INFORMATICA, 1997, Vol. 8, No. 3, 377–400

THE USE OF SPECIAL GRAPHS FOR OBTAINING LOWER BOUNDS IN THE GEOMETRIC QUADRATIC ASSIGNMENT PROBLEM

Gintaras PALUBECKIS

Kaunas University of Technology
Studentų 50, 3031 Kaunas, Lithuania
E-mail: gintaras@soften.ktu.lt

Abstract. In this paper we define a class of edge-weighted graphs having nonnegatively valued bisections. We show experimentally that complete such graphs with more than three vertices and also some special graphs with only positive edges can be applied to improve the existing lower bounds for a version of the quadratic assignment problem, namely with a matrix composed of rectilinear distances between points in the Euclidean space.

Key words: combinatorial optimization, quadratic assignment problem, lower bounds.

1. Introduction. Given a finite set $N = \{1, 2, \dots, n\}$ and three $n \times n$ matrices $W = (w_{ij})$, $D = (d_{ij})$ and $A = (a_{ij})$ with real entries, the Koopmans-Beckmann version of the quadratic assignment problem (QAP) is to find a permutation p of the set N such that the sum

$$f(p) = \sum_{i \in N} \sum_{j \in N} w_{ij} d_{p(i)p(j)} + \sum_{i \in N} a_{ip(i)} \tag{1}$$

is minimized. We call this problem a *geometric QAP* if the matrix D represents shortest distances between pairs of n points in the Euclidean space, computed using the rectilinear metric (so the distance between points (x, y) and (x', y') is equal to $|x - x'| + |y - y'|$). A typical example of such a QAP is the facility location problem, in which n given facilities are to be assigned to the same number of locations. In this interpretation, the matrix $W = (w_{ij})$ is the flow matrix, i.e., w_{ij} is the flow of materials from facility i to facility j, and $D = (d_{kl})$ is the distance matrix, i.e., d_{kl} represents the distance from location k to location l. The cost of simultaneously assigning facility i to location k

and facility j to location l is $w_{ij}d_{kl}$. The fixed cost of assigning facility i to location k is given by the entry a_{ik} of the matrix A. The objective is to find an assignment of n facilities to n locations, i.e., a permutation p, such that the total cost of the assignment is minimized. When referring to the QAP, we, occasionally, use the context of this location problem, assuming, of course, that the distances between locations are measured according to the rectilinear metric.

Many other QAPs coming from real applications (see, for example, reviews by Burkard (1984) and Finke *et al.*, (1987)) also fall under the definition of the geometric QAP. Moreover, some of the well-known benchmark problems, for example those due to Steinberg (1961) and Nugent *et al.*, (1968), are of this type. On the other hand, the geometric QAP itself includes as special cases other combinatorial optimization problems, e.g., the well-known linear arrangement problem.

The existing solution techniques for QAP often require lower bounds on the minimal value of f. The problem of obtaining sharp lower bounds has found considerable attention in the literature. The first one is due to Gilmore (1962) and Lawler (1963). Later many other lower bounds and bound computation algorithms have been proposed including those of Frieze and Yadegar (1983), Palubeckis (1988), Hadley *et al.*, (1992), Rendl and Wolkowicz (1992), Adams and Johnson (1994), Chakrapani and Skorin-Kapov (1994).

In this paper, we develop a new method for obtaining lower bounds for geometric QAPs. The method is based on the reduction of the matrix W and is similar to constructive bounding techniques described by Palubeckis (1988) and Chakrapani and Skorin-Kapov (1994). Our main contribution are the following two enhancements of the bound of Palubeckis (1988): the use of less simple weighted graphs than cycles with exactly one negative edge; further reduction of W according to some positively weighted graphs. It follows from computational results that for larger n our bounds compares favorably with the bounds obtained by Chakrapani and Skorin-Kapov (1994).

The bound computation method described here does not use the matrix A explicitly. This matrix is left for processing by some existing bounding technique which is applied to the problem with reduced matrix W. So, we may ignore A and treat f as a function having the quadratic part only. This assumption remains valid everywhere in the rest of the paper except a comment (in Section 3) concerning application of the technique mentioned above. We also assume without loss of generality that the lower triangle of W is zero, i.e., $w_{ji} = 0$ for all i, j such that $i < j$. Indeed, if $w_{ji} \neq 0$ for some pair i, j, $i < j$, then we can replace w_{ij} with $w_{ij} + w_{ji}$ (since for symmetric D $w_{ji}d_{p(j)p(i)} = w_{ji}d_{p(i)p(j)}$). Since $d_{ii} = 0$, $i = 1, \ldots, n$, we can assume, in addition, that $w_{ii} = 0$, $i = 1, \ldots, n$.

We end the introduction with some basic definitions and notations. We denote a graph by $G = (V, E)$ where V is the set of vertices and E is the set of edges (pairs of vertices). All the graphs considered in this paper are undirected and without loops (this means that each pair in E is unordered and consists of different vertices). Sometimes the edges of a graph $G = (V, E)$ will be supplied with weights c_{ij}, $(i, j) \in E$. In such cases we assume that c_{ij} and c_{ji} denote the same object – the weight of the edge $(i, j) \in E$. A graph $G' = (V', E')$ is a *subgraph* of $G = (V, E)$ (induced by the vertex set V') if $V' \subset V$, $E' \subseteq E$ and each edge of G with both vertices in V' is also an edge of G'. A vertex ν of $G = (V, E)$ is an *isolated vertex* if no edge in E is incident with ν. A graph $G = (V, E)$ is *complete* if E contains all (unordered) pairs of different vertices of G. An i-vertex complete graph is denoted by K_i. A *path* P_q of length $q \geqslant 1$ is a graph with vertex set $V = \{\nu_1, \ldots, \nu_{q+1}\}$ and edge set $E = \{(\nu_i, \nu_{i+1}) \mid i = 1, \ldots, q\}$. If $P_q = (V, E)$ is a path of length $q \geqslant 2$ with end vertices ν_1 and ν_{q+1} then the graph C_{q+1} obtained from P_q by way of adding additional edge (ν_1, ν_{q+1}) to E is a *cycle* of length $q + 1$. A cycle C_i is called *odd* if i is an odd number.

Given integers $n_x \geqslant 1$, $n_y \geqslant 1$, we define a *regular 2-dimensional* (or, more precisely, regular $n_x \times n_y$) *grid* as a set $\{(i, j) \mid i = 1, \ldots n_x, \ j = 1, \ldots, n_y\}$ of points on the plane. The distance d_{kl} between grid points $t_k = (i_1, j_1)$ and $t_l = (i_2, j_2)$ equals to $|i_1 - i_2| + |j_1 - j_2|$. A *rectangle* on the grid defined by points (i_1, j_1) and (i_2, j_2) is the set $\{(i, j) \mid i = k_1, \ldots, k_{\max}, \ j = l_1, \ldots, l_{\max}\}$, where $k_1 = \min(i_1, i_2)$, $k_{\max} = \max(i_1, i_2)$, $l_1 = \min(j_1, j_2)$, and $l_{\max} = \max(j_1, j_2)$. If either $n_x = n$ and $n_y = 1$ or $n_x = 1$ and $n_y = n$, then (1) with the distance matrix D corresponding to $n_x \times n_y$ grid is called a *linear arrangement problem.*

2. Special graphs used in the reduction. We can associate to any flow matrix W the graph $G(W) = (V(W), \ E(W))$, $V(W) = N$, $E(W) = \{(i, j) \mid 1 \leqslant i, \ j \leqslant n, \ w_{ij} \neq 0\}$, with edge weights $c_{ij} = w_{ij}$, $(i, j) \in E(W)$. This relation holds in the reverse direction as well, therefore, we can use any of

W, $G(W)$ together with D to specify an instance of QAP. In addition, we shall assume in what follows that the isolated vertices are eliminated from $V(W)$. In the context of facility location problem, the vertices of $G(W)$ can be interpreted as facilities and edge weights as flows of materials between pairs of facilities.

