ON LOWER BOUNDS FOR A CLASS OF QUADRATIC 0, 1 PROGRAMS

Arjang A. ASSAD

College of Business and Management, University of Maryland, College Park, MD 20742, USA

Weixuan XU

Institute of Applied Mathematics, Academia Sinica, Beijing, People's Republic of China

Received March 1985
Revised July 1985

We present a new method of obtaining lower bounds for a class of quadratic 0, 1 programs that includes the quadratic assignment problem. The method generates a monotonic sequence of lower bounds and may be interpreted as a Lagrangean dual ascent procedure. We report on a computational comparison of our bounds with earlier work in [2] based on subgradient techniques.

integer quadratic programming * quadratic assignment problem * lower bounding techniques

1. Introduction

Consider the quadratic 0, 1 program

$$\begin{aligned}\text{(QP):}\quad & \text{Min} \quad z(x) = x^T A x = \sum_{i=1}^{m} \sum_{j=1}^{m} a_{ij} x_i x_j, \\ & \text{s.t.} \quad x \in S \subseteq B_2^m,\end{aligned} \tag{1}$$

where the feasible set S is a subset of B_2^m, the space of all m-vectors with components in $\{0, 1\}$. For notational convenience, we denote the diagonal elements of the matrix $A = (a_{ij})$ by $b_i = a_{ii}$. Since $x_i^2 = x_i$, we can then rewrite the objective function as

$$z(x) = \sum_{i=1}^{m} \sum_{j \neq i}^{m} a_{ij} x_i x_j + \sum_{i=1}^{m} b_i x_i. \tag{2}$$

In this paper, we focus our attention on a class of 0, 1 programs (1) satisfying the following assumptions:

Assumption 1. $\sum_{i=1}^{m} x_i = K$ for all $x \in S$. (3)

Assumption 2. There is an efficient algorithm to solve the linear problem $\text{Min}\{cx \mid x \in S\}$.

Our goal is to develop a lower-bounding technique for (QP) under these two assumptions. In Section 4, we shall see that the celebrated Quadratic Assignment Problem (QAP) [1] satisfies the preceding assumptions. This allows us to generate lower bounds for the QAP which may be compared to bounds derived by subgradient techniques in [2].

To mention another instance of our class of quadratic problems, let $x_i = 1$ or 0 depending on whether arc i of a network G is selected or not. Let S be the set of all vectors x that correspond to a tree in G. Then (1) becomes the *quadratic minimum spanning tree* problem. Note that (3) holds since $\sum_{i=1}^{m} x_i = n - 1$ for any tree $x \in S$, where n = number of nodes in G. Moreover, the linear-objective problem over S is precisely the minimum spanning tree problem. In this paper, we shall focus on the QAP; the results of applying the algorithm to the tree problem can be found in Xu [6].

2. Development of the lower bound

Introduce a vector of parameters $u =$

$(u_1, \ldots, u_m)$ and let

$$a_{ij}(u) = a_{ij} + u_j, \qquad j \neq i, \qquad b_i(u) = b_i - (K-1)u_i. \tag{4}$$

Let $A(u)$ denote the matrix with these components (and $b_i(u)$'s on the diagonal). Direct computation and the use of (3) allows us to conclude the following:

Lemma 1. *Under the substitution given in* (4), *the value of* $z(x)$ *in* (2) *remains unchanged for any* $x \in S$, *that is,* $x^T A(u) x = z(x)$.

Define $S_i = \{x \in S \mid x_i = 1\}$, so that any vector in S_i has its ith component fixed at 1, and let

$$f_i(u) = \operatorname{Min}\left\{ b_i(u) + \sum_{j \neq i} a_{ij}(u) x_j \mid x \in S_i \right\}, \tag{5}$$

and

$$PB(u) = \operatorname{Min}\left\{ \sum_{i=1}^{m} f_i(u) x_i \mid x \in S \right\}. \tag{6}$$

By the definition of $f_i(u)$, we have

$$f_i(u) x_i \leq \left\{ b_i(u) + \sum_{j \neq i} a_{ij}(u) x_j \right\} x_i,$$

$i = 1, \ldots, m.$

If we sum these inequalities over i, the right-hand side will be equal to $x^T A(u) x$. Lemma 1 may then be used to obtain the following.

