PERIODICA POLYTECHNICA SER. EL. ENG. VOL. 49, NO. 3–4, PP. 209–221 (2005)

ROUTING PROTOCOLS WITH SMALLEST MAXIMAL LATENCY

Gábor KURUC* and Krisztina LÓJA**

*Vodafone Hungary Co.
e-mail: gabor.kuruc@vodafone.com
**Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
e-mail: loja@math.bme.hu

Received: May 26, 2005

Abstract

There are a lot of routing protocols that minimize the total latency of the network. These have the drawback of causing too high latency for some traffic. The goal of this paper is to suggest an approach that takes into account not only the total latency but also the greatest latency in the network. Providers of voice, video or other real-time services should use a routing method that guarantees a routing path with minimal SLA (service level agreement) via the entire network for each customer. The cost base routing selection method that this paper suggests, tries to increase the minimum SLA in the entire network. The goal is to create a cheaper transport network with higher SLA without more passive network capacity.

Keywords: selfish routing, latency, fairness, non-cooperative game theory, routing protocols, Nash flow.

1. Introduction

Service providers used to build and support separate networks to carry mission-critical and non-mission-critical voice and video traffic. There is a growing trend, however, toward the convergence of all these types of networks into a single, packet-based IP network.

Almost every routing protocol uses some kind of SPF (Shortest Path First) algorithm. SPF depends on the philosophy of the routing protocol. The metrics of difference between the paths is their cost. It can be hop based, bandwidth based, administrative cost based or other kind. This metrics of links can be additive, multiplicative or can use the minimum or maximum on a path.

Routing is a major problem since the new, intelligent systems make it possible to use multiple alternative paths simultaneously. Our aim is to minimize the latency-based characteristics of these networks, that is, minimizing the cost. We have to face the same problems in case of fixed and mobile networks.

The questions are: Is there an ideal solution? And if there is one, is there only one good solution? [1, 2]

The problems with the routing that minimizes total latency and the routing with the traffic at Nash equilibrium is already known. We suggest a solution which result a routing between these.

In section 2 we describe our abstraction, the subject of section 3 is game theory in routing problems. In section 4 we review the used routing protocols, the cost of the routing and the probability variables belonging to the paths, finally in the last section we draw the conclusions.

2. Assumptions and Model Abstraction

Our aim is to construct an abstract network, which allows us to test and compare the routing algorithms. In this case we may neglect the inter-router administrative communication and performance problem of routers' CPU. We analyze QoS, we are interested in traffic that we can characterize as traffic flows with fixed bandwidth requirements.

The QoS requirements can be described by maximum latency (d_p) and bandwidth claim (b_p) on a given path p.

Thus, in a closed network, where there are R routers that route all the traffic in the network, each amount of traffic that runs on $p_{s,t}$ originated from R_s and goes to R_t. The network topology can be described by graph $G(V, E)$. $\|V\| = R$, points in the graph represent routers in the network, E denotes the edge set in the network [2, 3].

2.1. Simplification

In this scenario the memory and processor resources are regarded as infinite. The inter-router communication does not result traffic overhead in the network.

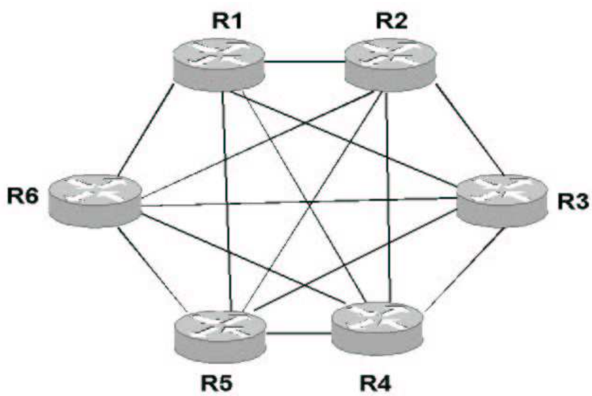

Fig. 1. Network with 6 routers.

