Journal of Global Optimization **14:** 357–364, 1999.

A New Semidefinite Programming Bound for Indefinite Quadratic Forms Over a Simplex

IVO NOWAK
Humboldt-Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany
(e-mail: ivo@mathematik.hu-berlin.de)

(Received 10 December 1997; accepted in revised form 3 September 1998)

Abstract. The paper describes a method for computing a lower bound of the global minimum of an indefinite quadratic form over a simplex. The bound is derived by computing an underestimator of the convex envelope by solving a semidefinite program (SDP). This results in a convex quadratic program (QP). It is shown that the optimal value of the QP is a lower bound of the optimal value of the original problem. Since there exist fast (polynomial time) algorithms for solving SDP's and QP's the bound can be computed in reasonable time. Numerical experiments indicate that the relative error of the bound is about 10 percent for problems up to 20 variables, which is much better than a known SDP bound.

Key words: Global optimization, Nonconvex quadratic programming, Semidefinite programming

1. Introduction

Let $f(x) := x^T F x$ be an indefinite quadratic form, where $F \in \mathbb{R}^{(n+1,n+1)}$ is an indefinite symmetric matrix. We consider the global quadratic optimization problem over the standard simplex:

$$\begin{array}{ll} \text{global minimize} & f(x) \\ \text{subject to} & x \in \Delta_n, \end{array} \tag{1}$$

where the admissible set is the n-dimensional standard simplex

$$\Delta_n := \{x \in \mathbb{R}^{n+1} : x_i \geqslant 0,\ 1 \leqslant i \leqslant n+1,\ e^T x = 1\}$$

and $e \in \mathbb{R}^n$ is the vector of ones. Although the structure of problem (1) is simple, finding a global solution – and even detecting a local solution – is known to be NP-hard (see [4], [6]). Problems of the type (1) occur for example in the search for a maximum (weighted) clique in an undirected graph. Problem (1) is also important for continuous (nonconvex) optimization because it is strongly related to the general quadratic optimization problem (QP) which has numerous applications (see also [1]).

Current solution techniques for QP often employ branch and bound methods. Most approaches for computing lower bounds of indefinite quadratic forms are based on linear programs and the poor quality of these bounds is a major cause of difficulties (see for example [3] for an overview of solution methods for QP).

In this paper we propose a new lower bound for (1) which is computed by solving a semidefinite program (SDP) and a QP. Semidefinite programming attracts currently many researchers since there exist fast (polynomial time) algorithms for solving SDP's and because it has been realized that many optimization problems can be expressed as an SDP (see for example [8, 9, 11]). In Section 2 we describe a lower bounding technique based on SDP, which has been applied recently to many hard combinatorial and quadratically constrained quadratic programs (which include (1)).

In Section 3 we present a new lower bound for (1). Our approach is based on approximating the convex envelope by solving an SDP and then solving the resulting QP to obtain a lower bound. Numerical results on random test problems, which we present in Section 4, indicate that the new bound is much better than the known SDP bound described in Section 2. The paper ends with some conclusions.

2. A known SDP bound

Shor and others proposed a lower bound for quadratically constrained quadratic programs (QQP) which is based on semidefinite programming. Consider the following QQP:

$$\begin{array}{ll} \min & f_0(x) \\ \text{subject to} & f_i(x) \leqslant 0, \quad i = 1, \dots, L, \end{array} \tag{2}$$

where $f_i(x) = x^T A_i x + 2b_i^T x + c_i$, $i = 1, \dots, L$. It is shown (see [10] and [11]) that a lower bound for (2) is given by the following SDP:

$$\begin{array}{ll} \min & \operatorname{tr} XA_0 + 2b_0^T x + c_0 \\ \text{subject to} & \operatorname{tr} XA_i + 2b_i^T x + c_i \leqslant 0, \quad i = 1, \dots, L, \\ & \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0, \end{array} \tag{3}$$

where the variables are $X = X^T \in \mathbb{R}^{(n,n)}$ and $x \in \mathbb{R}^n$. The notation $A \succeq 0$ means that the matrix A is positive semidefinite and tr A denotes the trace of a matrix A. Note that the only difference between (2) and (3) is the replacement of $X = xx^T$ with $X \succeq xx^T$.

