Termination and Verification for Ill-Posed Semidefinite Programming Problems

Christian Jansson

June 17, 2005

Keywords: Linear programming, semidefinite programming, combinatorial optimization, rounding errors, ill-posed problems, interval arithmetic, branch-bound-and-cut
AMS Subject classification: primary 90C22, secondary65G20

Abstract

We investigate ill-posed semidefinite programming problems for which Slater's constraint qualifications fail, and propose a new reliable termination criterium dealing with such problems. This criterium is scale-independent and provides verified forward error bounds for the true optimal value, where all rounding errors due to floating point arithmetic are taken into account. It is based on a boundedness qualification, which is satisfied for many problems. Numerical experiments, including combinatorial problems, are presented.

1 Introduction

Most algorithms in linear and semidefinite programming require that appropriate rank conditions are fulfilled and that strictly feasible solutions of the primal and the dual problem exist; that is, it is assumed that *Slater's constraint qualifications* hold. The algorithms are terminated if appropriate residuals are small. These residuals measure approximately the primal feasibility, the dual feasibility, and the duality gap (see for example Mittelmann [18]).

The acceptance of approximations with small residuals is based on the following observation for linear systems $Ax = b$, which was probably first proved by Wilkinson (see [9, Chapter

1]): An approximate solution $\widetilde{x}$ is approved if the residual

$$r(\widetilde{x}) := \frac{\|b - A\widetilde{x}\|}{\|A\| \, \|\widetilde{x}\|} \tag{1}$$

is small. For the 2-norm, the importance of this measure is explained by the identity

$$r(\widetilde{x}) = \min \left\{ \frac{\|\Delta A\|_2}{\|A\|_2} : (A + \Delta A)\widetilde{x} = b \right\}. \tag{2}$$

Thus, $r(\widetilde{x})$ measures how much A must be perturbed such that $\widetilde{x}$ is the exact solution to the perturbed system. This residual is scale-independent, i.e., it does not change if the data A and b are multiplied by a factor α, and it can be viewed as a backward error. For well-conditioned problems with uncertain input data, approximate solutions with small residual must be regarded as satisfactory, provided every perturbation ΔA is allowed.

Nevertheless, there are many applications where backward error analysis may not be suitable. The first class consists of ill-conditioned problems with dependencies in the input data. In this case only specific perturbations ΔA are allowed, and the residual may be much smaller than the right hand side of (2) with the minimum formed with respect to the allowed perturbations. The second class are ill-posed problems for which Slater's constraint qualifications are not fulfilled (see Gruber and Rendl [8], and Gruber et al. [7]). For such problems the solution does not depend continuously on the input data, and small perturbations can result in infeasibility and/or erroneous approximations. Several problems become ill-posed due to the modelling (for example problems with redundant constraints, identically zero variables, and free variables transformed to variables bounded on one side), others appear as ill-posed relaxations in combinatorial optimization. Relaxations are widely used for efficiently solving difficult discrete problems with branch-bound-and-cut methods (see for example Goemans and Rendl [6]).

Ill-conditioned and ill-posed problems are not rare in practice, they occur even in linear programming. In a recent paper, Ordóñez and Freund [26] stated that 71% of the lp-instances in the NETLIB Linear Programming Library [20] are ill-posed. This library contains many industrial problems. Last, we want to mention (see Neumaier and Shcherbina [25]) that backward error analysis has no relevance for integer programs, since slightly perturbed coefficients no longer produce problems of the same class. There, one can also find an innocent-looking linear integer problem for which the commercial high quality solver CPLEX [10] and several other state-of-the-art solvers failed. The reason is that the linear programming relaxations are not solved with sufficient accuracy by the linear programming algorithm.

The major goal of this paper is to present a reliable termination criterium dealing with ill-conditioned and ill-posed problems. This criterium uses a scale-independent residual, which relates violations of primal and dual feasibility to intrinsic properties of the optimization problem. Additionally, it provides a rigorous lower bound for the primal optimal value as well as a rigorous upper bound for the dual optimal value. In contrast to computed approximations of the primal and dual optimal value, these bounds always satisfy weak duality. In most cases the required computational effort is small compared to the effort for computing approximate solutions. All rounding errors due to floating point arithmetic are rigorously estimated. This criterium requires two weak boundedness qualifications, but not Slater's constraint qualifications or additional rank conditions. It may be used inside the code of semidefinite solvers for the purpose of termination. But it is of particular importance that it can be used outside the code of any imaginable solver as a reliable postprocessing routine, providing a correct output for the given input. Especially, we show in the example case of Graph Partitioning how branch-and-bound algorithms for combinatorial optimization problems can be made safe, even if ill-posed relaxations are used. Preliminary numerical results for some ill-posed and ill-conditioned problems are included.

The presented results can be viewed as an extension of methods for linear programming problems (Jansson [12] and Neumaier and Shcherbina [25]) and for convex programming problems [11] to the ill-conditioned and ill-posed cases.

The paper is organized as follows. In Section 2 an introductory example is discussed. In Section 3 the basic theoretical setting is given, and in Section 4 the example case of Graph Partitioning is investigated. Section 5 considers some numerical details for termination, verification, and also for problems with uncertain input data. Finally, in Section 6 we present some numerical results with problems from the SDPLIB collection of Borchers [2], and furthermore a conclusion in Section 7.

2 Example

With the following introductory example we want to demonstrate the behaviour and some typical properties of the new termination criterium. All numerical results were obtained by using MATLAB [16], the interval toolbox INTLAB [30], and the semidefinite solver SDPT3 (version 3.02) [31].

By $\langle ., . \rangle$ we denote the inner product on the linear space of symmetric matrices defined as the trace of the product of two matrices. $X \succeq 0$ means that X is positive semidefinite. Hence, $\succeq$

denotes the *Löwner partial order* on this linear space. For fixed δ the semidefinite program

$$
\begin{aligned}
f_p^* := \min \quad & \langle \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & \delta & 0 \\ 0 & 0 & \delta \end{pmatrix}, X \rangle \\
\text{s.t.} \quad & \langle \begin{pmatrix} 0 & -\frac{1}{2} & 0 \\ -\frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, X \rangle = 1, \\
& \langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, X \rangle = 2\,\delta, \\
& \langle \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, X \rangle = 0, \\
& \langle \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, X \rangle = 0, \\
& X \succeq 0
\end{aligned}
$$

has the *Lagrangian dual*

$$
f_d^* := \max y_1 + 2\delta y_2 \quad \text{s.t.} \quad Y := \begin{pmatrix} -y_2 & \frac{1+y_1}{2} & -y_3 \\ \frac{1+y_1}{2} & \delta & -y_4 \\ -y_3 & -y_4 & \delta \end{pmatrix} \succeq 0,
$$

and thus weak duality implies $f_d^* \leq f_p^*$. The linear constraints of the primal problem yield

$$
X = \begin{pmatrix} 2\,\delta & -1 & 0 \\ -1 & X_{22} & 0 \\ 0 & 0 & X_{33} \end{pmatrix},
$$

and X is positive semidefinite iff $X_{22} \geq 0$, $X_{33} \geq 0$, and $2\,\delta \cdot X_{22} - (-1)^2 \geq 0$. Hence, for $\delta \leq 0$, the problem is primal infeasible with $f_p^* = +\infty$. The dual problem is infeasible for $\delta < 0$ with $f_d^* = -\infty$.

