GENERALISED ANALYSIS OF A QOS-AWARE ROUTING ALGORITHM

Lachlan L. H. Andrew* and A. A. N. Ananda Kusuma†
*Department of Electrical and Electronic Engineering
University of Melbourne, Parkville, Victoria, 3052, Australia
L.Andrew@ee.mu.oz.au +61 3 9344 9201 (phone) +61 3 9344 9188 (fax)
†Direktorat Teknologi Elektronika dan Informatika
Badan Pengkajian dan Penerapan Teknologi.
Jl. M. H. Thamrin 8, Jakarta 10340, Indonesia.
Ananda@inn.bppt.go.id +62 21 3169819 (phone) +62 21 3169811 (fax)

Abstract— **Modern communications services have strict quality of service (QoS) requirements, with separate constraints on bandwidth, delay and error tolerance. The task of finding a route through a network satisfying multiple QoS constraints is intractable, but increasingly important for modern communications applications. This paper investigates an approximate algorithm, which has previously been analysed for the case of two constraints, and presents a generalisation of this analysis to the case of an arbitrary number of constraints. The blocking rate of this algorithm is then empirically compared to several other techniques**

I. Introduction

For many applications, it is important that a communication connection provide sufficient *Quality of Service*. That is, the connection must provide sufficient bandwidth, low delay, low error rate, and many other requirements. The actual requirements vary from one application to another. For example, interactive video requires low delay and a large bandwidth, while audio and video playback are less sensitive to the absolute delay as long as the delay variation (jitter) is small. For all applications, the monetary cost of the link must also be minimised. What is common to all of these services is that there are *multiple* requirements which must be satisfied simultaneously by the link. This paper will investigate the QoS routing problem, which requires the routing of a connection subject to multiple constraints, and will be formally defined in Section II.

This work assumes centralised routing, where the routing node has complete knowledge of the state of the entire network, and the problem reduces to one of graph theory. However, the problem of finding a path through a graph satisfying multiple constraints is known to be NP complete [2], and so approximate techniques must be used. Section III describes several approximate approaches to the problem of QoS routing, focussing in Section III-A on the approach suggested by Jaffe, and studied in this paper. This section also reviews Jaffe's analysis of this algorithm for the case of two constraints. These results are bounds on the amount by which the route found by the heuristic is worse than the best possible route. Section IV presents new results which extend these bounds to the case of more than two constraints. Although the analytical results of Section IV bound the amount by which a suboptimal route is suboptimal, they do not indicate the blocking probability, which is the proportion of routing requests for which a feasible route is found. Blocking probabilities for a range of heuristics are described in Section V.

II. Formal Definition of QoS Routing

A communication network can be modelled as a graph $G = (V, E)$, consisting of a set of vertices, V (representing nodes), connected by edges, $E = \{(v_1, v_2) : v_1, v_2 \in V\}$ (representing links between nodes). Each edge, $e \in E$, has n associated costs, $L_i(e)$, $i = 1, \ldots n$. Traditional shortest path algorithms assume $n = 1$, but for the QoS routing problem, $n > 1$. A path from s to d through the network is a sequence of vertices $s = v_0, v_1, \ldots, v_m = d$, such that $(v_{i-1}, v_i) \in E$ for $i = 1, \ldots, m$. The ith cost of a path $p = v_0, \ldots, v_m$ is given by $L_i(p) = \sum_{j=1}^{m} L_i(e_j)$, where $e_i = (v_{i-1}, v_i)$. The QoS routing problem can then be stated thus: Given a graph $G = (V, E)$, a source and destination, $s, d \in V$, and a set of maximum allowable costs $L_i > 0$, $i = 1, \ldots, n$, find a path p from s to d through the graph such that $L_i(p) \leq L_i$ for all $i = 1, \ldots, n$. Such a path is said to be feasible.