Let $f_0(W, D)$ denote an optimal value of f for given matrices W and D. First we state the following obvious fact.

Lemma 1. *If $W = W_1 + W_2$ and D is a rectilinear distance matrix, then*

$$f_0(W, D) \geqslant f_0(W_1, D) + f_0(W_2, D). \tag{2}$$

If one of the matrices, say W_2, in this decomposition corresponds to rather simple graph $G(W_2)$, and thus $f_0(W_2, D)$ can be efficiently established, the problem of finding a lower bound on $f_0(W, D)$ is reduced to the same problem with respect to W_1. Setting $W := W_1$ and applying (2) iteratively we can get a lower bound on the initial objective function.

Let M be a class of cycles with vertices in N, one edge having weight -1 and the rest having weight 1. Let C_k stand for a k-vertex cycle. The good candidates for $G(W_2)$ (and correspondingly for W_2 in (2)) are cycles $C_k \in M$, especially C_3, C_4, with weights multiplied by some positive number α. Such graphs were used in the bounds of Palubeckis(1988) and Chakrapani and Skorin-Kapov (1994).

The following definition characterizes a class of graphs some members of which are used in the alternative bound computation algorithm described in Section 3.

DEFINITION. A graph $G = (V, E)$ with edge weights c_{ij}, $(i, j) \in E$, is a *PB-graph* (has nonnegatively valued bisections) if the sum $S(G, V')$ of the weights in the set $\{c_{ij} \mid (i, j) \in E,\ i \in V',\ j \in V \backslash V'$ or $i \in V \backslash V',\ j \in V'\}$ is nonnegative for each subset $V' \subset V$.

It is easy to see that each member of M is a PB-graph.

The bound on the minimal value of f for PB-graphs is given by the following assertion.

Lemma 2. *If W is such that $G(W)$ is a PB-graph, then for any rectilinear distance matrix D*

$$f_0(W, D) \geqslant 0. \tag{3}$$

Proof. Since the matrix D is computed using the rectilinear metric, we can decompose, provided the dimension of the space is at least two, the objective function into two or more parts corresponding to different axes of a coordinate system

$$f_0(W, D) = f_{0x}(W, D) + f_{0y}(W, D) + \cdots.$$

Suppose we are given n points defining D and let $x_1, \ldots, x_n$ be x-coordinates of these points sorted nondecreasingly. We can write

$$f_{0x}(W, D) = \sum_{i=1}^{n-1} S(G(W), V_i)(x_{i+1} - x_i),$$

where V_i is the set of the vertices of $G(W)$ corresponding to facilities assigned by an optimal permutation to points (locations) with x-coordinates $x_1, \ldots, x_i$. Since $S(G(W), Vi) \geqslant 0$ for any subset of vertices, $f_{0x}(W, D) \geqslant 0$. The same holds for other directions as well, yielding (3).

Combining Lemma 1 with Lemma 2 for members of the class M leads to the following result of Palubeckis (1988), stated here using definitions and notations of this paper.

Lemma 3. *Let W be a nonnegative flow matrix and $W = W_1 + W_2$ be its decomposition satisfying the following conditions: 1) W_1 is nonnegative; 2) $W_2 = \alpha W_2^*$, where $\alpha > 0$ and W_2^* is such that $G(W_2^*) \in M$. Then for any rectilinear distance matrix D $f_0(W_1, D) \leqslant f_0(W, D)$.*

In order to get tighter lower bounds a reasonable strategy is to take in (2) such a matrix W_2 that $G(W_2)$ would be a PB-graph with the sum of edge weights as small as possible. This results in W_1 with larger sum of entries, so we may expect a larger bound for the residual problem defined by W_1 and D. As a measure of how good a PB-graph G is in this respect, the ratio $\rho(G) =$ (total weight of positive edges)/(-1)(total weight of negative edges) can be used. The goal is a PB-graph with small $\rho(G)$. For $C_i \in M$, $\rho(C_i) = i - 1$. Particularly, $\rho(C_3) = 2$. On the other hand, it is clear that a PB-graph G with $\rho(G) < 1$ cannot exist. Moreover, we now show that even $\rho(G) = 1$ is impossible.

Proposition. *For a PB-graph G, $\rho(G) > 1$.*

Proof. Suppose on contrary that $\rho(G) = 1$ for some PB-graph $G = (V, E)$ without isolated vertices. Let c_{ij}, $(i, j) \in E$, be the edge weights of G as before and $\sigma_i = \sum_{j,(i,j)\in E} c_{ij}$ be the degree of the vertex i. $\rho(G) = 1$ implies that $\sum_{i\in V} \sigma_i = 0$. Since $S(G, \{i\}) \geqslant 0$ for any $i \in V$ it follows that $\sigma_i = 0$ for all $i \in V$. Take the vertex $1 \in V$ and any $j \in V$ such that $c_{1j} > 0$. For this pair $S(G, \{1, j\}) = \sigma_1 + \sigma_j - 2c_{1j} < 0$, a contradiction.

To construct PB-graphs with $1 < \rho(G) < 2$, we suggest the following procedure which we call *lifting* of a PB-graph. Suppose a PB-graph $G = (V, E)$, $|V| = \widetilde{n}$, is given. Adjoin some set V_0 of new vertices to G. To this end, set $E := E \cup E_0$, $E_0 = \{(i, j) \mid i \in V,\ j \in V_0 \text{ or } i, j \in V_0\}$, $V := V \cup V_0$, and consider the inequalities $S(G, V') \geqslant 0$ for all $V' \subset V, |V'| \leqslant (\widetilde{n} + |V_0|)/2$, with unknowns $c_{ij}, (i, j) \in E_0$. Finding values for c_{ij}, $(i, j) \in E_0$, which are feasible for this system of inequalities and removing edges with $c_{ij} = 0$ from E yields an $(\widetilde{n} + |V_0|)$-vertex PB-graph.

EXAMPLE 1. Given a graph $G = (V, E)$, let E_+(resp. E_-) denote the set of positive (resp. negative) edges of G. Let $\widetilde{E}$ be the edge set of a complete graph. It is easy to see that by taking $C_3 \in M$ and applying the lifting procedure sequentially $i - 3 > 0$ times with V_0 containing only one new vertex each time we can obtain a complete i-vertex PB-graph with vertex set $V = V^1 \cup V^2$, $|V^1| = \lceil i/2 \rceil$, $V^2 = i - |V^1|$, positive and negative edge sets $E_+ = \{(i, j) \mid i \in V^1,\ j \in V^2\}$, $E_- = \widetilde{E} \backslash E_+$ and $|c_{ij}| = 1$ for all $(i, j) \in \widetilde{E}$. We denote this graph by $H_i = (V, \widetilde{E})$. The graph H_5 is shown in Fig. 1a. Clearly, $\rho(H_i) = i/(i-2)$ if i is even, and $\rho(H_i) = (i+1)/(i-1)$ if i is odd. For any H_i, $i \geqslant 4$, it is easy to select such a distance matrix D that equality is attained in (3).

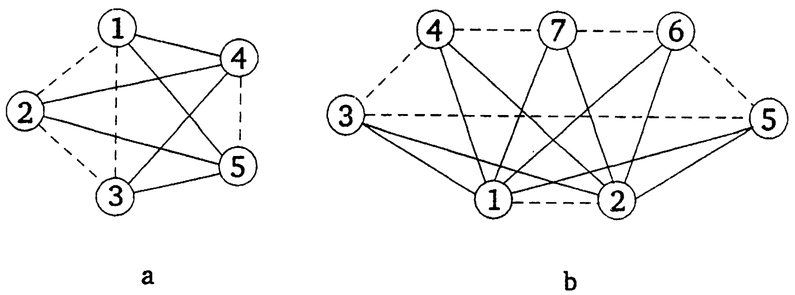


Fig. 1. Examples of PB-graphs (solid lines denote edges of weight 1, dashed – of weight –1): a – graph H_5; b – graph G^*.

