Improving Success Ratio in Multi-Constraint Quality of Service Routing

P. Khadivi, S. Samavi, H. Saidi and T. D. Todd

Abstract: Multi-constraint quality-of-service routing will become increasingly important as the Internet evolves to support real-time services. It is well known however, that optimum multi-constraint QoS routing is computationally complex, and for this reason various heuristics have been proposed for routing in practical situations. Among these methods, those that use a single mixed metric are the most popular. Although mixed metric routing discards potentially useful information, this is compensated for by significantly reduced complexity. Exploiting this tradeoff is becoming increasingly important where low complexity designs are desired, such as in battery operated wireless applications. In this paper, a novel single mixed metric multi-constraint routing algorithm is introduced. The proposed technique has similar complexity compared with existing low complexity methods. Simulation results are presented which show that it can obtain better performance than comparable techniques in terms of generating feasible multi-constraint QoS routes.

Keywords: QoS routing, Multi-constraint routing, Quality of service, Computer networks.

1 Introduction

Routing is one of the most basic and widely studied problems in computer networking. The current Internet however, uses only best-effort routing [1] and thus supports services without any quality of service (QoS) guarantees. For applications such as FTP and HTTP, it is clear that best-effort routing is sufficient. Applications such as real-time audio and video however, require strict performance guarantees in order to achieve acceptable performance. For these types of applications, a fundamental issue is how to find a feasible path that satisfies multiple constraints. This problem is known as multi-constraint QoS routing.

QoS routing is very complex, and dealing with multiple QoS requirements makes this problem NP-Complete [2]. Link and path constraints are the two types of QoS constraints considered. Link constraints specify the restrictions on the use of the individual links, while path constraints focus on the end-to-end QoS attributes of the entire path. In the multi-constraint case, each network link has multiple weights which can be classified as additive, multiplicative or concave. For additive weights, the end-to-end weight of the path is the sum of the individual link values. Delay is an example of

additive weights. A multiplicative path weight is the product of the link weights along the path. Path reliability is an example of multiplicative weights. Bandwidth belongs to the class of concave weights. The overall bandwidth of the path is equal to the minimum bandwidth of the links. Dealing with concave weights and constraints is very easy. In the bandwidth case for example, it is sufficient to delete the links with bandwidth less than the required value. It can be proven that optimum QoS routing with more than one constraint involving additive and/or multiplicative weights is an NP-Complete problem. For this reason it is difficult to have an algorithm which is computationally efficient in all possible situations [2].

In this paper we use *weight* and *metric* as synonymous. Since it is possible to transform the multiplicative weight case into the additive case by taking logarithms, we only consider cases with several additive constraints. Hence, the Multi-Constraint Path (MCP) problem can be stated as follows.

Definition: Consider a network that is represented by a graph $G=(V, E)$, where V is the set of nodes and E is the set of links. Each link $(i, j) \in E$ has K additive non-negative QoS weights, $w_m(i, j)$, $m = 1,...,K$. Given K constraints, C_m, $m = 1,2,...,K$, the MCP problem is to find a path P from a source node S to a destination node D such that,

$$w_m(P) \triangleq \sum_{(i,j)\in P} w_m(i, j) \le C_m, \quad \text{for} \quad m = 1, 2, ..., K \tag{1}$$

There are various methods for solving the MCP problem. *Depth-First Search* (DFS) is one approach. Although this method is able to find a feasible path if one exists, its worse-case time complexity is exponential. Shin and his coworkers have suggested a heuristic based on DFS which limits the number of crankbacks to control the worst case time complexity. Even though the time complexity is reduced it is possible that the algorithm will not find a feasible path even when one exists [3].

There are a number of methods that use a single mixed metric instead of dealing with multiple link weights. According to Wang and Crowcroft [2], a *single mixed metric* method at best can be used as a criterion in path selection but it does not contain sufficient information alone to determine if QoS requirements are satisfied. However, using a mixed metric can reduce the time complexity because we can employ a *single source single destination shortest path* algorithm such as Dijkstra's algorithm, but this is not as effective as it may appear. When a single mixed metric is used for routing, some information is lost [4][5][6][7]. The TAMCRA algorithm presented in [5] uses a single metric and a *k-shortest path* algorithm in order to solve a MCP problem. The k-shortest path algorithms are able to find multiple shortest paths between a given source and a destination. This method reduces the performance shortcomings of using a mixed metric. The H_MCOP algorithm presented in [6] uses mixed metrics. H_MCOP is used for solving a *Multi-Constraint Optimal Path* (MCOP) problem. The MCOP problem is a type of MCP, which tries to find a feasible optimal path based on a cost associated with each link and path.

The algorithms presented in [8][9][10][11] also use a mixed metric such as $w = w_1 + \alpha w_2$ for each link, where w_1 and w_2 are the weights for the two-constraint case, and w is the mixed weight of a link. These algorithms employ a method called *Lagrange Relaxation* and try to find the best value for α, leading to a feasible path after applying Dijkstra's algorithm. These methods however, need multiple runs of Dijkstra's algorithm in order to find α.

Yuan and Liu use a different definition of an optimal QoS path [12]. They present an extended version of the Bellman-Ford Algorithm to find all of the optimal QoS paths between a source and a destination. Then a feasible path is selected if one exists. In [13] all metrics except one are changed to quantized integer values and then a polynomial time solution is presented for these new metrics. There are also methods that are based on *distributed routing and flooding* [14][15].

