On Two Measures of Problem Instance Complexity and their Correlation with the Performance of SeDuMi on Second-Order Cone Problems*

Zhi Cai †and Robert M. Freund ‡

September 13, 2004

Abstract

We evaluate the practical relevance of two measures of conic convex problem complexity as applied to second-order cone problems solved using the homogeneous self-dual (HSD) embedding model in the software SeDuMi. The first measure we evaluate is Renegar's data-based condition measure $C(d)$, and the second measure is a combined measure of the optimal solution size and the initial infeasibility/optimality residuals denoted by S (where the solution size is measured in a norm that is naturally associated with the HSD model). We constructed a set of 144 second-order cone test problems with widely distributed values of $C(d)$ and S and solved these problems using SeDuMi. For each problem instance in the test set, we also computed estimates of $C(d)$ (using Peña's method) and computed S directly. Our computational experience indicates that

*This research has been partially supported through the MIT-Singapore Alliance.

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

‡MIT Sloan School of Management, 50 Memorial Drive, Cambridge, MA 01239, USA. (rfreund@mit.edu).

SeDuMi iteration counts and $\log(C(d))$ are fairly highly correlated (sample correlation $R = 0.676$), whereas SeDuMi iteration counts are not quite as highly correlated with S ($R = 0.600$). Furthermore, the experimental evidence indicates that the average rate of convergence of SeDuMi iterations is affected by the condition number $C(d)$ of the problem instance, a phenomenon that makes some intuitive sense yet is not directly implied by existing theory.

1 Introduction

The homogeneous self-dual (HSD) embedding model for linear optimization was originally developed by Ye, Todd, and Mizuno in [14], and has been extended to the conic case and implemented in software such as SeDuMi [13]. The HSD model has the very desirable property that it always has a strictly feasible primal/dual solution regardless of the feasibility of the original problem. Using a natural norm associated with the HSD model's starting point, the norms of approximately optimal solutions and their distances from the boundaries of the underlying cones are precisely controlled independent of the problem instance, see [5]. Furthermore, a reinterpretation of a standard stopping criterion for HSD interior-point methods shows that the performance of these methods is inherently related to the sizes of optimal solutions and the sizes of the initial infeasibilities, also see [5].

In this paper we evaluate the relevance of two measures of conic convex problem complexity as applied to second-order cone problems solved using the homogeneous self-dual (HSD) embedding model in the software SeDuMi. The first measure we evaluate is Renegar's data-based condition measure $C(d)$, and the second measure is a combined measure of the optimal solution size and the initial infeasibility/optimality residuals denoted by S (where the solution size is measured in a norm that is naturally associated with the HSD model).

Consider the primal-dual conic linear system:

$$\begin{array}{ll} (P_c) & \min_x\{c^T x \mid Ax = b,\ x \in C_X\} \\ (D_c) & \max_{y,z}\{b^T y \mid A^T y + z = c,\ z \in C_X^*\} \end{array} \tag{1}$$

where C_X is a closed convex cone and C_X^* is the corresponding dual cone. The complexity of computing approximately optimal solutions of (P_c) has been developed along two related approaches. The first approach is via the data-based condition measure theory of Renegar [1, 2, 9, 10, 11, 12]. The problem data d is the triplet $d = (A, b, c)$. The "distance to infeasibility" $\rho(d)$ is the minimum data perturbation Δd that renders either the perturbed primal (P_c) or dual (D_c) infeasible. The condition measure $C(d)$ is defined to be the ratio $\|d\|/\rho(d)$. This condition measure play a key role in the complexity analysis of (P_c) in [12]. In theory, the number of iterations of a suitably designed interior-point method (IPM) algorithm (but not the HSD model) needed to approximately solve (P_c) is bounded by $O(\sqrt{\vartheta}\log(C(d) + \ldots))$, see [12], where ϑ is the self-concordance parameter of the barrier function used for the cone C_X. Two efficient methods for estimating $C(d)$ have been developed, see Peña [8] and [3], [7].

The second approach to developing a complexity theory for (P_c) is via geometric measures of the problem, using quantities such as the norm of the largest ϵ-optimal primal and dual solutions. Let R_ϵ^P denote the norm of the largest ϵ-optimal solution of (P_c), with R_ϵ^D defined analogously for the dual cone variables z, and let $R_\epsilon := R_\epsilon^P + R_\epsilon^D$. These quantities appear in the complexity analysis of (P_c) in [4] as well as in [5]. In theory, with a choice of norm naturally connected to the starting point of the HSD model, the number of iterations of a suitably designed interior-point method (IPM) needed to approximately solve (P_c) via the HSD model is bounded by $O(\sqrt{\vartheta}\log(R_\epsilon + \ldots))$, see [5]. If the norm is chosen judiciously, then R_ϵ can be computed efficiently.

The explanatory value of $C(d)$ for non-HSD IPM algorithms was first explored in [7], which examined the relationship between the condition measure $C(d)$ and the IPM iterations of the CPLEX barrier software (a commercial IPM solver) on linear programming problems from the NETLIB suite. It was observed that 42% of the variation in the IPM iteration counts among the NETLIB suite problems is accounted for by $\log(C(d))$. The analysis in [7] was limited to linear programming instances, whose performance with interior-point methods tends to be different from more general conic linear systems.

In this paper, we explore the relationship between the condition measure $C(d)$ and the combined solution size and initial infeasibility measure S, and the number of iterations needed to approximately solve an SOCP problem using the HSD embedding in the IPM code SeDuMi. We constructed a set of 144 second-order cone test problems with widely distributed values of $C(d)$ and S and solved these problems using SeDuMi. For each problem instance in the test set, we also computed estimates of $C(d)$ (using Peña's method) and computed S directly. Our computational experience indicates that SeDuMi iteration counts and $\log(C(d))$ are fairly highly correlated (sample correlation $R = 0.676$), whereas SeDuMi iteration counts are not quite as highly correlated with S ($R = 0.600$). Furthermore, the experimental evidence indicates that the average rate of convergence of SeDuMi iterations is affected by the condition number $C(d)$ of the problem instance, a phenomenon that makes some intuitive sense yet is not directly implied by existing theory.

The paper is organized as following: Section 2 presents the notation for the standard form second-order cone optimization problem and the homogeneous self-dual embedding model. Section 3 describes the details of computing the condition measure $C(d)$ of a problem instance using Peña's method. Section 4 contains the method for creating the test problems and the computational evaluation of the correlation between $C(d)$ and SeDuMi iterations. Section 5 presents the combined measure of optimal solution size and initial infeasibility gap denoted by S, and the computational evaluation of the correlation between S and SeDuMi iterations. Section 6 contains brief concluding remarks.

Acknowledgement. We are grateful to Kim Chuan Toh for computational advice and for reading and editing an earlier draft of this paper.

2 Preliminaries

This section defines a standard SOCP problem, followed by a brief review of the homogeneous self-dual embedding model that is used in SeDuMi.

2.1 SOCP in standard form

The standard second-order cone in $I\!R^k$ is defined to be

$$K^k_{\text{SOC}} := \{v = (v^0, \bar{v}) \in I\!R \times I\!R^{k-1} : \|\bar{v}\|_2 \;\; \leq \;\; v^0\} \ ,$$

and we write "$v \succeq 0$" if $v \in K^k_{\text{SOC}}$. The standard form SOCP primal and dual problems are:

$$\begin{array}{lll}
\text{(P)}\;\; v^* := & \min & (c^l)^T x^l + (c^q)^T x^q \\
\text{s.t.} & & A^l x^l + A^q x^q \;=\; b \\
& & x^l_i \geq 0, \quad i = 1, \ldots, N^l, \quad x^q_i \succeq 0, \quad i = 1, \ldots, N^q \\
\text{(D)}\;\; w^* := & \max & b^T y \\
\text{s.t.} & & (A^l)^T y + z^l \;=\; c^l, \\
& & (A^q)^T y + z^q \;=\; c^q, \\
& & z^l_i \geq 0, \quad i = 1, \ldots, N^l, \quad z^q_i \succeq 0, \quad i = 1, \ldots, N^q
\end{array} \tag{2}$$

where $y \in R^m$ and the superscript "l" indicates the linear variables and the coefficients related to the linear variables: $x^l = [x^l_1; \ldots; x^l_{N^l}]$, N^l is the number of linear variables, and $A^l \in R^{m \times N^l}$ and $c^l \in R^{N^l}$ are the matrices and objective function vectors associated with the linear variables. Similarly, the superscript "q" indicates the second-order cone variables and the coefficients related to the second-order cone variables: N^q is the number of second-order cones, and n^q_i is the dimension of the i^{th} second-order cone and we write $x^q_i \in R^{n^q_i}$, $i = 1, \ldots, N^q$ and $x^q = [x^q_1; \ldots; x^q_{N^q}]$. The total number of second-order cone variables is denoted as $n^q = \sum_{i=1}^{N^q} n^q_i$. The matrices and objective function vectors associated with the second-order cone variables are $A^q \in R^{m \times n^q}$, $c^q \in R^{n^q}$. Analogous notation is used for the dual problem.