Table 1: Approximations $\widetilde{f}_p^*$, $\widetilde{f}_d^*$ and residuals $\widetilde{r}$, and r^*

δ	$\widetilde{f}_p^*$	$\widetilde{f}_d^*$	$\widetilde{r}$	r^*
1.0e-001	-5.00000e-001	-5.00000e-001	3.24751e-009	1.85517e-004
1.0e-002	-5.00000e-001	-5.00000e-001	1.93241e-009	1.26686e-006
1.0e-003	-5.00000e-001	-5.00000e-001	7.62771e-010	2.86746e-006
1.0e-004	-5.00032e-001	-5.00000e-001	1.29041e-008	1.27198e-003
1.0e-005	-7.69935e-001	-6.45752e-001	2.34663e-005	2.22265e+000
0	-1.00036e+000	-9.93546e-001	3.61091e-004	3.61555e+001

For $\delta = 0$ we obtain a duality gap with $f_p^* = +\infty$ and $f_d^* = -1$, and the problem is ill-posed. For $\delta > 0$ Slater's constraint qualifications are satisfied. The optimal value is $f_p^* = f_d^* = -1/2$ with corresponding optimal nonzero coefficients $X_{12}^* = -1, X_{22}^* = 1/(2\ \delta)$, and $y_2^* = -1/(4\ \delta)$. Especially, it follows that the optimal value is not continuous in $\delta = 0$.

This discontinuity is difficult in the following sense: If the factor 2 in the righthand side 2δ is replaced by some fixed positive factor k, then for every $\delta > 0$ the optimal value is $f_p^* = f_d^* = -1 + 1/k$ with optimal nonzero coefficients $X_{12}^* = -1, X_{22}^* = 1/(k\ \delta)$. Hence, each value in the interval $(-1, +\infty)$ is optimal. In other words, each value in $(-1, +\infty)$ is the exact optimal value for some arbitrarily small perturbed problems.

Numerical results for different values δ are summarized in Tables 1 and 2. The approximate primal and dual optimal value computed by SDPT3 are denoted by $\widetilde{f}_p^*$ and $\widetilde{f}_d^*$, respectively. The value $\widetilde{r}$ is the maximum of the relative gap and the measures for primal and dual infeasibility (for details see SDPT3 (version 3.02) [31]). In all experiments the default values of SDPT3 are used, and tc denotes the termination status, which is zero for normal termination without warning. The new residual is denoted by r^*, a rigorous lower bound of the primal optimal value is $\underline{f}_p^*$, and a rigorous upper bound of the dual optimal value is $\overline{f}_d^*$.

We see that SDPT3 is not backward stable, since in three cases $\widetilde{f}_p^* < \widetilde{f}_d^*$ violating the weak duality. For the two smallest values of δ no decimal digit of $\widetilde{f}_p^*$ or $\widetilde{f}_d^*$ is correct. In all cases the termination code is zero, and the approximate residual $\widetilde{r}$ leads to the suspicion that at least four decimal digits are correct. The new residual r^* (which uses the computed approximations of SDPT3) reflects much more the reliability of SDPT3, and the number of correct decimal digits for the computed result. The bounds $\overline{f}_d^*$ and $\underline{f}_p^*$ fulfill weak duality, and for $\delta > 0$ (where the duality gap is zero) the true optimal value $-1/2$ is inside the bounds, which is not the case for the approximations $\widetilde{f}_p^*$ and $\widetilde{f}_d^*$ corresponding to the values $\delta = 10^{-4}$ and $\delta = 10^{-5}$. For $\delta = 0$ the problem is ill-posed with infinite duality gap, but $\overline{f}_d^*$ is an upper bound of the dual optimal value, and $\underline{f}_p^*$ is a lower bound of the primal optimal value.

Summarizing, $r^*, \underline{f}_p^*$ and $\overline{f}_p^*$ check rigorously and reliable the output of SDPT3, and thus might improve this software significantly.

Table 2: Rigorous bounds $\overline{f}_d^*$, $\underline{f}_p^*$ and termination code tc

δ	$\overline{f}_d^*$	$\underline{f}_p^*$	tc
1.0e-001	-4.99814e-001	-5.00000e-001	0
1.0e-002	-4.99999e-001	-5.00000e-001	0
1.0e-003	-4.99997e-001	-5.00000e-001	0
1.0e-004	-4.98728e-001	-5.00000e-001	0
1.0e-005	1.57690e+000	-6.45752e-001	0
0	3.51614e+001	-9.94021e-001	0

3 Basic Theory

In this section we describe our theoretical setting for termination and verification in semidefinite programming. We consider the *(primal) semidefinite program* in *block diagonal form*

$$\begin{aligned} f_p^* := \min \sum_{j=1}^n \langle C_j, X_j \rangle \quad \text{s.t.} \quad & \sum_{j=1}^n \langle A_{ij}, X_j \rangle = b_i \quad \text{for } i = 1, \ldots, m, \\ & X_j \succeq 0 \quad \text{for } j = 1, \ldots, n, \end{aligned} \tag{3}$$

where $C_j, A_{ij}, X_j \in S^{s_j}$, the linear space of real symmetric $s_j \times s_j$ matrices, and $b \in \mathbf{R}^m$. We assign $f_p^* := +\infty$ if the set of feasible solutions is empty. In the case of only one block we suppress the subscript j, and write shortly C, X, A_i, and s. The input parameters are aggregated in the quantity $P = (A, b, C)$, where the subscripts are omitted.

If $s_j = 1$ for $j = 1, \ldots, n$ (i.e. C_j, A_{ij}, and X_j are real numbers), then (3) defines the standard linear programming problem. Hence, semidefinite programming is an extension of linear programming.

The *Lagrangian dual* of (3) is

$$f_d^* := \max b^T y \quad \text{s.t.} \quad \sum_{i=1}^m y_i A_{ij} \preceq C_j \quad \text{for } j = 1, \ldots, n, \tag{4}$$

where $y \in \mathbf{R}^m$. We assign $f_d^* := -\infty$ if the set of dual feasible solutions is empty. The constraints $\sum_{i=1}^m y_i A_{ij} \preceq C_j$ are called *linear matrix inequalities (LMI's)*.

The duality theory is similar to linear programming, but more subtle. The problems satisfy the *weak duality* condition

$$f_d^* \leq f_p^*, \tag{5}$$

but strong duality requires in contrast to linear programming additional conditions (see Ramana, Tunçel, and Wolkowicz [27] and Vandenberghe and Boyd [32]). It may happen that both optimal values are finite, but there is a nonzero duality gap, and an optimal solution does not exist. Also one optimal value may be finite and the other one may be infinite. If Slater's constraint qualifications are fulfilled, both optimal values are finite and strong duality holds.