Number of edges in graph:

$$\|E\| = \frac{\|V\|^2 - \|V\|}{2} \tag{1}$$

Cases where traffic flows travel through the same router multiple times are not accepted in this scenario. In this case, the number of links between two routers is maximum $R - 1$. Maximum k paths can exist between two routers:

$$k = \sum_{n=0}^{R-2} \frac{(R-2)}{(R-2-n)}. \tag{2}$$

The quality of link l ($l \in E$) depends on the maximum bandwidth B^l and the latency d^l. The quality of each path ($p \in P$) depends on the maximum bandwidth B_p and latency d_p. The parameters of path p depend on the parameters of those links (l) that are part of path p [4, 5].

$$B_p = \min_{l \in p} B^l, \tag{3}$$

$$d_p = \sum_{l \in p} d^l. \tag{4}$$

2.2. *Latency and Bandwidth*

The latency of a link depends on several different technical parameters as the finite speed of signal depending on the medium and the distance. In this case this latency is irrelevant.

The speed and the load indicate the relevant latency. In practice this means, that each output interface has a waiting queue. The length of this waiting queue depends on data speed that arrived in the queue and line speed. The latency depends on the length of this queue (see *Fig. 2*).

Taking the reciprocal of the bit speed gives the amount of time needed to transport one single bit. Each bit must be transported. Thus, the whole time is proportional to the summarized amount of each bit from each source:

$$d^l = \frac{\sum_i b_i^l}{B^l}, \tag{5}$$

The latency of link l is d^l, supposing $1/B^l$ is bit-time. We summarize bandwidth claims over all traffic on the link l, b_i^l is the the needed bandwidth of the ith traffic on the link l. Since packets arrive onto output interface at random as Poisson-distribution, the time of transmission depends on the product of the amount of arrived data and bit speed.

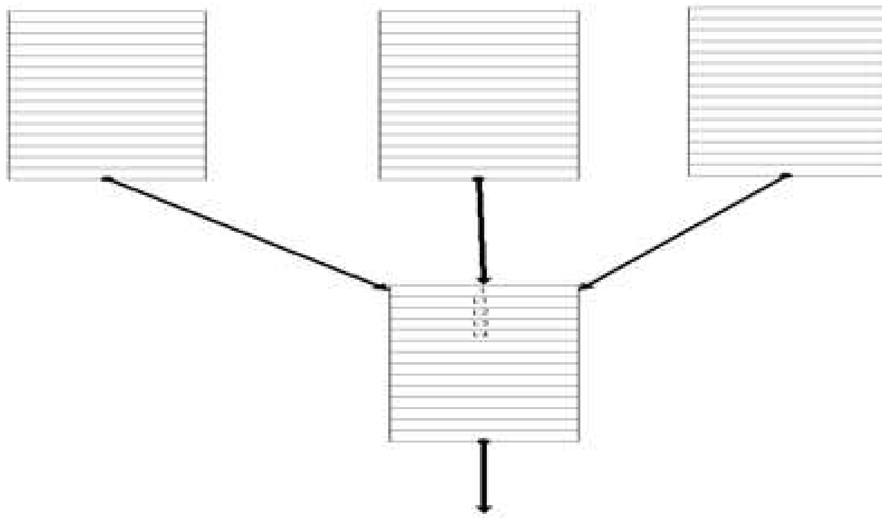

Fig. 2. Waiting queue on output interface.

The d_p is the latency of the whole path. Calculation of the whole path latency does not depend on the constant latency of non-traffic load. Thus, the path latency is the sum of the whole links latency along the path:

$$d_p = \sum_{l \in p} \frac{\sum_i b_i^l}{B^l}. \tag{6}$$

In this case we summarize the load of traffic that travels from different sources to different destinations, because one link can be a part of different paths. The latency of the link depends on the traffic that travels through the link. Where there is no connection between two routers, $B_{(s,t)} = 0$. We use two parameters of traffic flow, max d_i and min b_i.

All the possible paths between two routers are part of a set P. Each p path is a set of links.

$$P = \langle p^1, p^2, \ldots, p^{n_p} \rangle, \tag{7}$$

$$p^i = \langle l_1^i, l_2^i, \ldots, l_{n_i}^i \rangle. \tag{8}$$

By that, $p^k \subset E$, where $d_{p^k} = \sum_{l \in p^k} d^l \le d_i$ and $B_{p^k} = \min_{l \in p^k} B^l > b_i$.