In order to improve the performance of the routing algorithm, different single mixed metrics have been presented in the literature. Khadivi et al. have proposed in [16] to take into account *variations* among different weights of a path in the routing procedure. Simulation results show that this improves the *success ratio* of the algorithm.

In this paper, we propose a novel algorithm for the MCP problem using a single mixed metric. Simulation results show that our proposed method can have better performance than the existing algorithms with a similar computational complexity. We compared our results with those works that have used similar routing strategies to that of ours. The remainder of the paper is organized as follows. In Section 2 a brief review is presented of MCP routing based on a single mixed metric. In Section 3 our new algorithm is proposed. Simulation results are presented in Section 4 and some concluding remarks are given in Section 5.

2 Routing Based on a Single Mixed Metric

In [4] the following mixed metric was introduced for the two-constraint problem,

$$W(e) = d_1 w_1(e) + d_2 w_2(e), \qquad (2)$$

where *e* is a link with two different metrics, $w_1(e)$ and $w_2(e)$. Here, d_1 and d_2 are two constants and $W(e)$ is the mixed metric associated with link *e*. This formulation is known as Jaffe's method. If a graph has weights given by $W(e)$, and a *shortest path algorithm* such as Dijkstra Algorithm is used, then for a path $p = (S, u, ..., v, D)$, the path weight $W(p)$ can be written as follows,

$$W(p) = [d_1 w_1(S, u) + d_2 w_2(S, u)] + ...$$
$$+ [d_1 w_1(v, D) + d_2 w_2(v, D)] = d_1 [w_1(S, u) + \qquad (3)$$
$$... w_1(v, D)] + d_2 [w_2(S, u) + ... w_2(v, D)],$$

where $W(p)$ is minimal. Equation (3) describes parallel lines such as,

$$d_1 w_1(p) + d_2 w_2(p) = c. \qquad (4)$$

Figure 1 illustrates the parallel lines of Equation (4) and shows how this method searches for a feasible path. In this figure the horizontal axis is associated with the w_1 metric and the vertical axis is associated with w_2. The objective is to find a path, p^*, such that,

$$w_1(p^*) \le C_1 \text{ and } w_2(p^*) \le C_2. \qquad (5)$$

Constraints C_1 and C_2 are shown in Figure 1 as dashed lines. Each path *p* between source *S* and destination *D* has weights $w_1(p)$ and $w_2(p)$. Therefore, associated with each path is an achievable point in the $w_1(p)$-$w_2(p)$ plane. In Figure 1 there are some examples of these paths shown as black points. Clearly all of the points inside the rectangular region are associated with feasible paths and the Jaffe method searches for a path with the minimum $W(p)$. Figure 1(a) illustrates a situation where the Jaffe method finds a feasible path, i.e., the point shown closest to the origin. Since using mixed metrics discards some useful information, it is possible for the Jaffe method to fail. Figure 1(b), shows a simple case where this happens, i.e., the right-most path found does not satisfy the constraints.
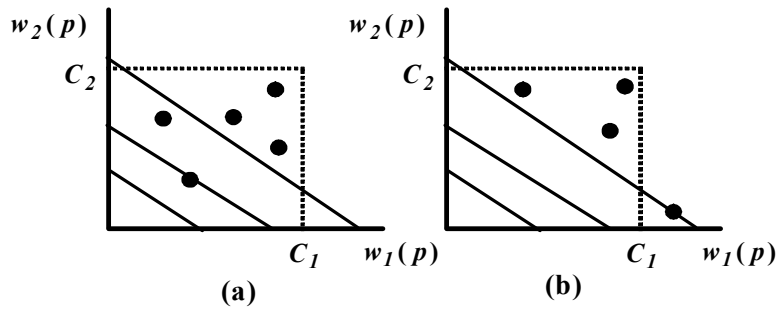
Fig. 1 Use of a single mixed metric. (a) feasible solution is found (b) algorithm fails.

An approach for improving the chance of finding a feasible path is to modify the path weight in order to influence the search region. For example, if d_1 and d_2 are selected based on the following equation,

$$\frac{d_1}{d_2} = \frac{C_2}{C_1}, \qquad (6)$$

then before leaving the feasible path region, half of it will be searched [5]. It is also possible to define the following weight for a path,

$$W(p) = \left(\frac{w_1(p)}{C_1}\right)^2 + \left(\frac{w_2(p)}{C_2}\right)^2. \qquad (7)$$

Figure 2 illustrates how the feasible path search region is affected by an algorithm that uses Equation (7) to find its mixed weights. It is clear from this figure that when we use squaring in the mixed weight computation, it will take longer for the algorithm to leave the feasible path region. Accordingly, for a K-constraint problem the following mixed weight can be defined as,

$$W_\lambda(p) = \sum_{j=1}^{K} \left(\frac{w_j}{C_j}\right)^\lambda, \qquad (8)$$

where p is a path that minimizes the mixed metric W_λ for a given $\lambda \geq 1$. It was proven in [6] that after using a mixed metric algorithm for a MCP problem in order to find a path p, when a feasible path p^* exists then,

$$\begin{cases} w_j(p) \leq C_j & \text{for some } j \\ w_j(p) \leq \sqrt[\lambda]{K}\, C_j & \text{for others} \end{cases} \qquad (9)$$

