Test instances for the traffic assignment problem

F. Babonneau* J.-P. Vial*

January 8, 2008

Abstract

This short note on the Traffic Assignment Problem (TAP) provides the relevant information on test problems previously used in the literature to facilitate benchmarking.

Keywords. Traffic assignment problem, BPR function, Kleinrock function, linear function.

Test problems for TAP

This short note on the Traffic Assignment Problem (TAP) provides the relevant information on test problems previously used in the literature to facilitate benchmarking. Problem data come from different sources [5, 7, 11] and often deal with a specific type of congestion function, e.g., BPR, Kleinrock or linear. To enlarge the test set, the authors of [1, 2] introduced missing arc capacities and set to be large enough to match the demands. Some mistakes were also made in reporting these elements [3] and it seems appropriate to put in a single place information on where to find the data, how to adjust them and what are the optimal values with five digit of accuracy. The proposed test set does not cover all problems in the literature, but the most challenging in the category of multiple users with single origin and single destination.

We briefly recall the mathematical formulation of the TAP problem with multiple users with single origin and single destination. Let $\mathcal{G}(\mathcal{N},\mathcal{A})$ be an oriented graph, where $\mathcal{N}$ is the set of nodes and $\mathcal{A}$ the set of arcs. The TAP problem is

$$\min_{x,y} \quad \sum_{a\in\mathcal{A}} g_a(y_a) \tag{1a}$$

$$\sum_{k\in\mathcal{K}} x_a^k = y_a, \quad \forall a \in \mathcal{A}, \tag{1b}$$

$$N x^k = d_k \delta^k, \quad \forall k \in \mathcal{K}, \tag{1c}$$

$$x_a^k \geq 0, \qquad \forall a \in \mathcal{A},\ \forall k \in \mathcal{K}. \tag{1d}$$

Here, N is the network incidence matrix; $\mathcal{K}$ is the set of "client" or "user" types; d_k is the demand for type $k \in \mathcal{K}$ of users; and δ^k is vector of zeros except a "1" at the origin node and a "-1" at the destination node. The vector $x^k = (x_a^k)_{a\in\mathcal{A}}$ represents the traffic flow of type k of users on the arcs of the network and y is the vector of total arc flow. The objective function components $g_a : \mathbb{R} \to \mathbb{R}$ is assumed to be convex and twice continuously

*ORDECSYS, Geneva, `http://www.ordecsys.com/`

differentiable. The TAP problem is also known as the multicommodity problem; there, a user type is a commodity and one talks of supply and demand nodes instead of origin and destination.

The literature essentially deals with three types of objective functions: the *Kleinrock* function (multicommodity version), the *BPR* (Bureau of Public Roads) function (TAP version) and the *linear* one with upper bounds on the flows (TAP and/or multicommodity version). The Kleinrock function is given by

$$g_a(y_a) = \frac{y_a}{c_a - y_a}, \quad \text{with } y_a \in [0, c_a), \tag{2}$$

where c_a is the arc capacity.

The *BPR* function is

$$g_a(y_a) = t_a y_a \left(1 + \frac{\alpha}{\beta+1}(\frac{y_a}{c_a})^\beta\right), \quad \text{with } y_a \in \mathbb{R}^+. \tag{3}$$

In general, the parameter α is very small and $\beta > 1$ does not exceed 5. When the flow y_a is less than c_a, the second term under the parenthesis in (3) is negligible. Thus $g_a(y_a) \approx t_a y_a$: the parameter t_a is called *free-flow travel time* and it can be interpreted as a fixed travel time on a congestion-free arc. For larger values of y_a the nonlinear contribution to congestion increases. The threshold value c_a for the flow y_a is usually named the *practical capacity* of the arc, beyond which congestion becomes effective.

The linear function is

$$g_a(y_a) = t_a y_a, \quad y_a \in [0, c_a], \tag{4}$$

where $t_a \geq 0$ is the linear constant.

In Table 1 we give data on four sets of problems. For each problem instance, we give the number of nodes $|\mathcal{N}|$, the number of arcs $|\mathcal{A}|$, the number of commodities $|\mathcal{K}|$, the optimal solution values to TAP $z^*_{Kleinrock}$ for the Kleinrock function, z^*_{BPR} for the BPR function and z^*_{linear} for the linear function, with a relative optimality gap less than 10^{-5}. For more details on the TAP and on the objective functions, we refer the reader to [1, 2].

