Solving Large Structured Semidefinite Programs Using an Inexact Spectral Bundle Method

Scott A. Miller and Roy S. Smith
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106
email: miller@engineering.ucsb.edu

Abstract

Semidefinite programs have received a great deal of attention because of the variety of problems that they can model and the rich theory that leads to polynomial-time algorithms to solve them. However, large practical problems are still hard to solve because most algorithms ignore the structure of the problem. In this paper we present an algorithm for solving semidefinite programs that exploits structure yet is not tailored *a priori* to any particular structure. It adapts a bundle method designed to solve structured LMI feasibility problems. Duality provides a tight lower bound for the optimal cost for use in a termination criterion. A numerical experiment demonstrates that the complexity is comparable to that of structured interior-point methods, and unlike those methods it applies to a general class of structures.

1 Introduction

A semidefinite program (or SDP) is an optimization problem of the form

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & F(x) \leq 0 \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$, F is an affine mapping from $\mathbb{R}^n$ to the space $\mathcal{S}^m$ of $m \times m$ symmetric matrices, and $F(x) \leq 0$ means $F(x)$ is negative semidefinite. The constraint $F(x) \leq 0$ is a linear matrix inequality (LMI), hence the SDP is a type of LMI problem. A wide variety of practical problems in engineering can be expressed as an SDP, and SDPs are successfully applied as relaxations of hard combinatorial problems. SDPs are considered tractable because they are convex programs and there are polynomial-time algorithms to solve them. However, these algorithms generally cost $O(m^6)$ flops per iteration in the common case of $n = O(m^2)$. This high order of complexity makes such general methods impractical for large problems. This paper describes a new algorithm designed to solve large SDPs efficiently.

A practical algorithm for large problems must take advantage of whatever structure the problem possesses to reduce the complexity of solving it. In [8] the authors discussed two types of LMI structure: *expression structure*, the ability to form expressions involving $F(x)$ quickly because of a special structure in the functional dependence, and *matrix structure*, a structure in the matrix $F(x)$ present for all x. An example of expression structure is the Lyapunov LMI $A^T P + PA + Q < 0$ for variable $P \in \mathcal{S}^m$; computing the left-hand-side expression costs $O(m^3)$ but to compute it from the generic LMI form

$$F(x) = F_0 + \sum_{i=1}^{n} x_i F_i \tag{2}$$

would cost $O(m^4)$. Less trivial examples can be derived from robust control design problems in which the plant, and possibly the controller, has a particular structure. An example of matrix structure is the Toeplitz-like form appearing in the LMI formulation of the time-domain model validation problem [9].

The inexact spectral bundle method (ISBM) [8] is designed to solve LMI feasibility problems (find x such that $F(x) \leq 0$) with expression and matrix structure. It is a nonsmooth optimization technique that uses estimates of the maximum eigenvalue and corresponding eigenvector(s) of $F(x)$, and controls the tolerance of those estimates to reduce the amount of computation. The algorithm only requires an implicit representation of the operator F, so it avoids forming large matrices and it can take maximal advantage of the structure of the problem while not being tailored to any particular structure. Structured interior-point solvers such as [2, 11] are designed for a specific expression structure, and they must form $m \times m$ matrices even when m is large. Hence, we adapt the ISBM to solve the optimization problem (1).

2 Notation

Recall that $\mathcal{S}^m$ denotes the Hilbert space of real symmetric $m \times m$ matrices. The inner product in $\mathcal{S}^m$ is given by $\langle X, Y \rangle = \operatorname{Tr} XY$, where $\operatorname{Tr} A$ is the trace of A. The eigenvalues of a matrix $A \in \mathcal{S}^m$ are ordered according to $\lambda_1(A) \geq \cdots \geq \lambda_m(A)$. In general, hat ($\hat{}$) accents mean approximations, and superscripts denote iteration counts. The LMI func-

tion $F : \mathbb{R}^n \to \mathcal{S}^m$ is affine, with its components given by (2). The symbol $\bar{F}$ represents the linear part of F, and $F^* : \mathcal{S}^m \to \mathbb{R}^n$ is the adjoint of $\bar{F}$, so $F^*(Z) = \big[\langle F_1, Z\rangle, \quad \ldots, \quad \langle F_n, Z\rangle\big]^T$.