Proposition 1. $PB(u)$ *is a lower bound for the optimal value of* (1). *Moreover, it is a piecewise linear concave function of* u.

Given that $PB(u)$ provides a lower bound for any value of u, we are interested in the largest lower bound generated in this manner, that is,

$$PB^* = \operatorname*{Max}_u PB(u). \tag{7}$$

In view of Proposition 1, one can apply standard subgradient techniques to find PB^*. We approach this problem in a different way. We first provide a good characterization of the optimal solution to (7) and use this as the basis of an algorithm proposed in Section 3.

Define

$$\underline{f}(u) = \operatorname*{Min}_i \{ f_i(u) \}, \quad \bar{f}(u) = \operatorname*{Max}_i \{ f_i(u) \}, \tag{8}$$

and let

$$\phi(u) = \bar{f}(u) - \underline{f}(u). \tag{9}$$

Theorem 1. *For any fixed* $\hat{u}$, *we have*

$$PB(u) \leq PB(\hat{u}) + K\phi(\hat{u}) \quad \textit{for all } u. \tag{10}$$

The characterization of solutions to (7) is an immediate corollary of this theorem.

Corollary. *Any* u^* *with* $\phi(u^*) = 0$ *maximizes* $PB(u)$ *over all* u. *Moreover, if* $\hat{u}$ *satisfies* $\phi(\hat{u}) \leq \epsilon/K$ *then it is an* ϵ*-optimal solution to* (7), *i.e.,*

$$PB^* - \epsilon \leq PB(\hat{u}).$$

Proof of Theorem 1. We need to relate $PB(u)$ and $PB(\hat{u})$. Let $d = u - \hat{u}$. By (4) we have $a_{ij}(u) = a_{ij}(\hat{u}) + d_j$, and $b_i(u) = b_i(\hat{u}) - (K-1)d_i$. We can use these substitutions in (5) to express $f_i(u)$ in terms of $\hat{u}$ and d.

$$\begin{aligned} f_i(u) &= \operatorname{Min}\Big\{ b_i(\hat{u}) - (K-1)d_i \\ &\qquad + \sum_{j \neq i} \big(a_{ij}(\hat{u}) + d_j\big) x_j \mid x \in S_i \Big\} \\ &= -Kd_i + \operatorname{Min}\Big\{ b_i(\hat{u}) + \sum_{j \neq i} a_{ij}(\hat{u}) x_j \\ &\qquad + \sum_{j=1}^{m} d_j x_j \mid x \in S_i \Big\} \\ &\leq -Kd_i + f_i(\hat{u}) + D, \end{aligned}$$

where $D = \operatorname{Max}\{\sum_{j=1} d_j x_j \mid x \in S\}$. (The fact that $x_i = 1$ for $x \in S_i$ was used here.) Multiplication of the last inequality by x_i and summation over i yields

$$\sum_{i=1}^{m} f_i(u) x_i \leq \sum_{i=1}^{m} f_i(\hat{u}) x_i - \sum_{i=1}^{m} K d_i x_i + KD,$$

since $\sum x_i = K$ for any $x \in S$.

Now take any $\bar{x} \in S$ satisfying $D = \sum d_j \bar{x}_j$ and let $x = \bar{x}$ in the preceding inequality. The last two terms in the resulting right-hand side cancel to yield

$$\sum_{i=1}^{m} f_i(u) \bar{x}_i \leq \sum_{i=1}^{m} f_i(\hat{u}) \bar{x}_i.$$

This relation together with (6) and (8) allows us to conclude that

$$PB(u) \leq K\bar{f}(\hat{u}).$$

Moreover, by (3) and (6) we have
$Kf(\hat{u}) \leq PB(\hat{u})$.

