On Routing Efficiency of a Network Design Algorithm

Kairat Jaroenrat
Rangsit University
52/347 Muang-Ake, Phaholyothin Rd., Lak-Hok,
Pathumtani, Thailand
(661)-907-8128
kairat@kairat.com

Pisit Charnkeitkong
Rangsit University
52/347 Muang-Ake, Phaholyothin Rd., Lak-Hok,
Pathumtani, Thailand
(662) 997-222 ext. 1601, 1606
pisitc@rangsit.rsu.ac.th

ABSTRACT
The topology of UMTS (Universal Mobile Telecommunications System)'s core and access networks are more like a partial mesh, which is difficult to obtain the optimal solution. Mesh Network Topological Optimization and Routing (MENTOR) algorithm is known as a low complexity heuristic used to design partial mesh networks. This study explores the relation between design parameters and performance of traffic routing of MENTOR algorithm. We analyze 432 MENTOR networks for 4 sets of 50 nodes each with equivalently distributed demand and randomly generated locations. For each of these networks, the performances at normal load and at congestion threshold of MENTOR flow assignment are calculated and compared with the optimum solution obtained by solving the linear programming. It is found that the performances of MENTOR strongly depend on "slack", a design parameter represented the different between the maximum and minimum utilization. For small value of slack, $s \leq 0.1$, the performances of MENTOR keep very close to that of the optimum solution, while the maximum utilization hardly impact the performances. As the slack get larger the performances of MENTOR become worse and more depend on the maximum utilization.

Categories and Subject Descriptors
C.2.1 [**COMPUTER-COMMUNICATION NETWORKS**]: Network Architecture and Design – *Network communications and Network topology*

General Terms
Measurement, Performance, Design, Experimentation

Keywords
MENTOR, Network Design, Flow assignment efficiency

1. INTRODUCTION
In the last decade the 2nd generation mobile systems, e.g. GSM (Global System for Mobile communication), got great popularity, and made a uniform and seamless surface for telecommunications around the world. The new demands hurry taking further steps, and have already started significant changes the third generation mobile systems, e.g. the UMTS. The UMTS differ from GSM networks in many ways, but the main difference is that the UMTS will support multimedia applications more efficiently. In the initial phases, transport networks of UMTS are based on virtual circuit packet switching technologies, e.g. ATM (Asynchronous Transfer Mode), MPLS (Multi-Protocol Label Switching). Furthermore, in RAN (Radio Access Network) of UMTS, RNCs (Radio Network Controllers) are allowed to directly connect to each others. Therefore, the topologies of core network and access network of UMTS are more like mesh. However, it is known that the computational complexity of solving the optimal mesh network is very high.

A heuristic network design algorithm called MENTOR (Mesh Network Topological Optimization and Routing) [1] [2] is known as a high-speed and very efficient design algorithm. MENTOR is often used to design virtual circuit packet switching networks such as Frame Relay, ATM or even MPLS. When MENTOR decides to installed a link, at the same time, traffic flow over it is assigned. Flow assignment of MENTOR is not always optimal and strongly depends on network design parameters such as maximum and minimum link utilization.

This study investigates the relation between design parameters and performance of flow assignment of MENTOR algorithm. We analyze 432 networks designed by MENTOR for 4 sets of 50 nodes each with equivalently distributed demand and randomly generated locations. For each of these networks, the performances at normal load and at threshold of congestion of MENTOR flow assignment are calculated and compared with the optimum solution obtained by solving the linear programming.

2. PROBLEM FORMULATION
2.1 MENTOR Algorithm
MENTOR algorithm is a low complexity heuristic network design algorithm. This low complexity is achieved by doing implicit routing over a link at the same time it is considered to be installed. For a given set of nodes N, demand matrix D and link cost matrix X, let $d_{s,t}$ and $x_{s,t}$ are the amount of traffic flow and link installation cost from s and t, respectively. The characteristics of MENTOR network are (1) traffic demands are routed on relatively direct paths (2) links have reasonable utilization and (3) relatively high capacity links are used.

MENTOR starts with clustering process. In this stage, nodes are classified in to end nodes and backbone nodes using a clustering algorithm. Examples of possible clustering algorithms are threshold clustering and K-mean clustering. However, we consider in this paper only the case where traffic demands are distributed equivalently among all nodes. Therefore, all nodes can be considered as backbone node.