The function $\bar{\lambda} : \mathbb{R}^n \to \mathbb{R}$ defined by $\bar{\lambda}(x) := \lambda_1(F(x))$ plays an important role, since $F(x) \leq 0$ iff $\bar{\lambda}(x) \leq 0$. The function $\bar{\lambda}$ is convex, but it is not differentiable where the maximum eigenvalue is multiple. Hence, we will employ some concepts from nonsmooth convex analysis.

If f is a convex functional on domain D and $x \in D$, then $\partial f(x)$ denotes the subdifferential (the set of subgradients) of f at x, and $\partial_\varepsilon f(x)$ denotes the ε-subdifferential (the set of ε-subgradients) of f at x. See [1] for definitions and properties of these objects. We also use the notation $\partial_1 f(x, y)$ to mean the subdifferential of the function $f(\cdot, y)$ for a fixed y. Similarly, $\partial_{1,\varepsilon} f(x, y)$ means the ε-subdifferential of the function $f(\cdot, y)$.

3 Solution Method

Bundle methods were originally developed to solve unconstrained nonsmooth convex optimization problems, but they have been extended in several ways to constrained problems. In his book [5], Kiwiel solves programs of the form $\min\{f(x) \mid g(x) \leq 0\}$ by minimizing the *improvement function*, defined by

$$h(y, x) := \max\{f(y) - f(x),\ g(y)\}.$$

At each feasible iterate x^k, descent of the function $h(\cdot, x^k)$ implies feasibility and cost reduction. The function $h(\cdot, x)$ is convex for every x, and its subdifferential is easily obtained from those of f and g. The point x^k is optimal if and only if $0 \in \partial_1 h(x^k, x^k)$.

Another common approach to solving constrained problems is an exact penalty method (e.g. [4]), which minimizes a penalty function $f(x) + \mu \max\{g(x), 0\}$ with increasing $\mu > 0$ to approach the optimum of the original problem. An alternative strategy [6] models the constraints directly with a bundle and simply uses the penalty function for step size selection; this has the advantage of being insensitive to scaling of the constraints, unlike the penalty function and improvement function approaches. However, both of these alternatives tend to generate infeasible iterates, a significant drawback especially for large problems.

3.1 Application of bundle method

Bundle methods generally work as follows. Let the current iterate be x^k. A lower-bounding model of the objective function, composed from a "bundle" of subgradients, is maintained at each iteration. Minimization of this model (plus a regularization term) gives the next trial point y^{k+1}, where a new function value and subgradient is obtained. This minimization is called the *search direction subproblem*. If the trial point reduces the objective sufficiently, a *serious step* is taken and $x^{k+1} = y^{k+1}$; otherwise, a *null step* is taken and $x^{k+1} = x^k$. In either case, the model is updated using the subgradient at y^{k+1}, so a better trial point can be obtained next time if a null step is taken. To avoid storing all past subgradients, the model includes an *aggregate subgradient*, an average of past subgradients which constitutes an ε-subgradient of the objective at the current iterate. This ε and the size of the aggregate subgradient converge to zero, indicating convergence of the bundle method to the minimum.

In the case of the SDP, the objective is the improvement function

$$\begin{aligned} h(y, x) &= \max\left\{c^T(y - x),\ \bar{\lambda}(y)\right\} \\ &= \lambda_1\left(\begin{bmatrix} c^T(y-x) & 0 \\ 0 & F(y) \end{bmatrix}\right), \end{aligned}$$

which suggests that we can apply the inexact spectral bundle method (ISBM) [8] to minimize h, adjusting some parameters as necessary when the second argument changes. We could augment the given code for implementing $F(y)$ with the extra diagonal component $c^T(y - x)$ and apply the unmodified ISBM to the augmented function. However, there is no need to develop a bundle model of the $c^T(y-x)$ block, since it is differentiable and the gradient is constant. Therefore, the bundling procedure is only applied to the $F(y)$ block, and the search direction subproblem is modified to account for the extra component in the objective function.