Let $N = N^l + N^q$, $A = [A^l \; A^q]$, $c = [c^l; c^q]$, $x = [x^l; x^q]$ and $z = [z^l; z^q]$. $x \succeq 0$ ($x \succ 0$) means that all x_i^l and x_i^q are in (the interior of) their defined cones. We also define, for $x_i^q, z_i^q \succeq 0$,

$$\gamma(x_i^q) = \sqrt{(x_i^0)^2 - \|\bar{x}_i\|_2^2}, \quad \gamma(z_i^q) = \sqrt{(z_i^0)^2 - \|\bar{z}_i\|_2^2}. \tag{3}$$

The self-concordant barrier function associated with (P) is

$$f(x) := -\sum_{i=1}^{N_l} \ln(x_i^l) - \sum_{i=1}^{N_q} \ln(\gamma^2(x_i^q))$$

whose complexity value ϑ is:

$$\vartheta = N^l + 2N^q \ , \tag{4}$$

see [6].

2.2 The Homogeneous Self-Dual Embedding Model

IPM solvers that apply the homogeneous self-dual embedding model embed the primal and dual problems into the self-dual optimization problem:

$$\begin{array}{llccccccc}
(HSD): & \min_{x,y,z,\tau,\kappa,\theta} & & & & \bar{\alpha}\theta & & & \\
\text{s.t.} & & & -Ax & +b\tau & -\bar{b}\theta & & & = 0 \\
& & A^T y & & -c\tau & -\bar{c}\theta & +z & & = 0 \\
& & -b^T y & +c^T x & & -\bar{g}\theta & & +\kappa & = 0 \\
& & \bar{b}^T y & +\bar{c}^T x & +\bar{g}\tau & & & & = \bar{\alpha} \\
& & & x \succeq 0 & \tau \ge 0 & & z \succeq 0 & \kappa \ge 0 \ . &
\end{array} \tag{5}$$

Let $(x^{(0)}, y^{(0)}, z^{(0)}, \tau^{(0)}, \kappa^{(0)}, \theta^{(0)})$ satisfy $x^{(0)} \succ 0$, $z^{(0)} \succ 0$, $\tau^{(0)} > 0$, $\kappa^{(0)} > 0$, and $\theta^{(0)} > 0$, and let $\bar{b}, \bar{c}, \bar{g}, \bar{\alpha}$ be defined as follows:

$$\begin{aligned}
\bar{b} &= \frac{b\tau^{(0)} - Ax^{(0)}}{\theta^{(0)}}; \quad \bar{c} = \frac{A^T y^{(0)} + z^{(0)} - c\tau^{(0)}}{\theta^{(0)}}; \\
\bar{g} &= \frac{c^T x^{(0)} - b^T y^{(0)} + \kappa^{(0)}}{\theta^{(0)}}; \quad \bar{\alpha} = \frac{(x^{(0)})^T z^{(0)} + \tau^{(0)}\kappa^{(0)}}{\theta^{(0)}}.
\end{aligned}$$

If the original problem is primal and dual feasible, (HSD) has an optimal solution $(x^*, y^*, z^*, \tau^*, \kappa^*, \theta^*)$ satisfying $\theta^* = 0$ and $\tau^* > 0$. In this case x^*/τ^* and $(y^*/\tau^*, z^*/\tau^*)$ will be optimal solutions to the original problems (P) and (D), respectively.

Let $(x, y, z, \tau, \kappa, \theta)$ be a feasible solution of HSD satisfying $\tau > 0$. Then the current test solutions for (P) and (D) are given by $\bar{x} := x/\tau$ and $(\bar{y}, \bar{z}) := (y/\tau, z/\tau)$, respectively. HSD IPM solvers, including SeDuMi, compute iterates until the primal and dual infeasibility and duality gaps of the current iterates' test solution are small. These are given by:

$$\begin{aligned} r_p &= b - A\bar{x} && \text{(primal infeasibility gap)} \\ r_d &= A^T\bar{y} + \bar{z} - c && \text{(dual infeasibility gap)} \\ r_g &= c^T\bar{x} - b^T\bar{y} && \text{(duality gap)} \end{aligned} \tag{6}$$

For example, SeDuMi uses the following criterion for terminating the algorithm:

$$2\frac{\|r_p\|_\infty}{1 + \|b\|_\infty} + 2\frac{\|r_d\|_\infty}{1 + \|c\|_\infty} + \frac{(r_g)^+}{\max(|c^T x|, |b^T y|, 0.001 \times \tau)} \;\leq\; r_{max} \;, \tag{7}$$

where $r_{max} = 10^{-9}$ is the default value.

3 Computing the Condition Measure of an SOCP Problem Instance Using Peña's Method

Consider the primal feasibility conditions associated with (1):

$$Ax = b \;, \quad x \in C_X \;. \tag{8}$$

The data for (1) is $d = (A, b, c)$, the data for the primal feasibility problem is $d_P = (A, b)$, and consider the norm on d_P the data given by

$$\|d_P\| := \|[A, \; -b]\|_2 := \max\{\|Ax - bt\| : \|x\|^2 + t^2 \;\; \leq \;\; 1\} \;,$$

where the Euclidean norm is used for all vectors in the above expression. Let

$$\mathcal{I}_P := \{(A, b) : \text{the system (8) is infeasible}\} \;.$$

The distance to primal infeasibility $\rho_P(d)$ is defined to be the norm of the smallest data perturbation $\Delta d_P = (\Delta A, \Delta b)$ that renders the resulting primal system infeasible:

$$\rho_P(d_P) := \inf\{\|[\Delta A, \Delta b]\|_2 : (A + \Delta A, b + \Delta b) \in \mathcal{I}_P\} \ .$$

We consider the dual problem in a similar way. The dual conic system is:

$$A^T y + z = c \ , \quad z \in C_X^* \ . \tag{9}$$

The data for this system is $d_D = (A, c)$, and consider the norm on the data given by

$$\|d_D\| := \|[A, \ -c]\|_2 := \max\{\|A^T y - ct\| : \|y\|^2 + t^2 \ \ \leq \ 1\} \ .$$

Let

$$\mathcal{I}_D := \{(A, c) : \text{the system (9) is infeasible}\} \ .$$

The distance to dual infeasibility $\rho_D(d)$ is defined to be the norm of the smallest data perturbation $\Delta d_D = (\Delta A, \Delta c)$ that renders the resulting dual system infeasible:

$$\rho_D(d_D) := \inf\{\|[\Delta A, \Delta c]\|_2 : (A + \Delta A, c + \Delta c) \in \mathcal{I}_D\} \ .$$

Since the primal (dual) distance to infeasibility is independent of the data c (b), we write $\rho_P(d) := \rho_P(d_P)$ and $\rho_D(d) := \rho_D(d_D)$. The condition measure $C(d)$ of the primal/dual system is the ratio:

$$C(d) = \frac{\|d\|}{\min\{\rho_P(d), \rho_D(d)\}} \ ,$$

where $\rho_P(d)$ and $\rho_D(d)$ are the primal and dual distances to infeasibility, respectively, $d = (A, b, c)$ is the data for the primal and dual problems, and $\|d\|$ is defined as:

$$\|d\| = \max\{\|d_P\| \, , \ \|d_D\|\} \ . \tag{10}$$

We remark that $C(d)$ is connected to a wide variety of behavioral, geometric, and algorithmic complexity bounds on the problems (P) and (D), see [11, 12, 1, 2, 3, 8, 9, 10], for example.

Computing $C(d)$ involves computing the four quantities $\|d_P\|$, $\|d_D\|$, $\rho_P(d)$, $\rho_D(d)$. Notice that the first two quantities are maximum eigenvalue computations, for example, $\|d_P\| = \|[A, -b]\|_2 = \sqrt{\lambda_{\max}([A, -b][A, -b]^T)}$, and so pose little computational burden. However, the primal and dual distances to infeasibility $\rho_P(d), \rho_D(d)$ are not as straightforward to compute. Peña [8] presents an efficient method for computing lower and upper bounds on the primal and/or dual distance to infeasibility that involves the solution of six convex optimization problems each of whose computational cost is similar to that of the original primal and/or dual problems. The computational details of Peña's method are summarized in the following subsections. For a theoretical justification of the method, see [8].

3.1 Estimating the distance to infeasibility $\rho_P(d)$ of the primal problem

In Peña's method, the estimation of the distance to infeasibility $\rho_P(d)$ involves the following steps.