Adding the last two inequalities results in (10) as required. □

3. The lower bound algorithm

We now state an algorithm for computing lower bounds on (1). The algorithm accepts a prespecified tolerance δ as input.

Step 0. Set $u^0 = 0$, $t = 0$.

Step 1. Solve the subproblems

$$\mathrm{SP}_i(u^t):\ f_i(u^t) = \mathrm{Min}\Big\{ b_i(u^t) + \sum_{j \neq i} a_{ij}(u^t) x_j \,|\, x \in S_i \Big\},$$

for i = 1,...,m. (11)

Step 2. Compute $\underline{f}(u^t)$, $\bar{f}(u^t)$, and $\phi(u^t)$ as in (8) and (9).

If $\phi(u^t) \leq \delta$, stop;
otherwise, solve for

$$PB(u^t) = \mathrm{Min}\Big\{ \sum_{i=1}^{m} f_i(u^t) x_i \,|\, x \in S \Big\}.$$

Step 3. Update the u-vector

$$u_i^{t+1} = u_i^t + \frac{1}{K} f_i(u^t), \qquad i = 1,\dots,m. \tag{12}$$

Put $t = t + 1$ and return to Step 1.

The results of Section 2 ensure that each u^t generates a lower bound $PB(u^t)$ on the optimal value of (1). In (12), the vector u is adjusted to provide a new lower bound. The advantage of this algorithm is that the sequence of lower bounds obtained in this way is monotonically non-decreasing in t. This is shown in the next result for which a slight change of notation is convenient.

The statement of the algorithm contains a number of quantities expressed as functions of u^t. In what follows, we suppress the argument u^t in favor of an indexing by t. Thus $PB^t = PB(u^t)$ and $\phi^t = \phi(u^t)$. By virtue of (4), the updating formula (12) translates into the following adjustment of the problem parameters (a_{ij}) and (b_i) in the new notation:

$$a_{ij}^{t+1} = a_{ij}^t + \frac{1}{K} f_j^t \quad \text{for } j \neq i,$$
$$b_i^{t+1} = b_i^t - \Big(1 - \frac{1}{K}\Big) f_i^t. \tag{13}$$

Theorem 2. *The sequence $\{PB^t\}$ generated by the preceding algorithm is non-decreasing in t. Moreover, $\{\phi^t\}$ forms a non-increasing sequence.*

This theorem states that $\phi^{t+1} \leq \phi^t$ and $PB^t \leq PB^{t+1}$ for all $t \geq 0$. Note that by the corollary to Theorem 1, if $\{\phi^t\}$ converges to 0, then $\{PB^t\}$ equals PB^* in the limit. We also observe that as ϕ^t decreases, the gap between the smallest and largest components of the vector $f^t = (f_1^t, \dots, f_m^t)$ becomes smaller. One visual interpretation is that the graph of f_i^t versus i becomes more level as t increases. For this reason, we have called the preceding procedure the *Levelling algorithm.*

As mentioned later in this paper, the vector u^t can be interpreted as a Lagrange multiplier. Relation (12) then shows that the Levelling algorithm is a 'multiplier adjustment method' for computing lower bounds.

Proof of Theorem 2. Fix i and let x^t and x^{t+1} be solutions to (11) for u^t and u^{t+1}. Define the function

$$v^t(y) = b_i^t + \sum_{j \neq i} a_{ij}^t y_j \quad \text{for all } y \text{ in } S_i.$$