The model of h at step k is

$$\hat{h}^k(y, x) := \max\left\{c^T(y - x),\ \hat{\lambda}^k(y)\right\},$$

where $\hat{\lambda}^k(y)$ is the *semidefinite cutting plane model* of $\bar{\lambda}(y)$ [3, 8] given by

$$\begin{aligned} \hat{\lambda}^k(y) &:= \max\left\{\langle F(y), W\rangle \mid W \in \mathcal{W}^k\right\} \\ \mathcal{W}^k &:= \Big\{\alpha\overline{W}^k + P^k V P^{kT} \;\Big|\; \alpha \in \mathbb{R},\ V \in \mathcal{S}^r, \\ &\qquad\qquad \alpha \geq 0,\ V \geq 0,\ \alpha + \operatorname{Tr} V = 1\Big\}, \end{aligned}$$

with r bounded by a constant $\ll m$. The bundle for $\lambda_1(F(y))$ is defined by the r approximate eigenvectors in the columns of P^k and the aggregate subgradient matrix $\overline{W}^k$. The search direction subproblem at step k is to find d that minimizes $\hat{h}^k(x^k + d, x^k) + \frac{1}{2}u^k\|d\|^2$, or equivalently

$$\begin{aligned} \min_d \Big[\tfrac{1}{2}u^k\|d\|^2 + \max\Big\{&\mu c^T d + \langle F(x^k + d), W\rangle \;\Big|\; \\ &W = \alpha\overline{W}^k + P^k V P^{kT}, \\ &\mu \geq 0,\ \alpha \geq 0,\ V \geq 0, \\ &\mu + \alpha + \operatorname{Tr} V = 1\Big\}\Big]. \end{aligned}$$

Using duality and the fact that d is unconstrained, this can be transformed into a small convex quadratic

problem with LMI constraints, and easily solved with an interior-point method.

Let μ^k, α^k and V^k solve the search direction subproblem at step k, let d^k be the resulting search direction, and let $y^{k+1} = x^k + d^k$ be the next trial point. In the original ISBM, the aggregate subgradient vector $w^k = F^*(\alpha^k \overline{W}^k + P^k V^k P^{kT})$ has the properties that $w^k \in \partial_{\tau^k} \bar{\lambda}(x^k)$ and $w^k \in \partial_{\eta^k} \bar{\lambda}(y^{k+1})$ for some nonnegative τ^k and η^k. To have the same subgradient properties for the modified problem, w^k must be rescaled to $F^*(\alpha^k \overline{W}^k + P^k V^k P^{kT})/(1-\mu^k)$. In the modified ISBM, the aggregate subgradient of $h(\cdot, x^k)$ is not w^k but $p^k := \mu^k c + (1-\mu^k) w^k$, and it satisfies $p^k \in \partial_{1,\tau^k} h(x^k, x^k)$ and $p^k \in \partial_{1,\eta^k} h(y^{k+1}, x^k)$. For convergence-testing purposes, the shifted quantity

$$\bar{\eta}^k := \eta^k + \max\{0, \bar{\lambda}(y^{k+1})\} - \max\{c^T d^k, \bar{\lambda}(y^{k+1})\}$$

is more appropriate than η^k, since $p^k \in \partial_{1,\bar{\eta}^k} h(y^{k+1}, y^{k+1})$. Then

$$\nu^k := \max\left\{\|p^k\|,\ \min\{\tau^k, \bar{\eta}^k\}\right\} \tag{3}$$

is a measure of suboptimality and $\nu^k \to 0$. In practice, the algorithm is considered converged if $\nu^k < \nu_{\text{tol}}$ for some prescribed tolerance $\nu_{\text{tol}} > 0$.

This minor modification to the ISBM does not affect the convergence theory in [8]. In case of finite termination or an infinite number of serious steps, the approximation of function values and subgradients does not affect the convergence proof, so convergence follows from the analysis in [4]. In case of a bounded number of serious steps but an infinite number of null steps, after some finite time x^k does not change, and the problem becomes one of minimizing $h(\cdot, x)$ for a fixed x. Since the objective function is fixed and the search direction subproblem is equivalent to the one in the unmodified ISBM, the analysis of [8] applies here as well.