Multiplying the constraints $x_i \geqslant 0$ and $x_j \geqslant 0$ and squaring the constraint $e^T x = 1$ in (1) yields the following QQP which is equivalent to (1):

$$\begin{array}{ll} \min & \operatorname{tr} FX \\ \text{subject to} & x_i x_j \geqslant 0, \quad 1 \leqslant i, j \leqslant n+1, \\ & (e^T x)^2 = 1, \quad X = xx^T. \end{array} \tag{4}$$

Relaxing this problem as in (3) leads to the following lower bound for (1):

$$\begin{aligned} b_{\text{old}} := \min \quad & \operatorname{tr} FX \\ \text{subject to} \quad & X \geqslant 0, \quad \operatorname{tr} JX = 1, \\ & X \succeq 0, \end{aligned} \tag{5}$$

where J is the matrix of ones.

3. A new lower bound

In this section we derive a new lower bound for (1). We begin with the following fundamental result:

LEMMA 1. *Let $F, W \in \mathbb{R}^{(n+1,n+1)}$ be symmetric matrices and $W \leqslant F$ (componentwise). Then*

$$\text{global} \min_{x \in \Delta_n} x^T W x \leqslant \text{global} \min_{x \in \Delta_n} x^T F x.$$

Proof. We have

$$x^T F x - x^T W x = x^T (F - W) x \geqslant 0 \quad \text{for } x \in \Delta_n$$

since $x \geqslant 0$ on Δ_n and $F - W \geqslant 0$. This implies the assertion. □

In [7] we used Bézier methods to compute lower bounds of multivariate polynomials over Δ_n. Here, we propose an improved lower bounding technique. The idea is to construct a convex quadratic function $w(x) := x^T W x$ such that $W \leqslant F$ and $w(x)$ approximates $f(x)$ in a special sense. From Lemma 1 it follows that the minimum of $w(x)$ over Δ_n is a lower bound for problem (1). The condition that a quadratic form is a convex over Δ_n can be formulated as a matrix inequality:

LEMMA 2. *Let $\Phi : \mathbb{R}^{(n+1,n+1)} \to \mathbb{R}^{(n,n)}$ be the linear map defined by $\Phi(G)_{ij} := G_{ij} + G_{n+1,n+1} - G_{n+1,i} - G_{n+1,j}$ $(1 \leqslant i, j \leqslant n)$, where $G \in \mathbb{R}^{(n+1,n+1)}$ is a symmetric matrix. A quadratic form $x^T G x$ is convex on Δ_n if and only if $\Phi(G) \succeq 0$.*

Proof. The matrix $\Phi(G)$ is the Hessian of the quadratic form $x^T G x$ with respect to the coordinates $e_i - e_{n+1}$ $(1 \leqslant i \leqslant n)$, i.e. $\Phi_{ij}(G) = \partial_{e_i - e_{n+1}} \partial_{e_j - e_{n+1}} (x^T G x)$, $1 \leqslant i, j \leqslant n$. □

Consider now the following optimization problem:

$$\begin{aligned} W := \operatorname{argmin} \quad & \operatorname{tr} J(F - G) \\ \text{subject to} \quad & G \leqslant F, \\ & \operatorname{diag} G = \operatorname{diag} F, \quad \Phi(G) \succeq 0, \end{aligned} \tag{6}$$

where Φ is defined as in Lemma 2. The new lower bound is defined by

$$b_{\text{new}} := \min_{x \in \Delta_n} x^T W x. \tag{7}$$

Before we prove some properties of the bound b_{new} we introduce the following notation. Let $A \in \mathbb{R}^{(n+1,n+1)}$ be a given matrix and denote by $A^l \in \mathbb{R}^{(n+1,n+1)}$ and by $A^c \in \mathbb{R}^{(n+1,n+1)}$ the matrices defined by

$$A^l_{ij} := \frac{1}{2}(A_{ii} + A_{jj}), \qquad A^c_{ij} := \frac{1}{2}(A_{ii} - 2A_{ij} + A_{jj}), \quad 1 \leqslant i, j \leqslant n+1. \tag{8}$$

We have obviously $A = A^l - A^c$. The quadratic form $x^T A^l x$ is linear over Δ_n and the entries of the matrix $2 \cdot A^c$ are the second-order derivatives of the quadratic form $x^T Ax$ along the edges of Δ_n, i.e. $\partial^2_{e_i - e_j} x^t Ax = 2 \cdot A^c_{ij}$ $(1 \leqslant i, j \leqslant n+1)$.

