primal infeasibilities. In Table 9.1, we see that the RAE-based IPM can drive the primal infeasibilities of all the problems to a level of 10^{-9} or smaller. For the `schedxxx` and `randxxx` problem sets, both the SCE-based IPMs in SDPT3 and SeDuMi cannot deliver accurate approximate solutions where the accuracies attained range from $\phi = -2.5$ to $\phi = -8.9$ for the `schedxxx` set and from $\phi = -4.5$ to $\phi = -7.6$ for the `randxxx` set. The RAE-based IPM, however, can achieve solutions with accuracy $\phi \leq -9.1$ for all the problems in these 2 sets. The improvement in the attainable accuracy is more than 5 orders of magnitudes in some cases. For the `firxxx` problems, the SCE approach can already produce accurate approximate solutions, and the RAE approach produces comparable accuracies.

The good performance in terms of accuracy of the RAE-based IPM on the `schedxxx`

and `randxxx` problem sets is consistent with the theoretical results established in Section 6. The SOCP problems in the `schedxxx` set are primal and dual nondegenerate, and strict complementarity holds at optimality. For the `randxxx` set, all the problems are primal non-degenerate, but 4 of the problems are dual degenerate. It is interesting to note that dual degeneracy does not seem to affect the performance of RAE on these degenerate problems. This fact is consistent with the observation we made in Remark 6.2.

By Theorem 5.3, the condition number of the reduced augmented matrix for the problems in `schedxxx` set is bounded when $\mu \downarrow 0$. But as noted in Remark 3.2, strict complementarity, and primal and dual nondegeneracy in an SOCP does not necessarily imply that the associated Schur complement matrix has bounded condition numbers when $\mu \downarrow 0$. The numerical results produced by the `schedxxx` problems concretely show the difference in numerical stability between the SCE and RAE approaches.

From the average CPU time taken per IPM iteration for the RAE and SCE approaches in Table 9.1, we see that the RAE approach is reasonably efficient in that the ratio (compared with SDPT3) is at most 6.0 for all the test problems, and 78% of them have ratios between 1.0 and 2.0.

The objective values obtained by the RAE-based IPM are given in Table 9.2.

As we are able to compute rather accurate approximate solutions for (1.1), it is worthwhile to gather information such as primal and dual degeneracy, and strict complementarity for some of the smaller SOCP problems we have considered in this paper. Such information is given in Table 9.3. We note that the degeneracies of the problems are determined by computing the numerical row and column rank (via the MATLAB command `rank`) of the matrices in Theorems 20 and 21 in [1], respectively.

9. Conclusion. We analyzed the accuracy of the search direction computed from the SCE approach, and how the residual norm in the computed solution affects the primal infeasibility and hence the achievable accuracy in the approximate optimal solution.

We also discussed the factors contributing to the good numerical performance of the very well implemented SCE-based IPM in the software SeDuMi.

A reduced augmented equation is proposed to compute the search direction at each IPM iteration when the SCE cannot be solved to sufficient accuracy. The proposed RAE approach can improve the robustness of IPM solvers for SOCP. It can be implemented efficiently by carefully preserving the sparsity structure in the problem data. Numerical results show that the new approach can produce more accurate approximate optimal solutions compared to the SCE approach.

Acknowledgment. The authors are grateful to Professor Robert Freund for valuable suggestions on the manuscript. The authors also thank the Associate Editor, Professor M. J. Todd, and the Referees for many helpful comments and suggestions to improve the paper.

REFERENCES

[1] F. Alizadeh and D. Goldfarb, *Second-order cone programming*, Mathematical Programming, 95 (2003), pp. 3–51.

[2] F. Alizadeh and S.H. Schmieta, *Optimization with Semi-definite, Quadratic and Linear Constraints*, Report 23-97, Rutgers Center for Operations Research, Rutgers University, 1997. Available from `http://rutcor.rutgers.edu/pub/rrr/reports97/23.ps`

[3] E.D. Andersen, C. Roos, and T. Terlaky, *On implementing a primal-dual interior-point method*

for conic quadratic optimization, Mathematical Programming, Series B, 95 (2003),pp. 249-277.

[4] K.D. Andersen, *A modified Schur complement method for handling dense columns in interior point methods for linear programming*, ACM Transactions on Mathematical Software, 22 (1996), pp. 348–356.

[5] K.M. Tsui, S.C., Chan, and K.S. Yeung, *Design of FIR digital filters with prescribed flatness and peak error constraints using second order cone programming*, IEEE Transactions on Circuits and Systems II, 52 (2005), pp. 601–605.