3.2 Proximity weight control

Adaptation of the proximity weight u^k in the search direction subproblem is critical to good performance of a bundle method. Very little theory is available to guide the choice of u^k, and published heuristic schemes such as [7, 10] can cause poor performance on eigenvalue problems. Our adaptation method is heuristic as well, but it is based on observing through numerical experiments how changing u^k affects the trajectories of the important quantities in the optimization: $c^T x^k$, $\|p^k\|$, τ^k and η^k. These heuristics also work well on random, unstructured SDPs.

According to our observations, there appears to be an optimal setting of u^k in the first phase of the minimization, where the cost is reduced the most. If u^k is too high, the serious steps are too small. If u^k is too low, null steps become more frequent, and in this phase null steps do not ultimately contribute to greater reduction of the cost; they simply dilute the progress of the serious steps. Therefore, a search procedure is employed during the minimization to try to find a nearly optimal setting of u^k. The procedure produces converging lower and upper bounds for the optimal u based on rough estimates of the slope of the cost trajectory, and terminates in a short time.

Thereafter, u^k is generally increased slightly after each null step, within the upper bound established in the previous search phase. The exception is when $\min\{\tau^k, \bar{\eta}^k\} < \nu_{\text{tol}}$, in which case we reduce u^k to help push down $\|p^k\|$ and encourage earlier convergence.

4 Stopping Criterion

In bundle methods for minimizing an unconstrained convex function $f(x)$, the usual stopping criterion is $\nu^k < \nu_{\text{tol}}$, where ν^k is defined such as in (3). This is motivated by the concept of ε-optimality, the property that for all x,

$$f(x) \geq f(x^k) - \varepsilon\|x - x^k\| - \varepsilon.$$

Thus $\nu^k < \nu_{\text{tol}}$ means that x^k is ε-optimal with $\varepsilon < \nu_{\text{tol}}$. Generally speaking, ε-optimality implies that locally the minimal value is within ε of $f(x^k)$, and globally the objective function cannot decrease at a rate greater than ε from the point x^k, so it is similar in spirit to conditions on the norm of the gradient in smooth optimization.

Nevertheless, when using an improvement function to solve an SDP this criterion is inadequate. Existing primal-dual methods for solving SDPs provide a converging lower bound on the optimal cost. But ε-optimality implies no such lower bound. Moreover, it does not even provide a bound on the decrease of the SDP cost, because it is the decrease of the improvement function h that is bounded, and that can have a much shallower slope. Therefore, we seek at least an approximate lower bound to the optimal cost for use in a stopping criterion, to be competitive with primal-dual methods. The lower bound is motivated by a duality result, and computed in a separate minimization phase that brings x^k toward the "center" of the feasible region at a fixed cost. This phase is described in the next section.

There is another good reason for using the lower bound estimate as a stopping criterion. It turns out that a significant portion of the computation time is spent essentially demonstrating that the optimum has been reached. Using the cost bound instead of ε-optimality as the criterion seems to shorten this "end game" considerably.

4.1 Centering phase

For the duration of this section we make the following assumptions about the SDP (1):

- γ^* is the optimal cost and $\gamma^* > -\infty$;

- the solution set is bounded;
- $\exists\, x \ni F(x) < 0$ (Slater condition).

Let $\gamma \geq \gamma^*$ be given. We say that $x_{\rm ctr}$ is a *center at* γ if

$$\bar{\lambda}(x_{\rm ctr}) = \min\{\bar{\lambda}(x) \mid c^T x = \gamma\}. \tag{4}$$

We call this optimization problem the *centering problem*. It can be shown that the first two assumptions imply the existence of a center, though it might not be unique.

