Overview of Routing Algorithms for MCP with Single or Mixed Metric

Yavor Tomov
Technical University of Sofia, Sofia, Bulgaria
qvor_tomov@abv.bg

Abstract: Nowadays more and more telecommunication services appear with increasing number of users. A large number of these services require a guaranteed Quality of Service (QoS). The main task of QoS routing is to find a path from source to destination satisfying all constraints. Our goal in this article is to make an overview of the algorithms able to solve MCP known as algorithms with single or mixed metrics.

Keywords: QoS, routing algorithm, MCP

1.Introduction

As we all know the modern Internet from the very beginning was created on the principle of "The Best Effort". This principle provides a fair usage of the net's resources, but it is not able to cope with the new challenges, related to the provision of given Quality of Service (QoS).

According to the ITU definition the QoS is "The collective effect of service performance which determines the degree of satisfaction of a user of the service".

Nowadays the number of services (especially such as the real time services) is constantly increasing. They need a guaranteed quality, which may be achieved by supporting one or more given parameters from source to destination.

The main tasks of routing are to find this path and to update and store the available data for a network.

A certain number of algorithms that are able to find the shortest path from a source to a destination are suggested in the literature [5][2]. These algorithms however can be applied only in cases when the parameter examined is one and only. When a certain service requires two or more parameters guaranteed from a source to a destination, the problem is known as Multi Constraint Problem (MCP). MCP is NP hard [7].

Depending on these parameters (such as delay, bandwidth, packet loss etc.) they may be classified as additive, multiplicative or concave. Delay is exemplified by additive parameter, while the bandwidth is an example for concave. This paper is about the additive parameters, where the weight of this parameter (end to end) is equal to the sum of the weights of all links on the path. A path which satisfies all constraints is called a feasible path.

The algorithms, which are able to find this path (if it exists) are exact algorithms. The general drawback is their high complexity, and they are inapplicable in practice. They are mainly used for evaluating other types of algorithms known as heuristic. The heuristic algorithms refer to experience-based techniques for problem solving, they have lower complexity but they don't guarantee the finding of solution. The other class is ε–approximation algorithms. They are not necessarily exact but can provide a solution quantifiably close to the exact solution.

The article is aimed at reviewing a large number of algorithms, known as algorithms with single or mixed metrics. Most of them are heuristic but there are also exact ones. Their advantage is the low complexity and easy implementation in practice. Their drawback is the insufficient information they give concerning the fulfillment of QoS requirements [17].

2. Problem Formulation

Each network can be presented as a graph $G(V, E)$, where V is the set of nodes and E is the set of links. The nodes are routers or switches, while the links are physical or logical connections between them. Each link e is associated with n-dimensional link vector $\vec{w}(w_1, w_2, ..., w_n)$. The source node will be noted as s, while the destination node will be noted as t. We will note the path from the source to the destination with p and the given constraints as L_i $(1 \le i \le n)$.The problem is to find a path from source node to destination node such that

$$w_i(p) = \sum_{e \in p} w_i(e) \le L_i \qquad (1)$$

A path that satisfies all n constraints is referred to as a *feasible* path.

Each path that has two metrics may be represented on the $((w_1(p), w_2(p))$ plane.

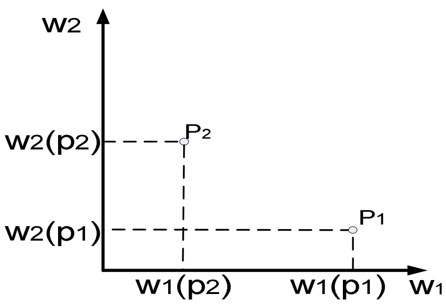

Figure 1: Non-dominant path

Fig. 1 shows that $w_1(p_2) < w_1(p_1)$ and $w_2(p_1) < w_2(p_2)$ and we cannot define whether $p_1 < p_2$ or not.