Step 1: Compute the analytic center (x_*, t_*) of the homogenized primal feasible region by solving the following problem:

$$\begin{array}{lll} (\text{P1}) & \min_{x,t} & f(x,t) := -\ln\left(1 - t^2 - \|x\|_2^2\right) - \ln(t) - \sum_{i=1}^{N^l} \ln\left(x_i^l\right) - \sum_{i=1}^{N^q} \ln\left(\gamma^2(x_i^q)\right) \\ \text{s.t.} & & Ax - bt = 0, \quad x \succ 0, \quad t > 0 \ . \end{array} \tag{11}$$

Step 2: Compute the minimum eigenvalue λ_P and its corresponding unit eigenvector V_P of the matrix $M_P = [A, -b]H(x_*, t_*)^{-1}[A, -b]^T$, where $H(x, t)$ is the Hessian of $f(x, t)$.

Step 3: Compute optimal solutions $(x_\pm^P, t_\pm^P)$ of the following two problems:

$$\begin{array}{lll} (\text{P2}) \ \delta_\pm^P := & \min & \delta \\ \text{s.t.} & & Ax - bt \ = \ \pm V_P, \\ & & x \succeq 0, \quad t \geq 0, \\ & & \sqrt{t^2 + \|x\|_2^2} \leq \delta \ . \end{array} \tag{12}$$

Step 4: Compute lower and upper bound for $\rho_P(d)$ and the associated perturbation to infeasibility:

$$L_P(d) =: \sqrt{\lambda_P} \quad \leq \quad \rho_P(d) \quad \leq \quad \min\left\{\frac{1}{\delta_+^P}, \frac{1}{\delta_-^P}, \vartheta\sqrt{\lambda_P}\right\} := U_P(d). \tag{13}$$

$$(\Delta A_P, \Delta b_P) := \begin{cases} \left(\frac{1}{\delta_+^P}\right)^2 (-V_P(x_+^P)^T, V_P t_+^P) & \text{if } \delta_+^P > \delta_-^P \\ \\ \left(\frac{1}{\delta_-^P}\right)^2 (V_P(x_-^P)^T, -V_P t_-^P) & \text{if } \delta_+^P \ \leq \ \delta_-^P \ . \end{cases} \tag{14}$$

According to Propositions 3.1 and 4.1 of [9], and by using some intermediary results of Peña's method, it follows that $(A + (1+\epsilon)\Delta A_P, b + (1+\epsilon)\Delta b_P) \in \mathcal{I}_\mathcal{P}$ for all $\epsilon > 0$.

3.2 Estimating the distance to infeasibility $\rho_D(d)$ of the dual problem

The estimation of the distance to infeasibility $\rho_D(d)$ is quite similar to that for $\rho_P(d)$, and involves the following steps.

Step 1: Compute the analytic center (y_*, p_*, z_*) of the homogenized dual feasible region by solving the following problem:

$$\begin{array}{lll} (\text{D1}) & \min_{y,p,z} & -\ln\left(1 - p^2 - \|y\|_2^2\right) - \ln(p) - \sum_{i=1}^{N^l} \ln\left(z_i^l\right) - \sum_{i=1}^{N^q} \ln\left(\gamma^2(z_i^q)\right) \\ \text{s.t.} & & A^T y + z - cp \ = \ 0, \quad z \succ 0, \quad p > 0. \end{array} \tag{15}$$

Step 2: Compute the minimum eigenvalue λ_D and its corresponding unit eigenvector V_D of the matrix $M_D = [A^T, -c]H(y_*, p_*)^{-1}[A^T, -c]^T + H(z_*)$, where $H(y,p)$ is the Hessian of the function $f(y,p) = -\ln(1 - \|y\|_2^2 - p^2) - \ln(p)$ and $H(z)$ is the Hessian of the function $f(z) = -\sum_{i=1}^{N^l} \ln\left(z_i^l\right) - \sum_{i=1}^{N^q} \ln\left(\gamma^2(z_i^q)\right)$.

Step 3: Compute optimal solutions $y_\pm^P, p_\pm^P, z_\pm^P$ of the following two problems:

$$\begin{array}{lll} (\text{P2}) \ \ \delta_\pm^D := & \min & \delta \\ \text{s.t.} & & A^T y + z - cp \ = \ \pm V_D, \\ & & z \succeq 0, \quad p \geq 0, \\ & & \sqrt{p^2 + \|y\|_2^2} \leq \delta \ . \end{array} \tag{16}$$

Step 4: Compute lower and upper bound for $\rho_D(d)$ and the associated perturbation to infeasibility:

$$L_D(d) =: \sqrt{\lambda_D} \quad \leq \quad \rho_D(d) \quad \leq \quad \min\left\{\frac{1}{\delta_+^D}, \frac{1}{\delta_-^D}, \vartheta\sqrt{\lambda_D}\right\} := U_D(d). \tag{17}$$

$$(\Delta A_D, \Delta c_D) := \begin{cases} \left(\frac{1}{\delta_+^D}\right)^2 (-y_+^D (V_D)^T, V_D p_+^D) & \text{if } \delta_+^D > \delta_-^D \\ \\ \left(\frac{1}{\delta_-^D}\right)^2 (y_-^D (V_D)^T, -V_D p_-^D) & \text{if } \delta_+^D \ \leq \ \delta_-^D \ . \end{cases} \tag{18}$$

As in the case of the primal distance to infeasibility, it follows that $(A+(1+\epsilon)\Delta A_D, c+(1+\epsilon)\Delta c_D) \in \mathcal{I_D}$ for all $\epsilon > 0$.

3.3 Estimating the Condition Measure and Computing the Associated Perturbation to Infeasibility

Once the bounds on $\rho_P(d)$ and $\rho_D(d)$ have been computed, lower and upper bounds for $C(d)$ are computed as follows:

$$C_L(d) := \frac{\|d\|}{\min\{U_P(d), U_D(d)\}} \quad \leq \quad C(d) \quad \leq \quad \frac{\|d\|}{\min\{L_P(d), L_D(d)\}} = C_U(d). \tag{19}$$

Finally, the associated perturbation associated with the above estimation procedure is:

$$(\Delta A, \Delta b, \Delta c) := \begin{cases} (\Delta A_P, \Delta b_P, 0) & \text{if } U_P(d) < U_D(d) \\ \\ (\Delta A_D, 0, \Delta c_D) & \text{if } U_P(d) \geq U_D(d) \ . \end{cases} \tag{20}$$

Using this perturbation, it is straightforward to show that

$$\min\{\rho_P(d+\alpha\Delta d), \rho_D(d+\alpha\Delta d)\} \quad \leq \quad (1-\alpha)\min\{U_P(d), U_D(d)\} \quad \text{for all } \alpha \in [0,1] \ . \tag{21}$$

4 Test Problems, Condition Measures, and Correlation with SeDuMi Iterations

We created 144 SOCP test problem instances with widely varying condition measures $C(d)$ as follows. We first created 12 random SOCP instances whose dimensions are described in Table 1. For each problem instance $d = (A, b, c)$ in this group of 12, we computed the condition measure $C(d)$ and the associated perturbation to infeasibility $\Delta d := (\Delta A, \Delta b, \Delta c)$ using Peña's method as described in Section 3. We then used this perturbation to create 11 increasingly ill-conditioned instances of the form $(A, b, c)+\alpha(\Delta A, \Delta b, \Delta c)$ for the 11 different values of $\alpha \in (0, 1)$, namely $\alpha = 0.1$, 0.5, 0.75, 0.9, 0.95, 0.97, 0.99, 0.995, 0.999, 0.9995, and 0.9999. Note from (21) that the distance to infeasibility of these 11 new instances will approach 0 as α approaches 1, thereby generating problems that are increasingly ill-conditioned. We applied this procedure to each of the 12 random SOCP problem instances, thus creating a total of 144 SOCP problems with widely varying condition measures. These problems were named according to their original instance from the first column of Table 1 and the value of α that was used to create the perturbed problem instance; for example problem instance sm_18_999 was created by perturbing problem instance sm_18 by its associated perturbation to infeasibility using $\alpha = 0.999$.

We computed the estimates of the condition measure $C(d)$ for each of the 144 problem instances using Peña's method. Table 2 shows the computed lower and upper bounds C_L and C_U, respectively, as well as the logarithm of their geometric mean $\bar{C} = \sqrt{C_L C_U}$. (all logarithm values are base 10). Notice that $\bar{C}$ ranges from 10^2 to 10^9. Also notice in general that in each problem group both C_L and C_U grow inversely proportional to $(1 - \alpha)$, where α is the extent of perturbation in (21). For example, C_L and C_U for "sm_18_9995" are approximately 2 times larger than the values for "sm_18_999" and are 10 times larger than the values for "sm_18_995". For those problems with C_U close to 10^9, the minimum eigenvalues of the associated matrices are in the range 10^{-15}

Table 1: Problem dimensions of 12 randomly generated SOCP problem instances. In the column "SOC Dimensions", "8×3" means 8 cones of dimension 3, for example.