The larger values of λ, lead to a higher probability of obtaining feasible results. The reason is that routing based on $\lambda > 1$ is naturally nonlinear. Therefore, it is impossible to find a polynomial time algorithm for this problem. For this reason heuristic methods are necessary for the routing procedure. It has also been proven that when $\lambda \to \infty$ the following metric can be used [6],

$$W_\infty(p) = \max\left\{\frac{w_1(p)}{C_1}, \ldots, \frac{w_K(p)}{C_K}\right\} \qquad (10)$$

In the following section, a new mixed metric is proposed which can improve the performance of the routing strategies based on single mixed metrics.
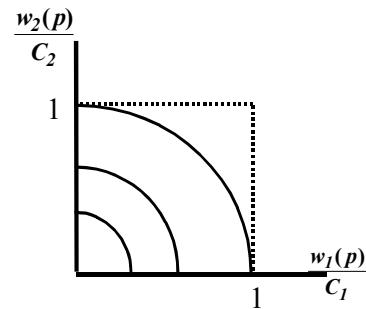

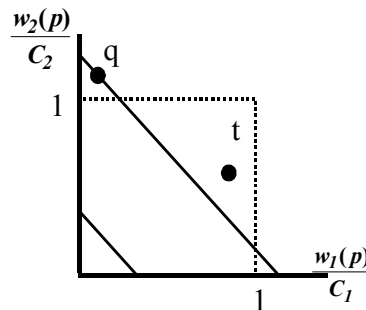**Fig. 2** Use of a quadratic single mixed metric.

Fig. 3 A feasible path exists but is not found.

3 Proposed Method

When mixed metric routing is used a part of the information is lost due to the process of mixing the metrics. Therefore, it is possible that while feasible paths do exist in the network none of them are discovered. In this section we propose a routing strategy that circumvents this shortcoming when mixed metrics are used.

Let us consider the mixed metric routing method when there are only two constraints and $\lambda = 1$. As shown in Figure 3 we assume a situation where two paths, t and q, exist between the source and destination nodes. Path t is a feasible path since it satisfies the constraints. On the other hand, path q minimizes the mixed metric but it does not satisfy the constraints. The mixed metric of path t is as follows,

$$W(t) = \frac{w_1(t)}{C_1} + \frac{w_2(t)}{C_2}, \qquad (11)$$

and for the path q we have

$$W(q) = \frac{w_1(q)}{C_1} + \frac{w_2(q)}{C_2} . \qquad (12)$$

Since $W(q) < W(t)$, it is clear that

$$\frac{w_1(q)}{C_1} + \frac{w_2(q)}{C_2} < \frac{w_1(t)}{C_1} + \frac{w_2(t)}{C_2} . \qquad (13)$$

Path t satisfies both constraints and therefore

$$\frac{w_1(t)}{C_1} \leq 1 \quad \text{and} \quad \frac{w_2(t)}{C_2} \leq 1 . \qquad (14)$$

Based on Equation (9), path q at least satisfies one of the constraints, hence

$$\frac{w_1(q)}{C_1} \leq 1 \quad \text{and} \quad \frac{w_2(q)}{C_2} > 1 . \qquad (15)$$

By rearranging Equation (13) we can write

$$\frac{w_1(q)}{C_1} < \frac{w_1(t)}{C_1} + \frac{w_2(t)}{C_2} - \frac{w_2(q)}{C_2} , \qquad (16)$$

but since

$$\frac{w_2(t)}{C_2} < \frac{w_2(q)}{C_2} , \qquad (17)$$

therefore,

$$\frac{w_1(q)}{C_1} < \frac{w_1(t)}{C_1} . \qquad (18)$$

Figure 4 shows the relative position of these four values. We now define the average of $w_1(t)/C_1$ and $w_2(t)/C_2$ as $\mu(t)$, i.e.,

$$\mu(t) \hat{=} \frac{1}{2}\left[\frac{w_1(t)}{C_1} + \frac{w_2(t)}{C_2} \right], \qquad (19)$$

and we define

$$\Delta(t) \hat{=} \left(\frac{w_1(t)}{C_1} - \mu(t) \right)^2 + \left(\frac{w_2(t)}{C_2} - \mu(t) \right)^2 . \qquad (20)$$

In the following Theorem, it is shown that for the example of Figure 3 we have:

$$\mu(t) > \mu(q) \quad \text{and} \quad \Delta(t) < \Delta(q) . \qquad (21)$$

Theorem 1: If in a two-constraint MCP problem a routing algorithm based on $W_1(p)$ finds an unfeasible path $q \in \pi$ where a feasible path $t \in \pi$ exists, then the following condition must hold:

$$\Delta(t) < \Delta(q) \qquad (22)$$

Proof: Because path t is feasible it satisfies both constraints. On the other hand, path q is unfeasible and based on Equation (9) in a two-constrained case, only one of the constraints is satisfied. Without loss of generality let us assume that:

$$\frac{w_1(t)}{C_1} \leq 1 \quad \text{and} \quad \frac{w_2(t)}{C_2} \leq 1$$

$$\frac{w_1(q)}{C_1} \leq 1 \quad \text{and} \quad \frac{w_2(q)}{C_2} > 1 \qquad (23)$$