Mobility 06, Oct. 25–27, 2006, Bangkok, Thailand.

Next, a good tree is formed to interconnect all (backbone) nodes. To build a spanning tree, Kershenbaum et. al. [1] suggests to a use a heuristic, which can be thought of as a modification of Prim and Dijkstra algorithm. The algorithms add a new node into the tree once at a time until all nodes are put into the tress. Prim Dijkstra's algorithm expands the tree by connecting a node i, which is already in tree, to an out of tree node j to such that $\zeta \cdot L_i + x_{ic}$ is minimized. Here ζ, $0 \leq \zeta \leq 1$, is parameter used to control the characteristic of the tree. L_i is the cost of path from root along the tree to node i. Note that $\zeta = 0$ and 1 is corresponding to Minimum Spanning Tree and Shortest Path Tree, respectively.

Given a tree, the objective of MENTOR is to consider adding a direct link between each pair of nodes if the utilization is between the predefined maximum and minimum utilization. Let the maximum utilization be ψ, and the minimum utilization be $(14s)\psi$, where slack s, $0 \leq s \leq 1$. Consider a pair of nodes A and B, let C_{AB} and l_{AB} be link capacity and accumulated load between A and B, respectively. If $l_{AB} < \psi C_{AB}$ (14s), no link is added and all traffic l_{AB} is overflowed to the next most direct path. A link is added and no over flow traffic if ψC_{AB}, (14s) $\leq l_{AB} \leq \psi C_{AB}$. However, if $l_{AB} > \psi C_{AB}$, a direct link is added only when splitting traffic among multiple route is possible. In this situation and a portion of traffic $l_{AB}\ 4\ \psi C_{AB}$ is overflow to the next most direct path. Otherwise, if splitting traffic is not possible, no link is added and l_{AB} is overflowed to the next most direct path. Node pairs are sequenced such that a link between a pair is considered only when all traffic flows that could overflow to the link are already considered.

MENTOR gives fairly good results and widely used to many type of networks, e.g. Frame Relay, ATM as well as MPLS. However, the impact of design parameters, e.g. ψ, s and ζ, on efficiency of traffic routing are not yet studied before.

2.2 Objective Function

Consider a directed network graph G = (N, A) with a capacity c_a for each $a \subset A$ and as define in previous section, d_{st} denote the amount of traffic flow between s and t. Let f_a^t indicate how much of the traffic flow to t over arc a, traffic load l_a over link $a \subset A$ is the sum of all f_a^t. It is suggested in [3] to measure the performance of network by cost function

$$\Phi = \sum_{a \subset A} \lambda_a(l_a, c_a), \tag{1}$$

where $\lambda_a(l_a, c_a)$ is an M/M/1 queuing theory style link cost function given by

$$\lambda_a(l_a, c_a) = l_a / (c_a\ 4\ l_a) \tag{2}$$

With this function, it is more expensive to send flow along arcs whose loads approach capacity, which is what we want. However, the function does not deal with overloaded links, i.e. $l_a \geq c_a$. To overcome this problem, $l_a / (c_a\ 4\ l_a)$ is approximated by a piece-wise linear function $\lambda_a(0) = 0$ and derivative

$$\lambda'_a(l_a, c_a) = \begin{cases} 1 \text{ for } 0 \ \Omega\ l_a/c_a < 1/3, \\ 3 \text{ for } 1/3 \ \Omega\ l_a/c_a < 2/3, \\ 10 \text{ for } 2/3 \ \Omega\ l_a/c_a < 9/10, \\ 70 \text{ for } 9/10 \ \Omega\ l_a/c_a < 1, \\ 500 \text{ for } 1 \ \Omega\ l_a/c_a < 11/10, \\ 5000 \text{ for } 11/10 \ \Omega\ l_a/c_a < \infty. \end{cases} \tag{3}$$

2.3 Optimum Solutions

Given the piece-wise linear cost function define by (3), the general routing problem can be formulated as the following linear programming [4].