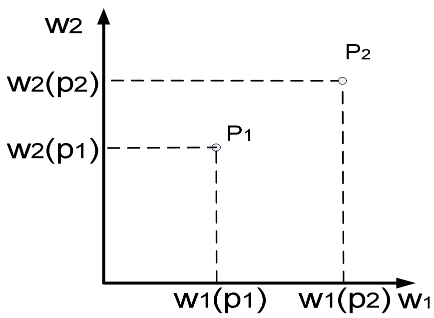

Figure 2: Dominant path

Fig. 2 shows that $w_1(p_1) < w_1(p_2)$ and $w_2(p_1) < w_2(p_2)$. In this case the path p_2 is dominated by p_1. If $w_i(p_1) < w_i(p_2)$ for all i, p_1 is *dominant* path [5]. If there is a dominant path, it is easy MCP to be solved since having found the shortest path with respect to one parameter, the same path is the shortest with respect to all other parameters.

The algorithms that will be considered in this article use two different path lengths – linear path length (ALPL) and non-linear path length (ANLPL).

3. Algorithms with Linear Path Length

3.1. Jaffe's Algorithm

In 1985 Jaffe presented his heuristic algorithm [10], based on Lagrange relaxation. He suggested that the weights of each link be presented as linear combination.

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

where d_1 and d_2 are positive multipliers.

The region between L_1 and L_2 (Fig. 3, Fig. 4) is feasible region of solutions. The black dots present the different paths. All paths that lie on the same parallel line have the same constant value *c.*

$$d_1 w_1(p) + d_2 w_2(p) = c \qquad (3)$$

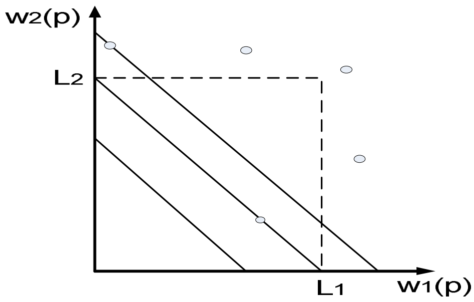

Figure 3: The algorithm works

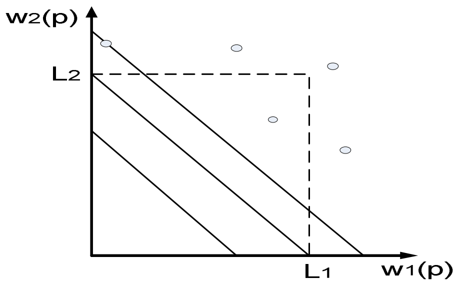

Figure 4: The algorithm fails

Each parallel line that lies above the other line has a higher constant value *c*. The main task in this method is defining the multipliers d_1 and d_2. Jaffe proposed these multipliers to be defined by the following equation:

$$\sqrt{\frac{L_1}{L_2}} = \frac{d_2}{d_1}. \qquad (4)$$

Fig. 3 indicates that Jaffe's algorithm works, while in Fig. 4 it fails.

3.2 Feng's Algorithm

Based on Jaffe's heuristic algorithm Feng proposed a new exact algorithm [6], applying in it two basic ideas:

- Reducing the search space;
- k-th shortest path.

This algorithm reduces the search space as follows: firstly it finds the shortest path with respect to each weight, and the feasible region is defined as $L_i - w_i(p)$, $L_i - w_i(p)$, Fig. 5.