For bounding rigorously the optimal value, we claim boundedness qualifications, which are more suitable for our purpose than Slater's constraint qualifications. The first one is called *dual boundedness qualification* (DBQ):

(i) Either the dual problem is infeasible,

(ii) or f_d^* is finite, and there are simple bounds $\overline{y} \in (\mathbf{R}_+ \cup \{ +\infty\})^m$, such that for every $\varepsilon > 0$ there exists a dual feasible solution $y(\varepsilon)$ satisfying

$$-\overline{y} \;\leq\; y(\varepsilon) \leq \overline{y}, \qquad \text{and} \qquad f_d^* - b^T y(\varepsilon) \;\leq\; \varepsilon. \tag{6}$$

Frequently, in applications $\overline{y}$ is finite. But we also treat the general case where some or all components of $\overline{y}$ may be infinite. Notice that DBQ is a rather weak condition. Even the existence of optimal solutions is not assumed. Only the following anomaly for finite $\overline{y}$ is kept out: problems where each sequence of dual feasible solutions is unbounded if their objective values converge to the dual optimal value. The following theorem provides a finite upper bound $\overline{f}_d^*$ for the dual optimal value.

Theorem 1 *Let $\widetilde{X}_j \in S^{s_j}$ for $j = 1, \ldots, n$, and assume that each $\widetilde{X}_j$ has at most k_j negative eigenvalues. Suppose that DBQ holds, and let*

$$r_i = b_i - \sum_{j=1}^{n} \langle A_{ij}, \widetilde{X}_j \rangle \quad \textit{for } i = 1, \ldots, m, \tag{7}$$

$$\underline{\lambda}_j = \lambda_{\min}(\widetilde{X}_j) \quad \textit{for } j = 1, \ldots, n, \qquad \textit{and} \tag{8}$$

$$\varrho_j = \sup\{\lambda_{\max}(C_j - \sum_{i=1}^{m} y_i A_{ij}) : \; -\overline{y} \leq y \leq \overline{y}, \; C_j - \sum_{i=1}^{m} y_i A_{ij} \succeq 0\} \tag{9}$$

for $j = 1, \ldots, n$. Then

$$f_d^* \leq \sum_{j=1}^{n} \langle C_j, \widetilde{X}_j \rangle - \sum_{j=1}^{n} k_j \underline{\lambda}_j^- \varrho_j + \sum_{i=1}^{m} |r_i| \overline{y}_i =: \overline{f}_d^*, \tag{10}$$

where $\underline{\lambda}_j^- := \min(0, \underline{\lambda}_j)$. Moreover, if

$$r_i = 0 \textit{ for } \overline{y}_i = +\infty \textit{ and } \underline{\lambda}_j \geq 0 \textit{ for } \varrho_j = +\infty, \tag{11}$$

then the right hand side $\overline{f}_d^*$ *is finite. If* $(\widetilde{X}_j)$ *is primal feasible then* $f_p^* \leq \overline{f}_d^*$*, and if* $(\widetilde{X}_j)$ *is optimal then* $f_p^* = \overline{f}_d^*$*.*

Notice, that the bound $\overline{f}_d^*$ sums up the violations of primal feasibility by taking into account the signs and multiplying these violations with appropriate weights. There are no further assumptions about the quality of the primal approximation $(\widetilde{X}_j)$. For well-posed as well as ill-posed problems with zero duality gap (for example linear programming problems or problems satisfying Slater's constraint qualifications) $\overline{f}_d^*$ is an upper bound of the primal optimal value. Moreover, it follows that an approximation close to a primal optimal solution yields an upper bound close to the primal optimal value. But for ill-posed problems with nonzero duality gap, we stress that $\overline{f}_d^*$ is only an upper bound for the dual optimal value, not for the primal one (see also the computational results in Section 2 for $\delta = 0$). We emphasize that the bound $\overline{f}_d^*$ does not verify primal feasibility. This is reflected by the property that only residuals are required and not a rigorous solution of a linear system, which would need much more computational effort. This is in contrast to the upper bound presented in [13].

The exact computation of the quantities $\underline{\lambda}_j$ and ϱ_j is in general not possible. But the inequality (10) and the finiteness of $\overline{f}_d^*$ is also satisfied if only bounds

$$\begin{aligned} \underline{\lambda}_j &\leq \lambda_{\min}(\widetilde{X}_j) \quad \text{for } j = 1, \ldots, n, \qquad \text{and} \\ \varrho_j &\geq \sup\{\lambda_{\max}(C_j - \sum_{i=1}^{m} y_i A_{ij}) : \; -\overline{y} \leq y \leq \overline{y}, \; C_j - \sum_{i=1}^{m} y_i A_{ij} \succeq 0\} \end{aligned} \tag{12}$$

are known. Hence, the computation of such bounds is sufficient, and some details are given in Section 5.

Proof. If the dual problem is not feasible, then $f_d^* = -\infty$ and each value is an upper bound. Hence, we assume that condition (ii) of DBQ holds valid. If $r_i \neq 0$ for some $\overline{y}_i = +\infty$ or $\underline{\lambda}_j < 0$ for some $\varrho_j = +\infty$, then $\overline{f}_d^* = +\infty$ is an upper bound. Hence, let (11) be satisfied, and let $\varepsilon > 0$. Then DBQ implies the existence of a dual feasible solution $y(\varepsilon)$ such that

$\sum_{i=1}^{m} y_i(\varepsilon)A_{ij} + Z_j = C_j,\ Z_j \succeq 0,\ -\overline{y}\ \le y(\varepsilon) \le\ \overline{y}$, and

$$
\begin{aligned}
f_d^* - \varepsilon &\le b^T y(\varepsilon) \\
&= \sum_{i=1}^{m} b_i y_i(\varepsilon) = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} \langle A_{ij}, \widetilde{X}_j \rangle + r_i \right) y_i(\varepsilon) \\
&= \left(\sum_{j=1}^{n} \left\langle \sum_{i=1}^{m} y_i(\varepsilon) A_{ij}, \widetilde{X}_j \right\rangle \right) + \sum_{i=1}^{m} r_i y_i(\varepsilon) \\
&= \sum_{j=1}^{n} \langle C_j - Z_j, \widetilde{X}_j \rangle + \sum_{i=1}^{m} r_i y_i(\varepsilon) \\
&= \sum_{j=1}^{n} \langle C_j, \widetilde{X}_j \rangle - \sum_{j=1}^{n} \langle Z_j, \widetilde{X}_j \rangle + \sum_{i=1}^{m} r_i y_i(\varepsilon).
\end{aligned}
$$

The symmetry of $\widetilde{X}_j$ yields an eigenvalue decomposition

$$\widetilde{X}_j = Q_j \Lambda(\widetilde{X}_j) Q_j^T, \quad Q_j Q_j^T = I,$$

and

$$-\langle Z_j, \widetilde{X}_j \rangle = \mathrm{trace}(-Z_j Q_j \Lambda(\widetilde{X}_j) Q_j^T) = \mathrm{trace}(Q_j^T(-Z_j) Q_j \Lambda(\widetilde{X}_j)).$$

Hence,

$$-\langle Z_j, \widetilde{X}_j \rangle = \sum_{k=1}^{s_j} \lambda_k(\widetilde{X}_j) Q_j(:,k)^T (-Z_j) Q_j(:,k). \tag{13}$$

By definition,

$$0 \preceq Z_j \preceq \lambda_{\max}(C_j - \sum_{i=1}^{m} y_i(\varepsilon) A_{ij}) \cdot I$$

implying

$$-\varrho_j I \preceq -Z_j \preceq 0, \text{ and } -\varrho_j \le Q_j(:,k)^T(-Z_j)Q_j(:,k) \le 0.$$

Therefore,

$$-\langle Z_j, \widetilde{X}_j \rangle \le k_j(-\varrho_j) \cdot \underline{\lambda}_j^-,$$

and it follows that

$$-\sum_{j=1}^{n}\langle Z_j, \widetilde{X}_j\rangle \leq -\sum_{j=1}^{n} k_j \varrho_j \underline{\lambda}_j^{-}.$$

Since

$$\sum_{i=1}^{m} r_i y_i(\varepsilon) \leq \sum_{i=1}^{m} |r_i| \overline{y}_i,$$

the inequality $f_d^* - \varepsilon \leq \overline{f}_d^*$ holds valid for every $\varepsilon > 0$. This proves formula (10).