The routing algorithm based on $W(p)$ finds path q. Therefore, $W(q) < W(t)$. Hence, it can be shown that
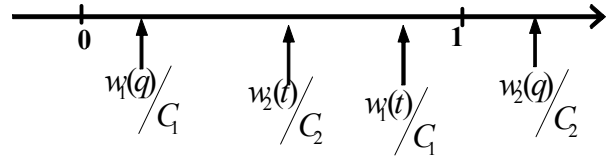

Fig. 4 Relative positions of the parameters in Equations 14 and 15.

$$\frac{w_2(q)}{C_2} > \max\left(\frac{w_1(t)}{C_1}, \frac{w_2(t)}{C_2} \right)$$

$$\frac{w_1(q)}{C_1} < \min\left(\frac{w_1(t)}{C_1}, \frac{w_2(t)}{C_2} \right) \qquad (24)$$

Therefore it is obvious that the normalized weights of path t are closer to each other than those of q. In other words

$$\left| \left(\frac{w_1(q)}{C_1} \right) - \left(\frac{w_2(q)}{C_2} \right) \right| > \left| \left(\frac{w_1(t)}{C_1} \right) - \left(\frac{w_2(t)}{C_2} \right) \right| \qquad (25)$$

By manipulating functions defined by Equations (19) and (20), it can be shown that:

$$\Delta(p) = \frac{1}{2}\left[\left(\frac{w_1(p)}{C_1} \right) - \left(\frac{w_2(p)}{C_2} \right) \right]^2 \qquad (26)$$

Therefore, based on (25) and (26), $\Delta(t) < \Delta(q)$. ∎

In existing algorithms, routing for a path, p, is based only on $\mu(p)$. We propose to use $\Delta(p)$ as well as $\mu(p)$ in the routing decisions. In the general multi-constraint case we define

$$\mu(p) = \frac{1}{K}\sum_{i=1}^{K} \frac{w_i(p)}{C_i} , \qquad (27)$$

and

$$\Delta(p) = \sum_{i=1}^{K}\left(\frac{w_i(p)}{C_i} - \mu(p) \right)^2 . \qquad (28)$$

Now we define the following mixed metric,

$$G(p) \hat{=} \mu(p)\left[\Delta(p) + \varepsilon \right], \qquad (29)$$

where ε is a constant and $0 \leq \varepsilon \leq 1$. By using $G(p)$ as a single metric, both $\mu(p)$ and $\Delta(p)$ are considered. A constant coefficient such as ε allows $\mu(p)$ to have a direct effect on the solution found. In addition, the product $\mu(p)\Delta(p)$ is taken into account and obviously

larger values of $\mu(p)$ and $\Delta(p)$ increases $G(p)$. The constant ε achieves the proper weighting for the role of $\mu(p)$. The best value of ε is dependent upon network size and its weights. In the next section simulation results are presented which give an indication of how ε should be chosen in practical situations.

A complete description of the routing algorithm is shown in Figure 5. The algorithm operates as a modified version of Dijkstra shortest path algorithm, however in this case $G(p)$ is used only as an indicator function. An alternative is to apply Equation 29 for each link, e, i.e., and then use $G(e)$ as the link weight for e. Simulation results that are presented in the next section show that using $G(p)$ is advantageous over $G(e)$.

In the algorithm description shown in Figure 5, SRC and DEST are the source and destination nodes. In each node V of the network, $W_j[V]$ is the j-th weight of the selected path between SRC and V. Also, $\mu[V]$ is the average of $W_j[V]$ for all j's. The corresponding Δ function is $\Delta(V)$. We define PREVIOUS[V] to hold the previous node of V when traversing on the path between SRC and V. The parameter $L_j(V,B)$ gives the j-th weight of the link between the nodes V and B. Initially all nodes are tentative, and when it is discovered that a link represents the shortest possible path from the source to that node, it is made permanent. PERMANENT[V] indicates if V is a permanent node. This description is similar to that commonly used for Dijkstra's algorithm. The routing starts at the SRC node and all of the weights are initialized to infinity. Also we initially assign a zero weight and average to the source node. TAG is the node whose neighbors are currently being examined.

The main part of the algorithm is between lines 16 to 39. The FOR loop begins at line 22, and the statements in lines 24 and 26 have an $O(K)$ complexity. The complexity of the overall algorithm is $O(KN^2)$ where N is the number of network nodes. The complexity of the single mixed metric algorithms with $\lambda > 1$ is not better because their structure does not differ from the new one and the difference is in the mixing functions. Hence, our new algorithm is comparable with existing low complexity algorithms in terms of time complexity.

4 Simulation Results

A large variety of simulation experiments have been performed using a wide range of different parameter values. In this section some representative results are presented which illustrate the relative performance of the proposed algorithm. In the results to be presented, the performance measure used for comparison is the Success Ratio (SR), which is defined as the percentage of time that the algorithm finds a feasible path when at least one exists. Single mixed metric routing methods introduced in Section 2, for $1 \le \lambda \le 4$ are simulated as well as our proposed algorithm. In addition, the method based on Equation (10) is simulated. In all of the existing algorithms of Section 2, the single mixed metric is used only as a path indicator. In the followings, "$Wi(P)$" indicates the routing performance based on a mixed metric, determined by Equation (8) with $\lambda = i$. Also, "$Maximum$" indicates the routing performance based on Equation (10). Results related with the proposed strategy are indicated by "$G(P)$".