Problem Instance	Rows (m)	Variables (n)	Nonnegative Variables	Second-Order Cones	SOC Dimensions
sm_18	54	913	411	69	[8×3, 7×4, 5×5, 7×6, 7×7
					8×8 11×9 8×10, 5×11, 3×12]
sm_19	55	228	57	42	[15×3, 13×4, 10×5, 4×6]
sm2_3	2	54	9	15	[15×3]
sm_5	60	912	429	66	[8×3 4×4, 10×5, 6×6, 9×7
					4×8 7×9 7×10, 3×11, 8×12]
md_1	57	409	67	114	[114×3]
lg_1	100	512	7	2	[1×5, 1×500]
md_2	6	733	91	214	[214×3]
sm2_1	7	1297	16	427	[427×3]
md_3	25	1060	10	105	[105×10]
md_5	60	2010	10	200	[200×10]
md_4	25	3010	10	300	[300×10]
md_6	100	2010	10	200	[200×10]

to 10^{-16}, which are close to the machine ϵ of 2.2×10^{-16}. As one would expect, the computational errors introduced in the analytic center computation or/and the smallest eigenvalue computation can easily affect the relative accuracy of the eigenvalue computed for these problems.

Table 2: Condition Measure Estimates for 144 SOCP Problem Instances.

Problem Instance	C_U	C_L	$\log(\bar{C})$	Problem Instance	C_U	C_L	$\log(\bar{C})$
sm_18	1.5E+5	1.1E+4	4.6	md_2	5.3E+4	3.2E+3	4.1
sm_18_1	1.6E+5	1.5E+4	4.7	md_2_1	6.0E+4	4.1E+3	4.2
sm_18_5	2.4E+5	3.1E+4	4.9	md_2_5	5.9E+4	9.3E+3	4.4
sm_18_75	4.4E+5	6.3E+4	5.2	md_2_75	8.4E+4	2.0E+4	4.6
sm_18_9	1.0E+6	1.6E+5	5.6	md_2_9	1.6E+5	5.1E+4	5.0
sm_18_95	1.9E+6	3.2E+5	5.9	md_2_95	2.8E+5	1.0E+5	5.2
sm_18_97	3.1E+6	5.3E+5	6.1	md_2_97	4.4E+5	1.7E+5	5.4
sm_18_99	8.9E+6	1.6E+6	6.6	md_2_99	1.2E+6	5.0E+5	5.9
sm_18_995	1.8E+7	3.2E+6	6.9	md_2_995	2.5E+6	1.0E+6	6.2
sm_18_999	8.8E+7	1.6E+7	7.6	md_2_999	1.2E+7	5.0E+6	6.9
sm_18_9995	1.8E+8	3.2E+7	7.9	md_2_9995	2.5E+7	1.0E+7	7.2
sm_18_9999	8.1E+8	1.6E+8	8.6	md_2_9999	1.2E+8	5.0E+7	7.9
sm_19	1.1E+4	1.3E+3	3.6	sm2_1	1.3E+5	6.9E+3	4.5
sm_19_1	1.2E+4	1.6E+3	3.6	sm2_1_1	1.5E+5	1.0E+4	4.6
sm_19_5	2.2E+4	3.0E+3	3.9	sm2_1_5	2.1E+5	2.3E+4	4.8
sm_19_75	3.9E+4	6.0E+3	4.2	sm2_1_75	3.0E+5	4.7E+4	5.1
sm_19_9	9.1E+4	1.5E+4	4.6	sm2_1_9	4.3E+5	1.2E+5	5.4
sm_19_95	1.8E+5	3.0E+4	4.9	sm2_1_95	7.7E+5	2.4E+5	5.6
sm_19_97	3.0E+5	5.1E+4	5.1	sm2_1_97	1.4E+6	4.0E+5	5.9
sm_19_99	8.9E+5	1.5E+5	5.6	sm2_1_99	3.5E+6	1.2E+6	6.3
sm_19_995	1.8E+6	3.0E+5	5.9	sm2_1_995	6.4E+6	2.4E+6	6.6
sm_19_999	8.9E+6	1.5E+6	6.6	sm2_1_999	3.4E+7	1.2E+7	7.3
sm_19_9995	1.8E+7	3.0E+6	6.9	sm2_1_9995	6.9E+7	2.4E+7	7.6
sm_19_9999	8.9E+7	1.5E+7	7.6	sm2_1_9999	3.4E+8	1.2E+8	8.3
sm2_3	1.6E+2	1.3E+2	2.2	md_3	1.6E+5	2.8E+4	4.8
sm2_3_1	1.8E+2	1.4E+2	2.2	md_3_1	2.0E+5	3.3E+4	4.9
sm2_3_5	3.8E+2	2.6E+2	2.5	md_3_5	3.8E+5	6.6E+4	5.2
sm2_3_75	9.2E+2	5.2E+2	2.8	md_3_75	6.8E+5	1.3E+5	5.5
sm2_3_9	2.3E+3	1.3E+3	3.2	md_3_9	1.5E+6	3.4E+5	5.9
sm2_3_95	4.2E+3	2.6E+3	3.5	md_3_95	2.7E+6	6.8E+5	6.1

Problem Instance	C_U	C_L	$\log(\bar{C})$	Problem Instance	C_U	C_L	$\log(\bar{C})$
sm2_3_97	6.7E+3	4.4E+3	3.7	md_3_97	4.2E+6	1.1E+6	6.3
sm2_3_99	1.9E+4	1.3E+4	4.2	md_3_99	1.2E+7	3.4E+6	6.8
sm2_3_995	3.9E+4	2.6E+4	4.5	md_3_995	2.3E+7	6.8E+6	7.1
sm2_3_999	1.9E+5	1.3E+5	5.2	md_3_999	1.1E+8	3.4E+7	7.8
sm2_3_9995	3.8E+5	2.6E+5	5.5	md_3_9995	2.3E+8	6.8E+7	8.1
sm2_3_9999	1.9E+6	1.3E+6	6.2	md_3_9999	9.0E+8	3.4E+8	8.7
sm_5	1.3E+5	1.0E+4	4.6	md_5	1.7E+6	2.1E+5	5.8
sm_5_1	1.5E+5	1.3E+4	4.6	md_5_1	2.1E+6	2.4E+5	5.9
sm_5_5	2.0E+5	2.5E+4	4.8	md_5_5	3.4E+6	4.8E+5	6.1
sm_5_75	3.3E+5	5.0E+4	5.1	md_5_75	6.0E+6	9.9E+5	6.4
sm_5_9	7.3E+5	1.3E+5	5.5	md_5_9	1.4E+7	2.5E+6	6.8
sm_5_95	1.4E+6	2.6E+5	5.8	md_5_95	2.7E+7	5.0E+6	7.1
sm_5_97	2.3E+6	4.3E+5	6.0	md_5_97	4.4E+7	8.4E+6	7.3
sm_5_99	6.7E+6	1.3E+6	6.5	md_5_99	1.3E+8	2.5E+7	7.8
sm_5_995	1.3E+7	2.6E+6	6.8	md_5_995	2.5E+8	5.0E+7	8.1
sm_5_999	6.8E+7	1.3E+7	7.5	md_5_999	9.0E+8	2.5E+8	8.7
sm_5_9995	1.4E+8	2.6E+7	7.8	md_5_9995	1.7E+9	5.0E+8	9.0
sm_5_9999	6.8E+8	1.3E+8	8.5	md_5_9999	2.1E+9	2.5E+9	9.4
md_1	5.1E+4	3.4E+3	4.1	md_4	2.3E+6	3.1E+5	5.9
md_1_1	5.4E+4	4.4E+3	4.2	md_4_1	3.1E+6	3.6E+5	6.0
md_1_5	7.8E+4	8.4E+3	4.4	md_4_5	5.5E+6	7.0E+5	6.3
md_1_75	1.3E+5	1.7E+4	4.7	md_4_75	9.1E+6	1.5E+6	6.6
md_1_9	3.1E+5	4.4E+4	5.1	md_4_9	1.9E+7	3.7E+6	6.9
md_1_95	6.2E+5	8.8E+4	5.4	md_4_95	3.2E+7	7.4E+6	7.2
md_1_97	1.0E+6	1.5E+5	5.6	md_4_97	5.0E+7	1.2E+7	7.4
md_1_99	3.1E+6	4.4E+5	6.1	md_4_99	1.4E+8	3.7E+7	7.9
md_1_995	6.1E+6	8.9E+5	6.4	md_4_995	2.7E+8	7.4E+7	8.2
md_1_999	3.0E+7	4.4E+6	7.1	md_4_999	1.2E+9	3.7E+8	8.8
md_1_9995	6.0E+7	8.9E+6	7.4	md_4_9995	2.0E+9	7.4E+8	9.1
md_1_9999	3.0E+8	4.4E+7	8.1	md_4_9999	6.6E+9	3.7E+9	9.7
lg_1	7.5E+2	4.2E+2	2.7	md_6	3.3E+6	2.6E+5	6.0
lg_1_1	8.3E+2	4.8E+2	2.8	md_6_1	3.8E+6	4.0E+5	6.1
lg_1_5	1.5E+3	9.3E+2	3.1	md_6_5	6.5E+6	8.0E+5	6.4
lg_1_75	2.9E+3	1.9E+3	3.4	md_6_75	1.2E+7	1.6E+6	6.6
lg_1_9	7.3E+3	4.8E+3	3.8	md_6_9	2.8E+7	4.1E+6	7.0
lg_1_95	1.5E+4	9.5E+3	4.1	md_6_95	5.5E+7	8.3E+6	7.3
lg_1_97	2.4E+4	1.6E+4	4.3	md_6_97	9.0E+7	1.4E+7	7.5