The number of possible paths between two routers is:

$$\|P\| = \sum_{n=0}^{R-2} \frac{(R-2)}{(R-2-n)}. \tag{9}$$

Thus, from the $\|P\|$ pieces of paths those are suitable which connect routers r and s and satisfy criterion of maximum latency d_i and minimum bandwidth b_i.

The Customer's goal is to minimize traffic latency. The path latency in the traffic originator source router, thus the latency of traffic transport depends on bandwidth of the path and the bit-rate of traffic; we can easily get the optimal solution for local cost. But possibly it is not the optimal path for the whole network.

The question is: If we have n traffic flows in the network, how can we arrange traffic so that produces lowest SC (Social Cost). This value means the state that realizes the minimal social cost that is the best possible cost in multi-path network.

$$SC(W, F) = \sum_{\langle p_1,\ldots,p_n\rangle = H^n} \left(\prod_{k=1}^{n} f_k^{p_k} \max_{p\in P} \sum_{l\in p} \frac{\sum_i b_{w_i}^l}{B^l} \right) \tag{10}$$

The H^n is a set of whole combination of path set's elements of n traffic (from P^1 to P^n and each $P^i = \langle p_1^i, p_2^i, \ldots, p_k^i\rangle$). Thus, we analyze each combination of possible paths of each traffic. (If we use the previously defined abstract network then each traffic flow has k possible paths. Thus, k^n combination exist.) The number of traffic flows is n, W is the set of traffic flows and F is the set of traffic share rate. In W there are n pieces of traffic flows $(w_1, w_2, \ldots, w_n)$. Each traffic flow has its own source router, destination router, bandwidth, and maximum latency parameters. The bandwidth claim of the traffic w_i is denoted by b_{w_i} [6, 7].

The F set determines the selection of probability of paths between source and destination routers $(f_1, f_2, \ldots, f_n)$. Thus, each path can transport the traffic flow with different probabilities. In the case if we have several traffic flows between two routers, the F set determines the rate of load sharing on the paths between these routers. Using a 'simple' routing protocol that cannot support load sharing, we can use only one independent path on traffic flow quantity. Thus, one f_k^p value equals to 1, each other selection probability is 0.

In each relation all the possible path combinations are analyzed, all combination probabilities are multiplied with the maximum latency of traffic flows in this combination, we get the social cost of this traffic set and routing method.

Of course, one f_k^p value only differs from 0 when the path suits the requirements of the traffic, i.e., there is enough bandwidth.

If we have R routers in a full-meshed network, and we distinguish directions then $p_{(a,b)} \neq p_{(b,a)}$, $w_{(a,b)} \neq w_{(b,a)}$. In this case, the number of paths among routers is:

$$\|P\| = (R-1)R \sum_{n=0}^{R-2} \frac{(R-2)}{(R-2-n)}. \tag{11}$$

$\|P\|$ is the possible number of paths among R routers including 0 capacity paths.

There is an important difference between the standard social cost calculation method and our solution. Usually they try to minimize the sum of latency. This is the optimal routing. We try to minimize latency of the slowest traffic flow. We think that it is a very important parameter in a service provider network, because they have uniform traffic flow that use paths in same condition, for example voice, video and other real-time traffic flow. In this approach, the service quality is the quality of the worst traffic flow.

3. Game Theory in Routing Problems

How can we apply game theory for this problem? The players are the units of traffic or equivalently the users belonging to them. They can decide which path they choose, so their strategy set is the set of the paths. The payoffs are inversely proportional to the latency. This is a non-cooperative game.

The most important equilibrium conception of game theory is the Nash equilibrium. A Nash equilibrium is an array of strategies, one for each player, so that no player has an incentive (in terms of improving his own payoffs) to deviate from his part of the strategy array.

3.1. Flows at Nash Equilibrium

The traffic on the different routes form a flow on the network. A flow is said to be in Nash equilibrium (or is a Nash flow) if no user can change its route in order to improve its latency. Nash flows always exist and are essentially unique, i.e., every Nash flow has the same total latency (we get it by summarizing the product of the latency and the amount of traffic on every link).