III. Review of Heuristics

There are many efficient algorithms for the single cost shortest path problem (e.g., [1]), and so a common approach to the QoS routing problem is to summarise the n costs of a path into a single "overall cost". There are many ways to do this, including such things as weighted geometric means, but most shortest path algorithms assume that the cost is linear ($f(p_1p_2) = f(p_1) + f(p_2)$, where p_1p_2 denotes the concatenation of p_2 to p_1), so linear combinations of the form $f(p) = \sum_{i=1}^{n} d_i L_i(p)$ for some set of d_is are the most promising.

An interesting and quite effective approach is to use each cost in turn [3], [6]. Initially, a path is found using

0-7803-4984-9/98/$10.00

$f(p) = L_1(p)$. If that path is not feasible, a second path is found using $f(p) = L_2(p)$, and the process is continued until all n costs have been used. Each of these overall cost functions corresponds to calculating the weighted sum where one weight is 1 and the rest are 0. The computational complexity of this scheme is clearly limited to n times the complexity of the shortest path algorithm used. This scheme is called "one cost, best of 3" in Section V.

An alternative [8] is to avoid path computation at connection setup time. For each source/destination pair, a fixed set of paths can be stored, and for each connection the path which "best" satisfies the QoS requirements is selected.

A. *Linear combination of costs*

A path which is not strictly feasible is still often useable. Jaffe [4] has investigated an approach in which all feasible paths are considered equivalent, while paths violating a constraint are assessed according to the amount by which the constraint is violated. For example, if the transmission delay is imperceptible by the user, its actual value is irrelevant, but if it is perceptible, the connection is still useable, but the delay should be minimised. The objective to be minimised is thus

$$f(p) = \sum_{i=1}^{n} \max(L_i(p), L_i) \tag{1}$$

where L_i is the constraint on the ith cost. Since this objective function is not linear, it cannot be used in standard shortest path algorithms. Instead, the algorithm minimises the weighted sum of the costs:

$$g(p) = \sum_{i=1}^{n} d_i L_i(p) \tag{2}$$

for some weights d_i. By appropriate choice of the d_is, this can ensure that $f(p)$ is not too far from its optimal value.

Because the true objective, $f(p)$, is an unweighted sum, the actual edge costs, $L_i(e)$, must be scaled to reflect the relative importance of each particular cost. For example, it may be sensible to scale the $L_i(e)$s such that the average value over the entire network is approximately the same for each i, which is the form of scaling assumed in Section V. More sophisticated schemes are also possible [7].

A key contribution of [4] was to derive bounds on the ratio $f(p')/f(p^*)$, where $f(\cdot)$ is given by (1), p' is the path which minimises $g(p)$ of (2), i.e., the path found by the algorithm, and p^* is the path which minimises $f(p)$, the true objective. These results were only derived in the case of two constraints per link, and are summarised below. Section IV describes our extension of these results to the general case.

Theorem 1: For $d_i = 1$ for all i,

$$f(p')/f(p^*) \leq 1 + \max\{L_1, L_2\}/(L_1 + L_2) \leq 2.$$

Lemma 1: For $d_1 = 1$,

$$f(p')/f(p^*) \leq 1 + \max\{L_1/d_2, L_2 d_2\}/(L_1 + L_2)$$

if a feasible path exists.

Theorem 2: The bound of Lemma 1 is minimised when $d_2 = (L_1/L_2)^{1/2}$ in which case

$$f(p')/f(p^*) \leq 1 + (L_1 L_2)^{1/2}/(L_1 + L_2) \leq 1.5.$$

IV. Extended Analysis: Multiple Constraints

This section will present an extension of the results quoted in Section III-A to the case of n constraints. For simplicity of notation, summation over i in this section is implicitly from 1 to n, a "path" is assumed to be a path through a graph $G = (V, E)$ from a source $s \in V$ to a destination $d \in V$, and functions $f(\cdot)$ and $g(\cdot)$ are those of (1) and (2) respectively.