Random network topologies for the simulations were generated using Waxman's method [17]. After generating each topology, weights for each link were selected randomly. A wide range of different link weights have also been considered. In the results presented below, some representative examples are given. For a link e, $w_1(e)$ is a uniformly selected random number from $[0,5]$ or from $[0,50]$, and $w_2(e)$ is uniformly selected from $[0,10]$ or $[0,200]$. Source, destination and constraints are generated based on the method of [6]. The source and destination are randomly generated such that the minimum hop-count between them is at least three. If p and q are the two shortest paths between the source and destination, and using weights w_1 and w_2, then the constraint C_1 is uniformly selected from $[0.8w_1(q), \ 1.2w_1(q)]$ and C_2 is uniformly selected from $[0.8w_2(p), \ 1.2w_2(p)]$. Networks with 10, 15, 20, 25, 30, 35 and 40 nodes are used and simulation results are shown for the 2-constraint cases.

Figure 6 compares the four existing methods with $1 \le \lambda \le 4$. It is clear from this diagram that the routing method with $\lambda = 4$ has the largest SR. In most of the other diagrams our proposed method is compared with this existing algorithm. Figure 7 shows the importance of the parameter, ε. When ε is zero, the performance of our method is very poor compared with the linear combination of weights. Also, the effect of ε is shown in Figures 8 and 9 where the value of the single mixed metric function (G) is shown for a two-constraint case. It is clear from these figures that when $\varepsilon = 0.5$, the value of the single metric function inside the feasible solution area is always less than its value outside of this area. This is not true when $\varepsilon = 0$.

Figure 10 shows that there is a range for ε where better performance of the proposed algorithm is observed compared with existing algorithms. Figures 11, 12 and 13 indicate that the proposed algorithm achieves better performance than the other mixed metric methods with $\lambda < 4$ and is comparable with the algorithm that uses $\lambda = 4$. In Figure 13 the proposed algorithm is compared with the one that uses Equation (10) as a path indicator.

Figure 14 shows the case where the new mixed metric is not used as a path indicator but is used for computing a single weight for each link. This situation is compared with the one that deploys a mixed metric only as a path indicator. When the mixed metric function that is a non-linear function of the weights is used as a path indicator, better performance is obtained. Single mixed metric methods work better when they use their mixed metric function as a path indicator.

Coefficient ε can take any value between 0 and 1. It is shown in Figure 7 that choosing ε to be zero does not

produce good results. By scanning the whole range of ε values through simulations, we found out that in networks with certain number of nodes specific values of ε produce better results than others. These results are illustrated in Table (1) from which Figure 15 is produced. Even though the performance of the routing algorithm based on these values are satisfactory, but these results are not necessarily optimum. The simulation results show that ε may depend on the size of the network. However, in-depth studies are required to analytically determine the effect of different parameters on the optimum value of ε and the behavior of the routing algorithm.

Table 1 Value of ε used in the simulations of Figure 15.

Number of Nodes	10	15	20	25	30	35	40
ε Value	0.3	0.3	0.4	0.5	0.5	0.6	0.6

```
        ROUTING ALGORITHM (NETWORK, SRC, DEST, CONSTRAINTS)
        {SRC is the source node. DEST is the destination node}
        { Lj(U,V) is the j-th weight of the link between nodes U and V}
        {Number of Constraints is K}
        BEGIN
01          FOR (all nodes V)  DO
02          BEGIN
03                  FOR (j=1 to K) DO
04                          Wj[V] ← ∞;
05                  μ(V) ← ∞;
06                  Δ(V) ← ∞;
07                  PREVIOUS(V) ←NULL;
08                  PERMANENT(V) ←FALSE;
09          END;
10          FOR (j=1 to K) DO
11                  Wj[SRC] ← 0;
12          μ[SRC] ← 0;
13          PREVIOUS[SRC] ← NULL;
14          PERMANENT[SRC] ← TRUE;
15          TAG ← SRC;
16          WHILE (TAG ≠DEST) DO
17          BEGIN
18                  FOR (all nodes, V, neighbors of TAG) DO
19                  BEGIN
20                          IF Not (PERMANENT[V]) THEN
21                          BEGIN
22                                  FOR (j=1 to K) DO
23                                          Wj[TEMP] ← Wj[TAG] + Lj(TAG,V)
24                                  FIND μ[TEMP] and Δ[TEMP]
25                                  G(TEMP) ← μ[TEMP]*(Δ[TEMP]+ε);
26                                  FIND μ[V] and Δ[V]
27                                  G(V) ← μ[V]*(Δ[V]+ε);
28                                  IF (G(TEMP) < G(V)) THEN
29                                  BEGIN
30                                          FOR (j=1 to K) DO
31                                                  Wj[V] ← Wj[TEMP];
32                                          PREVIOUS[V] ← TAG;
33                                  END;
34                          END;
35                  END;
36                  FIND NON-PERMANENT NODE V WITH SMALLEST G(V);
37                  TAG ← V;
38                  PERMANENT[TAG] ← TRUE;
39          END;
        END.
```