Problem Instance	C_U	C_L	$\log(\bar{C})$	Problem Instance	C_U	C_L	$\log(\bar{C})$
lg_1_99	7.3E+4	4.8E+4	4.8	md_6_99	2.5E+8	4.1E+7	8.0
lg_1_995	1.4E+5	9.5E+4	5.1	md_6_995	4.3E+8	8.2E+7	8.3
lg_1_999	7.2E+5	4.8E+5	5.8	md_6_999	8.3E+8	4.1E+8	8.8
lg_1_9995	1.4E+6	9.5E+5	6.1	md_6_9995	9.9E+8	8.2E+8	9.0
lg_1_9999	7.2E+6	4.8E+6	6.8	md_6_9999	1.3E+9	4.1E+9	9.4

4.1 Correlation of Condition Measures and SeDuMi Iterations

In this subsection we analyze the correlation between $\log(\bar{C})$ and the number of IPM iterations used by SeDuMi to solve a given problem instance. Table 3 shows the number of IPM iterations used by SeDuMi to solve each of the 144 test problem instances using SeDuMi default parameters. Notice from Table 3 that within each of the 12 groups of problems, the iterations grow with $\log(\bar{C})$, thereby suggesting that SeDuMi iterations should be positively correlated with $\log(\bar{C})$. Figure 1 presents a scatter plot of $\log(\bar{C})$ and SeDuMi iterations for all 144 problems. The figure clearly indicates such a trend. For small values of $\log(\bar{C})$ the SeDuMi iterations is almost precisely predictable. However, for larger values of $\log(\bar{C})$ there is greater variability in the SeDuMi iterations. We computed the sample correlation R for the 144 values of $\log(\bar{C})$ and SeDuMi iterations, which yielded a sample correlation $R = 0.676$, indicating a fairly strong linear relationship between these two values.

Figure 2 shows line plots of $\log(\bar{C})$ and SeDuMi iterations for each of the 12 groups of problem instances. This figure shows that within each problem group, there is a striking linear relationship between these two quantities. We ran simple linear regression models for each of the 12 sets of 12 data pairs, the results of which are shown in Table 4. Notice that the regression R^2 for each of the 12 regressions is at least 0.896, with half of these having $R^2 \geq 0.949$. However, as Figure 1 showed, when taken as a whole, the 144 pairs do not have

Table 3: SeDuMi IPM Iterations for the 144 SOCP Problem Instances.

Problem	Iterations	Problem	Iterations	Problem	Iterations	Problem	Iterations
sm_18	16	sm_5	16	md_2	16	md_5	15
sm_18_1	16	sm_5_1	15	md_2_1	16	md_5_1	15
sm_18_5	16	sm_5_5	15	md_2_5	16	md_5_5	15
sm_18_75	17	sm_5_75	16	md_2_75	18	md_5_75	16
sm_18_9	17	sm_5_9	17	md_2_9	18	md_5_9	16
sm_18_95	19	sm_5_95	18	md_2_95	18	md_5_95	17
sm_18_97	19	sm_5_97	18	md_2_97	18	md_5_97	17
sm_18_99	20	sm_5_99	19	md_2_99	19	md_5_99	19
sm_18_995	22	sm_5_995	20	md_2_995	21	md_5_995	20
sm_18_999	25	sm_5_999	23	md_2_999	26	md_5_999	23
sm_18_9995	26	sm_5_9995	24	md_2_9995	31	md_5_9995	24
sm_18_9999	30	sm_5_9999	28	md_2_9999	43	md_5_9999	27
sm_19	14	md_1	15	sm2_1	18	md_4	17
sm_19_1	14	md_1_1	16	sm2_1_1	18	md_4_1	17
sm_19_5	15	md_1_5	17	sm2_1_5	23	md_4_5	17
sm_19_75	16	md_1_75	16	sm2_1_75	19	md_4_75	18
sm_19_9	17	md_1_9	18	sm2_1_9	28	md_4_9	17
sm_19_95	17	md_1_95	19	sm2_1_95	29	md_4_95	18
sm_19_97	18	md_1_97	21	sm2_1_97	28	md_4_97	20
sm_19_99	19	md_1_99	23	sm2_1_99	29	md_4_99	22
sm_19_995	20	md_1_995	22	sm2_1_995	31	md_4_995	25
sm_19_999	21	md_1_999	25	sm2_1_999	34	md_4_999	30
sm_19_9995	22	md_1_9995	27	sm2_1_9995	36	md_4_9995	32
sm_19_9999	24	md_1_9999	29	sm2_1_9999	38	md_4_9999	32
sm2_3	10	lg_1	11	md_3	15	md_6	14
sm2_3_1	10	lg_1_1	11	md_3_1	15	md_6_1	14
sm2_3_5	11	lg_1_5	12	md_3_5	15	md_6_5	15
sm2_3_75	13	lg_1_75	12	md_3_75	16	md_6_75	14
sm2_3_9	15	lg_1_9	13	md_3_9	17	md_6_9	15
sm2_3_95	15	lg_1_95	13	md_3_95	16	md_6_95	15
sm2_3_97	17	lg_1_97	14	md_3_97	16	md_6_97	16
sm2_3_99	17	lg_1_99	14	md_3_99	18	md_6_99	17
sm2_3_995	19	lg_1_995	15	md_3_995	20	md_6_995	17
sm2_3_999	22	lg_1_999	17	md_3_999	23	md_6_999	20
sm2_3_9995	24	lg_1_9995	19	md_3_9995	24	md_6_9995	22
sm2_3_9999	26	lg_1_9999	20	md_3_9999	28	md_6_9999	25

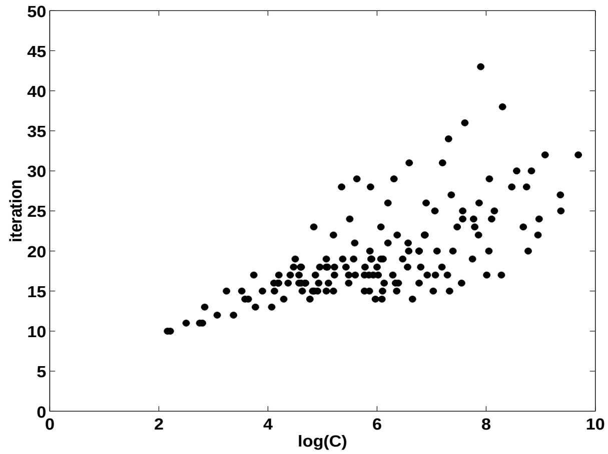

Figure 1: Scatter plot of $\log(\bar{C})$ and SeDuMi iterations for 144 SOCP test problem instances.

such a striking linear dependence.