A. *Equal weighting*

In the previous section it was indicated that, in the case of two constraints, minimising the simple sum of the costs of the edges in the path yields a path p' for which the measure $f(p')$ is worse than the optimum by a factor of at most two. That result also holds for the case of an arbitrary number of constraints, as shown by the following theorem.

Theorem 3: Let p^* be the path which minimises $f(p^*)$ and p' be the path which minimises $g(p')$ with $d_i = 1$ for all i. Then

$$\frac{f(p')}{f(p^*)} \leq 2 - \frac{\min_i L_i}{\sum_i L_i} \tag{3}$$

Proof: If p' is feasible, then $f(p') = f(p^*)$, and the result is proved. Otherwise $L_m(p') > L_m$ for some m, whence

$$\begin{aligned} \frac{f(p')}{f(p^*)} &\leq \frac{g(p') + \sum_{i \neq m} L_i}{f(p^*)} \\ &\leq \frac{g(p')}{g(p^*)} + \frac{\sum_{i \neq m} L_i}{\sum_i L_i} \\ &\leq 1 + \frac{\sum_{i \neq m} L_i}{\sum_i L_i} \\ &\leq 2 - \frac{\min_i L_i}{\sum_i L_i} \end{aligned}$$

as required. The second inequality occurs since $f(p^*) \geq g(p^*)$ and $f(p^*) \geq \sum_i L_i$. ∎

Note that this bound is always less than 2 but greater than or equal to $2 - 1/n$, with equality occurring when all of the constraints are equal, $L_i = L_j \; \forall i, j$.

B. *Unequal weighting*

In the case of two constraints, a substantial reduction in the upper bound on $f(p')/f(p^*)$ was achieved by minimising a weighted sum of the individual costs. An upper bound for the case of a weighted sum for more than two constraints, when a feasible path is known to exist, is given by the following generalisation of Lemma 1.

Lemma 2: Let p^* be the path which minimises $f(p^*)$ and p' be the path which minimises $g(p')$. If p^* is feasible, then

$$\frac{f(p')}{f(p^*)} \leq 1 + \max_m \frac{\sum_{i \neq m} (d_i/d_m) L_i}{\sum_i L_i}. \tag{4}$$

Proof: If p' is feasible, then $f(p') = f(p^*)$, and the result is proved. Otherwise there is a nonempty set V for which $L_i(p') > L_i$ for all $i \in V$. Let $m = \operatorname{argmin}_{i \in V} d_i$. Thus

$$\begin{aligned} f(p') &\leq \sum_{i \in V} L_i(p') + \sum_{i \neq m} L_i \\ &\leq \frac{g(p')}{d_m} + \sum_{i \neq m} L_i \\ &\leq \frac{g(p^*)}{d_m} + \sum_{i \neq m} L_i \\ &\leq \sum_{i \neq m} \frac{d_i}{d_m} L_i + \sum_i L_i \end{aligned}$$

where the third inequality follows since p' minimises $g(.)$ and the fourth inequality follows since p^* is feasible, whence $L_i(p^*) \leq L_i$. Since p^* is feasible, $f(p^*) = \sum_i L_i$, and so

$$\frac{f(p')}{f(p^*)} \leq 1 + \frac{\sum_{i \neq m} (d_i/d_m) L_i}{\sum_i L_i}. \tag{5}$$

However, since m is required to be the index of a constraint violated by p', the worst-case m must be assumed. Thus the bound must be maximised over all possible m. ■

C. Optimal weighting

Clearly the bound given in (4) can exceed that of (3) for some d_i and thus d_i must be chosen with care. For $n > 2$, the optimal weight d_i cannot in general be found in closed form, and will depend on all of the constraints L_j. This section presents suitable values for d_i which, although not generally optimal, are easy to compute and for which (4) is tighter than (3).