PROPOSITION 1.

(i) Problem (6) is well defined.

(ii) b_{new} is a lower bound of the global minimum of (1).

(iii) If $f(x)$ is concave on the edges of the simplex Δ_n then $W = F^l$ and b_{new} is exact.

(iv) Let

$$E^c := \{ij : F^c_{ij} > 0, \quad 1 \leqslant i < j \leqslant n+1\} \tag{9}$$

be the edge set of edges of Δ_n where $f(x)$ is strictly convex. Problem

$$\begin{aligned} W = \text{argmin} \quad & \text{tr}\, J(F - G) \\ \text{subject to} \quad & G_{ij} \leqslant F_{ij}, \quad ij \in E^c, \\ & \text{diag}\, G = \text{diag}\, F, \quad \Phi(G) \succeq 0 \end{aligned} \tag{10}$$

is equivalent to (6).

Proof.

(i) Let $q_{\min} := \min\{-F^c_{ij} : 1 \leqslant i, j \leqslant n+1\}$ and let $\hat{W} \in \mathbb{R}^{(n+1,n+1)}$ be the matrix defined by

$$\hat{W}_{ij} := F^l_{ij} + \begin{cases} q_{\min} & \text{if } i \neq j \\ 0 & \text{else} \end{cases}, \quad 1 \leqslant i, j \leqslant n+1.$$

Since $F^c_{ii} = 0$ it follows $q_{\min} \leqslant 0$ and therefore $\hat{W} \leqslant F$. Note that $\Phi(\hat{W}) = -q_{\min}(I + J)$ where I is the $(n+1) \times (n+1)$ identity matrix. Hence convexity of the quadratic form $x^T \hat{W} x$ over Δ_n follows immediately via Lemma 2 and $\hat{W}$ satisfies the constraints of problem (6). Therefore, problem (6) is well defined.

(ii) Since $W \leqslant F$ it follows from Lemma 1 that b_{new} is a lower bound of the optimal value of problem (1).

(iii) Let $G \in \mathbb{R}^{(n+1,n+1)}$ be a matrix which is in the feasible set of problem (6). Since $\operatorname{diag} F = \operatorname{diag} G$ we have $F^l = G^l$ and hence $F - G = G^c - F^c$ implying $\operatorname{tr} J(F - G) = \operatorname{tr} J(G^c - F^c) = -\operatorname{tr} JF^c + \sum_{1 \leqslant i,j \leqslant n+1} |G^c_{ij}|$. The last equality follows from $\Phi(G) \succeq 0$, which implies $G^c_{ij} \geqslant 0$ $(1 \leqslant i, j \leqslant n+1)$. Since $f(x)$ is concave on the edges of Δ_n we have $F^c_{ij} \leqslant 0$ $(1 \leqslant i, j \leqslant n+1)$ which makes the inequality $G \leqslant F$ superfluous. Therefore, problem (6) is equivalent to

$$W = F^l - W^c, \quad W^c = \operatorname{argmin} \sum_{1 \leqslant i,j \leqslant n+1} |G^c_{ij}| \quad \text{subject to} \quad \Phi(F^l - G^c) \succeq 0,$$

which has the unique solution $W^c = 0$. This implies that $W = F^l$ is the solution of (6) and the objective function in (7) is linear. Hence a solution of (7) is attained at a vertex of Δ_n, which proves that b_{new} is exact.

(iv) The number of inequalities in (6) can be reduced due to the following observation. Let $g(x) := x^T G x$ be a convex quadratic form on Δ_n where $\operatorname{diag} G = \operatorname{diag} F$. Since $F^c_{ij} \leqslant 0$ for all $ij \notin E^c$ and $\partial^2_{e_i - e_j} g(x) = 2G^c_{ij} \geqslant 0$ for $1 \leqslant i < j \leqslant n+1$ it follows

$$F_{ij} - G_{ij} = G^c_{ij} - F^c_{ij} \geqslant 0 \quad \text{for all } ij \notin E^c$$