$$\text{Min A} = \sum_{a \subset A} \lambda_a \tag{4}$$

Subject to:

$$\sum_{u:(u,v)\subset A} f^t_{(u,v)}\ 4 \sum_{u:(v,u)\subset A} f^t_{(v,u)} = \begin{cases} D^t, \text{ if } v = t. \\ \text{-}\ d_{s,t}, \text{ if } v = s \\ 0, \text{ otherwise,} \end{cases} \qquad v,s,t \subset N, \tag{5}$$

$$\lambda_a \varnothing l_a \qquad a \subset A, \tag{6}$$

$$\lambda_a \varnothing 3l_a - 2/3c_a \qquad a \subset A, \tag{7}$$

$$\lambda_a \varnothing 10l_a - 16/3c_a \qquad a \subset A, \tag{8}$$

$$\lambda_a \varnothing 70l_a - 178/3c_a \qquad a \subset A, \tag{9}$$

$$\lambda_a \varnothing 500l_a - 1468/3c_a \qquad a \subset A, \tag{10}$$

$$\lambda_a \varnothing 5000l_a - 19468/3c_a \qquad a \subset A, \tag{11}$$

$$l_a = \sum_{t \subset N} f_a^t \qquad a \subset A, \tag{12}$$

$$D^t = \sum_{s \subset N} d_{s,t} \tag{13}$$

$$f_a^t \varnothing 0 \qquad a \subset A;\ t \subset N. \tag{14}$$

Constraints (5) are flow conservation constraints, constraints (6) – (11) describe the cost function, constraints (12) define the load on each arc, and constraint (13) defines D^t that represents all traffic headed toward destination node t.

However, general optimum solution is not fairly comparable with other traffic routing that have limits maximum link utilization, e.g. MENTOR algorithm. This is because the purpose of limiting maximum link utilization is to reserve capacity to handle more traffic load when network get congest. To take in to account the capacity reservation, the optimum solution with maximum link utilization ψ is obtained by solving (4) subject to (5) – (14) and additional constraint

$$l_a / c_a\ \Omega\ \psi \tag{15}$$

2.4 Normalizing Routing Cost

Fortz and Thorup [5] proposed a normalizing scaling factor for the routing cost that makes possible comparisons across different network sizes and topologies:

$$\text{A}_{\text{UNCAP}} = \sum_{s,t \subset NxN} d_{s,t} h_{s,t} \tag{16}$$

where h_{st} = minimum hop count between s and t.

For any routing cost A, the scaled routing cost or normalized routing cost is defined as

$$\text{A}^* = \text{A} / \text{A}_{\text{UNCAP}} \tag{17}$$

The above program is a complete linear programming formulation of the general routing problem. We shall use A to denote the optimal general routing cost.

3. EXPERIMENTs

In order to evaluate the efficiency of flow assignment calculated by MENTOR algorithm, we analyze the performances of a number synthesized network and observe the relation between design parameters and performances.

3.1 Experiment Set Up

DELITE [6] is used to synthesize 4 sets of 50 nodes each with different node distribution obtained by varying SEED parameter. We shall refer to these set of nodes as N1, N2, N3 and N4. The traffic demand matrix for each set of nodes is also generated by DELITE with default setting and total traffic in and traffic out of each node are 100 Mbps.

By varying design parameters, a total of 432 MENTOR networks are generated for N1-N4 using the full-duplex link of capacity 45 Mbps. For each node sets, two groups of networks corresponding to Minimum Spanning Tree (ζ=0) and Shortest Path Tree (ζ=1) are generated. For each type of spanning tree, 54 networks are generated by varying of ψ∂#ψ⊂ (0.4, 0.5, 0.6, 0.7, 0.8, 1.0) and *s, s* ⊂ (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8).

3.2 Experiment Results

For each of 432 networks, optimum routing solution with maximum link utilization constraint proposed in section 2.3 is solved by GLPK[7]. On Intel Pentium IV Xeon 3.3 GHz machine, it takes maximum 2 hours to solve the optimal routing problem. Normalized cost A· of MENTOR flow assignment and optimal solution are calculated for different scaling of projected demand matrix.

Given network demands, performance of MENTOR flow assignment at normal load is measured by % of cost different from optimality

$$\Delta C = \frac{A^*_M - A^*_O}{A^*_O} \times 100,$$

where A_M and A^1_O are normalized cost of MENTOR flow assignment, and that of optimum solution, measured at demand used to design the network respectively.

