Journal of Global Optimization **10:** 391–403, 1997.

QAPLIB – A Quadratic Assignment Problem Library*

RAINER E. BURKARD[1], STEFAN E. KARISCH[2] and FRANZ RENDL[1]
[1]*Department of Mathematics, Graz University of Technology, Steyrergasse 30, A-8010 Graz, Austria,*
[2]*Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2100 Copenhagen, Denmark*

(Received: 3 May 1996; accepted: 16 September 1996)

Abstract. A collection of electronically available data instances for the Quadratic Assignment Problem is described. For each instance, we provide detailed information, indicating whether or not the problem is solved to optimality. If not, we supply the best known bounds for the problem. Moreover we survey available software and describe recent dissertations related to the Quadratic Assignment Problem.

Key words: Quadratic assignment problem, data instances, problem library.

1. Information

CONTACT ADDRESSES:

Stefan E. Karisch, Department of Computer Science, University of Copenhagen, Universitetsparken 1, DK-2100 Copenhagen, Denmark; E-Mail:
`karisch@diku.dk`

Franz Rendl, Department of Mathematics, Graz University of Technology, Steyrergasse 30, A-8010 Graz, Austria; E-Mail:
`rendl@opt.math.tu-graz.ac.at`

QAPLIB HOME PAGE:

This report, the data and also most of the best feasible solutions are available via World Wide Web. The URLs of the QAPLIB Home Page are
`http://www.opt.math.tu-graz.ac.at/~karisch/qaplib` and
`http://www.diku.dk/~karisch/qaplib`.

* September 1996. Updated version of "QAPLIB – A Quadratic Assignment Problem Library. *European Journal of Operational Research*, 55: 115–119, 1991".

2. Introduction

The *Quadratic Assignment Problem* (QAP) has remained one of the great challenges in combinatorial optimization. It is still considered a computationally non-trivial task to solve modest size problems, say of size $n = 20$. The QAPLIB was first published in 1991, in order to provide a unified testbed for QAP, accessible to the scientific community. It consisted of virtually all QAP instances that were accessible to us at that time. Due to the continuing demand for these instances, and the strong feedback from many researchers, we provided a major update in 1994, which was also accessible through anonymous ftp. In this update we also included many new problem instances, generated by several researchers for their own testing purposes. Moreover, we included a list of current champions, i.e. best known feasible solutions, and best lower bounds.

The current update reflects on one hand the big changes in electronic communication. It has become a World Wide Web site, the QAPLIB Home Page. The online version will be updated on a regular basis and also contains most of the currently best known permutations. On the other hand, we feel the update was necessary, due to the increased research activities around the QAP, carried out in the last years. Therefore we also include a short list of dissertations concerning QAP, which were written in the last few years.

3. Problem Instances

In this section we describe in some detail all the problem instances currently included in the QAPLIB. We have removed all the instances of size $n < 12$, because these can be solved quite efficiently by current state of the art programs. On the other hand, we included several larger instances, the largest one of size $n = 256$.

The instances are listed in alphabetical order by the names of their authors or contributors. We shortly characterize the examples by indicating their generation. All the instances in the current version are pure quadratic. If not stated otherwise the examples are symmetric.

The format of the problem data whose filenames have extension “dat” is

$$\begin{array}{l} n \\ A \\ B \end{array}$$

where n is the size of the instance, and A and B are either flow or distance matrix. This corresponds to a QAP of the form

$$\min_p \sum_{i=1}^n \sum_{j=1}^n a_{ij} b_{p(i),p(j)}$$

where p is a permutation.

We quote the filename under which it is stored in the library and report the size of the problem. Then the objective function value of the best known feasible solution is given. In parentheses we indicate whether this solution is provably optimal. Otherwise we indicate, by which heuristic the solution was found. The heuristics that are currently considered are

- genetic hybrids: (GEN) [13] and (GEN-2) [29],
- a greedy randomized adaptive search procedure: (GRASP) [25],
- simulated annealing: (SIM-1) [7] and (SIM-2) [41], and
- tabu search: reactive tabu search (Re-TS) [1], robust tabu search (Ro-TS) [39, 40], and strict tabu search (S-TS) [37].

If available we provide permutations corresponding to the feasible solutions in the QAPLIB Home Page. The files for these solutions have extension "sln" and their format is

$n \;\; sol$

p

where n gives the size of the instance, sol is the objective function value and p a corresponding permutation, i.e.

$$sol = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{p(i),p(j)}.$$

For problems solved to optimality, we enclose the optimal permutation. Otherwise we include the currently best known lower bounds. We also give explicit reference for who solved hard instances of size $n \geq 16$ first. The lower bounds given in the tables are

- the elimination bound: (ELI) [15],
- the Gilmore–Lawler bound: (GLB) [14, 22],
- an interior point based linear programming bound: (IPLP) [33]
- a semidefinite programming bound: (SDP) [18, 20], and
- a triangle decomposition bound: (TDB) [19].