[6] H. Ciria and J. Peraire, *Computation of upper and lower bounds in limit analysis using second-order cone programming and mesh adaptivity*, 9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, 2004.

[7] I.S. Duff and J.K. Reid, *The multifrontal solution of indefinite sparse symmetric linear equations*, ACM Transactions on Mathematical Software, 9 (1983), pp. 302–325.

[8] A. George, and K.H. Ikramov, *On the condition of symmetric quasi-definite matrices*, SIAM J. Matrix Analysis and Applications, 21 (2000), pp. 970–977.

[9] D. Goldfarb, and K. Scheinberg, *A product-form Cholesky factorization implementation of an interior-point method for second order cone programming*, Mathematical Programming, to appear.

[10] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, USA, 1996.

[11] M. Gu, *Primal-dual interior-point methods for semidefinite programming in finite precision*, SIAM J. Optimization, 10 (2000), pp. 462–502.

[12] N.J. Higham, *Accuracy and stability of numerical algorithms*, SIAM, Philadelphia, 1996.

[13] V. Kovacevic-Vujcic and M.D. Asic, *Stabilization of interior-point methods for linear programming*, Computational Optimzation and Applications, 14 (1999), pp. 331–346.

[14] M. S. Lobo, L. Vandenberghe, S. Boyd and H. Lebret, *Applications of Second-order Cone Programming*, Linear Algebra Appl., 284 (1998), pp.193–228.

[15] R.D.C. Monteiro and T. Tsuchiya, *Polynomial Convergence of Primal-Dual Algorithms for the Second-Order Cone Program Based on the MZ-Family of Directions*, Mathematical Programming, 88 (2000), pp. 61–83.

[16] E.G. Ng and B.W. Peyton, *Block sparse Cholesky algorithms on advanced uniprocessor computers*, SIAM J. Scientific and Statistical Computing, 14 (1993), pp. 1034–1056.

[17] G. Pataki and S.H. Schmieta, *The DIMACS Library of Mixed Semidefinite-Quadratic-Linear Programs*,
`http://dimacs.rutgers.edu/Challenges/Seventh/Instances/`

[18] J. Reid, and I.S. Duff, *MA47, a Fortran code for direct solution of indefinite sparse symmetric linear systems*, Report RAL-95-001, Rutherford Appleton Laboratory, Oxfordshire, England, January, 1995.

[19] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[20] D. Scholnik and J. Coleman, *An FIR Filter Optimization Toolbox for Matlab 5 & 6*, available from `http://www.csee.umbc.edu/~dschol2/opt.html`

[21] J.F. Sturm, *Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones*, Optimization Methods and Software, 11 & 12 (1999), pp. 625–653.

[22] J.F. Sturm, *Implementation of interior point methods for mixed semidefinite and second order cone optimization problems*, Optimization Methods and Software, 17 (2002), pp. 1105–1154.

[23] J.F. Sturm, *Avoiding numerical cancelation in the interior point method for solving semidefinite programs*, Mathematical Programming, Series B, 95 (2003), pp. 219–247.

[24] K.C. Toh, M.J. Todd, and R.H. Tütüncü, *SDPT3- A Matlab Software package for Semidefinite Programming*, Optimization Methods and Software, 11 (1999), pp. 545–581.

[25] R.H. Tütüncü, K.C. Toh and M.J. Todd, *Solving Semidefinite-Quadratic-Linear Programming Using SDPT3*, Mathematical Programming, Series B, 95 (2003), pp. 189–217.

[26] K.C. Toh, R.H. Tütüncü, and M.J. Todd, *On the implementation of SDPT3 (version 3.1) – a* MATLAB *software package for semidefinite-quadratic-linear programming*, invited paper, 2004 IEEE Conference on Computer-Aided Control System Design, Taipei, Taiwan.

[27] R.J. Vanderbei, *Symmetric Quasidefinite Matrices*, SIAM J. Optimization, 5 (1995), pp. 100–113.

[28] S.J. Wright, *Stability of linear equations solvers in interior-point methods*, SIAM J. Matrix Analysis and Applications, 16 (1995), pp. 1287–1307.

[29] S.J. Wright, *Stability of augmented system factorizations in interior-point methods*, SIAM J. Matrix Analysis and Applications, 18 (1997), pp. 191–222.