As seen in section 2, the cost function increase rapidly toward 5000 after the λ_a = 10 ⅔.The performance of MENTOR flow assignment at the threshold of congestion is measured by % of demand different from optimality

$$\Delta D = \frac{D_M - D_O}{D_O} \times 100,$$

where D_M and D_O are the scaling traffic demand of MENTOR flow assignment, and that of optimum solution measured when the cost A· = 10⅔, respectively

÷C and#-D for networks generated in 3.1 are presented in Figure 1-16. ÷C and#-D of the same#ζ and slack are averaged and summarized as shown in Table 1.

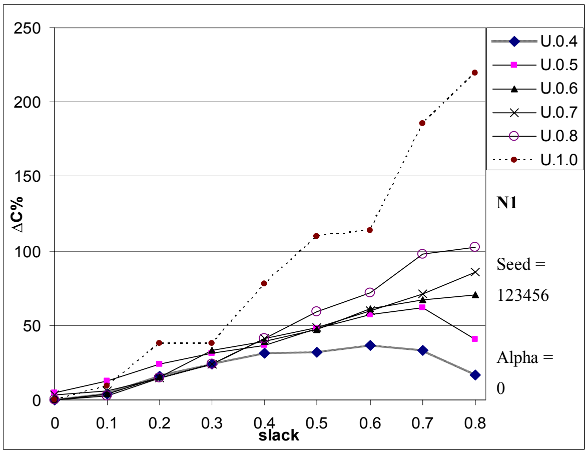

Figure 1. ÷C of networks with Alpha = 0 for N1 (Seed parameter = 123456).

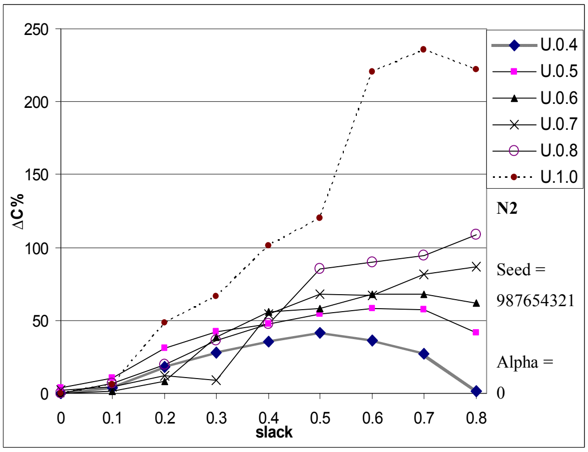

Figure 2. ÷C of networks with Alpha = 0 for N2 (Seed parameter = 987654321).

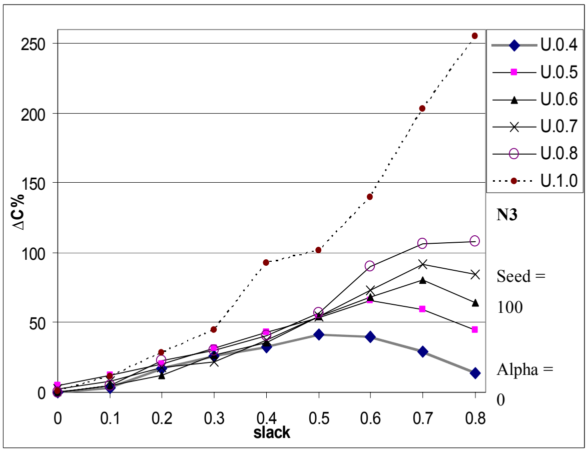

Figure 3. ÷C of networks with Alpha = 0 for N3. (Seed parameter = 100).

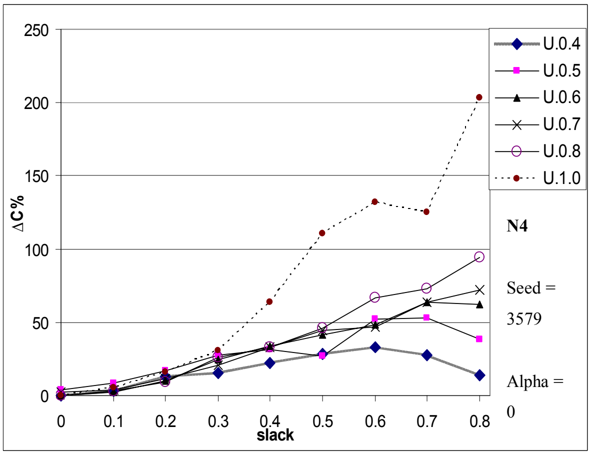

Figure 4. ÷C of networks with Alpha = 0 for N4 (Seed parameter = 3579).