For any $y \in S_i$, since $y_i = 1$, we can use (13) to write

$$v^{t+1}(y) = v^t(y) - f_i^t + \frac{1}{K} \sum_{j=1}^{m} f_j^t y_j. \tag{14}$$

Now let $y = x^t$ in (14) and note that $v^t(x^t) = f_i^t$ so that

$$f_i^{t+1} \leq v^{t+1}(x^t) = \frac{1}{K} \sum_j f_j^t x_j^t.$$

By substituting x^{t+1} for y in (14), and noting that $v^t(x^{t+1}) \geq f_i^t$, we obtain

$$\frac{1}{K} \sum f_j^t x_j^{t+1} \leq f_i^{t+1} = v^{t+1}(x^{t+1}).$$

Combining these inequalities yields

$$\frac{1}{K} \sum_{j=1}^{m} f_j^t x_j^{t+1} \leq f_i^{t+1} \leq \frac{1}{K} \sum_{j=1}^{m} f_j^t x_j^t. \tag{15}$$

This last relation implies that $\underline{f}^t \leq f_i^{t+1} \leq \bar{f}^t$, which in turn shows that $\phi^{t+1} \leq \phi^t$.

To show that $PB^t \leq PB^{t+1}$, we can use the optimality of PB^t and the second inequality of (15) to write

$$PB^t \leq \sum_{i=1}^{m} f_i^t x_i^{t+1} \leq K f_i^{t+1}.$$

Thus we have $PB^t/K \le f_i^{t+1}$ for all i. Combining this with (3), we conclude that $PB^t \le \sum f_i^{t+1} x_i$ for any $x \in S$. We must therefore have $PB^t \le PB^{t+1}$. □

4. Application to the QAP

Let $\mathscr{A}$ be the set of all $n \times n$ 0, 1 matrices $x = (x_{ip})$ satisfying the assignment constraints

$$\sum_i x_{ip} = 1 \quad \forall p \quad \text{and} \quad \sum_p x_{ip} = 1 \quad \forall i.$$

Using the notation of Frieze and Yadegar [2], the Quadratic Assignment Problem (QAP) may be formulated as

$$\text{(QAP)}: \text{Min} \sum_{i,\,p} \sum_{(j,\,q) \neq (i,\,p)} a_{ipjq} x_{ip} x_{jq} + \sum_{i,\,p} b_{ip} x_{ip}. \tag{16}$$

The sum of the variables x_{ip} over all i and p clearly equals n for any x in the feasible set $\mathscr{A}$. Moreover, the linear problem $\text{Min}\{cx \mid x \in \mathscr{A}\}$ is simply an assignment problem. Assumptions 1 and 2 of Section 1 are thus satisfied, showing that the Levelling algorithm can be applied to (QAP). Given parameters (u_{ip}), (4) may be written as

$$\begin{aligned} a_{ipjq}(u) &= a_{ipjq} + u_{jq}, \\ b_{ip}(u) &= b_{ip} - (n-1) u_{ip}. \end{aligned} \tag{17}$$

With these transformations, it is easy to rephrase the Levelling algorithm. Note that in Step 1, n linear assignment problems are solved to evaluate the f_{ip}'s, and one more to obtain PB in Step 2.

In [2], lower bounds for the QAP are derived from a Lagrangean relaxation of one integer programming formulation of (QAP). The relation of these bounds to our bound PB^* is discussed in the appendix. Briefly stated, we show that $PB(u)$ corresponds to the Lagrangean problem for another integer programming formulation of (QAP) that aggregates certain constraints of the one in [2].

Let FYB denote the maximum of the Lagrangean function of [2] over all choices of multipliers. The relation between our bounds and FYB may be summarized as follows.

(i) The Lagrangean problem of [2] involves $2n^3$ multipliers. The restriction of these to a certain n^2-dimensional subspace results in our Lagrangean with value $PB(u)$. This implies that $PB^* \le FYB$.
(ii) The maximization of the Lagrangean required to evaluate FYB is computationally very burdensome. We can use the Levelling algorithm to perform the dual ascent for our multipliers with significantly less effort.
(iii) The Levelling algorithm produces a monotone sequence of bounds (by Theorem 2), an advantage not shared by subgradient techniques proposed by [2].
(iv) Both approaches produce the Gilmore–Lawler bound (see [1] or [4]) when all multipliers are set equal to zero.