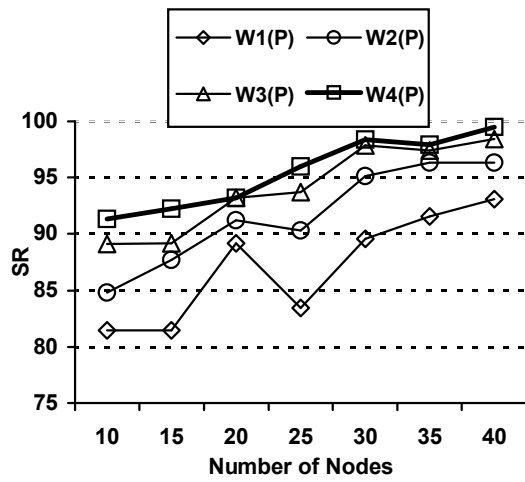**Fig. 5** Proposed Multi-Constraint routing algorithm
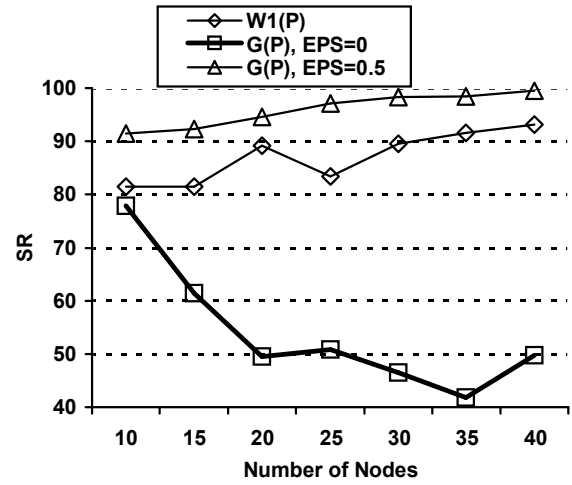
Fig. 6 The performance of existing methods
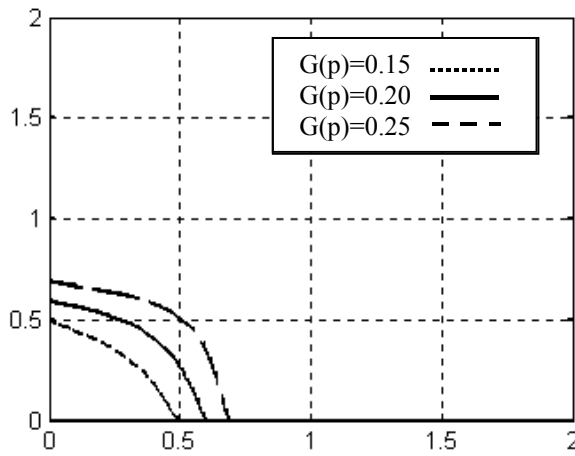

Fig. 7 The effect of ε

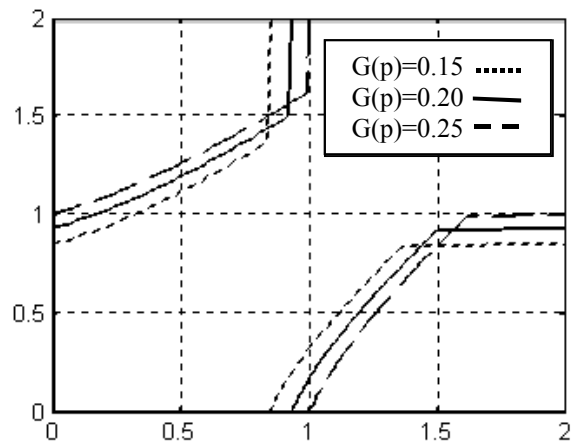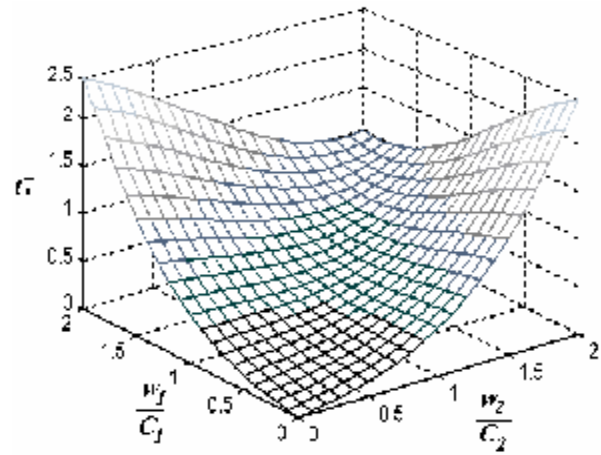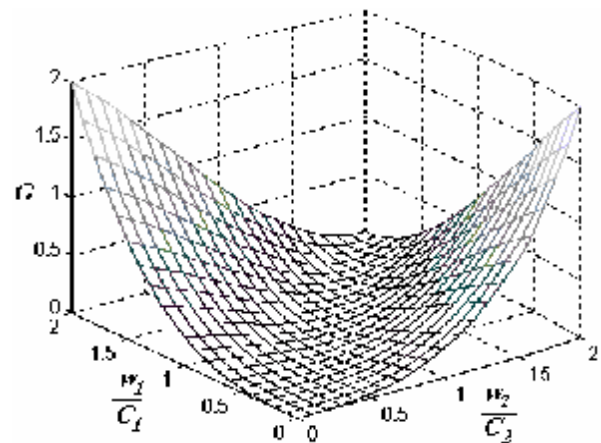**Fig. 8** Single mixed metric function in a two constraint case for ε = 0.5