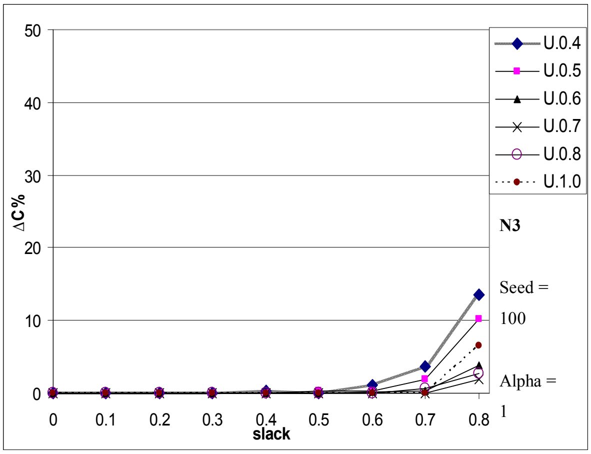

Figure 7. ÷C of networks with Alpha = 1 for N3 (Seed parameter = 100).

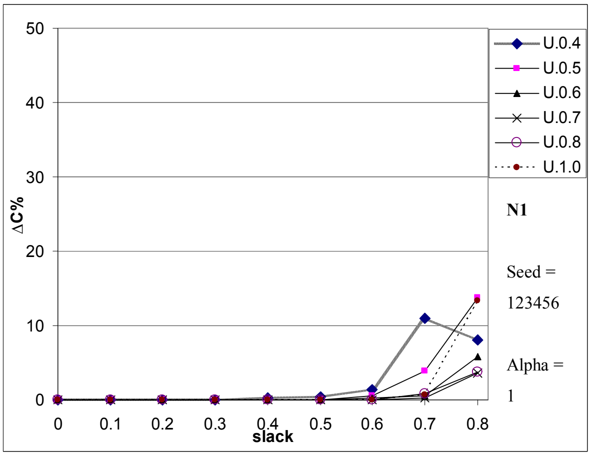

Figure 5. ÷C of networks with Alpha = 1 for N1 (Seed parameter = 123456).

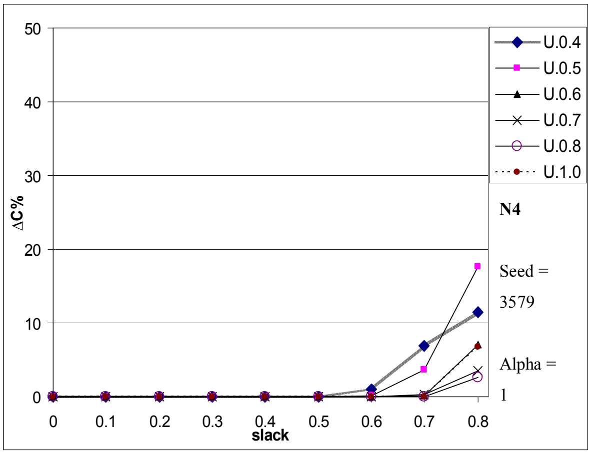

Figure 8. ÷C of networks with Alpha = 1 for N4 (Seed parameter = 3579).

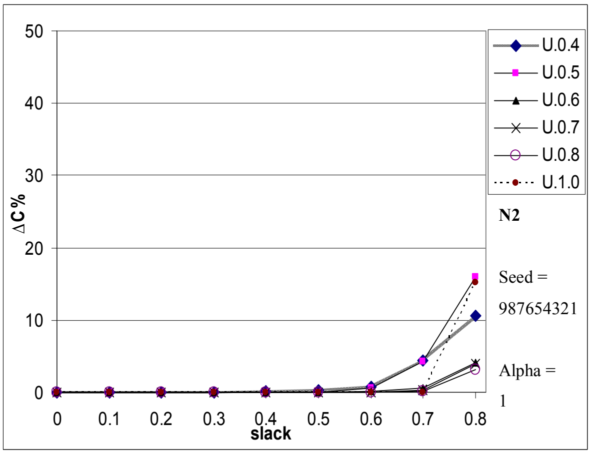

Figure 6. ÷C of networks with Alpha = 1 for N2. (Seed parameter = 987654321).