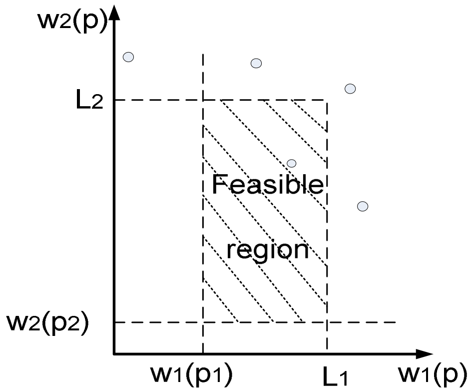

Figure 5: Reducing search space

Secondly the author suggests that the multipliers be defined as:

$$d_i = \begin{cases} \infty & : L_i = w_i(p_i) \\ \dfrac{L_0 - w_0(p_0)}{L_i - w_i(p_i)} & : \text{otherwise} \end{cases} \qquad (5)$$

The algorithm uses k-th shortest path, which means that if the first path returned by the algorithm is out of the feasible region it returns the second one and so on until the returned path by the algorithm is feasible or this path lies on a straight line with a higher constant value than the straight line with optimal constant value,

$$w_0(e)+\sum_{i=1}^{n-1} d_i w_i(e) > L_0 + \sum_{i=1}^{n-1} d_i L_i \qquad (6)$$

then the algorithm returns no solution.

3.3 Iwata's Algorithm

Iwata proposed a heuristic algorithm (8) using single metrics to solve MCP. This algorithm uses Dijkstra's algorithm. It first computes one shortest path, based on a QoS measure and checks whether the constraints are satisfied. If they are – the algorithm returns this path. If they are not – another measure is applied and this is repeated until a feasible path is found. Fig. 6 shows that algorithm fails.

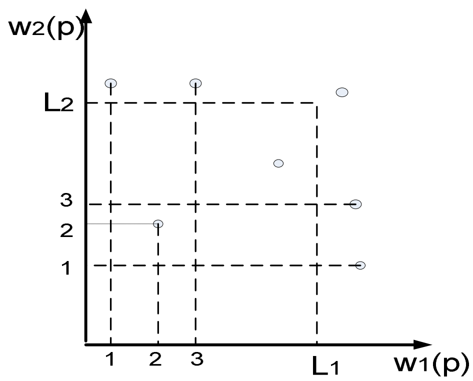

Figure 6: The algorithm fails

3.4 LARAC Algorithm

LARAC (Lagrange Relaxation based Aggregate cost) is an heuristic algorithm that is able to solve DCLC (Delay Constrained Least Cost) problem in polynomial time [11].

The first step in this method is to calculate the shortest path with respect to the cost-w_1 Fig. 7. In our case this path is p_2 and if this path meets the delay constraint $w_2(p_2) < L_d$, where L_d is constraint, this path is optimal. If this path doesn't satisfy this requirement, in the second step the algorithm finds the path with respect to delay constraint p_1 (if such a path exists). Thus the authors reduce the search space Fig. 7.

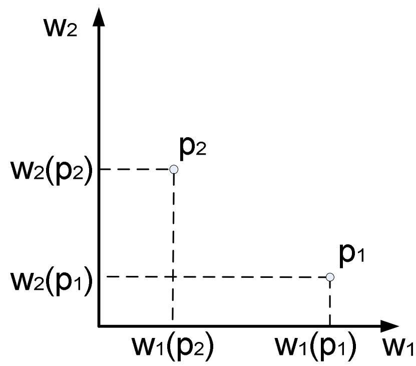

Figure 7: Two shortest paths with respect to cost p_2 and delay p_1

Based on the reduced search space the authors propose the multiplier to be defined in the following way:

$$d = \frac{w_1(p_2) - w_1(p_1)}{w_2(p_1) - w_2(p_2)}. \qquad (7)$$

where w_1, w_2 are the cost and delay respectively .

3.5 Khadivi's Algorithm

Khadivi proposed a heuristic algorithm [12], where he uses the single mixed metric as follow:

$$w(e) = \mu(e)[\Delta(e) + \varepsilon], \qquad (8)$$

where ε is constant $0 \le \varepsilon \le 1$,

$$\mu(e) = \frac{1}{n}\sum_{i=1}^{n} \frac{w_i(e)}{L_i}, \qquad (9)$$

and

$$\Delta(e)=\sum_{i=1}^{n}\left(\frac{w_i(e)}{L_i}-\mu(e)\right)^2, \qquad (10)$$

where $w(e)$ is a new mixed metric.