Lemma 3: Let p^* be the path which minimises $f(p^*)$ and p' be the path which minimises $g(p')$. If p^* is feasible, and d_i is of the form $d_i = L_i^{-1/k}$ with $k > 1$, then

$$\frac{f(p')}{f(p^*)} \leq 1 + \frac{k-1}{k^{k/(k-1)}} \frac{\|\mathbf{L}\|_{(k-1)/k}}{\|\mathbf{L}\|_1} \tag{6}$$

where $\mathbf{L} = (L_1, L_2, \ldots, L_n)$ and $\|\mathbf{L}\|_a = (\sum_i L_i^a)^{1/a}$, and moreover

$$\frac{f(p')}{f(p^*)} \leq 1 + \left(1 - \frac{1}{k}\right) \left(\frac{n}{k}\right)^{1/(k-1)}. \tag{7}$$

Note that, although $\|\mathbf{L}\|_a$ has the form of a Minkowski norm, it is not a norm for $a < 1$ since it does not satisfy the triangle inequality.

Proof: By Lemma 2,

$$\frac{f(p')}{f(p^*)} \leq 1 + \max_m \frac{\sum_{i \neq m} (L_m L_i^{k-1})^{1/k}}{\sum_i L_i}.$$

Letting $A = \sum_i L_i^{(k-1)/k}$ gives

$$\begin{aligned} \max_m \sum_{i \neq m} (L_m L_i^{k-1})^{1/k} &= \max_m (L_m^{1/k} A - L_m) \\ &\leq \max_s (s^{1/k} A - s) \end{aligned} \tag{8}$$

for some real number s. The unique stationary point of $s^{1/k} A - s$ is $(k-1)(A/k)^{k/(k-1)}$ at $s = (A/k)^{k/(k-1)}$, which is a maximum if $k > 1$. Substituting in for A gives

$$\begin{aligned} \frac{f(p')}{f(p^*)} &\leq 1 + \frac{(k-1)\left((1/k)\sum_i L_i^{(k-1)/k}\right)^{k/(k-1)}}{\sum_i L_i} \\ &\leq 1 + \frac{k-1}{k^{k/(k-1)}} \frac{\|\mathbf{L}\|_{(k-1)/k}}{\|\mathbf{L}\|_1}, \end{aligned}$$

which establishes relation (6). To establish inequality (7), note that for $a < 1$, $\|\mathbf{L}\|_a / \|\mathbf{L}\|_1$ has a maximum of $n^{(1/a)-1}$ when $L_i = L_j$ for all i, j. Substituting $n^{1/(k-1)}$ for the last factor in (6) and rearranging gives the required result. ■

Lemma 3 replaces the task of selecting n values for the d_is with the task of finding a single value for k. This is done by the following theorem, which is the main result of this section.

Theorem 4: For $n \geq 2$ the value of k in Lemma 3 which minimises the bound (7) is $k = n$, for which

$$f(p')/f(p^*) \leq 2 - 1/n. \tag{9}$$

Proof: Differentiating the right hand side of (7) gives

$$\frac{d}{dk}\left(1 + \left(1 - \frac{1}{k}\right)\left(\frac{n}{k}\right)^{1/(k-1)}\right) = \left(\frac{n}{k}\right)^{1/(k-1)} \left(\frac{\log(k/n)}{k(k-1)}\right)$$

which has a unique zero at $k = n$. This stationary point is a minimum since the derivative is positive for $k > n$. Substituting $k = n$ in (7) gives the required result. ■

As the number of constraints n increases, the bound (9) for well chosen weights approaches 2, which is an upper bound in the case of uniform weighting. This indicates that simple uniform weighting becomes more feasible for a larger number of constraints.

TABLE I
LINK COSTS AND PATH COST TARGETS

	Delay (s)	Delay variance (s)	Loss Ratio
Nominal link cost	5.5^{-4}	3.0^{-3}	1.0^{-5}
Path constraint	5.5^{-3}	3.0^{-2}	1.0^{-4}

V. SIMULATION

In this section, simulation results are presented to compare the blocking performance of the previous section with several alternative schemes.