Lemma 1. *Let $x_{\rm ctr}$ be a center at some $\gamma \geq \gamma^*$. Then there is a $w_{\rm ctr} \in \partial\bar{\lambda}(x_{\rm ctr})$ such that $w_{\rm ctr}$ is collinear with c.*

Proof. Suppose x^k is such that $c^T x^k = \gamma$. Then all solutions to $c^T x = \gamma$ have the form $x^k + Bz$, where the columns of $B \in \mathbb{R}^{n\times(n-1)}$ are a basis for $\mathrm{span}\{c\}^\perp$. The constraint in (4) may then be eliminated: there exists a $\bar{z}$ such that

$$\bar{\lambda}(x^k + B\bar{z}) = \min_z\ \bar{\lambda}(x^k + Bz) \tag{5}$$
$$x_{\rm ctr} = x^k + B\bar{z}.$$

Define $G(z) := F(x^k + Bz) = F(x^k) + \bar{F}(Bz)$, so that $\bar{G}(z) = \bar{F}(Bz)$ and $G^*(W) = B^T F^*(W)$. Then $\bar{\lambda}(x^k + Bz) = \lambda_1(G(z))$, and therefore $0 \in \partial(\lambda_1 \circ G)(\bar{z})$. This means $\exists\, W_{\rm ctr} \in \partial\lambda_1(G(\bar{z}))\ \ni G^*(W_{\rm ctr}) = 0$, and since $F(x_{\rm ctr}) = G(\bar{z})$,

$$\exists\, W_{\rm ctr} \in \partial\lambda_1(F(x_{\rm ctr}))\ \ni\ B^T F^*(W_{\rm ctr}) = 0. \tag{6}$$

The conclusion follows with $w_{\rm ctr} = F^*(W_{\rm ctr})$. □

We can now state the theorem providing a lower bound for γ^*.

Theorem 1. *Let $\gamma \geq \gamma^*$ be given. Let $x_{\rm ctr}$ be a center at γ and let $w_{\rm ctr}$ be as in Lemma 1, with $c^T w_{\rm ctr} < 0$. Then*

$$\gamma - \bar{\lambda}(x_{\rm ctr})\frac{\|c\|^2}{c^T w_{\rm ctr}}\ \leq\ \gamma^*,$$

with equality if $\gamma = \gamma^*$.

Proof. Define the Lagrangian for the SDP (1): $L(x,\theta) := c^T x + \theta\bar{\lambda}(x)$. The dual function is then $\varphi(\theta) := \min_x L(x,\theta)$, and $\varphi(\theta) \leq \gamma^*$ for all $\theta \geq 0$. Let $\theta_{\rm ctr} := -\|c\|^2/(c^T w_{\rm ctr})$, which is positive by assumption. Using the subgradient property of $W_{\rm ctr}$ in (6) one can show that $\varphi(\theta_{\rm ctr}) > -\infty$. Therefore, by boundedness and convexity of $L(\cdot,\theta_{\rm ctr})$, we have $\varphi(\theta_{\rm ctr}) = L(x,\theta_{\rm ctr})$ iff $0 \in \partial_1 L(x,\theta_{\rm ctr})$. This is equivalent to the existence of a $w \in \partial\bar{\lambda}(x)$ such that $w = -c/\theta_{\rm ctr}$. This condition is satisfied by choosing $x = x_{\rm ctr}$ and $w = w_{\rm ctr}$, leading to

$$\varphi(\theta_{\rm ctr}) = \gamma - \bar{\lambda}(x_{\rm ctr})\frac{\|c\|^2}{c^T w_{\rm ctr}}.$$

If $\gamma = \gamma^*$ the lower bound is exact because in this case $\bar{\lambda}(x_{\rm ctr}) = 0$. □

Remark. The assumption $c^T w_{\rm ctr} < 0$ is not very restrictive. It must hold in a neighborhood of the optimum, because under the Slater condition $c^T w < 0$ for all $w \in \partial\bar{\lambda}(x)$ with x optimal, and because the subdifferential is outer semicontinuous. We are mainly interested in the theorem for γ near γ^* anyway, and it is an easy condition to check.

Thus, from any x^k finding a center at $\gamma = c^T x^k$ with $c^T w_{\rm ctr} < 0$ gives a lower bound on the optimal cost according to Theorem 1. By (5) the centering problem may be solved via the ISBM, and we call this search a *centering phase*. Strictly speaking, Theorem 1 does not apply if we settle for a suboptimal solution to the centering problem. Nevertheless, we obtain an approximate lower bound for γ^* by obtaining an ε-optimal solution for small ε.