which proves the statement. □

Note that the number of variables in the semidefinite program (10) can be reduced by eliminating the constraints $\operatorname{diag} G = \operatorname{diag} F$. Define the linear map $\Psi : \mathbb{R}^{(n,n)} \to \{U \in \mathbb{R}^{(n+1,n+1)} : \operatorname{diag} U = 0\}$ by

$$\Psi(X)_{i,n+1} = \Psi(X)_{n+1,i} = -\frac{1}{2} X_{ii}, \quad 1 \leqslant i \leqslant n$$
$$\Psi(X)_{ij} = X_{ij} - \frac{1}{2}(X_{ii} + X_{jj}), \quad 1 \leqslant i, j \leqslant n$$
$$\Psi(X)_{n+1,n+1} = 0,$$

where $X \in \mathbb{R}^{(n,n)}$. We have $\Phi(\Psi(X) + F^l) = X$ and $\operatorname{diag}(\Psi(X) + F^l) = \operatorname{diag} F$. Substituting G by $\Psi(X) + F^l$ in problem (10) we obtain the following equivalent semidefinite program with the variable $X \in \mathbb{R}^{(n,n)}$:

$$\begin{aligned} W = \Psi(X^*) + F^l, \qquad X^* = \operatorname{argmin} \quad & \operatorname{tr} J(-F^c - \Psi(X)) \\ \text{subject to} \quad & \Psi(X)_{ij} \leqslant -F^c_{ij}, \quad ij \in E^c, \\ & X \succeq 0. \end{aligned} \tag{11}$$

4. Numerical results

We made numerical experiments on random test examples to compare the bounds b_{old} and b_{new}. In order for the reader to be able to reproduce these examples we include the source code, written in C++, that we used for generating random test examples of the type (1). We used the following procedures to compute the entries of the matrix F:

```
void rand_qps(int n,double dens,double dvert,int &seed,rmatrix &Fc,
   rmatrix &Fl,rmatrix &F)
{
   int i,j,l;
   double r=(4.*double(seed)+1.)/16384./16384.;
   Fc=0.0; seed++;
   for(i=0;i<n;i++)
      for(j=i+1;j<=n;j++)
      if(random(r,0,1)<dens)
        Fc(i,j)=Fc(j,i)=random(r,0.,10.);
      else
        Fc(i,j)=Fc(j,i)=random(r,-10.,0.);
   for(i=0;i<=n;i++) Fl(i,i)=random(r,0,dvert);
   for(i=0;i<n;i++)
      for(j=i+1;j<=n;j++) Fl(i,j)=Fl(j,i)=0.5*(Fl(i,i)+Fl(j,j));
   for(i=0;i<=n;i++)
      for(j=i;j<=n;j++) F(i,j)=F(j,i)=Fl(i,j)-Fc(i,j);
}

double random(double &r,double a,double b)
{
   r=fmod(r*41475557.,1.);
   return(r*(b-a)+a);
}
```

The parameter `seed` is initialized by one. The SDP's (5) and (11) were solved using the implementation of Borchers [2] of the interior point algorithm of [5]. The QP (7) was solved by a descent method. In order to compare the bounds we computed a local minimum of $f(x)$ over Δ_n, which we denote by f_{est}, by a descent method starting from the point

$$x^{(0)} := \arg\min_{x\in\Delta_n} x^T W x,$$

where W is the solution of the SDP (10). The table below displays the numerical results. We made always 50 runs and averaged the quantities. The parameters n, dens, d_{vert} denote the problem size, the density of the edge set E^c and the random deviation of $f(x)$ at the vertices from zero (see source code). The percentage

Table 1.