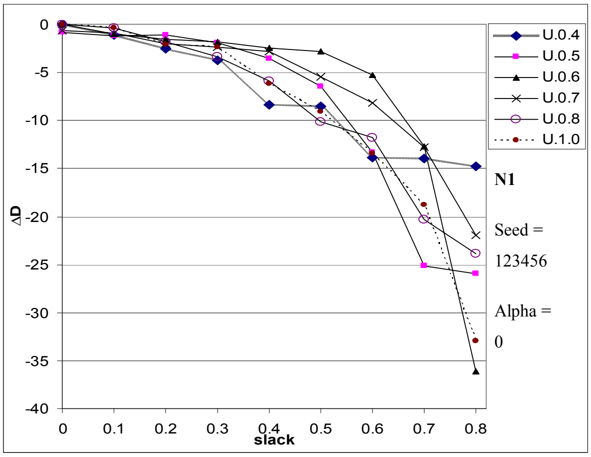

Figure 9. ÷D of networks with Alpha = 0 for N1 (Seed parameter = 123456).

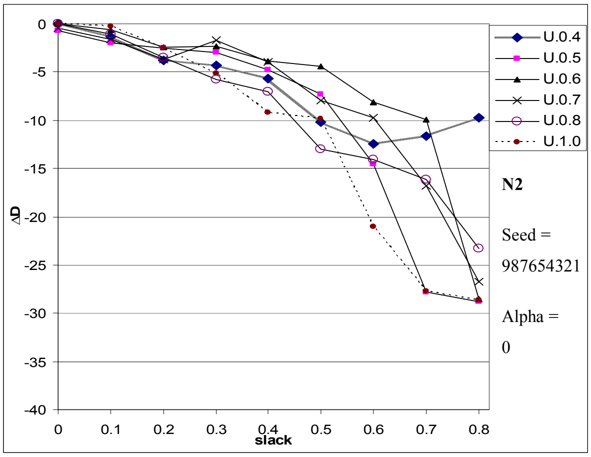

Figure 10. ÷D of networks with Alpha = 0 for N2 (Seed parameter = 987654321).

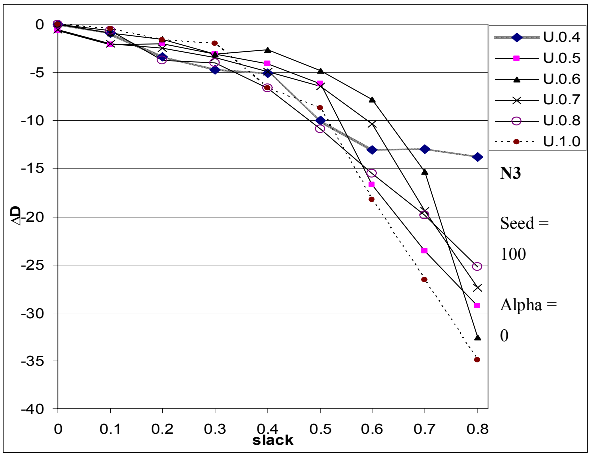

Figure 11. ÷D of networks with Alpha = 0 for N3 (Seed parameter = 100).

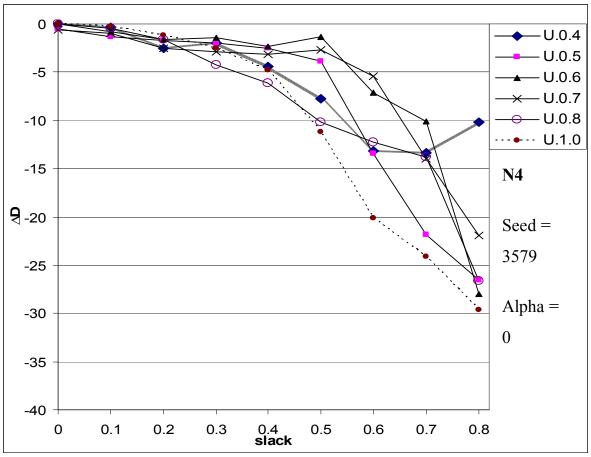

Figure 12. ÷D of networks with Alpha = 0 for N4 (Seed parameter = 3579).