During the centering phase we monitor the duality gap estimate

$$\beta(x,w) := \bar{\lambda}(x)\frac{\|c\|^2}{c^T w}$$

for $w \in \partial\bar{\lambda}(x)$. If it exceeds a given cost tolerance $\gamma_{\rm tol}$ the centering phase is terminated, because $\beta(x,w)$ tends to increase as $x \to x_{\rm ctr}$. Otherwise, a $\nu_{\rm tol}$-optimal center is found and $\gamma - \beta(x,w)$ is accepted as the lower bound estimate. The stopping condition for the SDP solver is that $\beta(x,w) \leq \gamma_{\rm tol}$ after a centering phase. If the condition is not met, the minimization of the improvement function continues from the approximate center.

We note that the centering phase is not actually required in order to evaluate the stopping criterion. At any iteration k, the fact that $w^k \in \partial_{\tau^k}\bar{\lambda}(x^k)$ means that $B^T w^k$ is a τ^k-subgradient for the centering problem (5) at x^k (or $z = 0$). Therefore, if $\tau^k < \nu_{\rm tol}$, $\|B^T w^k\| < \nu_{\rm tol}$ and $0 < \beta(x^k, w^k) < \gamma_{\rm tol}$, the stopping condition is satisfied. The algorithm must terminate because $\tau^k \to 0$, $\|p^k\| \to 0$ implies $\|B^T w^k\| \to 0$, $c^T w^k$ is negative and decreasing for large enough k, and $\bar{\lambda}(x^k) \to 0$. However, performing a centering phase at the right time can significantly reduce the time to detect suboptimality because the centering phase is aimed specifically at achieving the stopping criterion.

4.2 When to start centering

Since a failed centering phase (one that produces a duality gap estimate that is too big) wastes time, it is important to have a good guideline for deciding when to start centering. We have observed that in the late stages of the SDP optimization, the convergence behavior of the cost is quite linear. This permits a prediction of the limiting cost, which is presumably the optimal cost γ^*. We fit the tail of the cost trajectory with a stable linear process $\gamma^{i+1} = a\gamma^i + b$, where a and b are the parameters. The limit estimate is then $\hat{\gamma}^* := b/(1-a)$. If the fit is good and $c^T x^k - \hat{\gamma}^* < \gamma_{\rm tol}$ we call the centering phase to verify the bound. In

our experience this test generates few false positives. Moreover, $\|B^T w^k\|$ is typically quite large at the beginning of a centering phase, so without centering many further iterations of the bundle method on the improvement function would be required to satisfy the stopping criterion.

4.3 Proximity weight control

The strategy for setting u^k in the centering phase is different from that of Section 3.2. It is similar to the one described in [8], in that u^k is decreased after serious steps, and increased after null steps when necessary to get η^k to decrease if it is nondecreasing on average after 20 straight null steps. We also decrease u^k to push $\|w^k\|$ down if $\|w^k\| > (1 + \frac{100}{\nu_{\text{tol}}}(\tau^k - \frac{1}{2}\nu_{\text{tol}}))\tau^k$. This has the effect of keeping $\|w^k\|$ below a τ^k-dependent threshold that gets closer to τ^k as τ^k decreases, becoming equal to τ^k at $\frac{1}{2}\nu_{\text{tol}}$. The threshold generally does not take effect until late in the centering phase, when $\tau^k \approx \nu_{\text{tol}}$.

5 Numerical Results

Since the performance of bundle methods is difficult to determine analytically, we resort to numerical experiment to estimate the average-case complexity of the algorithm for a class of problems. The experiment is meant to serve as a demonstration of the viability of applying the ISBM to semidefinite programming, not as a complete performance analysis of the algorithm.

As in [8], we reproduce Vandenberghe and Boyd's first numerical experiment in [11] and compare the results with their structured interior-point solver. All problems are of the form

$$\begin{array}{ll} \underset{P \in \mathcal{S}^N}{\text{minimize}} & \langle E, P \rangle \\ \text{subject to} & -A_k P - P A_k^T \leq 0, \quad k = 1, \ldots, L-1 \\ & I - 2P \leq 0 \end{array}$$

so $n = N(N+1)/2$ and $m = LN$, though the block-diagonal structure of $F(x)$ means that it really takes $O(LN^2)$ space instead of $O(m^2)$ space. The A_k's are random square matrices whose eigenvalues have real parts in the interval $(0, 1)$, and $E = 2I + \sum_{k=1}^{L-1}(A_k + A_k^T)$. As in [11], we start with the strictly feasible point $P = I$ and we use as a stopping condition the reduction of the duality gap by a factor of 1000 (in [11] the initial duality gap is simply $\operatorname{Tr} E$). The test cases were drawn from combinations of $N \in \{20, 30, 40, 60\}$ and $L \in \{20, 30, 40, 60, 100\}$. Not all combinations were tried due to time constraints.