The graphs H_5 and H_7 are used in the lower bounding algorithm described in the next section. More precisely, they, after multiplying their edge-weights by some positive scalar, play the role of the graph $G(W_2)$ corresponding to the matrix W_2 in the decomposition $W = W_1 + W_2$ (see Lemma 1 and discussion below it).

EXAMPLE 2. Now we consider lifting of $G = H_4 = (V, \widetilde{E})$ with $V^1 = \{1, 2\}$, $V^2 = \{3, 4\}$. We take $V_0 = \{5, 6, 7\}$. It is easy to verify that the values $c_{15} = c_{16} = c_{17} = c_{25} = c_{26} = c_{27} = 1$, $c_{35} = c_{56} = c_{67} = c_{47} = -1$, $c_{36} = c_{37} = c_{45} = c_{46} = c_{57} = 0$ are feasible for the system $S(G, V') \geqslant 0, V' \subset \{1, \dots, 7\}$, $|V'| \leqslant 3$. Thus we obtain the PB-graph $G^* = (V \cup V_0, \widetilde{E})$ with $E_- = \{(1,2), (3,4), (3,5), (5,6), (6,7), (4,7)\}$, $E_+ = \widetilde{E} \backslash (E_- \cup \{(3,6), (3,7), (4,5), (4,6), (5,7)\})$ and $\rho(G^*) = 5/3$ (see Fig. 1b). Observe that any PB-subgraph obtained by removing one or more vertices from this graph has a larger value of ρ than the graph itself. This, however, does not hold for H_i for any even $i \geqslant 4$.

REMARK. For linear arrangement problem graphs violating condition "non-negativity of bisections" can be used. A good example is H_4 with $c_{ij} = -1$ replaced by $c_{ij} = -3/2$ for $(i, j) \in E_-$. For this modified graph, say H_4', we have $\rho(H_4') = 4/3$. It is easy to see that if D is the distance matrix of the linear arrangement problem and the flow matrix W is such that $G(W)$ is isomorphic to H_4', then $f_0(W, D) = 0$.

In the remainder of this section we restrict ourselves to a special case of geometric QAPs – problems whose matrix D represents distances between points of a regular grid of size $n_x \times n_y$, $n_x n_y = n$.

When applying (2) iteratively to derive a lower bound for a QAP in this subclass, together with PB-graphs we also use unweighted graphs having some special structure. An unweighted graph is obtained when all the entries of one of the right-hand matrices in the decomposition $W = W_1 + W_2$ belong to the set $\{0, \alpha\}$, where α is some positive constant. Suppose that W_2 satisfies this property. Let W_2^* be such that $W_2 = \alpha W_2^*$. Then $G(W_2^*) = (V(W_2^*),\ E(W_2^*))$, $|V(W_2^*)| = n^* \leqslant n$, is unweighted graph with vertices corresponding to nonzero rows of W_2 and edges corresponding to nonzero entries of W_2. Isolated vertices induced by zero rows of W_2 are eliminated from $G(W_2^*)$ (remember assumption made in the beginning of this section).

To bound $f_0(W_2, D)$ from below, we relax the QAP defined by W_2 and

D slightly. More precisely, we consider the problem of assigning the vertices of $G(W_2^*)$ to points of a sufficiently large, for example $n^* \times n^*$, regular grid. Let D' denote the distance matrix corresponding to this grid, and let W_2' be a matrix such that $G(W_2') = G(W_2^*)$ and the size of W_2' is the same as that of D'. Clearly, $\alpha f_0(W_2', D') \leqslant f_0(W_2, D)$. In the case of grids and simple graphs it seems, however, more appropriate instead of $f_0(W_2', D')$ to consider the function $\Delta(G(W_2^*)) = f_0(W_2', D') - |E(W_2^*)|$. Then we have the bound

$$\alpha\left|E(W_2^*)\right| + \alpha\Delta(G(W_2^*)) \leqslant f_0(W_2, D). \tag{4}$$

The value $\Delta(G(W_2^*))$ shows how much $f_0(W_2', D')$ is larger than the sum of lengths of edges $(i,j) \in E(W_2^*)$ in the case when each edge of $G(W_2^*)$ is assigned to a pair of points of the grid independently of the other edges. We are interested in graphs having a positive value of Δ. The simplest such graph is the triangle K_3. Obviously, $\Delta(K_3) = 1$. Applicability of triangles to obtaining lower bounds for QAP was pointed out by Chakrapani and Skorin-Kapov (1994). We restate their result (Lemma 3) using our definitions and notations.

Lemma 4. *Let p_0 be an optimal solution for QAP defined by the flow matrix W_1 and distance matrix D corresponding to regular 2-dimensional grid. If W is another flow matrix such that $W = W_1 + \alpha W_2^*$, $\alpha > 0$, $G(W_2^*) = K_3$ and the vertices of K_3 are assigned by p_0 to grid points i, j, k such that $d_{ik} = d_{kj} = 1$, $d_{ij} = 2$, then p_0 is still optimal for QAP defined by the pair W, D, with the optimal objective function value of $f_0(W_1, D) + 4\alpha$.*

Thus, under the conditions of this lemma, we have equality in (4), and the problem of obtaining a lower bound on $f_0(W, D)$ can be reduced to the same problem with respect to $f_0(W_1, D)$.

Other simple graphs with positive Δ are 4-vertex complete graph K_4 and odd cycles C_i, $i \geqslant 5$. Clearly, $\Delta(K_4) = 2$ and $\Delta(C_i) = 1$ for odd $i \geqslant 5$.

Let $P(q,r,s)$ be the graph composed of three paths having common end vertices, (pairwise) disjoint sets of internal vertices and lengths q, r and s respectively, i.e., the graph with vertex set $\{\nu_i \mid i = 1,\ldots,q+1\} \cup \{\nu_i'' \mid i = 2,\ldots,r\} \cup \{\nu_i' \mid i = 2,\ldots,s\}$ and edge set $\{(\nu_i, \nu_{i+1}) \mid i = 1,\ldots,q\} \cup \{(\nu_i', \nu_{i+1}') \mid i = 2,\ldots,r-1\} \cup \{(\nu_1, \nu_2'), (\nu_r', \nu_{q+1})\} \cup \{(\nu_i'', \nu_{i+1}'') \mid i = 2,\ldots,s-1\} \cup \{(\nu_1, \nu_2''), (\nu_s'', \nu_{q+1})\}$. Fig. 2 shows an example of such a graph.

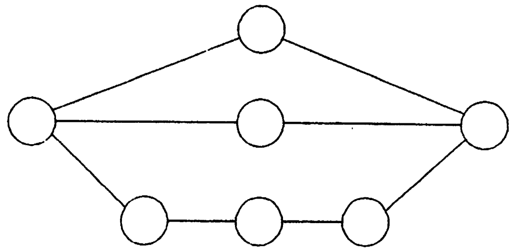


Fig. 2. Graph $P(2,2,4)$.

Theorem. *For integers q, r, s such that $q \leqslant r \leqslant s$, and $q \geqslant 2$ or $q = 1$, $r \geqslant 2$,*

$$\Delta(P(q,r,s)) \leqslant 2$$

with equality if and only if $q = r = s \leqslant 6$ or $q = r \in \{2,3,4\}$, $s = q+2$.

The proof is given in the Appendix.

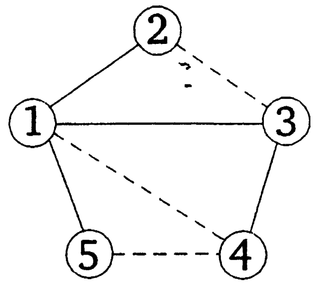
In the algorithm of the next section we use K_3, K_4, odd cycles of any length, $P(2,2,2)$ and $P(2,2,4)$.