If one of the paths of the Nash flow has a positive amount of traffic then its latency can not be greater than the latency of any other path in the Nash flow. In a Nash flow the product of the latency and the amount of traffic is equal on every path, provided that the traffic has arbitrary small units (we assume that there are lot of users and each of them controls a negligible fraction of the overall traffic).

Since there can be such a flow in which no user can improve its latency by changing its route but it can improve the latency of others, the Nash flows are not always optimal considering the total latency. For this there is a very simple example with one source and one destination node and two links between them. Let the latency of the first link be 1 independently from the traffic, and let the latency on the second link be the same as the amount of traffic on the edge. We have to send one unit of traffic from the source to the destination node. The optimal flow would be that in which half unit of traffic is on both edge (so the total latency is $0.5 \cdot 0.5 + 0.5 \cdot 1 = 0.75$), but the whole traffic is on the first link in the Nash flow, so the total latency is 1.

Braess gives a simple example on a network, in which giving a new path to the network increases the cost (the sum of latencies) in the Nash flow. This phenomenon is called Braess paradox [4, 8, 9].

3.2. Nash Flows and Optimal Flows

The following statements are known about the relationship between the Nash flow and the optimal flow. The total latency of a Nash flow is at most 4/3 times as much as that of the optimal flow, if the latency is a linear function of the amount of traffic on every link. This constraint is tight as proven by the example is section 3.1. As can be seen, network topology does not play an important role as the 4/3 ratio can appear in a network containing two links. If we assume that latency is only a non-negative, continuous and nondecreasing function of the traffic, then this ratio can be arbitrary large. For example if there are two links between the source and the destination, and the latency on the first link is the kth power of the traffic, the latency of the second link is 1 independently from the amount of traffic, then the cost of the Nash flow is 1 as the whole traffic is on the first link. Though, the flow could have arbitrary small cost if $1-\varepsilon$ unit of traffic is on the first link and ε on the other, where ε is a small positive number. A network with two links shows again that the complexity of the network plays no role. For almost any class of allowable latency functions, worst-case Nash/optimal ratio can be realized on a two-link network. The total latency incurred by a flow at Nash equilibrium is at most that of optimal flow forced to route twice as much traffic [4].

Optimal flow minimizes the total latency but it can be unfair to some traffic, i.e., some traffic might suffer a greater latency in the optimal flow than the Nash flow [8]. Let us suppose, that there are two links between the source and destination, the latency of the first is $2(1-\varepsilon)$ and the latency of the other is the same as the amount of traffic on the edge. At Nash equilibrium all traffic is on the second link, so total latency is 1. In the optimal flow ε units of traffic use the first link, $1-\varepsilon$ units of traffic use the second, the total latency is $1-\varepsilon^2$. As can be seen, the total latency of the Nash flow is greater but the latency of all traffic is 1 while in the optimal flow the latency of the traffic sent for the sake of optimality to the first link is $2-2\varepsilon$.

Let us assume that the amount of traffic (x) and the latency depending on the amount of traffic ($d_l(x)$) is convex on every link. The partial derivative of this with respect to the amount of traffic is $\partial/\partial x(x\, d_l(x))$ which we call marginal cost function. A flow is optimal if it is Nash equilibrium in the same network with the same amount of total traffic if the latency is the marginal cost function [2, 8, 10, 11].

4. The Decision Mechanism of the Routing Protocols Used Today (from QoS Point of View)

Nowadays, dynamic routing protocols try to find the shortest path, providing the widest bandwidth or some others are based on a subjective cost. These mechanisms tend to overrate a route or section of routes and make its characteristics worse with the exaggerated traffic directed to it. There are QoS-based routing protocols that select more paths in an ad-hoc manner and do not take the interdependence of the traffic into account.

4.1. Cost of Routing

Let be n data flows in the network. These data flows are represented by the $\langle w_1, w_2, \ldots, w_n \rangle$ elements of the set W. We assume hereafter that for every w_i and l

$$\max_i b_{w_i} < \min_l B^l, \tag{12}$$

where $w_i \in W$ and $l \in E$ and b_{w_i} is the bandwidth claim of w_i. This means that the needed bandwidth of every traffic is smaller than the bandwidth of any link. More different type of traffic can only cause problematic situation. A data flow can only be constrained to use another path by the bandwidth of another data flow.