Each bundle iteration performs an eigenvalue estimation and solves a search direction subproblem. The search direction subproblem is solved in $O(LN^2)$ flops (see [8]). The maximum eigenvalue is computed separately for each of the L blocks. A modified Lanczos process provides the estimate for a given block, performing one $N \times N$ matrix-vector multiplication each Lanczos iteration. It turns out that the average number of Lanczos iterations per block required to estimate an eigenvalue in this experiment is about $2.5N^{0.61}$, quite uniformly across all trials. Hence eigenvalue estimation is the dominant process, costing about $O(LN^{2.61})$ per bundle iteration, so we present the complexity results in terms of the number of $N \times N$ matrix-vector multiplications performed in the Lanczos process.

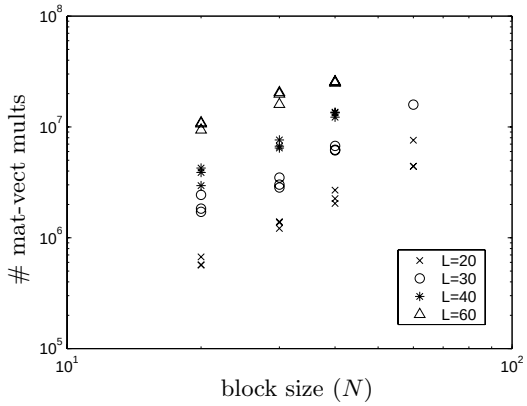

Figure 1: Power Law Dependence on N

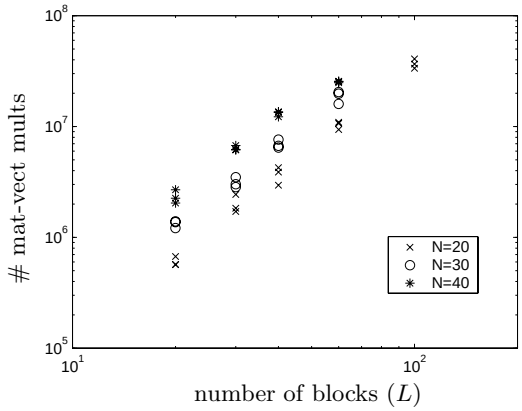

Figure 2: Power Law Dependence on L

Figures 1 and 2 plot the total number of $N \times N$ matrix-vector multiplications for each trial, excluding those performed in a centering phase. They show that the number of multiplications is well described by a power law in L and N. The best power law in a least-squares sense is $5.9L^{2.4}N^{1.8}$, and the fit is plotted in Figure 3. This shows that the complexity of reaching the solution (but not proving it is the solution) is about $O(L^{2.4}N^{3.8})$.

The reason for excluding the centering phase is depicted in Figure 4. The number of matrix-vector multiplications as a function of L and N is generally increasing but otherwise quite erratic. Clearly a more well-behaved procedure is required to evaluate the stopping