3. Computation of the lower bound. We now describe a lower bound computation algorithm for QAPs whose matrix D is composed of rectilinear distances between points of a regular $n_x \times n_y$ grid. Let $z_1 < z_2 < \ldots < z_h$ represent all distinct values in $\{w_{ij} \mid i = 1, \ldots, n-1,\ j = i+1, \ldots, n\}$. We assume that h is not very large as compared with n. If this would not be a case, it might probably be better instead of z_i, $i = 1, \ldots, h$, to consider intervals in some subdivision of $[\min w_{ij},\ \max w_{ij}]$.

Before presenting the algorithm, we give the definition of some special graph the subgraphs of which are used to select the second matrix in the decomposition defined in Lemma 1. Given a matrix W and reals T_1, T_2 such that $T_1 \geqslant T_2$, and $T_1 < z_h$ or (and) $T_2 > z_1$, let $\widetilde{G}(W, T_1, T_2)$ denote the graph with vertex set N, edge set $\widetilde{E}(W) = E_+ \cup E_-, E_+ = \{(i,j) \mid 1 \leqslant i < j \leqslant n,\ w_{ij} > T_1\}$, $E_- = \{(i,j) \mid 1 \leqslant i < j \leqslant n,\ w_{ij} < T_2\}$, and weights $c_{ij}(\widetilde{G}) = 1$, $(i,j) \in E_+$, $c_{ij}(\widetilde{G}) = -1$, $(i,j) \in E_-$. An illustration of such a graph is shown in Fig. 3. For a subgraph $G' = (V', E'_+ \cup E'_-)$ of $\widetilde{G}(W, T_1, T_2)$ and any positive α, let $\widetilde{W}(G', \alpha)$ be the matrix of size $n \times n$ with entries $\widetilde{w}_{ij} = \alpha c_{ij}(G')$ if $(i,j) \in E'_+ \cup E'_-$, and $\widetilde{w}_{ij} = 0$ otherwise. In

$$W = \begin{vmatrix} 0 & 3 & 3 & 0 & 4 \\ & 0 & 1 & 2 & 2 \\ & & 0 & 5 & 2 \\ & & & 0 & 0 \\ & & & & 0 \end{vmatrix}$$

a



b

Fig. 3. Example of the graph $\widetilde{G}(W, T_1, T_2)$: a – matrix W; b – graph $\widetilde{G}(W, 2, 2)$ (solid lines denote edges of weight 1, dashed – of weight -1).

the algorithm below, $\alpha = \alpha(G', W, T_1, T_2) = \min(\min\{w_{ij} - T_1 \mid (i,j) \in E'_+\}, \min\{T_2 - w_{ij} \mid (i,j) \in E'_-\})$.

For $k \in \{1, \ldots, h\}$, let $\eta_{k1} = \sum \max(0, w_{ij} - z_k)$, $\eta_{k2} = \sum \max(0, z_k - w_{ij})$ be the sums taken over all entries of W above the main diagonal. Define $\psi_k = |\eta_{k1}/2 - \eta_{k2}|$, $e = e' + 1$, where $e' = (n_x - 1)n_y + n_x(n_y - 1)$ is the number of pairs of neighboring grid points.

A description of the lower bounding algorithm (called LB) is given as follows.

ALGORITHM LB

1. Set $T := z_u$ where u is such that $\psi_u = \min\{\psi_i \mid 1 \leqslant i \leqslant h\}$.
2. Find a subgraph G' of $\widetilde{G}(W, T, T)$ isomorphic to H_7. If none exists, proceed to 3. Otherwise subtract $\widetilde{W}(G', \alpha)$, $\alpha = \alpha(G', W, T, T)$, from W and repeat 2.
3. Perform the same operations as in 2 with respect to H_5. Go to 4 (if no G' isomorphic to H_5 exists) or repeat 3.
4. Call RESIDUAL(b_0) (b_0 is a lower bound returned).
5. For $i = 1, \ldots, u - 1$
   5.1. Call CYCLES(i).
   5.2. Call RESIDUAL(b_i).

6. Save current W. Perform loop 5 calling CYCLES2 instead of CYCLES and using b_i' instead of b_i.
7. Restore W saved in 6. Perform loop 5 calling LONG_CYCLES instead of CYCLES and using b_i'' instead of b_i.
8. Stop, $b = \max(b_0, \max\{b_i, b_i', b_i'' \mid 1 \leqslant i \leqslant u-1\})$ is a lower bound on $f_0(W, D)$.

The following procedure reduces the flow matrix W by subtracting matrices corresponding to complete graphs, odd cycles and graphs $P(2,2,2)$, $P(2,2,4)$. To obtain the lower bound, it applies to the residual problem some known lower bounding technique.

PROCEDURE RESIDUAL(b^*)

1. Set $\delta := w_{\gamma\mu}$ where $w_{\gamma\mu}$ is the eth largest entry of W above the main diagonal. Set $\beta := 0$. Save W.
2. Find a subgraph $G' = (V', E')$ of $\widetilde{G}(W, \delta, -\infty)$ isomorphic to K_4 or K_3. If none exists, proceed to 3. Otherwise subtract $\widetilde{W}(G', \alpha)$, $\alpha = \alpha(G', W, \delta, 0)$, from W, set $\beta := \beta + (\Delta(G') + |E'|)\alpha$ and repeat 2.
3. Perform the same operations as in 2 with respect to $P(2,2,2)$ and $P(2,2,4)$. Go to 4 (if failure) or repeat 3.
4. Perform the same operations as in 2 with respect to odd cycles. Go to 5 (if failure) or repeat 4.
5. Compute a lower bound $\tilde{b}$ on the minimal value of f for the residual problem. Set $b^* := \tilde{b} + \beta$.
6. Optionally, set $\beta := 0$, restore W and repeat Steps 2–5 for smaller values of δ, keeping the largest b^* thus obtained.
7. Restore W saved in 1 and return with b^*.

The next procedure reduces the flow matrix W by subtracting matrices corresponding to cycles isomorphic to $C_3 \in M$ or $C_4 \in M$. This procedure invokes some heuristic for QAP. The heuristic we use in our implementation is described briefly at the end of this section.

PROCEDURE CYCLES(i)

1. If $i = 1$, apply a heuristic to the current problem.
2. Select a subgraph G' of $\widetilde{G}(W, T, z_{i+1})$ satisfying the following conditions: 1) G' is isomorphic to $C_3 \in M$; 2) current heuristic solution is optimal for a QAP with $\widetilde{W}(G', 1)$ and D. If none exists, proceed to 3.

Otherwise subtract $\widetilde{W}(G', \alpha)$, $\alpha = \alpha(G', W, T, z_{i+1})$, from W, apply a heuristic to the residual problem and repeat 2.

3. Perform the same operations as in 2 ignoring the second condition for G'. Go to 4 or repeat 3.
4. Perform the same operations as in 2 with respect to $C_4 \in M$ ignoring the second condition for G'. Return or repeat 4.

Procedure CYCLES2 is a slight embellishment of CYCLES. A subgraph G' in CYCLES2 is required to be isomorphic to C_3 (or C_4) $\in M$ without one positive edge. Moreover, it is required for such a nonedge pair (j, j'), $j < j'$, and $(k, k') \in E'_-$, $k < k'$, that $w_{jj'} = T$ and $w_{kk'} + T \geqslant 2z_{i+1}$.

Procedure LONG_CYCLES can be stated as Step 2 of CYCLES with the following simplifications: only the first condition with $C_i \in M$, $i > 4$, instead of C_3 is used, and heuristic is not invoked.

We end this section with the following comments and implementation details.

1. The choice of T in Step 1 of LB is motivated by (somewhat unrealistic) aim of obtaining the final W with all entries equal or close to some constant (that is T), using for this solely the subgraphs isomorphic to $C_3 \in M$. For all QAP instances used in our experiments $T = 2$.

2. Steps 6 and 7 of LB may be deemed as being optional. As it follows from our experiments, in most cases small or even no increase in b due to these steps was observed.