If we have a $w_{(s,t)}$ that is a traffic from s to t then it determines a $P^{s,t}$ subset of P which have $\langle p_1^{(s,t)}, p_2^{(s,t)}, \ldots, p_x^{(s,t)} \rangle$ elements. These are paths suitable to convey the traffic. So this set contains all the paths from s to t which have the demanded bandwidth. Every $p_x^{(s,t)}$ determines an $L^{p_x(s,t)}$ set, those elements $\langle l_1, l_2, \ldots, l_{i(s,t)} \rangle$ are the links between the routers. $L^{p_x(s,t)}$ is a subset of the set of the edges, E.

$$L^{p_x(s,t)} \subset E. \tag{13}$$

If two w traffic flows have different destination or source node then they have different P set and cannot have the same path in their sets. Two different paths however, may have common links (l). The traffic w_1 and w_2 cause a traffic congestion if $b_{w_1} + b_{w_2} > B^l$ on the link l. The aim of the routing protocols is to bypass this problem.

Let us accept that a routing protocol can be determined by a function that yields the F decision set (f^p is usually 1 or 0 in the simple routing protocols) from the information known about the network (set E and $\{B^l\}$). More sophisticated routing protocols can take into consideration the congestion of the paths, so indirectly also the congestion represented by the set W. So the W traffic set determines a traffic set containing w-s. A set P^{w_i} corresponding to the traffic control is determined by w_i. The elements of the set P^{w_i} are $\langle p_1^{w_i}, p_2^{w_i}, \ldots, p_{x_i}^{w_i} \rangle$. So the elements of the set $L^{p_{w_i}}$ have w_i congestion. This has an effect on the sets $P^{s,t}$ belonging to w traffic and on the F decision set. That routing protocol works most efficiently that can assign the $p^{s,t}$-s to the w-s so that the SC value will be as small as possible (see Eq.(10)).

4.2. Selecting Paths with Probability Variables

As we explained in the previous sections, the SC value represent the cost belonging to the traffic we want to convey, and the routing protocol depends on two factors. First is the maximal latency belonging to the paths combinations, the second is the probability of selecting the combination. The sum of the factors of these determines the collective cost. The maximal latency of the paths depends on how the selected combination of paths divides the traffic on the links. If it offers the shortest path to every traffic (SPF) then the latency of the sections will be high because of the congestion. If it adds too many links to the paths then the summarized latency of the links will be too large [6].

4.2.1. Best Variation

The question is, in which case one and only one good solution exists. We get variations by selecting one path from every set of paths belonging to w_i traffic. A simple protocol (SPF) finds only one path suitable, and it refuses all the others. The values of f will be 0 or 1. The factor of the f values will be 1, so different from 0 only when it is the combination that contains the paths, which were considered to be the best by the protocol.

In this case the decision is influenced by such events that are determined by the structure of the paths and the position of the sources and destinations, so the structure of the network. There is no probability event that has an influence on it, not even the timing of the traffic.

4.2.2. More Candidate Paths

We can talk about more paths in the case when the dynamic routing protocol gives a vote of confidence for not only a single path for some reason. For example when the selection of paths can be changed with the link-congestion (QoS-based routing). So the timing of the traffic from the set W can result different path-selections. There is a possibility to divide the traffic into more paths. In this case the value of f-s can be between 0 and 1 depending on with what probability it will assign the paths to the w_i traffic belonging to the set of paths (in which ratio it divides the traffic).

4.3. The Value of the Selections

4.3.1. Direct Selection

The selection is simple as we described in section 4.2.1. The routing protocol forwards the information in the decided direction according to the collected information from the network. The common cost is influenced here only by the efficiency of finding the suitable paths by the protocol.

These solutions can be important in such networks, where simple routing is needed. It is worths using in this case also the routing protocol that yields the smallest SC value with the given traffic conditions. Those protocols are able to do this, which have a thorough survey of the network structure, and take the deciding mechanism of the other devices into account.