When lower bounds are included we also give the relative gap between best feasible solution and best known lower bound in percent, i.e. gap $=$ $($solution $-$ bound$)/($solution$) * 100$ %. We also note that GLB can be calculated routinely for all instances of the QAPLIB. The bound ELI is only valid for symmetric instances. It can also be computed efficiently for all symmetric instances, but its computation time is (by a constant factor) higher than the time to compute GLB. The bound TDB can be applied only to instances where the distance matrix has a metric structure. It can be calculated efficiently for all metric instances in the QAPLIB. Finally, IPLP and SDP produce in general very strong bounds, but the computational effort by far outgrows the computation times for the other bounds. Currently, these bounds can not be considered efficient for problems of sizes larger than, say $n = 30$.

R. E. BURKARD AND J. OFFERMANN [6]

The data of the first matrix correspond to the typing-time of an average stenotypist, while the second matrix describes the frequency of pairs of letters in different languages taken over 100,000 pairs for examples a-f and over 187,778 pairs for examples g-h. (Note that the solutions of the latter instances are not scaled for a flow matrix of 100,000 pairs any more.) One also distinguishes between two types of typewriter keyboards. The instances are asymmetric.

name	n	feas. solution	bound	gap
Bur26a	26	5426670 (GRASP)	5334208 (IPLP)	1.71%
Bur26b	26	3817852 (GRASP)	3736954 (IPLP)	2.12%
Bur26c	26	5426795 (GRASP)	5359110 (IPLP)	1.25%
Bur26d	26	3821225 (GRASP)	3705831 (IPLP)	3.03%
Bur26e	26	5386879 (GRASP)	5315311 (IPLP)	1.33%
Bur26f	26	3782044 (GRASP)	3712627 (IPLP)	1.84%
Bur26g	26	10117172 (GRASP)	10047627 (IPLP)	0.69%
Bur26h	26	7098658 (GRASP)	7036448 (IPLP)	0.88%

N. CHRISTOFIDES AND E. BENAVENT [9]

One matrix is the adjacency matrix of a weighted tree the other that of a complete graph.

name	n	feas. solution	permutation
Chr12a	12	9552 (OPT)	$p^* = (7, 5, 12, 2, 1, 3, 9, 11, 10, 6, 8, 4)$
Chr12b	12	9742 (OPT)	$p^* = (5, 7, 1, 10, 11, 3, 4, 2, 9, 6, 12, 8)$
Chr12c	12	11156 (OPT)	$p^* = (7, 5, 1, 3, 10, 4, 8, 6, 9, 11, 2, 12)$
Chr15a	15	9896 (OPT)	$p^* = (5, 10, 8, 13, 12, 11, 14, 2, 4, 6, 7, 15, 3, 1, 9)$
Chr15b	15	7990 (OPT)	$p^* = (4, 13, 15, 1, 9, 2, 5, 12, 6, 14, 7, 3, 10, 11, 8)$
Chr15c	15	9504 (OPT)	$p^* = (13, 2, 5, 7, 8, 1, 14, 6, 4, 3, 15, 9, 12, 11, 10)$
Chr18a	18	11098 (OPT)	$p^* = (3, 13, 6, 4, 18, 12, 10, 5, 1, 11, 8, 7, 17, 14, 9, 16, 15, 2)$
Chr18b	18	1534 (OPT)	$p^* = (1, 2, 4, 3, 5, 6, 8, 9, 7, 12, 10, 11, 13, 14, 16, 15, 17, 18)$
Chr20a	20	2192 (OPT)	$p^* = (3, 20, 7, 18, 9, 12, 19, 4, 10, 11, 1, 6, 15, 8, 2, 5, 14, 16, 13, 17)$
Chr20b	20	2298 (OPT)	$p^* = (20, 3, 9, 7, 1, 12, 16, 6, 8, 14, 10, 4, 5, 13, 17, 2, 18, 11, 19, 15)$
Chr20c	20	14142 (OPT)	$p^* = (12, 6, 9, 2, 10, 11, 3, 4, 15, 18, 7, 13, 16, 5, 14, 17, 19, 1, 8, 20)$
Chr22a	22	6156 (OPT)	$p^* = (15, 2, 21, 8, 16, 1, 7, 18, 14, 13, 5, 17, 6, 11, 3, 4, 20, 19, 9, 22, 10, 12)$
Chr22b	22	6194 (OPT)	$p^* = (10, 19, 3, 1, 20, 2, 6, 4, 7, 8, 17, 12, 11, 15, 21, 13, 9, 5, 22, 14, 18, 16)$
Chr25a	25	3796 (OPT)	$p^* = (25, 12, 5, 3, 18, 4, 16, 8, 20, 10, 14, 6, 15, 23, 24, 19, 13, 1, 21, 11, 17, 2, 22, 7, 9)$