Table 1 contains a computational comparison of the bounds of [2] with those produced by the Levelling algorithm. The problems are taken from [3], [4] and [5]. They are identified by the author's name, followed by the size of the problem (n). The Gilmore–Lawler bound and the bound resulting

Table 1
Comparison of lower bounds *GLB*, *BYB* and *LVB*

Problem	(n)	*GLB*	*FYB1*	*FYB2*	*LVB*	*OPT*	*NIT*	*N1*	*N2*
Gavett	4	792	806	806	804	806	11	2	1
Lawler	7	499	559	511	541	559	7	23	50
Nugent	5	50	50	50	50	50	1	1	1
Nugent	6	82	86	82	82	86	1	166	1
Nugent	7	137	148	138	139	148	3	376	30
Nugent	8	186	194	187	188	214	3	411	20
Nugent	12	493	–	494	495	578	3	–	350
Nugent	15	963	–	963	968	1150	4	–	1
Nugent	20	2057	–	2057	2071	2570	4	–	1

from the Levelling algorithm are denoted by *GLB* and *LVB*.

Since the maximization of the Lagrangean is performed only approximately in [2] by using subgradient techniques, one does not obtain *FYB* in general but approximations to it denoted by *FYB1* and *FYB2*. The numerical results for *FYB1* and *FYB2* are taken from [2]. *N1T, N1,* and *N2* give the number of iterations performed to arrive at *LVB, FYB1,* and *FYB2*. *OPT* denotes the optimal value.

Table 1 shows that *FYB1* provides the largest bounds. However, the number of iterations it requires is very large (recall that each iteration involves solving n^2+1 assignment problems). Due to this computational burden, this bound could not be used for problems of dimension $n>9$ in [2]. The bound *FYB2* is also cumbersome to compute and fails to improve upon *GLB* for the largest two problems. The Levelling algorithms outperforms *FYB2* and takes only a few iterations.

5. Conclusions

The results of this paper may be interpreted as providing a multiplier adjustment algorithm, with only moderate computational requirements, for solving a Lagrangean dual for a class of quadratic 0, 1 programs.

In addition to the QAP, we have successfully applied this algorithm to the quadratic minimum spanning tree problem [6]. In our computational tests, we found that the Levelling algorithm converges rapidly (in the sense that ϕ' approaches 0 very quickly). We are currently investigating a convergence proof, but feel that the Levelling technique has already proved to be a viable alternative to approaches using subgradient techniques similar to the work in [2].

Appendix

This appendix relates the lower bounds of this paper to those derived from Lagrangean relaxation for the QAP as in [2]. Similar relations can also be derived for the general problem in (1) when (3) holds. We start with the following integer programming formulation of (QAP):

$$\text{(IP):}\quad \text{Min}\ \sum_{i,p}\sum_{(j,q)} a_{ipjq}y_{ipjq} + \sum_{i,p} b_{ip}x_{ip}, \tag{18}$$

$$\text{s.t.}\quad x\in\mathscr{A}, \tag{19}$$

$$\sum_{i,p} y_{ipjq} = nx_{jq} \quad \text{for all } j,\ q, \tag{20}$$

$$\sum_{j} y_{ipjq} = \sum_{q} y_{ipjq} = x_{ip} \quad \text{for all } i,\ p, \tag{21a}$$

$$y_{ipip} = x_{ip} \quad \text{for all } i,\ p, \tag{21b}$$

$$0 \le y_{ipjq} \le 1 \quad \text{for all } i,\ p,\ j,\ q. \tag{21c}$$

One can show that any feasible solution to (IP) satisfies $y_{ipjq}=x_{ip}x_{jq}$ for all i, p, j, q, and thus establish the equivalence of (IP) and (QAP).