4.3.2. Selection among More Paths

The situation is very different in that case. The probability variables of the different paths belonging to the w_i traffic are quite interesting. Their sum is 1 as the traffic has to select a path in any case. The router prefers the paths with smallest cost managing the traffic. The only elements that have influence on the common cost from the combinations of the traffic set of the f values are that ones which give some part of the path combination (H) to the traffic.

$$\prod_{k=1}^{n} f_k^p > 0 \tag{14}$$

Where this value is 0, the combination contains a path that is not suitable for the traffic. In this case we have to consider:

- the length of the paths of the selected combination
- the links used together by the traffic
- the bandwidth of the used links compared to the traffic directed to it

We can decrease the SC value with the consideration of these conditions.

If we conjugate the different traffic and examine the SC value, we can see the effects of the traffic on each other. If we scrutinize the change of the SC value with the change of the combinations we can determine the paths having common links. We have to optimize these. If we know the effects of the pairs of traffic to each other, we can calculate the rules used by the protocol to assign paths to the traffic, and determine the set F belonging to the demanded minimal SC value.

We can determine the effects of the elements of the W set with test bandwidth but there exist such procedures that can determine the topology. Using these we can calculate the rules of traffic control for the W set divided to traffic classes for the demanded SC values.

4.4. Examples on Routing Protocols

4.4.1. Common SPF (Shortest Path First) Routing

Let us consider a simple routing example (see *Fig. 3*). We assume, that all the links have the bandwidth of 1.5 Mb/s. Since the router considers only the costs of the paths, it sends all the traffic to one path, it uses only the best path, it handles all the traffic together.

The problem is, that the needed bandwidth is not available. Let us examine the SC value. $SC = 1.6/1.5 + 2.1/1.5 = 2.466$.

This solution is neither optimal nor a Nash flow since if one traffic went to the destination node via the other router, the latency of all traffic would decrease.

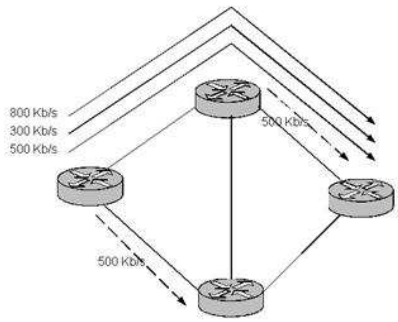

Fig. 3. Example on SPF routing.

4.4.2. QoS Routing

Let us consider the same example with QoS-based routing that examines the cost only locally (see *Fig. 4*). This takes into account the required bandwidth and the corresponding congestions. $SC = \max(1.3/1.5 + 1.3/1.5; 1.3/1.5 + 0.5/1.5 + 0.8/1.5; 0.8/1.5 + 0.8/1.5) = 1.73$.

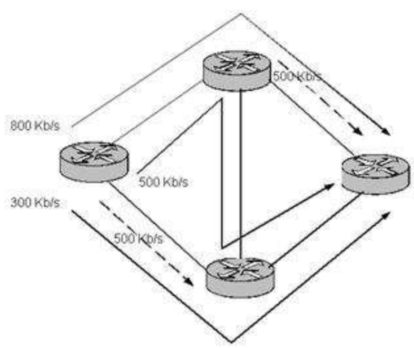

Fig. 4. Example on QoS routing.

This way the 1.5 Mb/s lines can be sufficient. QoS routing works but the total cost is too high, since one of the traffic flows has one more hop. This is a Nash flow, in this case no traffic could change its path, so that, its latency would decrease. We will see that this is still not the optimal solution.

4.4.3. An Optimal Routing

For the ideal solution the router should know the entire environment as the following (see *Fig. 5*):

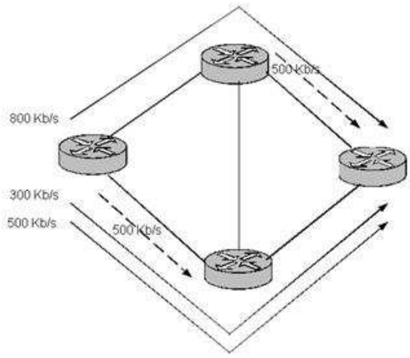

Fig. 5. Optimal routing.