A. N. ELSHAFEI [11]

The data describe the distances of 19 different facilities of a hospital and the flow of patients between those.

name	n	feas. solution	permutation
Els19	19	17212548 (OPT)[27]	$p^* = (9, 10, 7, 18, 14, 19, 13, 17, 6, 11, 4, 5, 12, 8, 15, 16, 1, 2, 3)$

B. ESCHERMANN AND H. J. WUNDERLICH [12]

These examples stem from an application in computer science, from the testing of self-testable sequential circuits. The amount of additional hardware for the testing should be minimized. (Note that problem instance Esc16f was removed from QAPLIB.)

name	n	feas. solution	permutation/bound	gap
Esc16a	16	68 (OPT)[10]	$p^* = (2, 14, 10, 16, 5, 3, 7, 8, 4, 6, 12, 11, 15, 13, 9, 1)$	–
Esc16b	16	292 (OPT)[10]	$p^* = (6, 3, 7, 5, 13, 1, 15, 2, 4, 11, 9, 14, 10, 12, 8, 16)$	–
Esc16c	16	160 (OPT)[10]	$p^* = (11, 14, 10, 16, 12, 8, 9, 3, 13, 6, 5, 7, 15, 2, 1, 4)$	–
Esc16d	16	16 (OPT)[10]	$p^* = (14, 2, 12, 5, 6, 16, 8, 10, 3, 9, 13, 7, 11, 15, 4, 1)$	–
Esc16e	16	28 (OPT)[10]	$p^* = (16, 7, 8, 15, 9, 12, 14, 10, 11, 2, 6, 5, 13, 4, 3, 1)$	–
Esc16g	16	26 (OPT)[10]	$p^* = (8, 11, 9, 12, 15, 16, 14, 10, 7, 6, 2, 5, 13, 4, 3, 1)$	–
Esc16h	16	996 (OPT)[10]	$p^* = (13, 9, 10, 15, 3, 11, 4, 16, 12, 7, 8, 5, 6, 2, 1, 14)$	–
Esc16i	16	14 (OPT)[10]	$p^* = (13, 9, 11, 3, 7, 5, 6, 2, 1, 15, 4, 14, 12, 10, 8, 16)$	–
Esc16j	16	8 (OPT)[10]	$p^* = (8, 3, 16, 14, 2, 12, 10, 6, 9, 5, 13, 11, 4, 7, 15, 1)$	–
Esc32a	32	130 (Ro-TS)	35 (GLB)	73.08%
Esc32b	32	168 (Ro-TS)	96 (GLB)	42.86%
Esc32c	32	642 (SIM-1)	464 (ELI)	27.73%
Esc32d	32	200 (Ro-TS)	106 (GLB)	47.00%
Esc32e	32	2 (OPT)[2]	$p^* = (1, 2, 5, 6, 8, 16, 13, 19, 9, 32, 7, 22, 24, 20, 4, 12, 3, 17, 29, 21, 11, 25, 27, 18, 30, 31, 23, 28, 14, 15, 26, 10)$	–
Esc32f	32	2 (OPT)[2]	$p^* = (1, 2, 5, 6, 8, 16, 10, 7, 9, 28, 30, 4, 32, 31, 22, 12, 3, 17, 26, 18, 13, 25, 29, 21, 23, 24, 19, 20, 14, 15, 27, 11)$	–
Esc32g	32	6 (SIM-1)	0 (GLB)	100.00%
Esc32h	32	438 (Ro-TS)	257 (GLB)	41.33%
Esc64a	64	116 (SIM-1)	47 (GLB)	59.49%
Esc128	128	64 (GRASP)	2 (GLB)	96.86%

S. W. HADLEY, F. RENDL AND H. WOLKOWICZ [15]

The first matrix represents Manhattan distances of a connected cellular complex in the plane while the entries in the flow matrix are drawn uniformly from the interval $[1, n]$.

name	n	feas. solution	permutation
Had12	12	1652 (OPT)	$p^* = (3, 10, 11, 2, 12, 5, 6, 7, 8, 1, 4, 9)$
Had14	14	2724 (OPT)	$p^* = (8, 13, 10, 5, 12, 11, 2, 14, 3, 6, 7, 1, 9, 4)$
Had16	16	3720 (OPT)[16]	$p^* = (9, 4, 16, 1, 7, 8, 6, 14, 15, 11, 12, 10, 5, 3, 2, 13)$
Had18	18	5358 (OPT)[2]	$p^* = (8, 15, 16, 6, 7, 18, 14, 11, 1, 10, 12, 5, 3, 13, 2, 17, 9, 4)$
Had20	20	6922 (OPT)[2]	$p^* = (8, 15, 16, 14, 19, 6, 7, 17, 1, 12, 10, 11, 5, 20, 2, 3, 4, 9, 18, 13)$