For this work, graphs were generated by the pure random algorithm of [9], in which the probability of a link is independent of the position of the nodes. The average degree of the nodes was 3.5. The costs assigned on each link as the link costs and QoS constraints are taken from data traffic class used in [3], and shown in Table I. The link costs are set to have log-uniform distribution with the range 0.1 to 10 times the nominal value. Note that loss ratio is not linear ($f(p_1p_2) \neq f(p_1) + f(p_2)$) and so $L_3 = -\log(1-\text{loss ratio})$, which is linear and monotonic increasing in loss ratio, was used in its stead.

As mentioned in Section III-A, the emphasis given to the ith cost by the shortest path algorithm depends on the average magnitude of the ith cost of the links in the network. If infeasible calls are blocked, all of the n costs are equally important. For this reason, additional simulations were conducted in which the costs of the network were scaled so that the nominal value was 1 in each case.

The routing algorithms compared were as follows: *minimum hop* selects the path containing the fewest links; *one cost* selects the path of least delay only; *one cost, best of 3* finds the paths minimising delay, delay variance and loss, and selects the first feasible of these; *Li, sqrt(Li),* and *cubr(Li)* select the path minimising $g(p')$ with $d_i = L_i^{-1}$, $L_i^{-1/2}$ and $L_i^{-1/3}$ respectively (cubr(Li) minimises the bound of (7)); *cubr(Li), best of 2, 3* calculate the paths with the two or three smallest values of $g(p)$, and select the first feasible of these; *brute force* performs an exhaustive search for a feasible solution, and thus provides a lower bound on the blocking probability, but can only be used for small networks.

The simulation was conducted on ten 10 node and 100 node randomly connected networks, and a connection attempt was made for each origin-destination pair in each network. The shortest path algorithm used is based on Dijkstra [1] with $O(n^2)$complexity. The second and third shortest paths were obtained based on the *k*th shortest path algorithm by Katoh *et al.* [5].

Clearly the blocking probability depends on the QoS requested; the "looser" the requirements, the lower the blocking probability. Since all of the algorithms tested depend only on the ratios of the L_is, rather than their actual values, the path selected does not depend on the absolute looseness of the constraints. Blocking was evaluated by finding a path, p', for some (L_1, L_2, L_3), and then determining whether p' would be feasible for a request of $(\lambda L_1, \lambda L_2, \lambda L_3)$, where λ is the looseness.

The problems associated with combining costs when incompatible units are used for the different costs are illustrated in Figures 1 and 2. In the case where costs are not normalised, schemes which use combined cost and have a good bound on $f(p')/f(p*)$ have poor blocking performance. The reason is that without normalisation too much emphasis is placed on the particular cost with the largest average magnitude. In this case, schemes *Li* and *sqrt(Li)* outperform *cubr(Li)*, although the latter minimises $f(p')/f(p*)$, because they perform a limited degree of scaling, since the mean value of L_i is proportional to the mean value of $L_i(e)$. The remarkably good performance of the scheme *one cost, best of 3* comes because this selects three paths which are all typically short, but which are essentially independent since they are based on different costs. In contrast, the three paths selected by *cubr(Li), best of 3* are likely to be very similar, since they attempt to minimise the same function, $g(p)$. This is similar to the concept of diversity in the reception of radio signals. It can be observed also in Figures 1 and 2 that without normalisation, the performance of the routing schemes relative to each other changes as the network size increases.

In the case where costs are normalised (Figures 3 and 4), the results are more promising. All routing schemes which use a linear combination of all costs (i.e., base decisions on all of the available information) show better performance than all of those which only consider one metric at a time (*min hop, one cost* and *one cost, best of 3*). The difference is more marked for larger networks, which are more representative or real networks. Even with normalisation, there is no clear improvement in blocking performance gained by using weights of $L_i^{-1/3}$, which was shown to optimise a performance bound, rather than simply using L_i. (Note that in this case, the L_is are all equal by the normalisation, and so this amounts to using an unweighted sum of the link costs.)