n	dens	d_{vert}	e_{new}	e_{old}	opt$_{\text{new}}$	opt$_{\text{old}}$	time$_{\text{new}}$	time$_{\text{old}}$
5	0.25	0	0.630642	127.301	86	0	0.0176	0.0282
5	0.25	2	2.20818	117.593	80	2	0.0174	0.029
5	0.5	0	5.98523	185.513	80	0	0.0224	0.0288
5	0.5	2	1.17532	159.883	76	0	0.0208	0.0286
5	0.75	0	1.66293	260.102	58	0	0.0264	0.0288
5	0.75	2	1.3963	233.698	62	0	0.0268	0.0278
10	0.25	0	15.7269	188.606	14	0	0.1212	0.2098
10	0.25	2	7.30263	163.564	28	0	0.1192	0.2106
10	0.5	0	7.70254	279.534	10	0	0.1636	0.2202
10	0.5	2	7.74051	253.067	12	0	0.1654	0.2202
10	0.75	0	3.4551	458.674	16	0	0.217	0.2174
10	0.75	2	3.65035	404.81	20	0	0.2184	0.223
20	0.25	0	16.5635	269.016	4	0	1.1584	3.1828
20	0.25	2	17.3237	239.572	2	0	1.151	3.101
20	0.5	0	8.85466	450.526	0	0	1.8094	3.3268
20	0.5	2	9.21408	393.978	0	0	1.8108	3.2972
20	0.75	0	5.59175	833.251	0	0	2.712	2.9648
20	0.75	2	8.08468	693.587	0	0	2.7348	3.22

relative error of the bound b_{new} and b_{old} is denoted by $e_{\text{new}} := 100 \cdot \frac{b_{\text{new}} - f_{\text{est}}}{f_{\text{est}}}$ and $e_{\text{old}} := 100 \cdot \frac{b_{\text{old}} - f_{\text{est}}}{f_{\text{est}}}$ respectively. The percentage averaged number of cases where the absolute error of b_{new} and b_{old} does not exceed 10^{-4} is denoted by opt_{new} and opt_{old}, respectively. The CPU time in seconds for computing the bounds b_{new} and b_{old} is denoted by time$_{\text{new}}$ and time$_{\text{old}}$ respectively. The computations were performed on a HP J 280 workstation.

The table shows that the relative error of b_{new} is between 0.6 and 18 percent which is much better than the relative error of b_{old} (by a factor between 10 and 200). The bound b_{new} is even sometimes exact for small size instances. In general, problems are bounded more accurately where the density of the edge set E^c is small. It seems that the parameter d_{vert} does not influence the results very much. The CPU time for computing b_{new} is slightly smaller than for computing b_{old}.

5. Conclusion

We presented a new technique for bounding indefinite quadratic forms over a simplex. Numerical experiments on random test examples show that the bound b_{new} is quite sharp and improves the known SDP bound b_{old}. The computational cost for b_{new} is not much higher than for b_{old} since it is only necessary to solve an additional convex quadratic program, which can be done in polynomial time. A branch and bound algorithm for solving (1) using this new bounding technique is in preparation.

Acknowledgments

We thank two anonymous referees for carefully reading the paper and for their valuable suggestions.

References

1. Bomze, I. (1998), On standard quadratic optimization problems, *Journal of Global Optimization* 13(4): 369–387.
2. Borchers, B. (1997), CSDP, a C library for semidefinite programming, manuscript, http://www.nmt.edu/∼borchers/csdp.html.
3. Floudas, C.A. and Visweswaran, V. (1995), Quadratic optimization, in *Handbook of Global Optimization* (R. Horst and P. Pardalos, eds.), Kluwer Academic Publishers, Dordrecht, pp. 217–269.
4. Garey, M.R. and Johnson, D.S. (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, New York.
5. Helmberg, C., Rendl, F., Vanderbei, R.J. and Wolkowicz, H. (1996), An interior-point method for semidefinite programming, *SIAM J. Opt.* 6(2): 342–361.
6. Horst, R., Pardalos, P. and Thoai, N. (1995), *Introduction to Global Optimization*, Kluwer Academic Publishers, Dordrecht.
7. Nowak, I. (1996), A branch-and-bound algorithm for computing the global minimum of polynomials by Bernstein–Bézier patches on simplices, *Preprint M-09*, BTU Cottbus.
8. Pardalos, M.P. and Wolkowicz, H. (eds.), (1998), Topics in Semidefinite and Interior-Point Methods, *Fields Institute Communications Series*, Vol. 18, American Mathematical Society.
9. Ramana, M. and Pardalos, P.M. (1996), Semidefinite programming, in *Interior Point Methods of Mathematical Programming*, T. Terlaky (ed.), Kluwer Academic Publishers, Dordrecht, 369–398.
10. Shor, N.Z. (1987), Quadratic optimization problems, *Soviet J. Circ. Syst. Sci.* 25(6): 1–11.
11. Vandenberghe, L. and Boyd, S. (1996), Semidefinite programming, *SIAM Review* 38: 49–95.