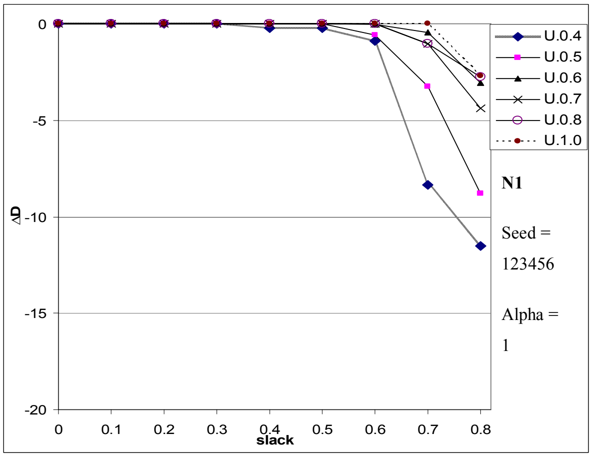

Figure 13. ÷D of networks with Alpha = 1 for N1 (Seed parameter = 123456).

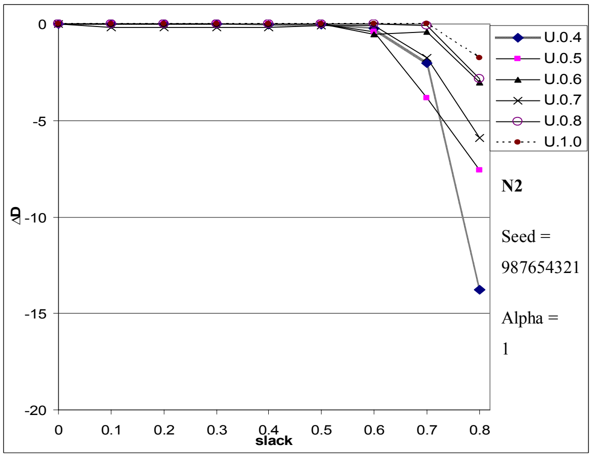

Figure 14. ÷D of networks with Alpha = 1 for N2 (Seed parameter = 987654321).

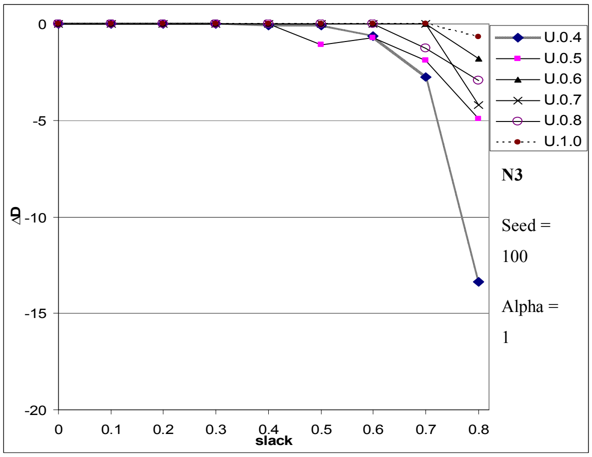

Figure 15. ÷D of networks with Alpha = 1 for N3 (Seed parameter = 100).

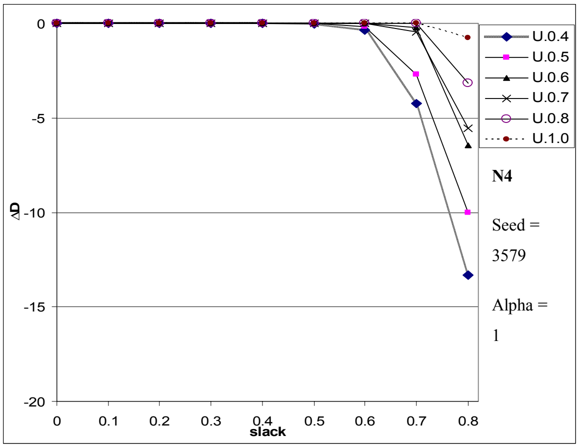

Figure 16. ÷D of networks with Alpha = 1 for N4 (Seed parameter = 3579).