4. Algorithms with Non - Linear Path Length.

4.1 TAMCRA(Tunable Accuracy Multiple Constraints Routing Algorithm)

Hans de Neve and Piet van Mieghem created a heuristic algorithm [4], based on three concepts:

- non-linear path length [1];
- k-th shortest path algorithm[3];
- non-dominated paths.

The authors replace the straight scanning lines used in the linear algorithms by curved equivalents lines Fig. 8.

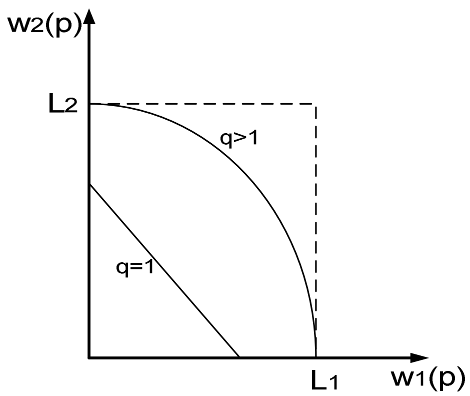

Figure 8: Linear and non-linear search space

In linear algorithms if the multipliers are calculated by the equation

$$\frac{L_1}{L_2}=\frac{d_2}{d_1} \qquad (11)$$

then the area that will be scanned before the algorithm is able to return any solution outside the feasible region, will be half of the whole area.

The path's length from source to destination is non-linear combination:

$$l(p)=\sum_{i=1\ldots n}\left(\frac{w_i(p)}{L_i}\right)^q \qquad (12).$$

The best match is obtained in the limit when $q\to\infty$.

If we use the non-linear definition of the path length, *the subsection of shortest paths are not necessarily the shortest paths.* In this reason if we use Dijkstra's algorithm, it can fail to find solution. TAMCRA avoids this drawback using the k-th shortest path. The third feature of TAMCRA is that it stores the paths only if they are not dominated by the others.

4.2. H_MCOP

Kormaz and Krunz proposed a heuristic algorithm based on three theorems that they proved [13]. The algorithm uses two modified versions of Dijkstra's algorithm – in the forward direction and in the backward direction. In backward direction the algorithm uses the linear path length (q = 1) with respect to

$$w(e)=\sum_{i=1\ldots m}\left(\frac{w_i(p)}{L_i}\right)^q \qquad (13)$$

from each intermediate node to destination node t. After that the algorithm evaluates the suitable paths. In the forward direction the algorithm discovers each intermediate node from the source node, using the non-linear function (q >

1).The path from s to t in the forward direction is heuristically found.

4.3. SAMCRA

SAMCRA (Self-Adaptive Multiple Constraints Routing Algorithm) is a further development of TAMCRA, it is improved and exact algorithm [16]. Two versions were created. According to the first version the basic difference with TAMCRA is the fact that TAMCRA stores an equal number of paths at each node that are predefined, while SAMCRA adaptively defines the number of the paths at each node. Predefining the number of paths is TAMCRA main drawback because if this number isn't sufficient then the algorithm fails to give solution.

In the second version the concept *look ahead* was added to reduce the search space. This concept was primarily proposed in [14][15].

5. Conclusions

The algorithms analyzed in this article are ALPL or ANLPL.

The major advantage of ANLPL is that when $q \rightarrow \infty$, they scan the feasible region precisely. Their drawback is that *subsections of shortest paths are not necessarily shortest paths*. Due to this, using Dijkstra's algorithm ANLPL are likely to fail. That is why algorithms using non-linear path length (TAMCRA, SAMCRA), need a modified Dijkstra's algorithm that stores k-number of paths. All this results in a higher algorithm complexity.