VI. CONCLUSION

This paper has presented a bound on the performance of a heuristic algorithm for finding a route through a network satisfying multiple QoS constraints. In particular it has shown that the "overall cost" of the path found by minimising the sum of the individual costs is at most twice the overall cost of the best possible path. Also, if a feasible path exists, the overall cost of the path minimising an appropriate weighted sum of the n individual costs is no more than $2 - 1/n$ times the overall cost of the best possible path.

In addition, the blocking performance of several QoS

Fig. 1. Blocking probability vs looseness for 10 node network without scaling

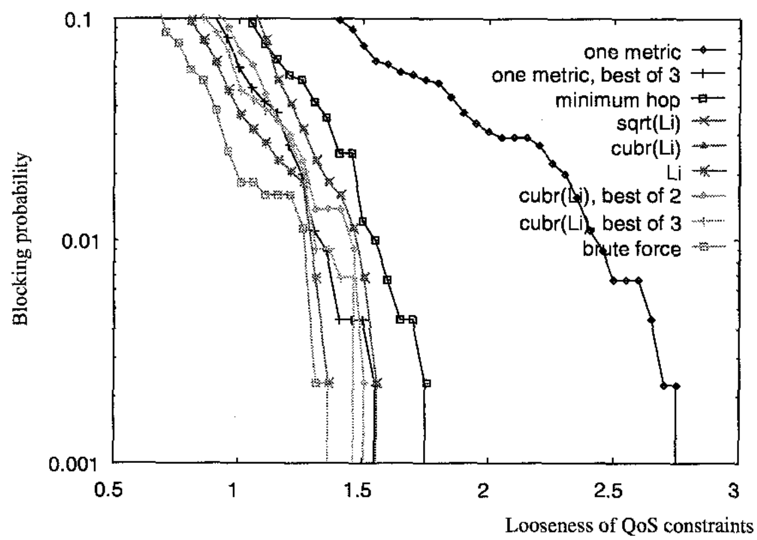
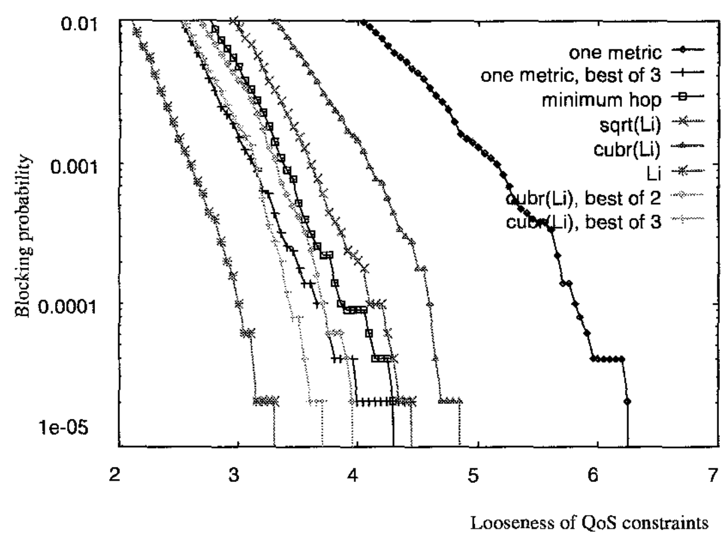


Fig. 2. Blocking probability vs looseness for 100 node network without scaling



routing algorithms has been investigated. It has been shown that the algorithm which minimises the bound on the overall cost also yields a low blocking probability, provided appropriate scaling of the individual costs is used. The importance of diversity in path selection has also been highlighted.

References

[1] E. W. Dijkstra, "A note on two problems in connexion with graphs", *Numerische Mathematik*, vol. 1, pp. 269-271, 1959.

[2] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, 1979.

[3] A. Iwata, R. Izmailov, H. Suzuki and B. Sengupta, "PNNI routing algorithms for multimedia ATM internet", *NEC Research & Development*, vol. 38, no. 1, pp. 60-73, Jan. 1997.