3. The choice of δ in Step 1 of RESIDUAL is based on the fact that only e' largest entries of W are multiplied by $\min_{i,j=1,\ldots,n,\ i\neq j} d_{ij} (= 1)$ in the minimal scalar product of the upper triangles of W and D, i.e., in the sum $\sum_{k=1}^{n'} \tau_k \chi_k$, where $n' = n(n-1)/2$, and τ (respectively, χ) is the vector constructed by taking the entries w_{ij}, $i = 1, \ldots, n-1$, $j = i+1, \ldots, n$ (respectively, d_{ij}, $i = 1, \ldots, n-1$, $j = i+1, \ldots, n$) as components and sorting them descendingly (respectively, ascendingly).

4. Step 5 of RESIDUAL is the only place where the matrix A, provided nonzero, is used. Any existing lower bounding method can be adopted to implement this step. Our computational results are obtained using the Gilmore-Lawler bound (Gilmore, 1962; Lawler, 1963). For the geometric QAP this bound can be formulated as follows. Let w^i, $i \in \{1, \ldots, n\}$, be the vector composed of w_{ij}, $j = i+1, \ldots, n$, and w_{ji}, $j = 1, \ldots, i-1$, ordered descendingly, and

d^j, $j \in \{1, \ldots, n\}$, be the vector obtained from the jth row of D by deleting d_{jj} and ordering the remaining components ascendingly. Denote by $O = (o_{ij})$ the matrix with entries $o_{ij} = a_{ij} + (\sum_{k=1}^{n-1} w_k^i d_k^j)/2$, $i, j = 1, \ldots, n$. Then the Gilmore-Lawler bound is defined to be the optimal value for the following linear assignment problem

$$\min_{p \in \Pi} \sum_{i \in N} o_{ip(i)},$$

where Π is the set of all permutations of N. The bound can be computed using efficient methods for this problem (see, for example, Papadimitriou and Steiglitz, 1982).

5. For a heuristic solution p and different $i, j, k, l \in N$, let $F_1 = d_{p(i)p(j)}$, $F_2 = d_{p(i)p(k)} + d_{p(k)p(j)}$, $F_3 = d_{p(i)p(k)} + d_{p(k)p(l)} + d_{p(l)p(j)}$, F_4 =(the number of grid points in the rectangle defined by the points $t_{p(i)}$ and $t_{p(j)}$), $F_5 = \min(\overline{w}_{ik}, \overline{w}_{jk})$, $F_6 = \max(\overline{w}_{ik}, \overline{w}_{jk})$, $F_7 = \min(\overline{w}_{ik}, \overline{w}_{kl}, \overline{w}_{lj})$, where $\overline{w}_{ik}$ is equal to w_{ik} (if $i < k$) or w_{ki} (if $i > k$), and $\overline{w}_{jk}, \overline{w}_{kl}, \overline{w}_{lj}$ are defined analogously. In CYCLES, the following criteria are used (if a pair is indicated, then the second for breaking the ties): for selecting the negative edge (i, j) of G', $\min F_1$ and $\min F_4$ at Step 2, and $\min F_1$ at Steps 3 and 4; for selecting the remaining one or two vertices, $\max F_5$ and $\max F_6$ at Step 2, $\min F_2$ and $\max F_5$ at Step 3, and $\min F_3$ and $\max F_7$ at Step 4.

6. No special rules are used for selecting subgraphs isomorphic to $H_7, H_5, K_4, K_3, P(2,2,2), P(2,2,4)$, odd cycles and long cycles, that is these subgraphs are processed in the order of appearance during the run of some search procedure specific for each case. For example, such a procedure for H_5 searches all pairs of negative edges of $\widetilde{G}(W, T, T)$, checks whether a pair induces H_4 and, if so, tries to find one more vertex to get H_5.

7. In CYCLES and CYCLES2 we have used a heuristic for QAP which can be characterized as a first step towards tabu search (see Glover, 1989; 1990) for a description of this general technique, and Skorin-Kapov (1990) for one of the first tabu search-based algorithms for QAP). The heuristic invokes repeatedly the following simple procedure the input to which includes, besides W and D, some initial assignment p_{init}.

PROCEDURE DESCENT

1. Set $p := p_{\text{init}}$.
2. Search the neighborhood of p defined as $J(p) = \{p' \in \Pi \mid$ there exist

$j, k \in N,\ j \neq k$, such that $p'(j) = p(k),\ p'(k) = p(j)$, and $p'(i) = p(i)$ for all $i \in N \backslash \{j, k\}\}$, where Π is the set of all permutations of N. If $p' \in J(p)$ is found for which $f(p') < f(p)$, then set $p := p'$ and repeat 2. Otherwise, stop with p.

The heuristic accepts flow matrix W, distance matrix D, starting solution p_{start} and some set of parameters as an input, builds some sequence of solutions each produced by DESCENT and stops with the best of them as an output. Each application of DESCENT together with some operations before and after it is called an iteration. At the first iteration, DESCENT is applied to p_{start}. At each subsequent iteration, the last solution in the sequence is perturbed in order to get a new initial permutation p_{init} and then DESCENT is applied. This process stops when the number i of successive iterations without an improvement of the best solution found reaches some specified limit I. To get p_{init}, each facility is exchanged with some other facility chosen to minimize the new value of the function f. This loop of n exchanges is repeated $\max(1, X + i)$ times, where X is some parameter. Note that in the case of $I = 0$ this heuristic stops after invoking the procedure DESCENT only once.

4. Computational results. The algorithm we have described has been coded in the C language and tested on the same benchmark QAPs as in Chakrapani and Skorin-Kapov (1994). The distance matrix of each test problem is defined by some regular 2-dimensional grid. All flow matrices except that of the problem given by Steinberg (1961) are generated randomly. The set of smallest QAPs is due to Nugent *et al.*, (1968). The sizes of problems and corresponding grids are the following (the first member of each pair is the value of n, whereas the second represents the values of n_x and n_y): $(6, 3 \times 2)$, $(8, 4 \times 2)$, $(12, 4 \times 3)$, $(15, 5 \times 3)$, $(20, 5 \times 4)$, $(30, 6 \times 5)$. The entries of W are taken from the set $\{0, 1, \ldots, 5, 6, 10\}$. To refer these problems, we use the names Nug6, ..., Nug30 each being a concatenation of "Nug" and dimension of the problem. The benchmark QAPs suggested by Skorin-Kapov (1990) mimic those by Nugent *et al.*, (1968) but have larger sizes. The entries of W are taken from the same set and appear with the same frequencies as in Nug12. The dimension of the first problem, named Sko42, is 42 and the second, named Sko49, is 49. The size of the corresponding grid is 7×6 and 7×7, respectively. The problem introduced by Steinberg (1961) is of different type. It is an instance of a real problem arising in the area of design automation. More precisely, this problem is concerned with the optimal placement of 36 components on a 9×4 grid of 36 backboard locations. The entries of W ranges from 0 to 316. We refer to this problem using the name Ste36.

The results of experimentation are presented in Tables 1 and 2. In Table 1, the second column gives the optimal if $n \leqslant 15$ or the best known objective function value. The next three columns display the lower bounds: the Gilmore-Lawler bound GLB, the bound CBLB from Chakrapani and Skorin-Kapov (1994), and the bound delivered by the algorithm LB for the case of $I = X = 0$ (remember that I and X are parameters of the heuristic used in the bound computation: I is the limit on the number of successive iterations without an improvement of the best solution already found, and X is the parameter used to obtain a perturbed solution). The last column in this triplet contains the main data for comparison of our bound with GLB and CBLB. We have applied our algorithm more than one time to each problem taking different values of I and X. The best results are presented in the column under heading 'LB best'. The last two columns give the values of I and X in the best trial. It should be noted that larger values of I and X not necessarily lead to better lower bounds. The results show that our bound is sharper than CBLB for all but first three problems.