TABLE 9.1

A comparison between 2 SCE-based IPMs and the RAE-based IPM for solving SOCP problems. The last column in the table gives the maximum number of columns in the matrix U in (7.5). The numbers in round brackets are those taken by the iterations using the RAE approach.

	SDPT3		SeDuMi		RAE approach							
problem	ϕ	$T_{\rm iter}$	ϕ	$T_{\rm iter}$	ϕ	$T_{\rm iter}$	iter	Time	relgap	p-inf	d-inf	nc(U)
sched-50-50-o	-4.5	0.18	-7.0	0.17	-10.0	0.26	37 (27)	9.7 (7.2)	10.0-11	1.2-12	1.2-11	86
sched-100-50-o	-4.3	0.39	-6.0	0.37	-10.6	0.90	41 (26)	36.9 (30.3)	2.6-11	1.7-11	1.7-11	494
sched-100-100-o	-2.5	0.76	-3.3	0.71	-9.1	1.94	41 (24)	79.5 (65.2)	7.5-10	7.1-11	1.2-10	245
sched-200-100-o	-4.0	2.08	-3.9	1.60	-10.1	7.30	49 (38)	357.8 (323.4)	8.2-11	4.7-11	1.9-11	578
sched-50-50-s	-6.2	0.17	-8.2	0.24	-10.6	0.28	31 (26)	8.6 (7.3)	2.6-11	4.6-12	4.2-15	85
sched-100-50-s	-7.0	0.41	-8.9	0.41	-10.4	1.15	33 (24)	37.9 (33.2)	4.4-11	9.8-12	6.6-14	495
sched-100-100-s	-5.9	0.76	-7.1	0.63	-9.4	2.12	33 (28)	70.0 (63.7)	3.4-11	4.2-10	2.5-14	254
sched-200-100-s	-6.2	2.23	-7.8	2.33	-10.5	13.32	31 (25)	412.9 (388.8)	2.9-11	3.8-12	2.0-13	578
firL1Linfalph	-9.9	6.92	-4.7	10.71	-9.9	8.23	35 (18)	288.0 (153.8)	4.5-11	1.4-10	0.8-15	10
firL1Linfeps	-10.2	6.67	-10.4	3.83	-10.2	8.22	39 (32)	320.7 (232.9)	5.8-11	1.6-12	7.3-16	646
firL1	-10.1	26.65	-9.0	24.35	-10.1	28.00	23 (0)	644.0 (0.0)	2.3-11	7.3-11	1.0-15	0
firL2a	-10.3	4.37	-12.6	4.29	-10.3	4.57	8 (0)	36.6 (0.0)	5.0-11	0.8-15	0.9-16	2
firL2L1alph	-10.1	5.13	-3.3	5.24	-10.1	5.27	23 (18)	121.1 (72.4)	8.0-11	1.6-11	6.2-16	4
firL2L1eps	-10.4	9.56	-9.3	9.51	-10.5	13.86	19 (13)	263.3 (185.0)	3.4-11	2.4-11	0.9-15	1
firL2Linfalph	-10.1	3.79	-9.5	8.88	-10.1	6.60	34 (29)	224.5 (201.6)	7.8-11	6.4-14	7.3-16	19
firL2Linfeps	-10.2	16.86	-9.1	7.74	-10.1	17.23	22 (0)	379.1 (0.0)	7.1-11	4.2-12	6.7-16	2
firL2	-11.3	0.03	-13.1	0.02	-11.3	0.04	8 (3)	0.3 (0.1)	5.2-12	3.3-16	2.0-16	1
firLinf	-8.9	17.30	-9.3	34.48	-8.9	22.40	29 (18)	649.6 (443.4)	3.6-10	1.3 -9	1.0-15	170
rand200-300-1	-7.6	0.22	-6.4	0.59	-10.5	0.26	14 (5)	3.6 (1.5)	3.3-11	1.9-13	5.9-15	47
rand200-300-2	-6.0	0.23	-5.0	1.07	-10.1	0.28	16 (6)	4.5 (2.0)	8.1-11	4.6-14	5.3-14	62
rand200-800-1	-5.6	0.45	-5.0	2.11	-10.1	0.54	16 (8)	8.7 (4.7)	7.6-11	1.2-14	1.8-14	19
rand200-800-2	-4.9	0.45	-5.8	4.67	-10.1	0.55	18 (8)	9.9 (4.8)	8.1-11	1.0-13	7.6-14	19
rand400-800-1	-5.8	1.60	-5.1	2.63	-10.3	1.80	14 (6)	25.3 (11.5)	5.3-11	5.4-14	2.6-14	40
rand400-800-2	-5.8	1.68	-4.5	5.66	-10.4	1.80	15 (6)	27.0 (11.6)	3.9-11	2.0-13	5.1-14	40
rand700-1e3-1	-6.1	5.67	-5.7	10.93	-10.1	6.42	17 (8)	109.1 (55.1)	7.3-11	4.5-14	2.4-14	123
rand700-1e3-2	-5.3	5.84	-4.6	18.07	-10.1	6.59	20 (10)	131.8 (71.6)	8.6-11	5.2-13	5.8-14	151
rand1000-2e3	-5.7	20.95	-5.0	50.00	-11.3	24.66	14 (6)	345.3 (151.3)	4.7-12	1.6-13	6.9-14	100
rand1500-3e3	-7.0	67.69	-7.0	192.69	-10.1	69.27	15 (6)	1039.0 (388.9)	8.5-11	6.9-13	9.2-14	450