If condition (11) is satisfied, then the two sums on the right hand side of (10) are finite. Hence, $\overline{f}_d^*$ is finite. If $\widetilde{X}_j$ is primal feasible, then $\underline{\lambda}_j^-$ and r_i are zero for all i, j. Hence, $f_d^* \leq \sum_{j=1}^{n}\langle C_j, \widetilde{X}_j\rangle = \overline{f}_d^*$, and equality holds iff $(\widetilde{X}_j)$ is optimal. ■

The second qualification is called the *primal boundedness qualification* (PBQ):

(i) Either the primal problem is infeasible,

(ii) or f_p^* is finite, and there are simple bounds $\overline{x} \in (\mathbf{R}_+ \cup \{+\infty\})^n$ such that for every $\varepsilon > 0$ exists a primal feasible solution $(X_j(\varepsilon))$ satisfying

$$\lambda_{\max}(X_j(\varepsilon)) \leq \overline{x}_j \text{ for } j = 1, \dots, n, \text{ and } \sum_{j=1}^{n}\langle C_j, X_j(\varepsilon)\rangle - f_p^* \leq \varepsilon. \tag{14}$$

The following theorem provides a finite lower bound $\underline{f}_p^*$ of the primal optimal value.

Theorem 2 *Assume that PBQ holds, and let $\widetilde{y} \in \mathbf{R}^m$. Let*

$$D_j = C_j - \sum_{i=1}^{m} \widetilde{y}_i A_{ij}, \text{ and } \underline{d}_j := \lambda_{\min}(D_j) \text{ for } j = 1, \dots, n. \tag{15}$$

Assume that D_j has at most l_j negative eigenvalues. Then

$$f_p^* \geq b^T \widetilde{y} + \sum_{j=1}^{n} l_j \underline{d}_j^- \overline{x}_j =: \underline{f}_p^*. \tag{16}$$

Moreover, if

$$\underline{d}_j \geq 0 \text{ for } \overline{x}_j = +\infty, \tag{17}$$

then the right hand side $\underline{f}_p^*$ is finite. If $\widetilde{y}$ is dual feasible then $f_d^* \geq \underline{f}_p^*$, and if $\widetilde{y}$ is optimal then $f_d^* = \underline{f}_p^*$.

As before, the bound $\underline{f}_p^*$ sums up the violations of dual feasibility by taking into account the signs and multiplying these violations with appropriate primal weights. Obviously, the inequality (16) is also satisfied for any $\underline{d}_j$ with $\underline{d}_j \leq \lambda_{\min}(D_j)$.

Proof. If the primal problem is not feasible, then $f_p^* = +\infty$, and each value is a lower bound. Hence, we assume that condition (ii) of PBQ holds valid. If $\underline{d}_j < 0$ for $\overline{x}_j = +\infty$, then $\underline{f}_p^* = -\infty$ is a lower bound. Hence, let (17) be satisfied, and let $\varepsilon > 0$. Then PBQ implies the existence of a primal feasible solution $(X_j) = (X_j(\varepsilon))$ with

$$\sum_{j=1}^{n} \langle C_j, X_j \rangle - f_p^* \leq \varepsilon.$$

Let

$$D_j = C_j - \sum_{i=1}^{m} \widetilde{y}_i A_{ij} \quad \text{for } j = 1, \ldots, n,$$

then

$$\sum_{j=1}^{n} \langle C_j, X_j \rangle = \sum_{j=1}^{n} \langle D_j + \sum_{i=1}^{m} \widetilde{y}_i A_{ij}, X_j \rangle = b^T \widetilde{y} + \sum_{j=1}^{n} \langle D_j, X_j \rangle.$$

Let D_j have the eigenvalue decomposition

$$D_j = Q_j \Lambda(D_j) Q_j^T, \quad Q_j Q_j^T = I,$$

where $\Lambda(D_j)$ is the diagonal matrix with eigenvalues of D_j on the diagonal. Then

$$\langle D_j, X_j \rangle = \text{ trace } (Q_j \Lambda(D_j) Q_j^T X_j) = \text{ trace } (\Lambda(D_j) Q_j^T X_j Q_j).$$

Hence,

$$\langle D_j, X_j \rangle = \sum_{k=1}^{s_j} \lambda_k(D_j) Q_j(:,k)^T X_j Q_j(:,k), \tag{18}$$

and since the eigenvalues of X_j are bounded, we have $0 \leq Q_j(:,k)^T X_j Q_j(:,k) \leq \overline{x}_j$, yielding

$$\langle D_j, X_j \rangle \geq \sum_{k=1}^{s_j} \lambda_k^-(D_j) \cdot \overline{x}_j \ \geq \ l_j \cdot \underline{d}_j^- \cdot \overline{x}_j.$$

Hence, we obtain

$$\sum_{j=1}^{n} \langle D_j, X_j \rangle \geq \sum_{j=1}^{n} l_j \cdot \underline{d}_j^- \cdot \overline{x}_j,$$

which implies $b^T \widetilde{y} + \sum_{j=1}^{n} l_j \underline{d}_j^- \overline{x}_j \leq f_p^* + \varepsilon$ for every $\varepsilon > 0$. This proves (16). If condition (17) is satisfied then all products $\underline{d}_j^- \cdot \overline{x}_j$ are finite, and therefore $\underline{f}_p^*$ is finite. If $\widetilde{y}$ is dual feasible then $\underline{d}_j = 0$ for every $j = 1, \ldots, n$. Hence $\underline{f}_p^* = b^T \widetilde{y} \leq f_p^*$, and equality holds if $\widetilde{y}$ is optimal. ■

The computation of lower bounds of $\underline{\lambda}_j$ suffices for computing a rigorous lower bound of the optimal value. Details for computing rigorously the quantities $|r_i|, \underline{\lambda}_j, \varrho_j$, and $\underline{d}_j$ are described in Section 5.

For judging the accuracy of the approximate solution $\widetilde{X}, \widetilde{y}$ of a semidefinite programming problem we define the residual

$$r^*(\widetilde{X}, \widetilde{y}) = (\overline{f}_d^* - \underline{f}_p^*)/\|P\|. \tag{19}$$

The following theorem describes the main properties of r^*, and shows that this measure provides a reasonable termination criterium, also for ill-posed problems.