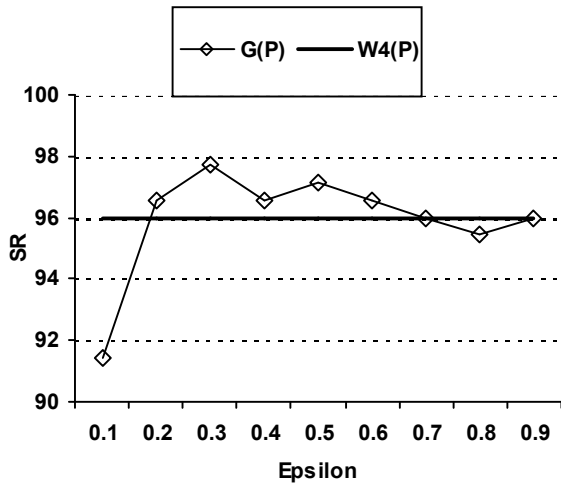**Fig. 9** Single mixed metric function in a two constraint case for ε = 0

Fig. 10 SR vs. change in ε in a 25-node network
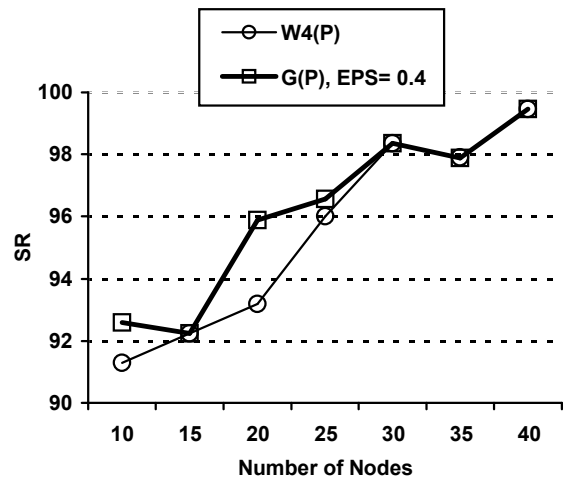

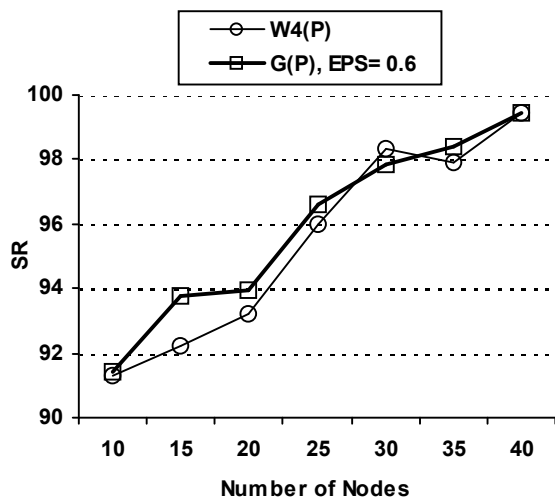**Fig. 11** SR with fixed ε

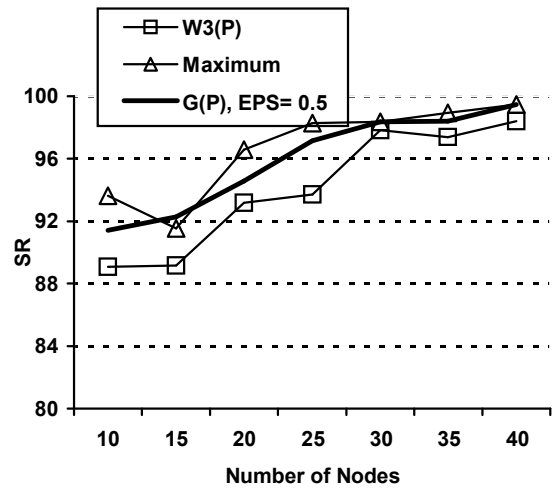
Fig. 12 SR with fixed ε

Fig. 13 SR of the new method compared with existing algorithms
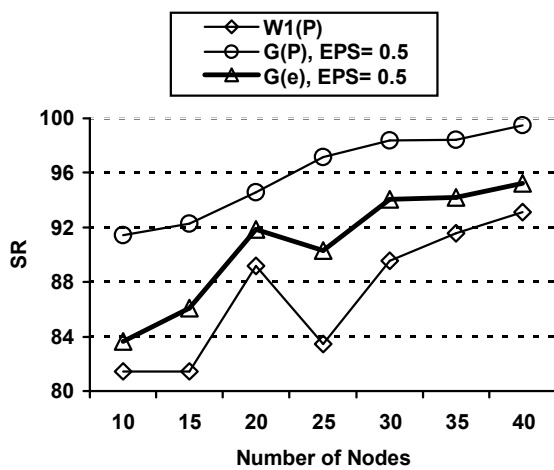