J. KRARUP AND P. M. PRUZAN [21]

The instances contain real world data and were used to plan the Klinikum Regensburg in Germany.

name	n	feas. solution	bound	gap
Kra30a	30	88900 (S-TS)	76003 (IPLP)	14.51%
Kra30b	30	91420 (Ro-TS)	76752 (IPLP)	16.05%

Y. LI AND P. M. PARDALOS [24]

These instances come from problem generators described in [24]. The generators provide asymmetric instances with known optimal solutions.

name	n	feas. solution
Lipa20a	20	3683 (OPT)
Lipa20b	20	27076 (OPT)
Lipa30a	30	13178 (OPT)
Lipa30b	30	151426 (OPT)
Lipa40a	40	31538 (OPT)
Lipa40b	40	476581 (OPT)
Lipa50a	50	62093 (OPT)
Lipa50b	50	1210244 (OPT)
Lipa60a	60	107218 (OPT)
Lipa60b	60	2520135 (OPT)
Lipa70a	70	169755 (OPT)
Lipa70b	70	4603200 (OPT)
Lipa80a	80	253195 (OPT)
Lipa80b	80	7783962 (OPT)
Lipa90a	90	360630 (OPT)
Lipa90b	90	12490441 (OPT)

C. E. NUGENT, T. E. VOLLMANN AND J. RUML [28]

The following problem instances are probably the most frequently used. The distance matrix contains Manhattan distances of rectangular grids. The instances of size $n \in \{14, 16, 17, 18, 21, 22, 24, 25\}$ were constructed out of the larger ones by deleting certain rows and columns, see Clausen and Perregaard [10].

name	n	feas. solution	permutation/bound	gap
Nug12	12	578 (OPT)	$p^* = (12, 7, 9, 3, 4, 8, 11, 1, 5, 6, 10, 2)$	–
Nug14	14	1014 (OPT)	$p^* = (9, 8, 13, 2, 1, 11, 7, 14, 3, 4, 12, 5, 6, 10)$	–
Nug15	15	1150 (OPT)	$p^* = (1, 2, 13, 8, 9, 4, 3, 14, 7, 11, 10, 15, 6, 5, 12)$	–
Nug16a	16	1610 (OPT)[10]	$p^* = (9, 14, 2, 15, 16, 3, 10, 12, 8, 11, 6, 5, 7, 1, 4, 13)$	–
Nug16b	16	1240 (OPT)[10]	$p^* = (16, 12, 13, 8, 4, 2, 9, 11, 15, 10, 7, 3, 14, 6, 1, 5)$	–
Nug17	17	1732 (OPT)[10]	$p^* = (16, 15, 2, 14, 9, 11, 8, 12, 10, 3, 4, 1, 7, 6, 13, 17, 5)$	–
Nug18	18	1930 (OPT)[10]	$p^* = (10, 3, 14, 2, 18, 6, 7, 12, 15, 4, 5, 1, 11, 8, 17, 13, 9, 16)$	–
Nug20	20	2570 (OPT)[10]	$p^* = (18, 14, 10, 3, 9, 4, 2, 12, 11, 16, 19, 15, 20, 8, 13, 17, 5, 7, 1, 6)$	–
Nug21	21	2438 (OPT)[2]	$p^* = (4, 21, 3, 9, 13, 2, 5, 14, 18, 11, 16, 10, 6, 15, 20, 19, 8, 7, 1, 12, 17)$	–
Nug22	22	3596 (OPT)[2]	$p^* = (2, 21, 9, 10, 7, 3, 1, 19, 8, 20, 17, 5, 13, 6, 12, 16, 11, 22, 18, 4, 14, 15)$	–
Nug24	24	3488 (SIM-1)	3251 (TDB)	6.80%
Nug25	25	3744 (SIM-1)	3486 (TDB)	6.89%
Nug30	30	6124 (S-TS)	5772 (TDB)	5.75%

C. ROUCAIROL [35]

The entries of the matrices are chosen from the interval $[1, 100]$.

name	n	feas. solution	permutation
Rou12	12	235528 (OPT)	$p^* = (6, 5, 11, 9, 2, 8, 3, 1, 12, 7, 4, 10)$
Rou15	15	354210 (OPT)	$p^* = (12, 6, 8, 13, 5, 3, 15, 2, 7, 1, 9, 10, 4, 14, 11)$
Rou20	20	725522 (OPT)[2]	$p^* = (1, 19, 2, 14, 10, 16, 11, 20, 9, 5, 7, 4, 8, 18, 15, 3, 12, 17, 13, 6)$