Theorem 3 *Suppose that the boundedness qualifications PBQ and DBQ are fulfilled, and assume that for the approximations $\widetilde{X} = (\widetilde{X}_j)$, $\widetilde{y}$, and for all $i = 1, \ldots, m$, $j = 1, \ldots, n$ the conditions*

$$r_i = 0 \quad \textit{for} \quad \overline{y}_i = +\infty, \tag{20}$$

$$\underline{\lambda}_j \geq 0 \quad \textit{for} \quad \varrho_j = +\infty, \tag{21}$$

$$\underline{d}_j \geq 0 \quad \textit{for} \quad \overline{x}_j = +\infty \tag{22}$$

hold valid. Then:

(i) The residual

$$r^*(\widetilde{X}, \widetilde{y}) = \left(\sum_{j=1}^{n} \langle C_j, \widetilde{X}_j \rangle - b^T \widetilde{y} - \left(\sum_{j=1}^{n} k_j \underline{\lambda}_j^- \varrho_j + l_j \underline{d}_j^- \overline{x}_j - \sum_{i=1}^{m} |r_i| \overline{y}_i \right) \right) / \|P\| \tag{23}$$

is finite.

(ii) *If* $\widetilde{X}$, $\widetilde{y}$ *are optimal, then*

$$r^*(\widetilde{X}, \widetilde{y}) = (f_p^* - f_d^*)/\|P\|, \tag{24}$$

and for problems with zero duality gap $r^*(\widetilde{X}, \widetilde{y}) = 0$.

(iii) *If* $\sum_{j=1}^{n} \langle C_j, \widetilde{X}_j \rangle \geq b^T \widetilde{y}$ *and* $0 \leq r^*(\widetilde{X}, \widetilde{y}) \leq \varepsilon$ *then for all* i, j

$$0 \leq -\underline{\lambda}_j^- \varrho_j,\ -\underline{d}_j^- \overline{x}_j,\ |r_i| \overline{y}_i,\ |\sum_{j=1}^{n} \langle C_j, \widetilde{X}_j \rangle - b^T \widetilde{y}| \leq \varepsilon \|P\|. \tag{25}$$

(iv) *For problems with zero duality gap and* $0 \leq r^*(\widetilde{X}, \widetilde{y})$ *it is* $\underline{f}_p^* \leq f_p^* = f_d^* \leq \overline{f}_d^*$*; that is the bounds* $\underline{f}_p^*$ *and* $\overline{f}_d^*$ *provide an enclosure of the exact optimal value.*

(v) *If* $r^*(\widetilde{X}, \widetilde{y}) < 0$*, then a nonzero duality gap with* $f_p^* - f_d^* \geq |r^*(\widetilde{X}, \widetilde{y})| \cdot \|P\|$ *exists.*

(vi) r^* *is scale-independent.*

Proof. (i) and (ii) are immediate consequences of Theorems 1 and 2.

(iii) follows, because $0 \leq -\left(\sum_{j=1}^{n} k_j \varrho_j \underline{\lambda}_j^- + l_j \underline{d}_j^- \overline{x}_j - \sum_{i=1}^{m} |r_i| \overline{y}_i \right) \leq \varepsilon \|P\|$.

(iv) Since $r^* \geq 0$ we obtain $\underline{f}_p^* \leq \overline{f}_d^*$, and the relations $\underline{f}_p^* \leq f_p^*$, $\overline{f}_d^* \geq f_d^*$, $f_d^* = f_p^*$ yield $\underline{f}_p^* \leq f_d^* = f_p^* \leq \overline{f}_d^*$.

(v) Because $r^*(\widetilde{X}, \widetilde{y}) < 0$ by definition, it is $\overline{f}_d^* < \underline{f}_p^*$. Theorems 1 and 2 show that $\overline{f}_d^* \geq f_d^*$ and $\underline{f}_p^* \leq f_p^*$, yielding the duality gap.

(vi) If we multiply the input data $P = (A, b, C)$ by a positive factor $\alpha > 0$, then some short computations show the following relations for $\widehat{P} := \alpha P$:

$\widehat{f}_p^* = \alpha f_p^*$, $\widehat{f}_d^* = \alpha f_d^*$, $\widehat{r}_i = \alpha r_i$, $\widehat{\varrho}_j = \alpha \varrho_j$, $\widehat{d}_j = \alpha d_j$ and $\widehat{\underline{\lambda}} = \lambda_j$.

Hence, the bounds fulfill $\widehat{\overline{f}}_d^* = \alpha \overline{f}_d^*$ and $\widehat{\underline{f}}_p^* = \alpha \underline{f}_p^*$ yielding the scale independency of r^*. ∎

The property (ii) shows that for an optimal solution $\widetilde{X}$, $\widetilde{y}$ the residual $r^*(\widetilde{X}, \widetilde{y})$ gives the exact normalized duality gap, and (iii) demonstrates that a small nonnegative residual implies that the primal feasibility, dual feasibility and optimality are satisfied apart from the small quantity $\varepsilon \|P\|$. Here, the measure of infeasibility is weighted and related to appropriate upper bounds ϱ_j, $\overline{x}_j$, $\overline{y}_i$ and $\|P\|$. In other words, the numbers $\underline{\lambda}_j^-$, $-\underline{d}_j^-$ and $|r_i|$ have no absolute quantitative meaning, only together with the corresponding multipliers and the norm of the problem a reasonable judgment is possible. It is an open problem, whether $r^*(\widetilde{X}, \widetilde{y})$ measures how much the input data P of a well-posed semidefinite problem must be perturbed w.r.t. some norm such that $r^*(\widetilde{X}, \widetilde{y})$ is the exact solution to the perturbed problem.

Properties (iv) and (v) show that r^* serves as a forward error bound, which can be used also for ill-conditioned or ill-posed problems. No perturbation arguments are necessary. We mention that the error bounds used for termination in many semidefinite programming codes are not scale-independent and provide no rigorous forward error bounds (see Mittelmann [18]).

Last, we want to remind that in the special case of linear programming the duality gap is always zero, and parts of the previous theory are simplified. Since all matrices are one-dimensional, a lower bound of an eigenvalue is just the matrix itself.

4 Graph Partitioning Problems

In this section we consider Graph Partitioning as a first application of the previous theory. These combinatorial problems are known to be NP-hard, and finding an optimal solution is difficult. Graph Partitioning has many applications among those is VLSI design; for a survey see Lengauer [15]. Several relaxation techniques can be used for efficiently solving this problem. Because of the nonlinearity introduced by the positive semidefinite cone, semidefinite relaxations provide tighter bounds for many combinatorial problems than linear programming relaxations. Here, we investigate the special case of the Equicut Problem and the semidefinite relaxations proposed by Gruber and Rendl [8]. These have turned out to deliver tight lower bounds.