Table 1. Average ÷*C* and ÷*D* of networks with ζ=0, 1

Slack	Alpha = 0		Alpha = 1	
	ΔC%	ΔD%	ΔC%	ΔD%
0	1.189094339	-0.209231862	0.001792464	-0.000277185
0.1	5.915411496	-0.962888524	0.003686283	-0.007729361
0.2	18.92519146	-2.304549027	0.005015377	-0.007924555
0.3	30.07653909	-3.042986299	0.005805637	-0.007609262
0.4	46.277338	-4.87835726	0.036562493	-0.023758658
0.5	59.60022267	-7.464423244	0.04862677	-0.065836352
0.6	74.80195614	-12.45952308	0.269307118	-0.197585068
0.7	85.7515248	-17.86446987	1.852873051	-1.494986418
0.8	87.97719211	-25.30072376	7.872259754	-5.633103577

3.3 Experiment Analysis

Figure 1-4 present ÷*C* of MENTOR networks with ζ=0 designed for N1-N4, respectively. It is clear that ÷*C* is more depend on *s* than ψ. For small *s*, *s* = 0.1, ÷*C* is small with average 5.9154%, and hardly change with ψ. As *s* increase, ÷*C* get worse and more depend on ψ. ÷*C* achieve maximum of 250% at *s*=0.8 and ψ= 1.

Figure 5-8 present ÷*C* of MENTOR networks with ζ=1 designed for N1- N4, respectively. For small *s*, *s* = 0.1, ÷*C* is very small with average 0.0037%, and also hardly change with ψ. ÷*C* get worse as *s* increase and has no obvious relation with ψ6

Figure 9-12 present ÷*D* of MENTOR networks with ζ=0 designed for N1-N4, respectively. For small *s*, *s* = 0.1, ÷*D* is small with average 0.9629%, and also hardly change with ψ. As *s* increase, ÷*D* get worse and more depend on ψ. ÷*D* achieve maximum of ~35% at *s*=0.8 and ψ= 1

Figure 13-16 present ÷*D* of MENTOR networks with ζ=1 designed for N1-N4, respectively. For small *s*, *s*=0.1, ÷*D* is very small with average 0.0077%, and hardly change with ψ. As *s* increase, ÷*D* get worse as *s* increase and has no obvious relation with ψ6

4. CONCLUSIONS

In this paper, we have explored the relations between design parameters and the efficiency of traffic assignment of MENTOR algorithm. 432 networks designed by MENTOR for 4 sets of 50 nodes each with equivalently distributed demand and randomly generated locations have been analyzed. For each of these networks, the performances at normal load and at threshold of congestion of MENTOR flow assignment are calculated. It is found that the performances of MENTOR strongly depend on "slack", a design parameter represented the different between the maximum and minimum utilization. For small value of slack, e.g. s ≤0.1, the performances of MENTOR keep very close to that of the optimum solution, while the maximum utilization hardly impact the performances. As the slack get larger the performances of MENTOR become worse and more depend on the maximum utilization.

5. ACKNOWLEDGMENTS

Our thanks to Assistant Professor Dr. Krung Sinapiromsaran for helping us to verify GLPK MathProg language.

6. REFERENCES

[1] A. Kershenbaum, P. Kermani, and G. A. Grover, MENTOR: An Algorithm for Mesh Network Topological Optimization and Routing, *IEEE Transaction on Communications*, Vol.39, April 1991, pp.503-513.

[2] R. S. Cahn, *Wide Area Network Design*. Morgan Kaufmann Publisher, San Francisco, CA, 1998.

[3] B. Fortz, J. Rexford, and M. Thorup, Traffic engineering with traditional IP routing protocols, *IEEE Commun. Mag.*, vol. 40, Oct. 2002, pp. 118–124.

[4] A. Sridharan, R. Gu´erin, and C. Diot, Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/IS-IS Networks, In *Proceedings of IEEE INFOCOM*, San Francisco, USA, April 2003.

[5] B. Fortz and M. Thorup, Internet traffic engineering by optimizing OSPF weights, in *Proc. IEEE INFOCOM*, vol. 2, Mar. 2000, pp.519–528.

[6] R. S. Cahn, The Design Tool: Delite (software), 1998, http://www.kt.agh.edu.pl/~mwagrow/projektowa niesiecitelekomunikacyjnych/DeLite/setupex.exe

[7] A. Makhorin. GLPK (GNU Linear Programming Kit), 2000, http://www.gnu.org/software/glpk/