M. SCRIABIN AND R. C. VERGIN [36]

The distances of these problems are rectangular.

name	n	feas. solution	permutation
Scr12	12	31410 (OPT)	$p^* = (8, 6, 3, 2, 10, 1, 5, 9, 4, 7, 12, 11)$
Scr15	15	51140 (OPT)	$p^* = (15, 7, 11, 8, 1, 4, 3, 2, 12, 6, 13, 5, 14, 10, 9)$
Scr20	20	110030 (OPT)[27]	$p^* = (20, 7, 12, 6, 4, 8, 3, 2, 14, 11, 18, 9, 19, 15, 16, 17, 13, 5, 10, 1)$

J. SKORIN-KAPOV [37]

The distances of these problems are rectangular and the entries of the flow matrices are pseudorandom numbers.

name	n	feas. solution	bound	gap
Sko42	42	15812 (Ro-TS)	14934 (TDB)	5.56%
Sko49	49	23386 (Ro-TS)	22004 (TDB)	5.91%
Sko56	56	34458 (Ro-TS)	32610 (TDB)	5.37%
Sko64	64	48498 (Ro-TS)	45736 (TDB)	5.70%
Sko72	72	66256 (Ro-TS)	62691 (TDB)	5.38%
Sko81	81	90998 (GEN)	86072 (TDB)	5.41%
Sko90	90	115534 (Ro-TS)	108493 (TDB)	6.10%
Sko100a	100	152002 (GEN)	142668 (TDB)	6.14%
Sko100b	100	153890 (GEN)	143872 (TDB)	6.51%
Sko100c	100	147862 (GEN)	139402 (TDB)	5.73%
Sko100d	100	149576 (GEN)	139898 (TDB)	6.47%
Sko100e	100	149150 (GEN)	140105 (TDB)	6.07%
Sko100f	100	149036 (GEN)	139452 (TDB)	6.43%

L. STEINBERG [38]

The three instances model the backboard wiring problem. The distances in the first one are Manhattan, in the second squared Euclidean, and in the third one Euclidean distances.

name	n	feas. solution	bound	qap
Ste36a	36	9526 (Ro-TS)	7124 (GLB)	25.22%
Ste36b	36	15852 (S-TS)	8653 (GLB)	45.42%
Ste36c	36	8239.11 (Ro-TS)	6393.63 (GLB)	22.40%

É. D. TAILLARD [39, 40]

The instances Taixxa are uniformly generated and were proposed in [39]. The other problems were introduced in [40]. Problems Taixxb are asymmetric and randomly generated. Instances Taixxc occur in the generation of grey patterns.

name	n	feas. solution	permutation/bound	gap
Tai12a	12	224416 (OPT)	$p^* = (8, 1, 6, 2, 11, 10, 3, 5, 9, 7, 12, 4)$	–
Tai12b	12	39464925 (OPT)	$p^* = (9, 4, 6, 3, 11, 7, 12, 2, 8, 10, 1, 5)$	–
Tai15a	15	388214 (OPT)	$p^* = (5, 10, 4, 13, 2, 9, 1, 11, 12, 14, 7, 15, 3, 8, 6)$	–
Tai15b	15	51765268 (OPT)	$p^* = (1, 9, 4, 6, 8, 15, 7, 11, 3, 5, 2, 14, 13, 12, 10)$	–
Tai17a	17	491812 (OPT)[2]	$p^* = (12, 2, 6, 7, 4, 8, 14, 5, 11, 3, 16, 13, 17, 9, 1, 10, 15)$	–
Tai20a	20	703482 (OPT)[2]	$p^* = (10, 9, 12, 20, 19, 3, 14, 6, 17, 11, 5, 7, 15, 16, 18, 2, 4, 8, 13, 1)$	–
Tai20b	20	122455319 (Ro-TS)	14857089 (GLB)	87.87%
Tai25a	25	1167256 (Ro-TS)	962417 (GLB)	17.55%
Tai25b	25	344355646 (Ro-TS)	51401950 (GLB)	85.08%
Tai30a	30	1818146 (Ro-TS)	1504688 (GLB)	17.25%
Tai30b	30	637117113 (Ro-TS)	40947945 (GLB)	93.58%
Tai35a	35	2422002 (Ro-TS)	1951207 (GLB)	19.44%
Tai35b	35	283315445 (Ro-TS)	32611838 (GLB)	88.49%
Tai40a	40	3139370 (Re-TS)	2492850 (GLB)	20.60%
Tai40b	40	637250948 (Ro-TS)	46143753 (GLB)	92.77%
Tai50a	50	4941410 (GEN)	3854359 (GLB)	22.00%
Tai50b	50	458821517 (Ro-TS)	40296004 (GLB)	91.23%
Tai60a	60	7208572 (Ro-TS)	5555095 (GLB)	22.94%
Tai60b	60	608215054 (Ro-TS)	50113782 (GLB)	91.77%
Tai64c	64	1855928 (Ro-TS)	896398 (ELI)	51.71%
Tai80a	80	13557864 (Ro-TS)	10329674 (GLB)	23.82%
Tai80b	80	818415043 (Ro-TS)	89169828 (GLB)	89.11%
Tai100a	100	21125314 (Re-TS)	15824355 (GLB)	25.10%
Tai100b	100	1185996137 (Ro-TS)	174687926 (GLB)	86.28%
Tai150b	150	499348972 (Ro-TS)	63007151 (GLB)	87.39%
Tai256c	256	44919020 (GEN-2)	41291996 (ELI)	8.08%