Now if the constraints in (20) are relaxed and brought into the objective with multipliers u_{jq}, we obtain

$$L(u) = \quad \text{Min}\ \sum_{i,p}\sum_{(j,q)} a_{ipjq}(u)y_{ipjq} + \sum_{i,p} b_{ip}(u)x_{ip}, \tag{22}$$

$$\text{s.t.}\quad (19),\ (21\text{a}–21\text{c}),$$

where the cost coefficients in (22) are defined by (17). The coefficient $n-1$ in $b_{ip}(u)$ deserves some comment. Recall that the double sum in (18) and (22) is over all $(j,q)\neq(i,p)$. Thus the term $u_{ip}y_{ipip}$ arising in the Lagrangean function is written as $u_{ip}x_{ip}$ by (21b) and incorporated into $b_{ip}(u)x_{ip}$. Now the Lagrangean problem $L(u)$ decomposes over (i,p) once the x_{ip}'s are fixed. One can therefore re-express it as

$$\text{Min}\left\{\sum_{i,p} f_{ip}(u)x_{ip} \mid x\in\mathscr{A}\right\}, \quad \text{where}$$

$$f_{ip}(u) = \text{Min}\left\{ b_{ip}(u) + \sum a_{ipjq}(u)\theta_{jq} \mid \theta = (\theta_{jq}) \in \mathscr{A}\right\}$$

with the sum in the minimand running over all $(j,q)\neq(i,p)$. Comparing these with the equations of Section 2 shows that

$$L(u) = PB(u) \quad \text{for all } u. \tag{23}$$

Frieze and Yadegar [2] use a different integer programming formulation for the QAP in which constraint (20) is replaced with

$$\sum_i y_{ipjq} = x_{jq} \quad \text{for all } p,\ j,\ q, \tag{24a}$$

$$\sum_p y_{ipjq} = x_{jq} \quad \text{for all } i,\ j,\ q. \tag{24b}$$

Relaxing these constraints with multipliers v_{pjq} and w_{ijq} results in the Lagrangean problem

$$\begin{aligned} \bar{L}(v, w) = \quad \text{Min} \quad & \sum_{i,p} \sum_{(j,q)} \bar{a}_{ipjq} y_{ipjq} \\ & + \sum_{i,p} \bar{b}_{ip} x_{ip}, \\ \text{s.t.} \quad & (19), (21\text{a}–21\text{c}), \end{aligned} \tag{25}$$

where

$$\begin{aligned} \bar{a}_{ipjq} &= a_{ipjq} + v_{pjq} + w_{ijq}, \\ \bar{b}_{ip} &= b_{ip} - \sum_{k \neq p} v_{kip} - \sum_{l \neq i} w_{lip}. \end{aligned} \tag{26}$$

Now if we let $v_{pjq} = \bar{v}_{jq}$ for all p, $w_{ijq} = \bar{w}_{jq}$ for all i, and stipulate that $v_{jq} + w_{jq} = u_{jq}$, we see that $\bar{L}(v, w) = L(u)$ for this choice of v and w.

This shows that the Lagrangean problem in (22) is a restriction of the one in (25) that also appears in [2]. Now by our notation

$$\operatorname*{Max}_{u} L(u) = PB^* \quad \text{and} \quad \operatorname*{Max}_{v,w} L(v, w) = FYB.$$

We can therefore conclude that

$PB^* \leq FYB$.

References

[1] R. Burkard, "Quadratic assignment problems", *European Journal of Operational Research* **15**, 283–289 (1984).

[2] A. Frieze and J. Yadegar, "On the quadratic assignment problem", *Discrete Applied Mathematics* **5**, 89–98 (1983).

[3] J. Gavett and N. Plyter, "The optimal assignment of facilities to locations by branch and bound", *Operations Res.* **14**, 210–232 (1966).

[4] E. Lawler, "The quadratic assignment problem", *Management Science* **9**, 586–599 (1963).

[5] C. Nugent, T. Vollmann and J. Ruml, "An experimental comparison of techniques for the assignment of facilities to locations", *Operations Res.* **16**, 150–173 (1968).

[6] W. Xu, "The Quadratic minimum spanning tree problem and related topics", Ph.D. Dissertation, Department of Mathematics, University of Maryland, College Park, MD, 1984.