Given an edge-weighted graph G with an even number of vertices, the problem is to find a partitioning of the vertices into two sets of equal cardinality which minimizes the weight of the edges joining the two sets. The algebraic formulation is obtained by representing the partitioning as an integer vector $x \in \{-1, 1\}^n$ satisfying the parity condition $\sum_i x_i = 0$. Then the Equicut Problem is equivalent to

$$\min \sum_{i<j} a_{ij} \frac{1 - x_i x_j}{2} \quad \text{subject to} \quad x \in \{-1, 1\}^n, \ \sum_{i=1}^{n} x_i = 0,$$

where $A = (a_{ij})$ is the symmetric matrix of edge weights. This follows immediately, since $1 - x_i x_j = 0$ iff the vertices i and j are in the same set. The objective can be written as

$$\frac{1}{2} \sum_{i<j} a_{ij}(1 - x_i x_j) = \frac{1}{4} x^T (\mathrm{Diag}(Ae) - A)x = \frac{1}{4} x^T L x,$$

where e is the vector of ones, and $L := \mathrm{Diag}(Ae) - A$ is the *Laplace matrix* of G. Using $x^T Lx = \mathrm{trace}(L(xx^T))$ and $X = xx^T$, it can be shown that this problem is equivalent to

$$f_p^* = \min \frac{1}{4}\langle L, X\rangle \quad \text{subject to} \quad \mathrm{diag}(X) = e,\ e^T Xe = 0,\ X \succeq 0,\ \mathrm{rank}(X) = 1.$$

Since $X \succeq 0$ and $e^T Xe = 0$ implies X to be singular, the problem is ill-posed, and for arbitrarily small perturbations of the right hand side the problem becomes infeasible. By definition, the Equicut Problem has a finite optimal value f_p^*, and a rigorous upper bound of f_p^* is simply obtained by evaluating the objective function for a given partitioning integer vector x.

In order to compute a rigorous lower bound, the nonlinear rank one constraint is left out yielding an ill-posed semidefinite relaxation, where the Slater condition does not hold. The related constraints $\mathrm{diag}(X) = e$ and $e^T Xe = 0$ can be written as

$$\langle A_i, X\rangle = b_i,\ b_i = 1,\ A_i = E_i \text{ for } i = 1, \ldots, n, \text{ and } A_{n+1} = ee^T,\ b_{n+1} = 0.$$

where E_i is the $n \times n$ matrix with a one on the ith diagonal position and zeros otherwise. Hence, the dual semidefinite problem is

$$\max \sum_{i=1}^{n} y_i \quad \text{s.t.} \quad \mathrm{Diag}(y_{1:n}) + y_{n+1}(ee^T) \preceq \frac{1}{4}L, \quad y \in \mathbf{R}^{n+1}.$$

Since the feasible solutions $X = xx^T$ with $x \in \{-1, 1\}^n$ satisfy $\overline{x} = \lambda_{\max}(X) = n$, the constraint qualification PBQ are fulfilled. Hence, using Theorem 2 we obtain:

Corollary 4.1 *Let $\widetilde{y} \in \mathbf{R}^{n+1}$, and assume that the matrix*

$$D = \frac{1}{4}L - \mathrm{Diag}(\widetilde{y}_{1:n}) - \widetilde{y}_{n+1}(ee^T).$$

has at most l negative eigenvalues, and let $\underline{d} \leq \lambda_{\min}(D)$. Then

$$f_p^* \geq \sum_{i=1}^{n} \widetilde{y}_i + l \cdot n \cdot \underline{d}^- =: \underline{f}_p^*.$$

In Table 3 we display some numerical results for problems which are given by Gruber and Rendl [8]. Matlab m-files can be found at *http://uni-klu.ac.at/groups/math/optimization/*. The number of nodes is n, and the accuracy is measured by

$$\mu(a, b) := \frac{|a - b|}{\max\{1.0, (|a| + |b|)/2\}},$$

n	$\widetilde{f}_d^*$	$\underline{f}_p^*$	$\mu(\widetilde{f}_p^*, \widetilde{f}_d^*)$	$\mu(\widetilde{f}_d^*, \underline{f}_p^*)$
100	-3.58065e+003	-3.58065e+003	7.11732e-008	2.09003e-008
200	-1.04285e+004	-1.04285e+004	7.01770e-008	6.86788e-008
300	-1.90966e+004	-1.90966e+004	2.57254e-008	3.78075e-007
400	-3.01393e+004	-3.01393e+004	1.63322e-008	3.82904e-007
500	-4.22850e+004	-4.22850e+004	1.43078e-008	6.45308e-007
600	-5.57876e+004	-5.57876e+004	5.41826e-009	1.05772e-006

Table 3: Approximate and rigorous results

n	t	t_1	t_2
100	2.98	0.08	0.61
200	8.81	0.19	3.66
300	21.19	0.44	11.11
400	41.27	0.89	27.22
500	71.63	1.64	57.78
600	131.47	2.69	93.11

Table 4: Times in seconds

For this suite of ill-posed problems with about up to 600 constraints and 180000 variables SDPT3 has computed approximate lower bounds $\widetilde{f}_d^*$ of the optimal value, which are close to $\widetilde{f}_p^*$ (see the column $\mu(\widetilde{f}_p^*, \widetilde{f}_d^*)$). The small quantities $\mu(\widetilde{f}_d^*, \underline{f}_p^*)$ show that the overestimation of the rigorous lower bound $\underline{f}_p^*$can be neglected. SDPT3 gave $tc = 0$ (normal termination) for the first five examples. Only in the last case $n = 600$ the warning $tc = -5$*: Progress too slow* was returned, but a close rigorous lower bound is computed.

In Table 4, we display the times t, t_1, and t_2 for computing the approximations with SDPT3, for computing $\underline{f}_p^*$ by using Corollary 4.1, and for computing $\underline{f}_p^*$ by using Theorem 2, respectively. The additional time t_1 for computing the rigorous bound $\underline{f}_p^*$ is small compared to the time t needed for the approximations. The time t_2 is much larger because the special structure of the problems is not taken into account. This large increase is due to special MATLAB and INTLAB characteristics in our current implementation. A C program should be much more efficient.

Summarizing, Corollary 4.1 facilitates cheap and rigorous lower bounds for the optimal value of graph partitioning problems, yielding reliable results for related branch-bound-and-cut methods.

5 Verification

It is obvious how the quantities $|r_i|, \underline{\lambda}_j, \varrho_j, \underline{d}_j$ can be computed approximately. Following, we discuss rigorous estimates.

First, we consider the computation of rigorous lower bounds $\underline{\lambda}_j, \underline{d}_j$ for the symmetric matrices X_j and D_j, respectively. This can be done by using an approach due to Rump [29] for solving rigorously sparse linear systems. We describe shortly the part required for computing a lower bound $\underline{\lambda}$ of a symmetric matrix X. The matrix $X - \lambda I$ is positive definite iff $\lambda_{\min}(X) > \lambda$. Given an approximation $\widetilde{\lambda}$ of the smallest eigenvalue of X, we take some $\lambda < \widetilde{\lambda}$ (for example $\lambda := 0.99\widetilde{\lambda}$ if $\widetilde{\lambda} > 0$ and $\lambda := 1.01\widetilde{\lambda}$ if $\widetilde{\lambda} < 0$), and compute an approximate Cholesky decomposition LL^T of $X - \lambda I$. If LL^T can be computed, perturbation analysis for eigenvalues of symmetric matrices yields with $E := LL^T - (X - \lambda I)$

$$|\lambda_{\min}(LL^T) - \lambda_{\min}(X - \lambda I)| = |\lambda_{\min}(X - \lambda I + E) - \lambda_{\min}(X - \lambda I)| \leq \|E\|.$$

Since $LL^T = X - \lambda I + E \preceq X - (\lambda - \|E\|) \cdot I$ is positive semidefinite, we obtain the lower bound

$$\lambda_{\min}(X) > \lambda - \|E\| =: \underline{\lambda}.$$

If the Cholesky decomposition cannot be computed, we decrease the λ as above and repeat the process. Notice that for computing $\underline{\lambda}$, only the upper bound of $\|E\|$ must be rigorously estimated (for example by using the monotonic rounding modes), all other computations are performed by using floating point arithmetic with rounding to nearest.