5 Correlation of SeDuMi Iterations and Geometric Measure of Solution Size and Initial Infeasibility

In this section we analyze the correlation between SeDuMi iterations and a geometric measure of solution size and starting point infeasibility presented in [5]. We first summarize this theory, for details see [5]. The solution size measure stems from using a natural norm $\|\cdot\|$ associated with the starting point cone variables $(x^{(0)}, z^{(0)}, \tau^{(0)}, \kappa^{(0)})$ of the HSD embedding model. SeDuMi uses the

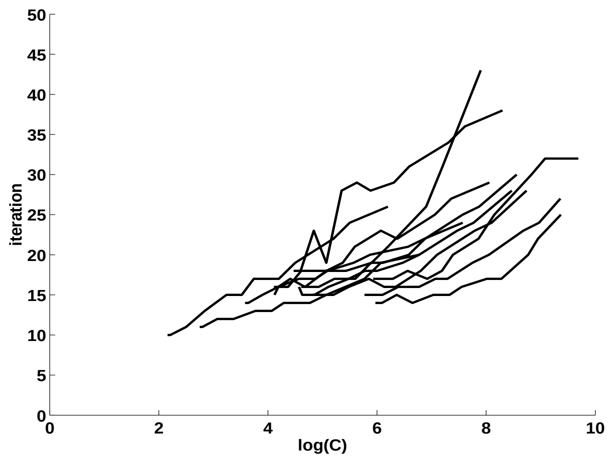

Figure 2: Line plot of $\log(\bar{C})$ and SeDuMi iterations for each of the 12 groups of SOCP test problem instances.

Table 4: Linear regression output of SeDuMi iterations as a function of $\log(\bar{C})$, for each of the 12 groups of SOCP problem instances.

Problem Instance	R^2	Slope	Intercept	Problem Instance	R^2	Slope	Intercept
sm_18	95.6%	3.4	-1.1	md_2	80.3%	5.8	-10.7
sm_19	99.3%	2.4	5.5	sm2_1	89.6%	5.2	-3.6
sm2_3	98.9%	4.0	1.3	md_3	90.0%	3.1	-1.6
sm_5	94.9%	3.0	0.5	md_5	93.2%	3.2	-4.5
md_1	97.5%	3.5	0.8	md_4	91.1%	4.7	-12.9
lg_1	95.8%	2.2	4.6	md_6	82.7%	2.8	-4.0

following starting point:

$$
\begin{array}{rcl}
x^{(0)} & = & (1+\|b\|_\infty)\psi \\
y^{(0)} & = & 0 \\
z^{(0)} & = & (1+\|c\|_\infty)\psi \\
\tau^{(0)} & = & 1 \\
\kappa^{(0)} & = & (1+\|b\|_\infty)(1+\|c\|_\infty) \\
\theta^{(0)} & = & \sqrt{(1+\|b\|_\infty)(1+\|c\|_\infty)}(\psi^T\psi+1)
\end{array}
\tag{22}
$$

where $\psi = [\psi_1^l, \ldots, \psi_{N^l}^l, \psi_1^q, \ldots, \psi_{N^q}^q]$, $\psi_i^l = 1$ for each x_i^l, $i = 1, \ldots, N^l$, and $\psi_i^q = [\sqrt{2}; 0; \ldots; 0]$ for each x_i^q, $i = 1, \ldots, N^q$. Note that $\psi^T\psi = \vartheta$ where ϑ is the complexity value of the cone of the primal variables, see (4). According to [5], the natural norm associated with this starting point for the HSD cone variables is:

$$
\begin{array}{rl}
\|(x,z,\tau,\kappa)\| & := \; (1+\|c\|_\infty)\left(\sum_{i=1}^{N^l}|x_i^l| + \sqrt{2}\sum_{i=1}^{N^q}\max\{|x_i^0|, \|\bar{x}_i\|_2\}\right) \\
& \quad + (1+\|b\|_\infty)\left(\sum_{i=1}^{N^l}|z_i^l| + \sqrt{2}\sum_{i=1}^{N^q}\max\{|z_i^0|, \|\bar{z}_i\|_2\}\right) \\
& \quad + (1+\|b\|_\infty)(1+\|c\|_\infty)|\tau| + |\kappa| \; .
\end{array}
\tag{23}
$$

This norm then can be broken up into different norms for the different variables, in particular

$$\|x\| := (1+\|c\|_\infty)\left(\sum_{i=1}^{N^l}|x_i^l| + \sqrt{2}\sum_{i=1}^{N^q}\max\{|x_i^0|, \|\bar{x}_i\|_2\}\right)$$

and

$$\|z\| := (1+\|b\|_\infty)\left(\sum_{i=1}^{N^l}|z_i^l| + \sqrt{2}\sum_{i=1}^{N^q}\max\{|z_i^0|, \|\bar{z}_i\|_2\}\right)$$

which conveniently specialize to

$$\|x\| = (z^{(0)})^T x \text{ and } \|z\| = (x^{(0)})^T z$$

for $x \succeq 0, z \succeq 0$. Let R_ϵ^P and R_ϵ^D denote the maximum norm among ϵ-optimal solutions of (P) and (D), respectively, measured among the cone variables x and

z, respectively:

$$
\begin{aligned}
R^P_\epsilon := \quad \max \quad & \|x\| := (1+\|c\|_\infty)(\textstyle\sum_{i=1}^{N^l} x^l_i + \sqrt{2}\sum_{i=1}^{N^q} x^0_i) \\
\text{s.t.} \quad & Ax \;=\; b, \;\; c^T x \;\le\; v^* + \epsilon \\
& x \succeq 0
\end{aligned}
\tag{24}
$$

$$
\begin{aligned}
R^D_\epsilon := \quad \max \quad & \|z\| := (1+\|b\|_\infty)(\textstyle\sum_{i=1}^{N^l} z^l_i + \sqrt{2}\sum_{i=1}^{N^q} z^0_i) \\
\text{s.t.} \quad & A^T y + z - c \;=\; 0, \;\; b^T y \;\ge\; w^* - \epsilon \\
& z \succeq 0
\end{aligned}
\tag{25}
$$

where v^* (w^*) is the primal (dual) optimal value of the SOCP problem. Let R_ϵ be the sum of the maximum norms of the primal and dual ϵ-optimal solutions:

$$R_\epsilon := R^P_\epsilon + R^D_\epsilon \; .$$

Recalling SeDuMi's stopping criterion (7), let S denote the following quantity:

$$S := \frac{(R_\epsilon + \kappa^{(0)})}{\bar{\alpha}} \left(2\frac{\|\bar{b}\|_\infty}{1+\|b\|_\infty} + 2\frac{\|\bar{c}\|_\infty}{1+\|c\|_\infty} + \frac{(\bar{g} - \frac{\kappa^{(f)}}{\theta^{(f)}})^+}{\max\{|c^T\bar{x}|, |b^T\bar{y}|, 0.001 \times \tau\}} \right) \; ,$$

where $\kappa^{(f)}, \theta^{(f)}$ denote the values of κ, θ in the final iteration of SeDuMi. Then the analysis in [5] indicates that the number of iterations T of SeDuMi is approximately:

$$T \;\approx\; \frac{\log(S) + |\log(r_{max})|}{|\log(\beta)|} \tag{26}$$

where $\beta = \sqrt[T]{\frac{\theta^{(f)}}{\theta^{(0)}}}$ is the (geometric) average decrease in θ over all iterations. The above approximation is valid under mild assumptions, see [5] for details.

Notice in (26) that T will be positively correlated with $\log(S)$ to the extent that $|\log(\beta)|$ is relatively constant. In order to test the correlation between T and $\log(S)$, we computed R_ϵ and $\log(S)$ for the 144 test problem instances, whose values are shown in Table 5. A scatter plot of the values of $\log(S)$ and the SeDuMi iterations T (from Table 3) is shown in Figure 3. Like the analysis of the condition measure $C(d)$ shown in Figure 1, Figure 3 shows a definitive upward trend in SeDuMi iterations for increasing values of $\log(S)$, but with much more variance, particularly when comparing relatively smaller values of $\log(S)$ and $\log(\bar{C})$. We computed the sample correlation R for the 144 values of $\log(S)$ and SeDuMi iterations, which yielded a sample correlation $R = 0.600$, indicating a modest linear relationship between these two values.

Table 5: R_ϵ and $\log(S)$ for 144 SOCP Problem Instances.