Korkmaz and Krunz [13] avoid the k-th shortest path concept. Instead they apply the "*look ahead*" concept to obtain paths which could be a possible solution. The usage of k-th shortest path guarantee that a shortest path will be found (if it exists) while "*look ahead*" does not.

When using ALPL only half of the feasible region is scanned before the algorithm is able to return the solution outside this region.

The main task is to choose the multipliers that define the ratio between the metrics, inside the mixed metric.

Iwata's algorithm [9] computes the shortest path with respect to each metric without mixing them. This algorithm has low complexity but the probability to find a solution is low too.

The authors of LARAC [11] use a similar method to find the shortest path with respect to cost and delay reducing the search space and based on this space to define a multiplier for a subsequent search with mixed metric.

Feng [6] applies the same approach, adding the concept of k-th shortest path to create an exact algorithm.

References

1. Ahuja R.K., Magnanti T.L., and. Orlin J.B. Network Flows: Theory, Algorithms, and Applications. Prentice Hall Inc., 1993
2. Bellman Richard. On a routing problem. Quarterly Applied Mathematics, XVI(1): 87- 90, 1958.
3. Chong E.I. , Maddila S., and Morley S.. On finding single-source single-destination k shortest paths. Journal of Computing and Information, special issue ICCI' 95, pages 40—47, July 1995.
4. De Neve H. and Van Mieghem P.. TAMCRA: a tunable accuracy multiple constraints routing algorithm. Computer Communications, 23: 66 - 679, 2000.
5. Dijkstra E.W. A note on two problems in connexion with graphs. Numerische Mathematik, (1):269—271, 1959.
6. Feng Gang, " Exact algorithms for multi-constrained QoS routing, "International Conference on Computer, Communication and Control Technologies (CCCT'03), Orlando, USA, Vol. 2, pp. 340 – 345, July 31 – Aug 2, 2003.
7. Garey M.R. and Johnson D.S.. Computers and Intractability: A guide to the Theory of NP-Completeness. ISBN 0 -

7167–1044-7. W. H. Freeman and Company, San Francisco, 1979.
8. Henig M.I. The shortest path problem with two objective functions. European Journal of Operational Research, 25:281-291, 1985.
9. Iwata A., Izmailov R., Lee D., Sengupta B., Ramamurthy G., and Suzuki H.. ATM routing algorithms with multiple QoS requirements for multimedia internetworking. IEICE Trans. Commun., E79 - B(8):999 - 1007, Aug. 1996.
10. J.M. Jaffe. Algorithms for finding paths with multiple constraints. Networks,14:95—116, 1984.
11. Juttner A., Szviatovszki B., Mecs I., Rajko Z., “Lagrange relaxation based method for the QoS routing problem”, Proceedings of the INFOCOM 2001 Conference, vol. 2, IEEE, 2001, pp. 859–868
12. Khadivi P., S.Samavi, Todd T.D., "Multi-constraint QoS routing using a new single mixed metrics", Journal of Network and Computer Applications, Vol. 31, Issue 4, pp 656-676, Nov. 2008.
13. Korkmaz T. , M. Krunz , “Multi-constrained optimal path selection”,Proceedings of the INFOCOM 2001 Conference, vol. 2, IEEE, Anchorage, Alaska, 2001, pp. 834–843.
14. Lin S., “Computer solutions of the traveling salesman problem,” Bell Syst. Tech. J., vol. 44, no. 10, pp. 2245–2269, 1965.
15. Newell A. and Ernst G., “The search for generality,” in Proc. IFIP Congr., vol. 1, 1965, pp. 17–24
16. Van Mieghem P. and F. A. Kuipers, “Concepts of Exact Quality of Service Algorithms,” IEEE/ACM Trans. Net Vol. 12, pp. 851 - 862, 2004.
17. Wang Z. and Crowcroft J.. Quality of service routing for supporting multimedia applications. IEEE Journal on Selected Areas in Communications, 14(7):1228—1234, September 1996.

.

.

.