U. W. THONEMANN AND A. BÖLTE [41]

The distances of these instances are rectangular.

name	n	feas. solution	bound	gap
Tho30	30	149936 (SIM-2)	136447 (TDB)	9.00%
Tho40	40	240516 (SIM-2)	214218 (TDB)	10.94%
Tho150	150	8134030 (GEN)	7620628 (TDB)	6.32%

M. R. WILHELM AND T. L. WARD [42]

The distances of these problems are rectangular.

name	n	feas. solution	bound	gap
Wil50	50	48816 (SIM-2)	47098 (TDB)	3.52%
Wil100	100	273038 (GEN)	263909 (TDB)	3.35%

4. Surveys and Dissertations Concerning QAP since 1990

SURVEYS

R. E. Burkard and E. Çela provide the most recent survey on QAP [4]. Their paper is an annotated bibliography on all aspects of the QAP. Another recent survey on QAP is due to P. M. Pardalos, F. Rendl and H. Wolkowicz [30]. It appeared in 1994 in a proceedings book of the DIMACS workshop on QAP edited by P. M. Pardalos and H. Wolkowicz [31]. R. E. Burkard reviews the QAP in the context of facility location in the survey paper [3].

DISSERTATIONS

The following list of dissertations considering the quadratic assignment problem shows that there is still a broad interest in this difficult combinatorial optimization problem. Even though there has not been substantial improvement regarding the solvability of larger problem instances, these dissertations contain many ideas which are certainly a strong foundation for successful future work on QAP.

E. Çela [8] investigated the computational complexity of specially structured quadratic assignment problems. Moreover, she considered a generalization of QAP, the so called biquadratic assignment problem.

T. A. Johnson [17] introduced solution procedures based on linear programming. The linear formulation derived in her thesis theoretically dominates alternate linear formulations for QAP.

S. E. Karisch [18] presented nonlinear approaches for QAP. These provide the currently strongest lower bounds for problems instances whose distance matrix contains distances of a rectangular grid and for smaller sized general problems.

Y. Li [23] introduced beside other ideas lower bounding techniques based on reductions, GRASP and a problem generator for QAP.

F. Malucelli [26] proposed a lower bounding technique for QAP based on a reformulation scheme and implemented it in a branch and bound code. Some new applications of QAP in the field of transportation were also presented.

T. Mautor [27] focused on parallel implementations and exploited the metric structure of the Nugent instances to reduce the branching tree considerably.

M. Rijal [34] investigated structural properties of the QAP polytope. The starting point is the quadric Boolean polytope.

5. Fortran Codes for QAP

The following Fortran codes are available through the QAPLIB Home Page on WWW. We intend to extend this list of codes, and would like to include also further software, contributed by other researchers.

Unless otherwise stated, the following programs are selfcontained, i.e. compiling them should result in an executable main program. The input convention is the same for all files. The main program expects a QAP instance (in the format of the QAPLIB) from the primary input.

qapbb.f
The Branch and Bound code from [5] solves QAPs to optimality. The code qapbb.f is a modified version of it (a linear term can be included) and is quite efficient on problems of sizes $n \leq 15$. Running it on larger problems may result in unpredictably long computation times. Currently the code is dimensioned to handle problems of sizes at most $n \leq 33$. A typical call might look like

```
bbqap < nug12.dat
```

qapglb.f
The Gilmore–Lawler bound can be computed quite efficiently for all instances of the QAPLIB. Currently the code is dimensioned for problems with $n \leq 256$. It uses some of the subroutines from [5] in modified form.

qapeli.f
This routine computes the elimination bound. It is applicable only if the problem is symmetric. It is also dimensioned for $n \leq 256$.

GRASP
These are the GRASP heuristics of [25, 32]. The code is obtainable from the home page of M. G. C. Resende (URL:
`ftp://netlib.att.com/netlib/att/math/resende/home.html`),
and consists of compressed tar-files.

qapsim.f
This is the code from [7], and produces heuristic solutions for symmetric QAPs of dimension $n \leq 256$, based on simulated annealing.