Using this method, the only available bound for the number of negative eigenvalues is the dimension; that is, k_j and l_j must be set equal to s_j. In all numerical experiments we have used this approach. Another possibility is to compute bounds for all eigenvalues, for example by applying Gershgorin's Theorem to the matrix VXV^T, where V contains approximate orthonormal eigenvectors. Then we obtain better estimates of k_j and l_j, but the computational costs increase. Interesting references for computing rigorous bounds of some or all eigenvalues are Floudas [3], Mayer [17], Neumaier [24], and Rump [28, 29].

In Theorem 1 upper bounds of the residuals $|r_i|$ are required. One possibility is to calculate the residuals by using interval arithmetic, and taking the supremum of the computed interval quantities. This is done in our numerical experiments. There are a number of textbooks on interval arithmetic that can be highly recommended to readers. These include Alefeld and Herzberger [1], Kearfott [14], Moore [19], and Neumaier [21], [22]. Of course, there are a variety of other methods yielding upper bounds; for example by using appropriately the rounding modes, or by using some higher precision arithmetic. The advantage of using interval arithmetic is the possibility of including uncertain interval data b_i, A_{ij}. The advantage of using exact arithmetic [4] or multi precision arithmetic [5] is that the exact solution of the linear system is computed yielding $|r_i| = 0$ for $i = 1, ..., m$. Hence, there might be an evident improvement of the bounds for the optimal value, which has to be compared with the computational costs.

Last, we consider the computation of an upper bound for ϱ_j. From Perron-Frobenius theory (see for example [21]) it follows that for each y with $-\overline{y} \leq y \leq \overline{y}$ it is

$$\lambda_{\max}(C_j - \sum_{i=1}^{m} y_i A_{ij}) \leq \varrho(C_j - \sum_{i=1}^{m} y_i A_{ij}) \leq \varrho(|C_j| + \sum_{i=1}^{m} \overline{y}_i |A_{ij}|),$$

where ϱ denotes the spectral radius. Defining

$$U_j := |C_j| + \sum_{i=1}^{m} \overline{y}_i |A_{ij}|$$

we obtain the upper bound

$$\varrho_j \leq \varrho(U_j) \leq \|U_j\|.$$

This rough bound may be improved by incorporating in special cases the relaxed constraints $C_j - \sum_{i=1}^{m} y_i A_{ij} \succeq 0$. However, this improvement increases the computational costs for ϱ_j.

The bounds ϱ_j are finite for problems with finite $\overline{y}$. Hence, condition (11) is trivially satisfied, and $\overline{f}_d^*$ is finite. If components of $\overline{y}$ are infinite, in general a verification of condition (11) requires the existence of interior primal feasible solutions. Otherwise, $\overline{f}_d^*$ would be infinite, because numerically either $r_i \neq 0$ or $\underline{\lambda}_j < 0$ for some i, j. A method for computing in this case a finite upper bound is discussed in [12], [13]. There, also the case of infinite bounds $\overline{x}$ is treated.

Finally, we want to mention that the previous theory also allows to consider problems with interval input data. Corresponding corollaries can be formulated in a canonical way by using the inclusion isotonicity principle of interval arithmetic.

6 SDPLIB Problems

In practice, there are frequently situations where details of modelling a problem or the generation of input data may not be known precisely, and may cause ill-posed problems. For example because of redundant constraints, identically zero variables, describing free variables as the difference of nonnegative variables, or replacing a vector by its outer product as in Section 4, the constraints do not satisfy Slater's constraint qualifications, but the boundedness of optimal solutions is not touched. Therefore, the previous theory may be used if either the user has a rough idea about the order of magnitude of the optimal solutions, or if he accepts that the absolute value of the optimal solutions is not much larger than the absolute value of the computed approximations multiplied by some positive factor, i.e., he trusts the order of magnitude:

$$\overline{x}_j = \mu \cdot \lambda_{\max}(\widetilde{X}_j) \text{ for } j = 1, \ldots, n, \text{ and } \overline{y}_i = \mu \cdot |\widetilde{y}_i| \text{ for } i = 1, \ldots, m.$$

These bounds can be viewed as a form of a-posteriori regularization for judging the computed approximate solution of an ill-posed optimization problem. They are not used for computing a regularized approximation. Because this boundedness assumption is not (completely) verified, the results are not fully rigorous. Nevertheless, this stage of rigor is with rounding error control and without uncontrolled approximations, and we may speak of a *rounding error controlled weak verification.* An interesting presentation of stages of rigor for numerical methods can be found in Neumaier [23].

For the examples in Section 2, we have used the upper bound 10^5 for all variables. By inspection of the optimal solutions it follows that DBP and PBQ are satisfied, yielding fully rigorous results.

Following, we describe in Tables 5 and 6 the numerical results on some ill-conditioned or ill-posed problems from the SDPLIB collection of Borchers [2], where we use the above bounds with a factor $\mu = 10$. These problems were examined in [13], and in all cases the rigorous upper bound was infinite, indicating primal infeasibility or an ill-posed problem. The computational times in seconds are t, $\bar{t}$, and $\underline{t}$ for computing the approximations with SDPT3, for the upper bound $\overline{f}_d^*$, and for the lower bound $\underline{f}_p^*$, respectively. The termination code was in all cases $tc = 0$ (normal termination) with exception of problem $hinf7$ where $tc = -4$, which means that the Schur complement matrix becomes singular.

We see that in many cases both, $\widetilde{r}$ and r^*, indicate the useability of the computed approximations, although the factor $\mu = 10$ introduces some overestimation in r^* (increasing μ increases safety but in addition overestimation). For some problems both values differ drastically, for example *hinf12.* SDPT3 has computed for this problem the approximate optimal values $\widetilde{f}_d^* = -0.78304$, $\widetilde{f}_p^* = -1.56488$ and the values $1.4810^{-8}, 4.6210^{-8}, 2.2410^{-10}$ for the relative gap, primal infeasibility, and dual infeasibility, respectively. Especially, it follows that weak duality is violated by these approximations. However, the residual r^* indicates that there is something wrong and that there can be no confidence.