The first set, the `planar` problems, contains 10 instances that have been generated by Di Yuan to simulate telecommunication problems. Nodes are randomly chosen as points in the plane, and arcs link neighbor nodes in such a way that the resulting graph is planar. Commodities are pairs of origin and destination nodes, chosen at random. Demands and capacities are uniformly distributed in given intervals. The second set, the `grid` problems, contains 15 networks that have a grid structure such that each node has four incoming and four outgoing arcs. Commodities, and demands are generated in a way similar to that of `planar` networks. These two sets of problems are calibrated to solve the linear TAP. The data include arc capacities and linear costs and can be downloaded from `http://www.di.unipi.it/di/groups/optimize/Data/MMCF.html`. We use directly these arc capacities in the definition of the Kleinrock function. To solve TAP with BPR function, we use the capacity as *practical capacity* and the linear cost as *free-flow travel time*. As suggested in [12], we use the parameter values $\alpha = 0.15$ and $\beta = 4$. These two sets of instances are used in [1, 2, 10].

The third collection of problems is composed of telecommunication problems of various sizes. The small problems `ndo22` and `ndo148` are two practical problems solved in [1, 2, 7, 8]. Problem `904` is based on a real telecommunication network and was used in [1, 2, 11]. This problem set is adapted to solve TAP with Kleinrock function. To solve TAP with BPR function, we use the capacity as *practical capacity* and also use it as *free-flow travel*

Problem ID	$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	$\|\mathcal{K}\|$	$z^*_{Kleinrock}$	z^*_{BPR}	z^*_{linear}
			`planar` problems			
`planar30`	30	150	92	40.5668	4.44549×10^7	4.43508×10^7
`planar50`	50	250	267	109.478	1.21236×10^8	1.22200×10^8
`planar80`	80	440	543	232.321	1.81906×10^8	1.82438×10^8
`planar100`	100	532	1085	226.299	2.29114×10^8	2.31340×10^8
`planar150`	150	850	2239	715.309	5.27985×10^8	5.48089×10^8
`planar300`	300	1680	3584	329.120	6.90748×10^8	6.89982×10^8
`planar500`	500	2842	3525	196.394	4.83309×10^9	4.81984×10^8
`planar800`	800	4388	12756	354.008	1.16952×10^9	1.16737×10^8
`planar1000`	1000	5200	20026	1250.92	3.41859×10^9	3.44962×10^9
`planar2500`	2500	12990	81430	3289.05	1.23827×10^{10}	1.26624×10^{10}
			`grid` problems			
`grid1`	25	80	50	66.4002	8.33599×10^5	8.27323×10^5
`grid2`	25	80	100	194.512	1.72689×10^6	1.70538×10^6
`grid3`	100	360	50	84.5618	1.53241×10^6	1.52464×10^6
`grid4`	100	360	100	171.331	3.05543×10^6	3.03170×10^6
`grid5`	225	840	100	236.699	5.07921×10^6	5.04970×10^6
`grid6`	225	840	200	652.877	1.05075×10^7	1.04007×10^7
`grid7`	400	1520	400	776.566	2.60669×10^7	2.58641×10^7
`grid8`	625	2400	500	1542.15	4.21240×10^7	4.17113×10^7
`grid9`	625	2400	1000	2199.83	8.36394×10^7	8.26533×10^7
`grid10`	625	2400	2000	2212.89	1.66084×10^8	1.64111×10^8
`grid11`	625	2400	4000	1502.75	3.32475×10^8	3.29259×10^8
`grid12`	900	3480	6000	1478.93	5.81488×10^8	5.77189×10^8
`grid13`	900	3480	12000	1760.53	1.16933×10^9	1.15932×10^9
`grid14`	1225	4760	16000	1414.39	1.81297×10^9	1.80268×10^9
`grid15`	1225	4760	32000	1544.15	3.61568×10^9	3.59353×10^9
			Telecommunication-like problems			
`ndo22`	14	22	23	103.412	1.86767×10^3	1.85488×10^3
`ndo148`	61	148	122	151.926	1.40233×10^5	1.39500×10^5
`904`	106	904	11130	33.4931	1.29197×10^7	1.37850×10^7
			Transportation problems			
`Sioux-Falls`	24	76	528	600.679	4.23133×10^6	1.71969×10^6
`Winnipeg`	1067	2836	4344	1527.41	8.25673×10^5	420.163
`Barcelona`	1020	2522	7922	845.872	1.22856×10^6	240.940
`Chicago-sketch`	933	2950	93135	614.726	1.67484×10^7	6.43520×10^6
`Chicago-region`	12982	39018	2296227	3290.49	2.58457×10^7	3.75942×10^6
`Philadelphia`	13389	40003	1149795	2557.42	2.24926×10^8	2.55777×10^7