[30] Y. Ye, M.J. Todd, and S. Mizuno, *An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm*, Math. Oper. Res., 19 (1994), pp. 53–67

TABLE 9.2
Primal and dual objective values obtained by the IPM using the RAE approach.

Problem	primal objective	dual objective	Problem	primal objective	dual objective
sched-50-50-o	2.6673000979 4	2.6673000977 4	firL2Linfalph	-7.0591166471 -3	-7.0591167258 -3
sched-100-50-o	1.8188993937 5	1.8188993936 5	firL2Linfeps	-1.4892049051 -3	-1.4892049762 -3
sched-100-100-o	7.1736778669 5	7.1736778615 5	firL2	-3.1186645862 -3	-3.1186645914 -3
sched-200-100-o	1.4136044650 5	1.4136044649 5	firLinf	-1.0068176528 -2	-1.0068176895 -2
sched-50-50-s	7.8520384401 0	7.8520384399 0	rand200-300-1	-1.5094030119 2	-1.5094030119 2
sched-100-50-s	6.7165031103 1	6.7165031100 1	rand200-300-2	-1.2861024800 2	-1.2861024801 2
sched-100-100-s	2.7330785593 1	2.7330785592 1	rand200-800-1	1.8086048337 0	1.8086048335 0
sched-200-100-s	5.1811961028 1	5.1811961027 1	rand200-800-2	-2.3277765222 1	-2.3277765220 1
firL1Linfalph	-3.0673166232 -3	-3.0673166686 -3	rand400-800-1	6.6607764189 0	6.6607764185 0
firL1Linfeps	-2.7112896665 -3	-2.7112897249 -3	rand400-800-2	6.3708631137 1	6.3708631135 1
firL1	-2.9257813804 -4	-2.9257816083 -4	rand700-1e3-1	-7.1501954797 1	-7.1501954802 1
firL2a	-7.1457742547 -4	-7.1457747536 -4	rand700-1e3-2	-5.5374169002 1	-5.5374169007 1
firL2L1alph	-5.7634914619 -5	-5.7634994782 -5	rand1000-2e3	-2.4138366508 4	-2.4138366508 4
firL2L1eps	-8.4481294535 -4	-8.4481297976 -4	rand1500-3e3	1.7396653464 4	1.7396653465 4

TABLE 9.3
Primal nondegeneracy ("p.n.d") and dual nondegeneracy ("d.n.d"), and strict complementarity ("s.c.") information of approximate solutions of some SOCPs. A "1" means true and a "0" means false. A number of the form (34/35) *in the second column means that at the computed approximate optimal solution, the column rank of* $\widetilde{A}_1$ *is* 34*, and the number of columns in* $\widetilde{A}_1$ *is* 35*.*

Problem	p.n.d	d.n.d	s.c.	Problem	p.n.d	d.n.d	s.c.
sched-50-50-orig	1	1 (79/79)	1	rand200_800_1	1	1 (19/19)	1
sched-50-50-scaled	1	1 (83/83)	1	rand200_800_2	1	1 (19/19)	1
firL2a	1	1 (1/1)	1	rand400_800_1	1	1 (40/40)	1
firL2Linfalph	1	1 (15/15)	1	rand400_800_2	1	1 (40/40)	1
firL2	1	1 (1/1)	1	rand700_1e3_1	1	0 (84/85)	1
rand200_300_1	1	0 (34/35)	1	rand700_1e3_2	1	0 (126/130)	1
rand200_300_2	1	0 (62/65)	1				