problem	$\overline{f}_d^*$	$\underline{f}_p^*$	$\widetilde{r}$	r^*
gpp100	4.49436e+001	4.49435e+001	1.03536e-007	5.49639e-007
gpp124-1	7.34313e+000	7.34303e+000	2.35293e-008	7.61278e-007
gpp124-4	4.18988e+002	4.18987e+002	3.61726e-006	4.58830e-006
gpp250-1	1.54454e+001	1.54443e+001	4.84606e-008	4.36724e-006
qap5	4.36000e+002	4.36000e+002	6.07211e-007	1.38375e-007
qap6	3.81728e+002	3.81404e+002	5.52529e-006	3.72957e-004
qap7	4.25093e+002	4.24790e+002	4.40110e-006	2.70701e-004
qap8	7.57719e+002	7.56865e+002	9.34198e-006	5.42096e-004
qap9	1.41070e+003	1.40988e+003	2.70468e-005	2.72918e-004
hinf1	-2.03091e+000	-2.03281e+000	1.37891e-007	1.31953e-003
hinf4	-2.74729e+002	-2.74768e+002	8.95745e-006	6.55338e-004
hinf7	-3.90502e+002	-3.90827e+002	2.61417e-002	3.16988e-003
hinf10	-1.07428e+002	-1.08863e+002	8.30106e-006	1.14212e-002
hinf11	-6.51630e+001	-6.59384e+001	1.78695e-005	5.21135e-003
hinf12	6.25348e+000	-7.83044e-001	4.61644e-008	1.12113e-001
hinf15	-1.54011e+000	-2.60852e+001	1.85567e-004	2.25385e+000

Table 5: Bounds and residuals for some SDPLIB problems

problem	t	$\overline{t}$	$\underline{t}$	tc
gpp100	3.80	1.75	0.59	0
gpp124-1	3.55	2.23	0.89	0
gpp124-4	3.59	2.39	0.98	0
gpp250-1	10.74	19.78	6.75	0
qap5	0.88	0.52	0.22	0
qap6	1.61	0.89	0.41	0
qap7	2.17	1.88	0.73	0
qap8	3.19	4.00	1.44	0
qap9	6.55	7.80	2.61	0
hinf1	1.42	0.19	0.06	0
hinf4	1.31	0.19	0.06	0
hinf7	0.86	0.19	0.08	-4
hinf10	1.98	0.22	0.11	0
hinf11	1.88	0.28	0.13	0
hinf12	2.84	0.38	0.19	0
hinf15	1.67	0.59	0.44	0

Table 6: Required times for some SDPLIB problems

7 Conclusion

Of particular interest is the property of the new residual that a very rough knowledge of the order of magnitude of the optimal solution provides guaranteed decimal digits of the true optimal value. Hence, an a posteriori validation of numerical optimization algorithms is possible, even for ill-posed problems.

References

[1] G. Alefeld and J. Herzberger. *Introduction to Interval Computations.* Academic Press, New York, 1983.

[2] B. Borchers. SDPLIB 1.2, A Library of Semidefinite Programming Test Problems. *Optimization Methods and Software*, 11(1):683–690, 1999.

[3] C.A. Floudas. *Deterministic Global Optimization - Theory, Methods and Applications*, volume 37 of *Nonconvex Optimization and Its Applications.* Kluwer Academic Publishers, Dordrecht, Boston, London, 2000.

[4] K. Fukuda. *cddlib reference manual, cddlib version 092a*, 2001. Programm available at `http://www.cs.mcgill.ca/~fukuda/software/cdd_home/cdd.html`.

[5] GNU multiple precision arithmetic library (GMP), version 4.1.2., 2003. Code and documentation abailable at `http://swox.com/gmp`.

[6] M. Goemans and F. Rendl. Combinatorial Optimization. In H. Wolkowicz et al., editor, *Handbook of Semidefinite Programming.* Kluwer Academic Publishers, 2000.

[7] G. Gruber, S. Kruk, F. Rendl, and H. Wolkowicz. Presolving for Semidefinite Programs Without Constraint Qualifications. In G. Gruber et al., editor, *Proceedings of HPOPT97, Second Workshop on High Performance Optimization Techniques, Rotterdam, Netherlands*, 1997.

[8] G. Gruber and F. Rendl. Computational experience with ill-posed problems in semidefinite programming. *Computational Optimization and Applications*, 21(2):201–212, 2002.

[9] N.J. Higham. *Accuracy and Stability of Numerical Algorithms.* SIAM Publications, Philadelphia, 2nd edition, 2002.

[10] ILOG CPLEX 7.1, User's manual. *ILOG.* France, 2001.

[11] C. Jansson. A rigorous lower bound for the optimal value of convex optimization problems. *J. Global Optimization*, 28:121–137, 2004.

[12] C. Jansson. Rigorous Lower and Upper Bounds in Linear Programming. *SIAM J. Optim.*, 14(3):914–935, 2004.

[13] C. Jansson and C. Keil. Rigorous Error Bounds for the Optimal Value in Semidefinite Programming. 2005. Submitted, and electronically published `http://www.optimization-online.org/DB_HTML/2005/01/1047.html`.

[14] R.B. Kearfott. *Rigorous Global Search: Continuous Problems.* Kluwer Academic Publisher, Dordrecht, 1996.

[15] T. Lengauer. *Combinatorial algorithms for integrated circuit layout.* John Wiley and Sons, Chichester, 1990.

[16] MATLAB User's Guide, Version 7. The MathWorks Inc., 2004.

[17] G. Mayer. Result verification for eigenvectors and eigenvalues. In J. Herzberger, editor, *Topics in validated computations. Proceedings of the IMACS-GAMM international workshop, Oldenburg, Germany, 30 August - 3 September 1993*, Stud. Comput. Math. 5, pages 209–276, Amsterdam, 1994. Elsevier.

[18] H.D. Mittelmann. An independent benchmarking of SDP and SOCP solvers. *Math. Programming Ser. B*, 95:407–430, 2003.

[19] R.E. Moore. *Methods and Applications of Interval Analysis.* SIAM, Philadelphia, 1979.

[20] NETLIB Linear Programming Library. `http://www.netlib.org/lp`.

[21] A. Neumaier. *Interval Methods for Systems of Equations.* Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1990.

[22] A. Neumaier. *Introduction to Numerical Analysis.* Cambridge University Press, 2001.

[23] A. Neumaier. Grand challenges and scientific standards in interval analysis. *Reliable Computing*, 8(4):313–320, 2002.

[24] A. Neumaier. Complete Search in Continuous Global Optimization and Constraint Satisfaction. *Acta Numerica*, 13:271–369, 2004.

[25] A. Neumaier and O. Shcherbina. Safe bounds in linear and mixed-integer programming. *Mathematical Programming, Ser. A*, 99:283–296, 2004.

[26] F. Ordóñez and R.M. Freund. Computational experience and the explanatory value of condition measures for linear optimization. *SIAM J. Optimization*, 14(2):307–333, 2003.

[27] M.V. Ramana, L. Tunçel, and H. Wolkowicz. Strong duality for semidefinite programming. *SIAM J. on Optimization*, 7(3):641–662, 1997.

[28] S.M. Rump. Validated Solution of Large Linear Systems. In R. Albrecht, G. Alefeld, and H.J. Stetter, editors, *Validation numerics: theory and applications*, volume 9 of *Computing Supplementum*, pages 191–212. Springer, 1993.

[29] S.M. Rump. Verification Methods for Dense and Sparse Systems of Equations. In J. Herzberger, editor, *Topics in Validated Computations — Studies in Computational Mathematics*, pages 63–136, Elsevier, Amsterdam, 1994.

[30] S.M. Rump. INTLAB - Interval Laboratory, a Matlab toolbox for verified computations, Version 3.1, 1998. http://www.ti3.tu-harburg.de/rump/intlab/index.html.

[31] R.H. Tütüncü, K.C. Toh, and M.J. Todd. Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Program.*, 95B(2):189–217, 2003.

[32] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996.