Problem	R_ϵ	$\log(S)$	Problem	R_ϵ	$\log(S)$	Problem	R_ϵ	$\log(S)$
sm_18	1.8E+4	1.0	md_1	1.3E+3	0.5	md_3	5.7E+1	0.5
sm_18_1	1.8E+4	1.0	md_1_1	1.3E+3	0.5	md_3_1	5.7E+1	0.5
sm_18_5	1.8E+4	1.0	md_1_5	1.5E+3	0.6	md_3_5	5.8E+1	0.5
sm_18_75	1.9E+4	1.1	md_1_75	1.9E+3	0.7	md_3_75	6.3E+1	0.6
sm_18_9	2.2E+4	1.1	md_1_9	2.5E+3	0.8	md_3_9	7.3E+1	0.7
sm_18_95	2.4E+4	1.2	md_1_95	3.7E+3	0.9	md_3_95	8.3E+1	0.7
sm_18_97	2.7E+4	1.2	md_1_97	5.3E+3	1.1	md_3_97	9.3E+1	0.8
sm_18_99	3.4E+4	1.3	md_1_99	8.2E+3	1.3	md_3_99	1.3E+2	0.9
sm_18_995	4.5E+4	1.4	md_1_995	1.1E+4	1.4	md_3_995	1.6E+2	1.1
sm_18_999	8.6E+4	1.7	md_1_999	2.3E+4	1.7	md_3_999	3.2E+2	1.5
sm_18_9995	1.2E+5	1.8	md_1_9995	3.1E+4	1.9	md_3_9995	4.3E+2	1.7
sm_18_9999	2.4E+5	2.2	md_1_9999	7.0E+4	2.3	md_3_9999	9.1E+2	2.2
sm_19	8.9E+2	0.5	lg_1	2.1E+1	0.3	md_5	1.3E+2	0.6
sm_19_1	8.9E+2	0.5	lg_1_1	2.1E+1	0.3	md_5_1	1.3E+2	0.6
sm_19_5	9.3E+2	0.6	lg_1_5	2.6E+1	0.4	md_5_5	1.3E+2	0.6
sm_19_75	1.1E+3	0.6	lg_1_75	3.5E+1	0.5	md_5_75	1.5E+2	0.6
sm_19_9	1.7E+3	0.8	lg_1_9	5.2E+1	0.6	md_5_9	1.6E+2	0.7
sm_19_95	2.4E+3	0.9	lg_1_95	7.0E+1	0.8	md_5_95	1.8E+2	0.8
sm_19_97	3.2E+3	1.1	lg_1_97	8.7E+1	0.9	md_5_97	2.0E+2	0.8
sm_19_99	6.5E+3	1.4	lg_1_99	1.4E+2	1.1	md_5_99	2.7E+2	1.0
sm_19_995	9.7E+3	1.6	lg_1_995	2.0E+2	1.3	md_5_995	3.4E+2	1.1
sm_19_999	2.5E+4	2.1	lg_1_999	4.3E+2	1.5	md_5_999	6.1E+2	1.4
sm_19_9995	3.7E+4	2.3	lg_1_9995	6.0E+2	1.6	md_5_9995	8.1E+2	1.6
sm_19_9999	8.4E+4	2.9	lg_1_9999	1.3E+3	1.8	md_5_9999	1.7E+3	2.1
sm2_3	4.6E+1	0.6	md_2	1.7E+3	1.0	md_4	6.5E+1	0.5
sm2_3_1	4.6E+1	0.6	md_2_1	1.7E+3	1.0	md_4_1	6.5E+1	0.5
sm2_3_5	4.6E+1	0.7	md_2_5	1.7E+3	1.0	md_4_5	6.7E+1	0.5
sm2_3_75	1.3E+2	1.2	md_2_75	1.8E+3	1.0	md_4_75	7.1E+1	0.5
sm2_3_9	1.3E+2	1.2	md_2_9	1.8E+3	1.0	md_4_9	8.0E+1	0.5
sm2_3_95	1.4E+2	1.2	md_2_95	1.8E+3	1.0	md_4_95	9.4E+1	0.6

Problem	R_ϵ	$\log(S)$	Problem	R_ϵ	$\log(S)$	Problem	R_ϵ	$\log(S)$
sm2_3_97	1.8E+2	1.4	md_2_97	1.9E+3	1.0	md_4_97	1.0E+2	0.7
sm2_3_99	3.1E+2	1.8	md_2_99	2.1E+3	1.1	md_4_99	1.4E+2	0.8
sm2_3_995	4.3E+2	2.0	md_2_995	2.3E+3	1.1	md_4_995	1.7E+2	0.8
sm2_3_999	9.6E+2	2.6	md_2_999	3.3E+3	1.3	md_4_999	3.1E+2	1.1
sm2_3_9995	1.4E+3	2.7	md_2_9995	4.1E+3	1.4	md_4_9995	3.9E+2	1.3
sm2_3_9999	3.0E+3	2.2	md_2_9999	7.2E+3	1.6	md_4_9999	7.1E+2	1.6
sm_5	1.9E+4	1.0	sm2_1	4.4E+2	0.7	md_6	2.3E+2	0.7
sm_5_1	1.9E+4	1.0	sm2_1_1	4.4E+2	0.7	md_6_1	2.3E+2	0.7
sm_5_5	1.9E+4	1.0	sm2_1_5	4.5E+2	0.7	md_6_5	2.4E+2	0.7
sm_5_75	1.9E+4	1.0	sm2_1_75	4.7E+2	0.7	md_6_75	2.5E+2	0.7
sm_5_9	2.1E+4	1.1	sm2_1_9	6.3E+2	0.9	md_6_9	2.8E+2	0.8
sm_5_95	2.3E+4	1.1	sm2_1_95	7.4E+2	0.9	md_6_95	3.1E+2	0.8
sm_5_97	2.6E+4	1.2	sm2_1_97	9.0E+2	1.0	md_6_97	3.4E+2	0.9
sm_5_99	3.3E+4	1.3	sm2_1_99	1.3E+3	1.2	md_6_99	4.4E+2	1.0
sm_5_995	4.0E+4	1.3	sm2_1_995	1.5E+3	1.2	md_6_995	5.4E+2	1.1
sm_5_999	6.9E+4	1.6	sm2_1_999	2.0E+3	1.3	md_6_999	9.6E+2	1.5
sm_5_9995	8.9E+4	1.7	sm2_1_9995	2.6E+3	1.5	md_6_9995	1.3E+3	1.7
sm_5_9999	1.8E+5	2.0	sm2_1_9999	3.7E+3	1.6	md_6_9999	2.6E+3	2.2

Figure 4 shows line plots of $\log(S)$ and SeDuMi iterations for each of the 12 groups of problem instances. This figure shows that within each problem group, there is a definite linear relationship between these two quantities. We ran simple linear regression models for each of the 12 sets of 12 data pairs, the results of which are shown in Table 6. Notice that the regression R^2 for each of the 12 regressions is at least 0.875, with half of these having $R^2 \geq 0.981$. However, as Figure 3 showed, when taken as a whole, the 144 pairs do not have such a definitive linear dependence.

The absence of a very strong linear correlation between SeDuMi iterations and $\log(S)$ indicates from (26) that β, which is the average decrease in the objective function (and duality gap) over all iterations, cannot be approximately constant over the test problem instances. Table 7 shows the values of β for all 144 test problem instances. Notice that in general one observes within each of

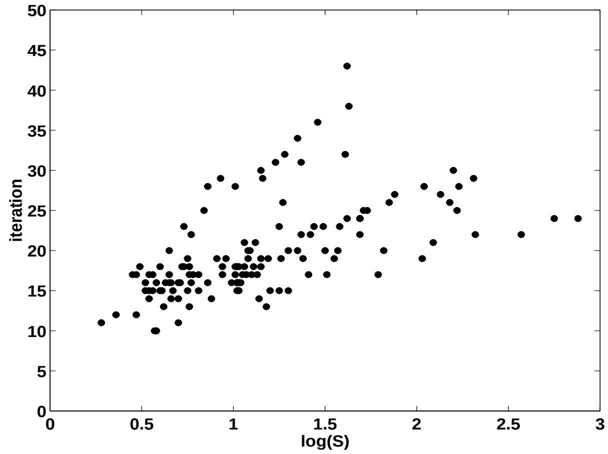

Figure 3: Scatter plot of log(S) and SeDuMi iterations for 144 test problem instances.

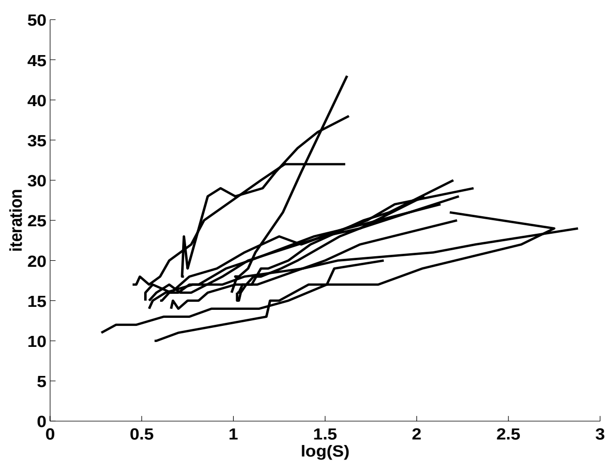

Figure 4: Line plot of log(S) and SeDuMi iterations for each of the 12 groups of SOCP test problem instances.

Table 6: Linear regression output of SeDuMi iterations as a function of $\log(S)$, for each of the 12 groups of SOCP problem instances.