Table 1. Comparison of LB with previous lower bounds

Problem	Best value	GLB	CBLB	LB	LB best	I	X
Nug6	86	82	84	84	84	0	0
Nug8	214	186	206	196	198	2	0
Nug12	578	493	528	528	530	3	0
Nug15	1150	963	1044	1054	1062	5	−1
Nug20	2570	2057	2262	2312	2316	3	0
Nug30	6124	4539	5450	5526	5558	4	1
Ste36	9526	7124	7480	7722	7766	2	0
Sko42	15812	11311	14192	14280	14280	0	0
Sko49	23386	16161	20910	21052	21160	1	0

In Table 2, we compare the standard version of LB (second column) with the following simplifications: without Steps 2 and 3 of LB (the bound LB2 in the table), with RESIDUAL replaced by Gilmore-Lawler bound (the bound

Table 2. Standard algorithm LB versus its simplifications ($I = X = 0$)

Problem	LB	LB2	LB3	LB4	LB time	$\#H_7$	$\#H_5$
Nug6	84	84	82	84	0	0	0
Nug8	196	196	196	196	0	0	0
Nug12	528	528	528	528	0	0	0
Nug15	1054	1046	1050	1054	1	0	2
Nug20	2312	2292	2294	2276	2	0	5
Nug30	5526	5414	5518	5516	16	0	24
Ste36	7722	7724	7618	7740	54	0	4
Sko42	14280	13910	14252	14158	70	2	57
Sko49	21052	20320	21044	21092	111	6	90

LB3 in the table), and with applying a heuristic only once, that is in Step 1 of CYCLES (the bound LB4). The results are given for the case of $I = X = 0$. The running time (in seconds under an IBM PC-486/66) is reported for the standard version only. For LB4 computation it is about 3–4 times shorter. The last two columns display the number of subgraphs selected in Steps 2 and 3 of LB.

Inspection of Table 2 reveals that for $n \geqslant 15$ each relaxation of LB leads, in most cases, to inferior lower bounds than LB. For Ste36, LB is smaller than LB2 and LB4. However, for the values of I and X given in Table 1 we obtain a different picture – $\mathrm{LB} = 7766$, $\mathrm{LB2} = 7722$, $\mathrm{LB3} = 7636$, $\mathrm{LB4} = 7746$.

5. Concluding remarks. In this paper we have defined a class of edge-weighted graphs with nonnegatively valued bisections. We have shown that complete such graphs with more than three vertices and also some special graphs with only positive edges can be applied to strengthen the existing lower bounds for the geometric QAP. We provided an algorithm which for test QAPs with $n \geqslant 15$ improves the bounds obtained by Chakrapani and Skorin-Kapov (1994). We should note that other algorithms based on the same ideas could be devised. For smaller size QAPs one way is to employ linear programming techniques.

Given $W, T, \widetilde{G}(W, T, T)$ and some collection Ω of PB-graphs, define $\Theta = \{G_l = (V_l, E_l) \mid G_l$ is a subgraph of $\widetilde{G}(W, T, T)$ isomorphic to some member of $\Omega\}$, and let L denote the index set of the subgraphs in Θ. For $G_l \in \Theta$, let

Λ_l be the number of negative edges of G_l. Define $E_+ = \{(i,j) \mid 1 \leqslant i < j \leqslant n,\ w_{ij} > T\}$, $E_- = \{(i,j) \mid 1 \leqslant i < j \leqslant n,\ w_{ij} < T\}$. For $(i,j) \in E_+ \cup E_-$ let $L(i,j) = \{l \in L \mid (i,j) \in E_l\}$. Now we can write the following linear program

$$\max \sum_{l \in L} \Lambda_l \alpha_l,$$

$$\sum_{l \in L(i,j)} \alpha_l \leqslant |w_{ij} - T| \quad \text{for all} \quad (i,j) \in E_+ \cup E_-, \tag{5}$$

$$\alpha_l \geqslant 0 \quad \text{for all} \quad l \in L.$$

Subtracting $\widetilde{W}(G_l, \alpha_l)$ from W for all G_l with $\alpha_l > 0$ in an optimal solution to this program and applying procedure RESIDUAL, we can get a lower bound on $f_0(W, D)$. However, this approach may not be computationally tractable for larger QAPs.

We have solved (5) for Nug6–Nug12 and obtained the following bounds: 84, 198 and 526. The first two bounds in Table 1 under heading "LB best" are the same and the third is better. This can be explained by the fact that the algorithm LB tries first to reduce in W the number of entries equal to z_1, then the number of entries equal to z_2 and so on. For Nug12, the final matrix W obtained using (5) has 7 zero entries, while W at the end of Step 5 of LB has no zero. Thus LB outperforms the algorithm based on (5) for this particular QAP.

Finally, we note that other existing lower bounds rather than Gilmore-Lawler bound could be used in Step 5 of RESIDUAL. For example, the bounds described by Hadley *et al.*, (1992) and Rendl and Wolkowicz (1992) could be tried.

Appendix: *Proof of Theorem.* We can consider an unbounded grid $\{(i,j) \mid i,j = 0, \pm 1, \pm 2, \ldots\}$. Let us define a *grid path* as a sequence of different grid points. For a grid path $Q = \{(x_i, y_i) \mid i = 1, \ldots, m\}$, let $\lambda_i = |x_i - x_{i+1}| + |y_i - y_{i+1}|$, $i \in \{1, \ldots, m-1\}$. The sum $\lambda(Q) = \sum\{\lambda_i \mid i = 1, \ldots, m-1\}$ is called the length of Q. We say that Q is *minimal* if $\lambda(Q) = m - 1$.

First we establish the following two simple facts.

Fact 1. *Given any pair ξ, ζ of the grid points, the minimal grid paths between ξ and ζ are either all of even length or all of odd length.*

Proof. Suppose $\xi = (x, y)$ and $\zeta = (x', y')$ are such that $|x - x'| + |y - y'|$ is even. We will show that each minimal grid path between ξ and ζ has even length. To the contrary, assume that for a minimal grid path $Q = \{t_i = (x_i, y_i) \mid i = 1, \ldots, m,\ t_1 = \xi,\ t_m = \zeta\}$ $\lambda(Q)$ is odd. Each pair $(t_i, t_{i+1}),\ i \in \{1, \ldots, m-1\}$, can be assigned to precisely one of four groups according to what of the following conditions it satisfies: $x_{i+1} > x_i,\ x_{i+1} < x_i,\ y_{i+1} > y_i,\ y_{i+1} < y_i$. Let $m_i,\ i = 1, \ldots, 4$, stand for the number of pairs in each of these groups. Since $|m_1 - m_2| = |x - x'|,\ |m_3 - m_4| = |y - y'|$ and $\lambda(Q) = m_1 + m_2 + m_3 + m_4$ it follows that $\lambda(Q)$ must be even, a contradiction.

Fact 2. *Let $q,\ r,\ s$ be as in Theorem and let p be an optimal assignment of the vertices of $P(q, r, s)$ to points of the grid. If*

$$r - q = 0 \ mod\ 2,\ s - q = 0 \ mod\ 2 \tag{6}$$

and $\Delta(P(q, r, s)) < 2$, then all three grid paths obtained by restricting of p to paths of $P(q, r, s)$ are minimal (and thus actually $\Delta(P(q, r, s)) = 0$).

Proof. Suppose on contrary that one of three grid paths given by p is not minimal (since $\Delta(P(q, r, s)) < 2$ at most one such can exist). Let $Q = \{t_i = (x_i, y_i) \mid i = 1, \ldots, m\}$ denote this path, and suppose that Q is of even length while the remaining two paths, by (6), are of odd length. Clearly, $\lambda_j = 2$ for some $j \in \{1, \ldots, m-1\}$ and $\lambda_i = 1$ for all $i \neq j,\ i \in \{1, \ldots, m-1\}$. Let U be the set consisting of one or two common neighbors of t_j and t_{j+1}. If $U \backslash Q$ is nonempty, we can insert any point $t \in U \backslash Q$ between t_j and t_{j+1}. Otherwise, we can assume without loss of generality that Q contains a subpath $Q' = \{t_i \mid i = j+1, \ldots, k\}$ such that $Q' \cap U = \{t_k\}$. We can remove from Q all the points of Q' except t_k. In each case we obtain a minimal grid path of even length, which is a contradiction to Fact 1.