Li–Pardalos generator
The problem generator of Y. Li and P. M. Pardalos [24] can be obtained by sending

an E-Mail to `coap@math.ufl.edu` and putting “send 92006" in the body of the mail message.

6. Conclusion

Even though the research activities around the QAP have significantly increased during the last years, we feel that the QAP is still a serious challenge for scientists. There are very efficient heuristics available, that find in acceptable computation times seemingly good solutions. To prove their optimality, there are a variety of bounds available. Unfortunately, it seems to be the case that the bounds with low computational cost, like GLB or ELI are not strong enough on larger problems, to prove optimality with limited enumeration.

The more advanced, and only recently investigated polyhedral and semidefinite relaxations seem to be stronger, but their current implementations are prohibitive for even moderately sized problems. Their advantage lies also in the fact, that dual information is available, which can be used to guide the branching process.

A breakthrough to solve larger QAP instances to optimality can in our opinion only be expected, if these stronger bounds can also be implemented to run much faster than the current implementations. It will be interesting to follow the progress on QAP in the near future.

Acknowledgements

We thank all colleagues who contributed to QAPLIB over the last years. For this update we are particularly grateful to Eranda Çela for pointing out several references, and to Charles Fleurent, Michael Perregaard, Mauricio Resende and Éric Taillard for making their data and solutions available to us.

The second author acknowledges financial support from Christian Doppler Laboratory for Discrete Optimization at Graz University of Technology. Most of this work was done while he was employed in the laboratory. He is now supported by the Austrian “Fonds zur Förderung der wissenschaftlichen Forschung, Erwin-Schrödinger-Auslandsstipendium Nr. J01204-MAT".

References

1. R. Battiti and G. Tecchiolli. The reactive tabu search. *ORSA Journal on Computing* **6**(2): 126–140, 1994.
2. A. Brüngger, J. Clausen, A. Marzetta and M. Perregaard. Joining forces in solving large-scale quadratic assignment problems. DIKU Technical Report, University of Copenhagen, 1996.
3. R.E. Burkard. Locations with spatial interactions: the quadratic assignment problem. In P.B. Mirchandani and R.L. Francis, editors, *Discrete Location Theory*. Wiley, Berlin, 1991.
4. R.E. Burkard and E. Çela. Quadratic and three-dimensional assignment problems. In M. Dell'Amico, F. Maffioli, and S. Martello, editors, *Annotated Bibliographies in Combinatorial Optimization*. 1996. To appear. Available as SFB Report 63, Graz University of Technology, Graz, Austria.