Problem Instance	R^2	Slope	Intercept	Problem Instance	R^2	Slope	Intercept
sm_18	98.6%	11.9	4.4	md_2	98.8%	41.0	-24.5
sm_19	95.0%	4.0	13.0	sm2_1	87.5%	20.2	6.4
sm2_3	89.1%	6.8	6.3	md_3	98.4%	7.7	11.0
sm_5	98.1%	12.2	3.5	md_5	98.3%	8.1	10.6
md_1	97.9%	7.9	-9.0	md_4	93.7%	15.6	10.0
lg_1	92.4%	5.3	9.4	md_6	98.7%	7.0	9.6

the 12 groups that β increases as the condition measure $C(d)$ increases. Figure 5 shows line plots of $\log(\bar{C})$ and $|\log(\beta)|$, which confirms the intuition that $|\log(\beta)|$ is decreasing in $\log(\bar{C})$. This figure indicates that for the HSD embedding IPM algorithm SeDuMi, there is at least a loosely defined relationship between the condition number and the rate of convergence of the algorithm.

Table 7: (Geometric) Average Decrease in the Duality Gap β for all 144 test problem instances.

Problem	β	Problem	β	Problem	β	Problem	β
sm_18	0.22	sm_5	0.22	md_2	0.23	md_5	0.22
sm_18_1	0.21	sm_5_1	0.21	md_2_1	0.21	md_5_1	0.22
sm_18_5	0.22	sm_5_5	0.19	md_2_5	0.22	md_5_5	0.21
sm_18_75	0.23	sm_5_75	0.23	md_2_75	0.26	md_5_75	0.23
sm_18_9	0.24	sm_5_9	0.26	md_2_9	0.27	md_5_9	0.24
sm_18_95	0.27	sm_5_95	0.26	md_2_95	0.25	md_5_95	0.24
sm_18_97	0.28	sm_5_97	0.27	md_2_97	0.25	md_5_97	0.24
sm_18_99	0.29	sm_5_99	0.29	md_2_99	0.27	md_5_99	0.27
sm_18_995	0.34	sm_5_995	0.30	md_2_995	0.29	md_5_995	0.27
sm_18_999	0.37	sm_5_999	0.34	md_2_999	0.38	md_5_999	0.31
sm_18_9995	0.37	sm_5_9995	0.35	md_2_9995	0.40	md_5_9995	0.32

Problem	β	Problem	β	Problem	β	Problem	β
sm_18_9999	0.39	sm_5_9999	0.38	md_2_9999	0.55	md_5_9999	0.37
sm_19	0.18	md_1	0.22	sm2_1	0.29	md_4	0.27
sm_19_1	0.18	md_1_1	0.24	sm2_1_1	0.26	md_4_1	0.27
sm_19_5	0.21	md_1_5	0.26	sm2_1_5	0.36	md_4_5	0.28
sm_19_75	0.21	md_1_75	0.24	sm2_1_75	0.28	md_4_75	0.29
sm_19_9	0.25	md_1_9	0.29	sm2_1_9	0.44	md_4_9	0.26
sm_19_95	0.24	md_1_95	0.28	sm2_1_95	0.45	md_4_95	0.30
sm_19_97	0.27	md_1_97	0.33	sm2_1_97	0.44	md_4_97	0.31
sm_19_99	0.27	md_1_99	0.32	sm2_1_99	0.45	md_4_99	0.35
sm_19_995	0.28	md_1_995	0.32	sm2_1_995	0.47	md_4_995	0.37
sm_19_999	0.29	md_1_999	0.35	sm2_1_999	0.50	md_4_999	0.42
sm_19_9995	0.30	md_1_9995	0.36	sm2_1_9995	0.50	md_4_9995	0.46
sm_19_9999	0.31	md_1_9999	0.40	sm2_1_9999	0.49	md_4_9999	0.46
sm2_3	0.06	lg_1	0.14	md_3	0.23	md_6	0.20
sm2_3_1	0.06	lg_1_1	0.14	md_3_1	0.22	md_6_1	0.20
sm2_3_5	0.10	lg_1_5	0.16	md_3_5	0.22	md_6_5	0.19
sm2_3_75	0.14	lg_1_75	0.16	md_3_75	0.24	md_6_75	0.19
sm2_3_9	0.19	lg_1_9	0.17	md_3_9	0.25	md_6_9	0.21
sm2_3_95	0.19	lg_1_95	0.17	md_3_95	0.25	md_6_95	0.21
sm2_3_97	0.24	lg_1_97	0.19	md_3_97	0.24	md_6_97	0.22
sm2_3_99	0.22	lg_1_99	0.17	md_3_99	0.27	md_6_99	0.24
sm2_3_995	0.21	lg_1_995	0.19	md_3_995	0.28	md_6_995	0.24
sm2_3_999	0.26	lg_1_999	0.23	md_3_999	0.31	md_6_999	0.29
sm2_3_9995	0.31	lg_1_9995	0.25	md_3_9995	0.34	md_6_9995	0.29
sm2_3_9999	0.35	lg_1_9999	0.26	md_3_9999	0.36	md_6_9999	0.32

6 Concluding Remarks

Our computational experience indicates that SeDuMi iteration counts and $\log(C(d))$ are fairly highly correlated (sample correlation $R = 0.676$), whereas SeDuMi iteration counts are not quite as highly correlated with the combined measure of initial infeasbility/optimality residuals S ($R = 0.600$).

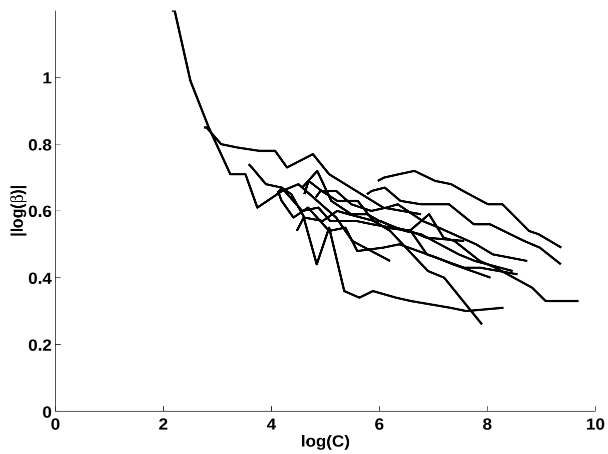

Figure 5: Line plots of $\log(\bar{C})$ and $|\log(\beta)|$ for each of the 12 groups of SOCP test problem instances.

The theory of interior-point methods only points to one factor, namely the complexity value ϑ of the underlying self-concordant barrier for the cone, that can have a provable influence on the rate of convergence in theory. Yet as Table 7 and Figure 5 have shown, there is evidence of some systematic effect of increasingly ill-behaved problems on the average convergence rate of SeDuMi. We believe that this evidence bears further analysis.

References

[1] R.M. Freund and J. Vera, *Some characterization and properties of the 'distance to ill-posedness' and the condition measure of a conic linear system*, Mathematical Programming, 86 (1999), pp. 225–260.

[2] R.M. Freund and J. Vera, *Condition-based complexity of convex optimization in conic linear form via the ellipsoid algorithm*, SIAM Journal on Optimization, 10 (1999), pp. 155-176.

[3] R.M. Freund and J. Vera, *On the complexity of computing estimates of condition measures of a conic linear system*, Mathematics of Operations

Research, 28 (2003), pp. 625-648.

[4] R.M. Freund, *Complexity of convex optimization using geometry-based measures and a reference point*, Mathematical Programming, 99 (2004), pp. 197–221.

[5] R.M. Freund, *On the behavior of the homogeneous self-dual model for conic convex optimization*, Manuscript, 2004.

[6] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1994.

[7] F. Ordóñez and R.M. Freund, *Computational experience and the explanatory value of condition measures for linear optimization*, SIAM Journal on Optimization, 14 (2004), pp. 307-333.

[8] J. Peña, *Computing the distance to infeasibility: theoretical and practical issues*, Technical report, Center for Applied Mathematics, Cornell University, 1998.

[9] J. Peña, *Understanding the geometry of infeasible perturbation of a conic linear system*, SIAM Journal on Optimization, 10 (2000), pp. 534–550.

[10] J. Peña, *Two properties of condition numbers for convex programs via implicitly defined barrier functions*, Mathematical Programming, 93 (2002), pp. 55–57.

[11] J. Renegar, *Some perturbation theory for linear programming*, Mathematical Programming, 65 (1994), pp. 73–91.

[12] J. Renegar, *Linear programming, complexity theory, and elementary functional analysis*, Mathematical Programming, 70 (1995), pp. 279–351.

[13] 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.

[14] Y. Ye, M.J. Todd and S. Mizuno, *On $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm*, Mathematics of Operations Research, 19 (1994), pp. 53–67.