Proof of Theorem. Let $P_q,\ P_r,\ P_s$ be the paths in $P(q, r, s)$ and $g_1,\ g_2$ be their common end vertices. Let $R = (x_1, y_1),\ x_1, y_1 \geqslant 2$, denote the rectangle on the grid defined by the grid points (1,1) and (x_1, y_1). If $x_1 + y_1 - 2 = (i + j)/2$ for some paths $P_i, P_j,\ i, j \in Z := \{q, r, s\}$, then we say that $R = (x_1, y_1)$ is *compatible* with P_i, P_j. For paths $P_i, P_j,\ i, j \in Z$, and a rectangle R compatible with $P_i,\ P_j$, we let $B(P_i, P_j, R; x_2, y_2; x_3, y_3)$ denote the assignment of the vertices of P_i and P_j to the grid points located on the boundary of R such that the vertex g_1 is assigned to $(x_2, y_2), g_2$ to (x_3, y_3), and both the resulting grid paths are minimal (see Fig. 4a for an example; in this and the rest figures only relevant grid points are shown, i.e., those to which the vertices of $P(q,r,s)$ are assigned). If $R=(x_1,y_1)$ is compatible with $P_i, P_j,\ i,j \in Z$, and $B(P_i,P_j,R;x_2,y_2;x_3,y_3)$ is an assignment of P_i and P_j, then we can extend $B(P_i,P_j,R;x_2,y_2;x_3,y_3)$ to an assignment of $P(q,r,s)$ by assigning the internal vertices of P_k, $\{k\}=Z\backslash\{i,j\}$, taken in the direction from g_1 to g_2, to points $(0,l)$, $l=y_1,\dots,y_1-k+2$. We will write $B'(P_i,P_j,R;x_2,y_2;x_3,y_3)$ for this assignment. An illustration of how the assignment of Fig. 4a can be extended is shown in Fig. 4b.

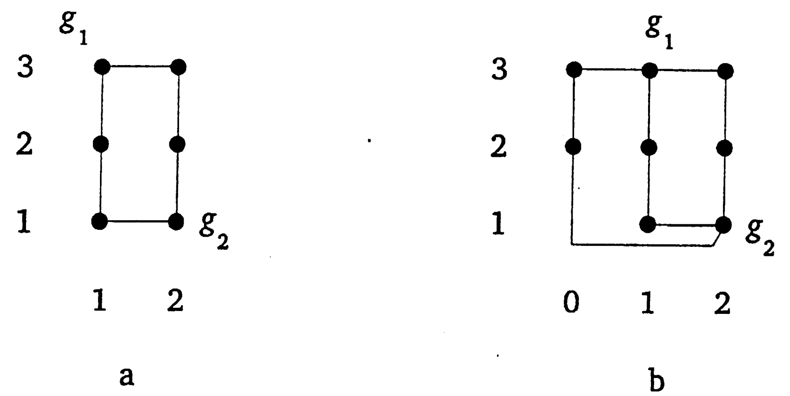


Fig. 4. Assignments of paths (bullets denote grid points, and lines – the edges of $P(3,3,3)$): a – $B(P_3,P_3,R;1,3;2,1)$, $R=(2,3)$; b – $B'(P_3,P_3,R;1,3;2,1)$.

We distinguish between the following three cases.

Case 1. $2 \leqslant q=r=s$. Suppose that $\Delta(P(q,r,s))<2$. Then, by Fact 2, $\Delta(P(q,r,s))=0$. This implies that P_q must belong to some rectangle compatible with P_r, P_s. Let l be the number of grid points strictly inside such a rectangle. It is easy to see that $l \leqslant 4$ if $q=6$, $l \leqslant 2$ if $q=5$, $l \leqslant 1$ if $q=4$, and $l=0$ if $q \leqslant 3$. In each case this number is too small to construct the third minimal grid path of length q, a contradiction. Thus for $q \leqslant 6$, $\Delta(P(q,r,s)) \geqslant 2$. On the other hand, the assignment $B'(P_r,P_s,R;1,q;2,1)$, $R=(2,q)$, (for $q=3$ given in Fig. 4b) shows that $\Delta(P(q,r,s)) \leqslant 2$. Hence $\Delta(P(q,r,s))=2$ if $q \leqslant 6$.

For $q \geqslant 7$ we can take $B(P_r,P_s,R;\lfloor q/4\rfloor+1,\ \lceil q/2\rceil+1;\ \lfloor (q+2)/4\rfloor+1,1)$, $R=(\lfloor q/2\rfloor+1,\lceil q/2\rceil+1)$, and assign the internal vertices of P_q to the grid points inside R. For instance, in the most stringent case, that is for $q = 7$, the following grid points are used (in the direction from g_1 to g_2): $(2,4),(3,4),(3,3),(2,3),(2,2),(3,2)$. The corresponding assignment is shown in Fig. 5. Thus $\Delta(P(q,r,s)) = 0$ if $q \geqslant 7$.

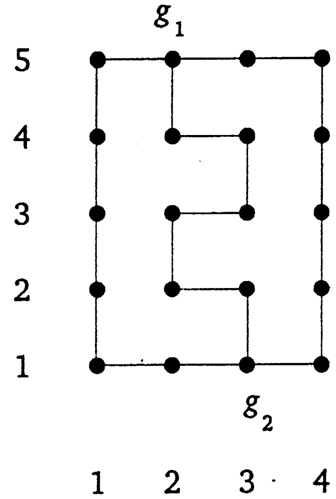


Fig. 5. Assignment of $P(7,7,7)$.

Case 2. $2 \leqslant q = r < s$. If $s = q+1$, then $B'(P_q, P_r, R;\ 1,q;\ 2,1)$, $R = (2,q)$, shows that $\Delta(P(q,r,s)) \leqslant 1$. If $s = q+1+2i$, $i \geqslant 1$, then we can apply the path shifting operation with respect to P_s in $B'(P_q, P_r, R;\ 1,q; 2,1)$ which consists of replacing the points $(0,j)$, $j = 1,\dots,q$, by $(-i,j)$, $j = 1,\dots,q$, and adding $(j,1)$, (j,q), $j = -i+1,\dots,0$, as new grid points for P_s. Fig. 6a illustrates this operation. Therefore, $\Delta(P(q,r,s)) \leqslant 1$ once again.

Now suppose $s = q+2$. If $q \leqslant 4$, then no rectangle compatible with P_r, P_s can contain $q-1$ or more points inside. Thus as in Case 1, $\Delta(P(q,r,s)) \geqslant 2$. The equality is provided by $B'(P_r, P_s, R;\ 1,q+1;\ 1,1)$, $R = (2,q+1)$. If $q \geqslant 5$, we can take $B(P_r, P_s, R;\ 2,q-1;\ 2,1)$, $R = (4,q-1)$, and realize P_q inside R showing that $\Delta(P(q,r,s)) = 0$ (as illustrated in Fig. 6b). If $s = q+2i$, $i \geqslant 2$, we obtain a similar situation. In this case we can consider $B(P_r, P_s, R;\ 2,q+1;\ 1,2)$, $R = (i+1,q+1)$ (see example in Fig. 6c).