[4] J. M. Jaffe, "Algorithms for finding paths with multiple con-

Fig. 3. Blocking probability vs looseness for 10 node network with scaling

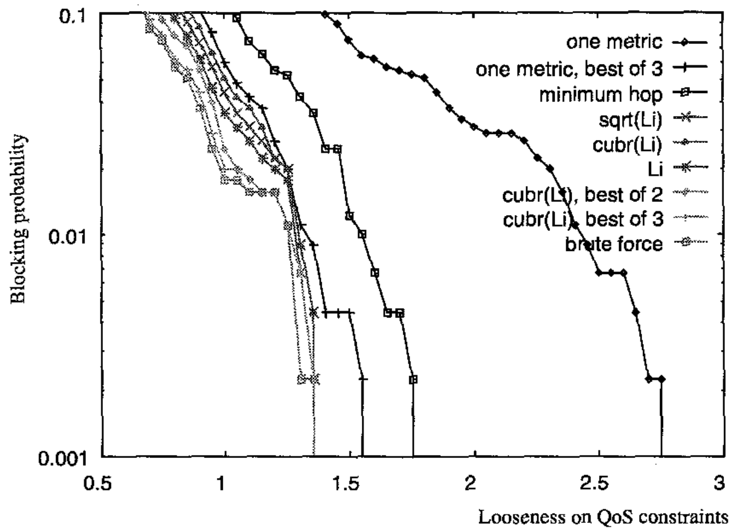
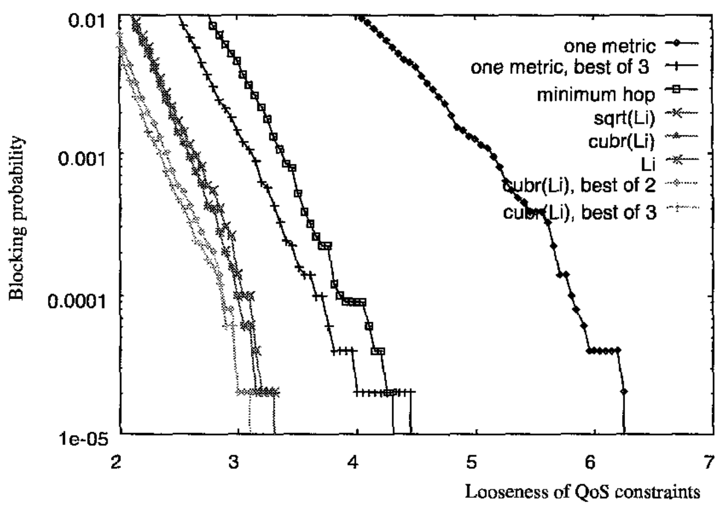


Fig. 4. Blocking probability vs looseness for 100 node network with scaling



straints", *Networks*, vol. 14, pp. 95-116, 1984.

[5] N. Katoh, T. Ibaraki and H. Mine, "An efficient algorithm for k shortest simple paths", *Networks*, vol. 12, pp. 411-427, 1982.

[6] W. C. Lee, M. G. Hluchyi and P. A. Humblet, "Routing subject to quality of service constraints in integrated networks", *IEEE Network*, vol. 9, no. 4, pp. 46-55, July/August 1995.

[7] G. S. Russell and R. R. Mankarious, "Methodology for obtaining a cost estimate in a multiple media network architecture", *MILCOM'90*, vol. 2, pp. 719-723, 1990.

[8] R. Vogel, R. G. Herrtwich, W. Kalfa, W. Wittig and L. C. Wolf, "QoS-based routing of multimedia streams in computer networks", *IEEE J. Select. Areas Commun.*, vol. 14, no. 7, pp. 1235-1244, Sept. 1996.

[9] E. W. Zegura, K. L. Calvert and S. Bhattacharjee, "How to model an internetwork", in *Proc. IEEE INFOCOM'96*, vol. 2, pp. 594-602, 1996.