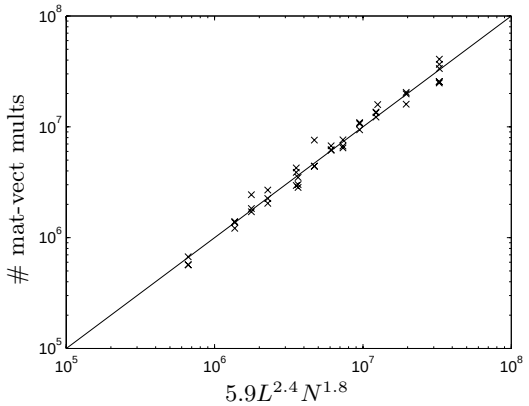

Figure 3: Least-Squares Fit of # Mat-Vect Mults

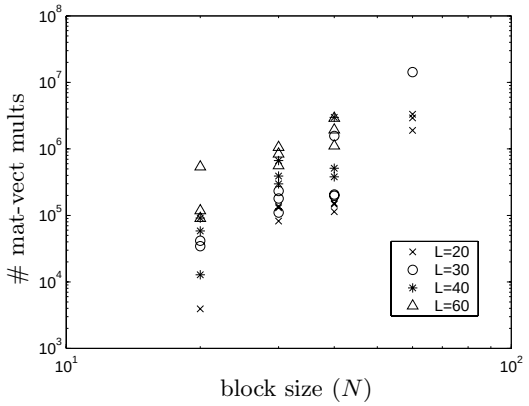

Figure 4: Mat-Vect Mults in Centering Phase

condition before the total complexity of the algorithm can be assessed in this way.

Compared to Vandenberghe and Boyd's structured interior-point algorithm [11], which achieves a complexity of about $O(L^{1.2}N^{4.2})$ experimentally, the bundle method described here seems to have a complexity that grows more quickly with L but less quickly with N. The higher-order dependence on L may be explained by the observation that the cost $\langle E, P\rangle$ grows linearly with L on average (because of the definition of E), but the rate of decrease of the cost achieved by the bundle method does not appear to grow much with L. Thus it takes longer to reach the optimum as L is increased. The rate of decrease of the cost can be adjusted by rescaling the cost or the constraint. This underscores the advantage of scale-independence that the constraint linearization method of [6] possesses but our method does not.

6 Conclusion

There is much room for improvement in this algorithm. The dependence of the method on the proper scaling of the constraints may have contributed to the observed quadratic growth with respect to L. Performance might be improved by employing an approach similar to the constraint linearization method of [6] but using the improvement function instead of the penalty function as the merit function for deciding serious vs. null steps. This would remove the scaling dependence yet preserve the feasible-point nature of the algorithm.

The experimental results show that the performance of the centering phase as currently implemented is unpredictable. Either better tuning rules for the proximity weight are needed, or a more efficient stopping criterion must be devised.

Still, these preliminary results show the promise of using the ISBM to solve large structured SDPs. The complexity is comparable to that achieved by an interior-point method specifically designed for the type of problem described in Section 5, but the ISBM applies to SDPs with any expression or matrix structure.

Acknowledgment: This work was supported by NSF grants ECS–9634498 and ECS–9978562.

References

[1] F. H. Clarke. *Optimization and Nonsmooth Analysis*. Wiley-Interscience, New York, 1983.

[2] Y. Ge, L. T. Watson, and E. G. Collins, Jr. Exploiting structure in a projective algorithm for solving strict linear matrix inequalities. In *Proc. Amer. Contr. Conf.*, pages 3089–90, Philadelphia, PA, June 1998.

[3] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. Technical Report SC 97-37, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Oct. 1997.

[4] K. C. Kiwiel. An exact penalty function algorithm for nonsmooth convex constrained minimization problems. *IMA J. Numer. Anal.*, 5(1):111–119, 1985.

[5] K. C. Kiwiel. *Methods of Descent for Nondifferentiable Optimization*, volume 1133 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1985.

[6] K. C. Kiwiel. A constraint linearization method for nondifferentiable convex minimization. *Numer. Math.*, 51(4):395–414, 1987.

[7] K. C. Kiwiel. Proximity control in bundle methods for convex nondifferentiable minimization. *Math. Prog.*, 46(1):105–122, Jan. 1990.

[8] S. A. Miller and R. S. Smith. A bundle method for efficiently solving large structured linear matrix inequalities. In *Proc. Amer. Contr. Conf.*, Chicago, IL, June 2000.

[9] K. Poolla, P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal. A time-domain approach to model validation. *IEEE Trans. Auto. Control*, 39(5):951–959, May 1994.

[10] H. Schramm and J. Zowe. A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results. *SIAM J. Opt.*, 2(1):121–152, Feb. 1992.

[11] L. Vandenberghe and S. Boyd. A primal-dual potential reduction method for problems involving matrix inequalities. *Math. Prog.*, 69(1):205–36, July 25, 1995.