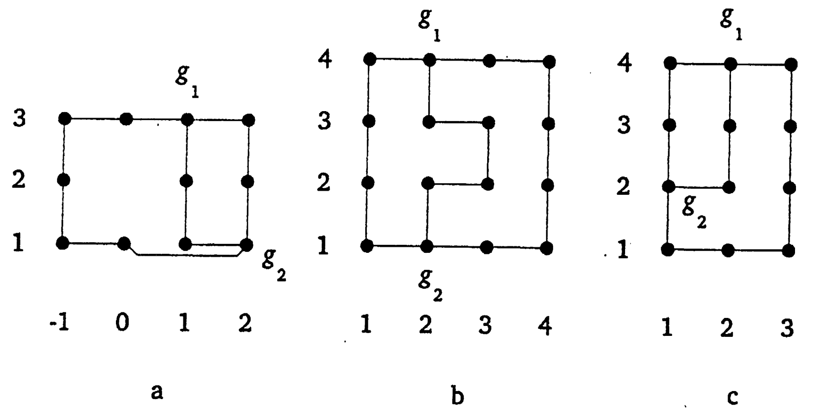


Fig. 6. Assignments for the case when $q = r < s$: a – assignment of $P(3,3,6)$; b – assignment of $P(5,5,7)$; c – assignment of $P(3,3,7)$.

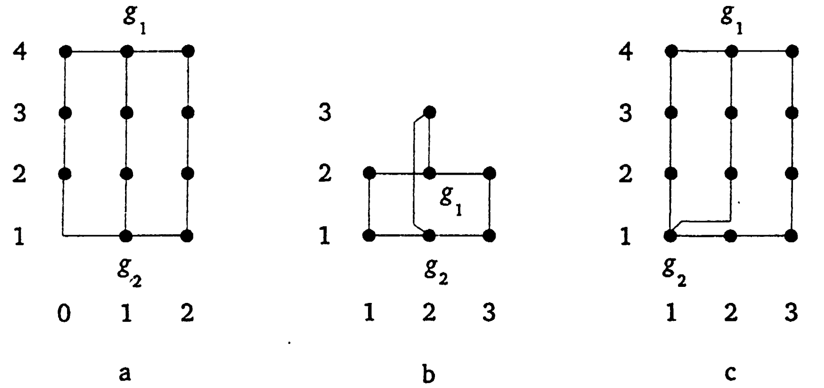


Fig. 7. Assignments for the case when $q < r$: a – assignment of $P(3,4,5)$; b – assignment of $P(2,3,3)$; c – assignment of $P(3,4,6)$.

Case 3. $1 \leqslant q < r \leqslant s$. Suppose $s-q = i$, i even. If $r-q \leqslant 2$, the assignment $B'(P_q, P_s, R;\ 1, q+1;\ 1,1)$, $R = (i/2+1, q+1)$, gives $\Delta(P(q,r,s)) \leqslant 1$ (see Fig. 7a). If $r - q > 2$, we can apply the shifting operation as described in Case 2 to get the same bound. Similarly, $\Delta(P(q,r,s)) \leqslant 1$ if $r-q = j$, j even.

Thus we may assume that both $i = s - q$ and $j = r - q$ are odd. Suppose $i = j = 1$. If $q \geqslant 5$, we can consider an assignment similar to that of Case 1 for $q \geqslant 7$: we only replace q by $q+1$. Since P_q can be realized inside the rectangle R related to this assignment, $\Delta(P(q,r,s)) = 1$. If $q \leqslant 4$, we can take the assignment $B(P_r, P_s, R;\ q, 2;\ 2, 1)$ or, if $q = 1, B(P_r, P_s, R; 2, 2;\ 1, 1)$ with $R = (q+1, 2)$ and assign the internal vertices of P_q to $(k, 3)$, $k = 2, \dots, q$. For $q = 2$, the assignment appears in Fig. 7b. In each case, $\Delta(P(q,r,s)) = 1$.

Finally, suppose $i + j \geqslant 4$. If $j = 1$, we obtain $\Delta(P(q,r,s)) = 1$ by considering the assignment $B(P_r, P_s, R;\ 2, q+1;\ 1, 1)$, $R = ((i+1)/2 + 1, q+1)$, and observing that there is sufficient room for P_q in R (as illustrated in Fig. 7c). If $j \geqslant 3$, we can take the above assignment and apply (with respect to P_r) a shifting operation similar to that defined in Case 2. Thus $\Delta(P(q,r,s)) = 1$ for all odd i and j. The proof is complete.

Acknowledgement. The author thanks an anonymous referee for comments and valuable suggestions that led to improvements in the presentation of the paper.

REFERENCES

Adams, W.P., and T.A. Johnson (1994). Improved linear programming-based lower bounds for the quadratic assignment problem. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science,* **16**, 43–75.

Burkard, R.E. (1984). Quadratic assignment problems. *European Journal of Operational Research,* **15**(3), 283–289.

Chakrapani, J., and J. Skorin-Kapov (1994). A constructive method for improving lower bounds for a class of quadratic assignment problems. *Operations Research,* **42**(5), 837–845.

Finke, G., R.E. Burkard, F. Rendl (1987). Quadratic assignment problems. *Annals of Discrete Mathematics,* **31**, 61–82.

Frieze, A.M., and J. Yadegar (1983). On the quadratic assignment problem. *Discrete Applied Mathematics,* **5**(1), 89–98.

Gilmore, P.C. (1962). Optimal and suboptimal algorithms for the quadratic assignment problem. *J. SIAM,* **10**(2), 305–313.

Glover, F. (1989). Tabu search, Part I. *ORSA Journal on Computing,* **1**(3), 190–206.

Glover, F. (1990). Tabu search, Part II. *ORSA Journal on Computing,* **2**(1), 4–32.

Hadley, S.W., F. Rendl, H. Wolkowicz (1992). A new lower bound via projection for the quadratic assignment problem. *Mathematics of Operations Research,* **17**(3), 727–739.

Lawler, E.L. (1963). The quadratic assignment problem. *Management Science,* **9**, 586–599.

Nugent, C.E., T.E. Vollmann, J. Ruml (1968). An experimental comparison of techniques for the assignment of facilities to locations. *Operations Research,* **16**(1), 150–173.

Palubeckis, G.S. (1988). A generator of test quadratic assignment problems with known optimal solution. *USSR. Comput. Maths. Math. Phys.,* **28**(6), 97–98.

Papadimitriou, C.H., and K. Steiglitz (1982). *Combinatorial Optimization: Algorithms and Complexity.* Prentice-Hall, Englewood Cliffs, New Jersey.

Rendl, F., and H. Wolkowicz (1992). Applications of parametric programming and eigenvalue maximization to the quadratic assignment problem. *Mathematical Programming,* **53**(1), 63–78.

Skorin-Kapov, J. (1990). Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing,* **2**(1), 33–45.

Steinberg, L. (1961). The backboard wiring problem: a placement algorithm. *SIAM Rev.,* **3**(1), 37–50.

Received July 1997

G. Palubeckis received the degree of Doctor of Technical Sciences from the Kaunas Polytechnic Institute, Kaunas, Lithuania, in 1987. Currently, he is an Associate Professor at the Kaunas University of Technology. His major research interests are in graph theory, combinatorial optimization, and computer-aided design.

SPECIALIŲJŲ GRAFŲ PANAUDOJIMAS APATINIŲ RIBŲ GAVIMUI GEOMETRINIAME KVADRATINIO PASKIRSTYMO UŽDAVINYJE

Gintaras PALUBECKIS

Šiame straipsnyje apibrėžiama klasė, susidedanti iš grafų su briaunoms priskirtais svoriais, turinčių tokią savybę: grafo kiekvieno suskaidymo į dvi dalis vertė yra neneigiama.

Eksperimentiškai yra parodoma, kad pilni tokie grafai, turintys daugiau negu tris viršūnes, o taip pat tam tikri specialūs grafai, kurių visos briaunos teigiamos, gali būti panaudoti egzistuojančių apatinių ribų pagerinimui kvadratinio paskirstymo uždavinio versijai, kuomet viena iš matricų yra sudaryta iš atstumų stačiakampėje metrikoje tarp Euklidinės erdvės taškų.