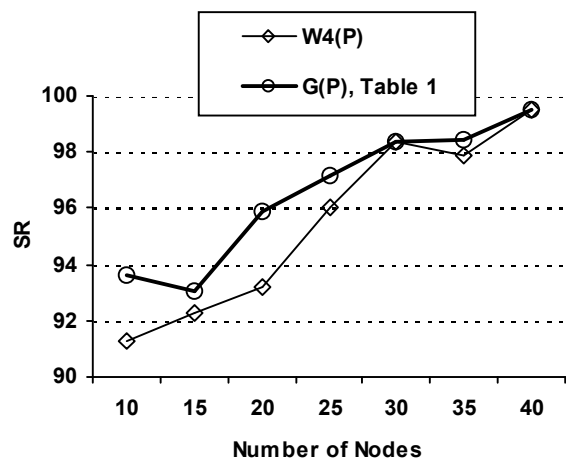**Fig. 14** Comparison between link-based and path-based routing, i.e. G(e) and G(P).

Fig. 15 SRs for different size networks.

5 Conclusions

In this paper a new single mixed metric routing algorithm was presented for solving the MCP problem. The proposed method takes into account variations of the link weights in performing path selection, and the results suggest that this is as important as the linear combination of these weights. A wide range of simulation results show that this new method has better performance in terms of success ratio than existing algorithms with comparable time complexity. The proposed algorithm uses a parameter, ε, that helps control the region over which searching is performed and simulation results were presented which indicate how this parameter should be selected in practice. It is possible to use the single mixed metric function as either a path indicator or for computing a single weight for each link. Simulation results show that if the single mixed function is used as a path indicator, better results may be obtained.

References

[1] Z. Wang, *Internet QoS: Architectures and Mechanisms for Quality of Service*, Morgan Kaufmann Publishers, 2001.

[2] Z. Wang, J. Crowcroft "Quality-of-service routing for supporting multimedia applications", *IEEE Journal on Selected Areas in Communications*, Vol. 14, Pp. 1228 –1234, Sept. 1996.

[3] D.W. Shin, E.K.P.Chong, H.J.Siegel, "A multiconstraint QoS routing scheme using the depth-first search method with limited crankbacks", *IEEE Workshop on High Performance Switching and Routing*, Pp. 385 –389, 2001.

[4] J. M. Jaffe, "Algorithms for Finding Paths with Multiple Constraints", *Networks*, Vol. 14, Pp. 95-116, 1984.

[5] H. Neve, P. Mieghem, "A multiple quality of service routing algorithm for PNNI", *1998 IEEE ATM Workshop Proceedings*, Pp. 324 –328, 1998.

[6] T. Korkmaz, M. Krunz "Multi-constrained optimal path selection", *Proceedings of INFOCOM 2001*, Vol.2, Pp. 834 –843, 2001.

[7] L.Costa, S.Fdida, O.Duarte, "A scalable algorithm for link-state QoS-based routing with three metrics", *IEEE International Conference on Communications ICC 2001*, Vol.8, Pp. 2603 – 2607, 2001.

[8] G. Feng, C.Douligeris, K. Makki, N. Pissinou, "Performance evaluation of delay-constrained least-cost QoS routing algorithms based on linear and nonlinear lagrange relaxation", *IEEE International Conference on Communications ICC 2002*, Vol.4, Pp. 2273 –2278, 2002.

[9] A. Juttner, B. Szviatovski, I. Mecs, Z. Rajko, "Lagrange relaxation based method for the QoS routing problem*", Proceedings INFOCOM 2001*, Vol. 2, Pp. 859 –868, 2001.

[10] G. Feng, K. Makki, N. Pissinou, C. Douligeris, "An efficient approximate algorithm for delay-cost-constrained QoS routing", *Proceedings of the Tenth International Conference on Computer Communications and Networks*, Pp. 395 –400, 2001.

[11] L. Guo, I. Matta, "Search space reduction in QoS routing", *Proceedings of the 19th IEEE International Conference on Distributed Computing Systems*, Pp. 142 –149, 1999.

[12] X. Yuan, X. Liu, "Heuristic algorithms for multi-constrained quality of service routing", *Proceedings of INFOCOM 2001*, Vol. 2, Pp. 844 – 853, 2001.

[13] S. Chen, K. Nahrstedt, "On finding multi-constrained paths", *IEEE International Conference on Communications ICC 98*, Vol. 2, Pp. 874 –879, 1998.

[14] J. Song, H. Pung, L. Jacob, "A multi-constrained distributed QoS routing algorithm", *IEEE International Conference on Networks ICON 2000*, Pp. 165 –171, 2000.

[15] A. Fei, M. Gerla, "Smart forwarding technique for routing with multiple QoS constraints", *IEEE Global Telecommunications Conference GLOBECOM '00*, Vol. 1, Pp. 599 –604, 2000.

[16] P.Khadivi, S.Samavi, T.D.Todd, H.Saidi, "Multi-Constraint QoS Routing Using a New Single Mixed Metric", *Proceedings of IEEE International Conference on Communications (ICC 2004)*, Vol. 4 , 20-24 June 2004, Paris, France, Pages:2042 – 2046.

[17] B.M. Waxman, "Routing of Multipoint Connections", *IEEE Journal on Selected Areas in Communications*, Vol. 6, No. 9, Pp. 1617-1622, 1988