$SC = \max(1.3/1.5 + 0.8/1.5; 1.3/1.5 + 0.8/1.5) = 2.1/1.5 = 1.4$. As *Fig. 5* depicts, the link between the two intermediate routers remains unused. If we delete the link from the network seen in the previous case, we get a better solution. Braess gives a simple example like this on a network, in which giving a new path to the network increases the cost (the sum of latencies) in the Nash flow. This phenomenon is called Braess paradox [4, 8, 9].

In this example, the allocations for every traffic are determined. In the case of real flows however, the goal of deciding mechanism is minimizing the costs. It may send a traffic with small probability in a roundabout way to minimizing the SC value, but causing a much too high latency for the selected traffic.

5. Conclusion

According to the above standings it is possible to construct a protocol that finds the traffic assignment belonging to the minimal social cost using local statistics and information it gets from the routers. Knowing the network topology and the traffic classes created by other participants with statistics or with other intelligence are required for calculating the paths.

As we have seen, the Nash flow is not always optimal. Though, the optimal flow minimizes the summarized latency only, the latency of an individual packet can be much higher as in the Nash flow. This shows that however appealing is the optimal routing it is not applicable in practice if we want to provide the supply for everyone. On the other hand the Nash flows may induce with 30% greater cost on the network what is inadmissible in a cost-sensitive situation. The best solution is somewhere between these two. So our SC value takes the greatest latency into account because all traffic of a telecommunication network have to reach the destination in time. In the cost calculation of the optimal routing the common cost is interpreted as the sum of the cost of the individuals. Optimizing this can result high latencies at some traffic if it has greater advantage at some other. This cannot be used at real-time traffic. There are such elements however, as the resource requirements of routing protocols where such an approach can be helpful.

References

[1] CSOPAKI, GY. – KURUC, G., Útvonalválasztás, kapcsolástechnika: merre haladunk? *Híradástechnika*, **10** (2003), pp. 16–19.

[2] BOYCE, D. E. – FARHI, A. – WEISCHEDEL, R., Optimal Network Problem: a Branch and Bound Algorithm, *Environment Planning*, **5** (1973), pp. 519–533.

[3] MA, Q. – STEENKISTE, P. – ZHANG, H., Routing High-Bandwidth Traffic in Max-Min Fair Share Networks, *Proc. of ACM SIGCOMM*, August 1996, pp. 206–217.

[4] ROUGHGARDEN, T. – TARDOS, É., How Bad is Selfish Routing? *Journal of the ACM*, **49** (2) (2002), 236–259.
http://www.cs.cornell.edu/timr/papers/routing.ps

[5] AHUJA, R. K. – MAGNANTI, T. L. – ORLIN, J. B., *Network Flows - Theory, Algorithms, and Applications*, Prentice - Hall, New Jersey, 1993.

[6] MAVRONICOLAS, M., *Game-Theoretic Approaches to Network Routing: A Primer Tutorial for Euro-Par*, Paderborn, Germany, 2002.
http://europar.upb.de/tutorials/tutorial01.html

[7] JÓZSA, B. G. – KIRÁLY, Z. – MAGYAR, G. – SZENTESI, Á., An Effcient Algorithm for Global Path Optimization in MPLS Networks, *Optimization and Engineering*, **2** (3) (2001), pp. 321–347.

[8] ROUGHGARDEN, T., How Unfair is Optimal Routing? *Proc.of 13th Annual Symposium on Discrete Algorithms*, 2002, pp. 203–204.
http://www.cs.cornell.edu/timr/papers/unfair.pdf

[9] TUMER, G. – WOLPERT, D., *Collective Intelligence and Braess' Paradox*, NASA Ames Research Center Moffett Field, CA 94035.
http://ic.arc.nasa.gov/people/kagan/pubs/aaai00_braess.ps

[10] GUERIN, R. – ORDA, A. – WILLIAMS, D., Williams, QoS Routing Mechanisms and OSPF Extensions, *Proc. of Global Internet (Globecom)*, Phoenix, Arizona, November 1997.

[11] LORENZ, D. H. – ORDA, A., QoS Routing in Networks with Uncertain Parameters, *IEEE/ACM Transactions on Networking*, **6** (6) (1998), pp. 768–778.
http://www-ee.technion.ac.il/ deanh/infocom98.ps.gz