Table 1: Test problems.

time. We choose the parameter values $\alpha = 0.15$ and $\beta = 4$. For the linear case, we use the capacity as linear cost.

The last collection of problems is composed of six realistic transportation problems used in [1, 2, 4, 5, 6, 9]. The data are adapted for the BPR function. They include *free-flow travel time*, *practical capacity* and the tuning parameters α and β. These problems, can be downloaded from `http://www.bgu.ac.il/~bargera/tntp/`. To solve TAP with Kleinrock and linear functions we use *practical capacity* as capacity and to turn these problems feasible with respect to the capacity, the demands are divided by a given factor. The scaling factors are 2 for Sioux-Falls, 5100 for Barcelona, 2000 for Winnipeg, 2.5 for Chicago-sketch, 6 for Chicago-region and 7 for Philadelphia. For the linear TAP, we use *free flow time* as linear cost. Note that the authors in [1] use different values for the linear cost.

In Table 2, we report alternate instances used in [2]. These data can be downloaded from `http://www.ordecsys.com/oboe/AlternateData.tar.gz`

Problem ID	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{K}\rvert$	$z^*_{Kleinrock}$	z^*_{BPR}	z^*_{linear}
Alternate instances						
`ndo22-alt`	14	22	23	11.5631	1.87110×10^3	1.88237×10^3
`Barcelona-alt`	1020	2522	7922	845.872	1.23277×10^6	240.940
`Philadelphia-alt`	13389	40003	1151166	2557.42	1.27810×10^8	2.55777×10^7

Table 2: Test problems.

Acknowldegements We are indebted to K. Kiwiel who helped us to check the optimal values for all instances with Kleinrock and BPR functions.

References

[1] F. Babonneau, O. du Merle, and J.-P. Vial. Solving large scale linear multicommodity flow problems with an active set strategy and Proximal-ACCPM. *Operations Research*, 54(1):184–197, 2006.

[2] F. Babonneau and J.-P. Vial. ACCPM with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems. Forthcoming *Mathematical Programming*, 2007.

[3] F. Babonneau and J.-P. Vial. ACCPM with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems: A corrigendum. Forthcoming *Mathematical Programming*, 2007.

[4] F. Babonneau and J.-P. Vial. An efficient method to compute traffic assignment problems with elastic demands. Forthcoming *Transportation Science*, 2007.

[5] H. Bar-Gera. Origin-based algorithm for traffic assignment problem. *Transportation Science*, 36(4):398–417, 2002.

[6] M. Daneva and P.O. Lindberg. The stiff is moving - conjugate direction Franck-Wolfe methods with applications to traffic assignment. Technical report, Linkoping University, Department of Mathematics, 2004.

[7] E.M. Gafni and D.P. Bertsekas. Two-metric projection methods for constrained optimization. *SIAM Journal on Control and Optimization*, 22(6):936–964, 1984.

[8] J.-L. Goffin, J. Gondzio, R. Sarkissian, and J.-P. Vial. Solving nonlinear multicommodity flow problems by the analytic center cutting plane method.

[9] T. Larsson and M. Patriksson. An augmented lagrangean dual algorithm for link capacity side constrained traffic assignment problems. *Transportation Research*, 29B:433–455, 1995.

[10] T. Larsson and Di Yuan. An augmented lagrangian algorithm for large scale multicommodity routing. *Computational Optimization and Applications*, 27(2):187–215, 2004.

[11] A. Ouorou, P. Mahey, and J.-P. Vial. A survey of algorithms for convex multicommodity flow problems. *Management Science*, 46:126–147, 2000.

[12] Y. Sheffi. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Models*. Prentice-Hall, New Jersey, 1985.