5. R.E. Burkard and U. Derigs. *Assignment and Matching Problems: Solution Methods with Fortran Programs*, volume 184 of *Lecture Notes in Economics and Mathematical Systems*. Springer, Berlin, 1980.
6. R.E. Burkard and J. Offermann. Entwurf von Schreibmaschinentastaturen mittels quadratischer Zuordnungsprobleme. *Zeitschrift für Operations Research* **21**: B121–B132, 1977.
7. R.E. Burkard and F. Rendl. A thermodynamically motivated simulation procedure for combinatorial optimization problems. *European Journal of Operations Research* **17**(2): 169–174, 1984.
8. E. Çela. *The quadratic assignment problem: special cases and relatives*. PhD thesis, Graz University of Technology, Graz, Austria, 1995.
9. N. Christofides and E. Benavent. An exact algorithm for the quadratic assignment problem. *Operations Research* **37-5**: 760–768, 1989.
10. J. Clausen and M. Perregaard. Solving large quadratic assignment problems in parallel. *Computational Optimization and Applications*, 1994. To appear.
11. A.N. Elshafei. Hospital layout as a quadratic assignment problem. *Operations Research Quarterly* **28**: 167–179, 1977.
12. B. Eschermann and H.J. Wunderlich. Optimized synthesis of self-testable finite state machines. In *20th International Symposium on Fault-Tolerant Computing (FFTCS 20)*, Newcastle upon Tyne, 26-28th June, 1990.
13. C. Fleurent and J.A. Ferland. Genetic hybrids for the quadratic assignment problem. In P. Pardalos and H. Wolkowicz, editors, *Quadratic Assignment and Related Problems*, volume 16, pages 173–187. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1994.
14. P.C. Gilmore. Optimal and suboptimal algorithms for the quadratic assignment problem. *SIAM Journal on Applied Mathematics* **10**: 305–31, 1962.
15. S.W. Hadley, F. Rendl, and H. Wolkowicz. A new lower bound via projection for the quadratic assignment problem. *Mathematics of Operations Research* **17**: 727–739, 1992.
16. P. Hahn, T. Grant, and N. Hall. Solution of the quadratic assignment problem using the Hungarian method. *European Journal of Operational Research*, to appear, 1995.
17. T.A. Johnson. *New linear programming-based solution procedures for the quadratic assignment problem*. PhD thesis, Clemson University, Clemson, USA, 1992.
18. S.E. Karisch. *Nonlinear approaches for quadratic assignment and graph partition problems*. PhD thesis, Graz University of Technology, Graz, Austria, 1995.
19. S.E. Karisch and F. Rendl. Lower bounds for the quadratic assignment problem via triangle decompositions. *Mathematical Programming* **71**(2): 137–152, 1995.
20. S.E. Karisch, F. Rendl, H. Wolkowicz, and Q. Zhao. Semidefinite programming relaxations for the quadratic assignment problem. Working paper, CDL-DO, Department of Mathematics, Graz University of Technology, Graz, Austria, 1995.
21. J. Krarup and P.M. Pruzan. Computer-aided layout design. *Mathematical Programming Study* **9**: 75–94, 1978.
22. E. Lawler. The quadratic assignment problem. *Management Science* **9**: 586–599, 1963.
23. Y. Li. *Heuristic and exact algorithms for the quadratic assignment problem*. PhD thesis, The Pennsylvania State University, USA, 1992.
24. Y. Li and P.M. Pardalos. Generating quadratic assignment test problems with known optimal permutations. *Computational Optimization and Applications* **1**: 163–184, 1992.
25. Y. Li, P.M. Pardalos, and M.G.C. Resende. A greedy randomized adaptive search procedure for the quadratic assignment problem. In P. Pardalos and H. Wolkowicz, editors, *Quadratic Assignment and Related Problems*, volume 16, pages 237–261. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1994.
26. F. Malucelli. *Quadratic assignment problems: solution methods and applications*. PhD thesis, University of Pisa, Pisa, Italy, 1993.
27. T. Mautor. *Contribution à la résolution des problèmes d'implanation: algorithmes séquentiels et parallèles pour l'affectation quadratique*. PhD thesis, Université Pierre et Marie Curie, Paris, France, 1992.
28. C.E. Nugent, T.E. Vollman, and J. Ruml. An experimental comparison of techniques for the assignment of facilities to locations. *Operations Research* **16**: 150–173, 1968.

29. T. Ostrowski and V.T. Ruoppila. Genetic annealing search for index assignment in vector quantization. Technical Report, Digital Media Institute, Tampere University of Technology, Tampere, Finland, 1996.
30. P.M. Pardalos, F. Rendl, and H. Wolkowicz. The quadratic assignment problem: a survey of recent developments. In P. Pardalos and H. Wolkowicz, editors, *Quadratic Assignment and Related Problems*, volume 16, pages 1–42. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 1994.
31. P.M. Pardalos and H. Wolkowicz, editors. *Quadratic Assignment and Related Problems*, volume 16 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science. 1994.
32. M.G.C. Resende, P.M. Pardalos, and Y. Li. FORTRAN subroutines for approximate solution of dense quadratic assignment problems using GRASP. *ACM Transactions on Mathematical Software* **22**(1): 104–118, 1996.
33. M.G.C. Resende, K.G. Ramakrishnan, and Z. Drezner. Computing lower bounds for the quadratic assignment problem with an interior point algorithm for linear programming. *Operations Research* **43**: 781–791, 1995.
34. M. Rijal. *Scheduling, design and assignment problems with quadratic costs*. PhD thesis, New York University, New York, USA, 1995.
35. C. Roucairol. *Du sequentiel au parallele: la recherche arborescente et son application à la programmation quadratique en variables 0 et 1*, 1987. Thèse d'Etat, Université Pierre et Marie Curie, Paris, France.
36. M. Scriabin and R.C. Vergin. Comparison of computer algorithms and visual based methods for plant layout. *Management Science* **22**: 172–187, 1975.
37. J. Skorin-Kapov. Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing* **2**(1): 33–45, 1990.
38. L. Steinberg. The backboard wiring problem: a placement algorithm. *SIAM Review* **3**: 37–50, 1961.
39. É.D. Taillard. Robust tabu search for the quadratic assignment problem. *Parallel Computing* **17**: 443–455, 1991.
40. É.D. Taillard. Comparison of iterative searches for the quadratic assignment problem. *Location Science*, 1994. To appear.
41. U.W. Thonemann and A. Bölte. An improved simulated annealing algorithm for the quadratic assignment problem. Working paper, School of Business, Department of Production and Operations Research, University of Paderborn, Germany, 1994.
42. M.R. Wilhelm and T.L. Ward. Solving quadratic assignment problems by simulated annealing. *IIE Transaction* **19**/1: 107–119, 1987.