Optimized Routing Adaptation in IP Networks Utilizing OSPF and MPLS

Anton Riedl
Munich University of Technology, Institute of Communication Networks
Munich, Germany
Anton.Riedl@ieee.org

Abstract— **In this paper routing adaptation methodologies are investigated, which utilize conventional routing protocols such as OSPF in combination with MPLS. While having the majority of traffic routed along optimized shortest paths, MPSL is only partly introduced to complement the adaptation process.**

We present a novel algorithm based on simulated annealing to optimize link metrics in OSPF networks. The algorithm takes into account the original routing configuration and allows tradeoff considerations between routing optimality and adaptation impact. For the setup of complementary MPLS paths, two mixed-integer programming models are proposed. It can be shown that already a relatively small number of MPLS paths is sufficient to greatly improve a network's quality of service.

Keywords -- Traffic engineering, routing optimization, mixed-integer programming, simulated annealing.

I. INTRODUCTION

Routing adaptation provides a means to keep up a network's quality of service (QoS) in times of changing load conditions. As long as there are still enough unutilized resources in the network, adjusting the routes of individual traffic flows can help to avoid regionally limited congestion spots. Rerouting for traffic engineering purposes is usually performed on a medium to long-term basis, i.e., the time between rerouting processes is at least a few minutes or hours.

In this paper we investigate procedures for optimized routing adaptation, which are based on OSPF and are only complemented by MPLS. This way the administrational effort can be kept low while the flexibility of MPLS is still utilized. In order to carry out routing adaptation in OSPF networks, we present an algorithm, which not only tries to optimize the overall QoS but also lets us consider the impact of the adaptation process. Thus, it is possible to trade in some of the QoS gain for a lower amount of rerouted traffic. If routing optimization based on OSPF is not sufficient, a number of MPLS paths can be set up to further improve QoS. We propose two mixed-integer programming models for the complementary MPLS routing problem and show their applicability in practical scenarios. We consider the maximum link utilization within the network as the relevant network QoS measure.

The paper is organized as follows: In Section II we introduce the routing protocols OSPF and MPLS and discuss relevant optimization issues. In Section III the optimization procedures for OSPF as well as for complementary MPLS are presented. Their applicability and results are discussed in Section IV. Section V concludes the paper.

II. ROUTING PROTOCOLS AND ROUTING ADAPTATION ISSUES

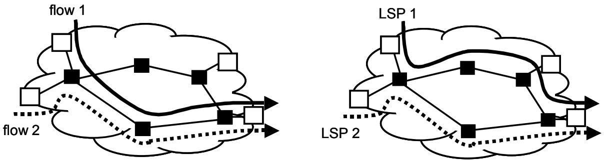
When dealing with routing optimization in the context of traffic engineering, we restrict our view to autonomous systems or even smaller network domains. Since the optimization process requires coordinated modification of network configuration settings, a homogeneous environment under the control of a single operator is necessary. Currently only little information about network internal parameters is exchanged between peers, making it impossible to optimize routing across network borders. In the following, we will refer to homogeneous network regions as "network domains" or just "networks".

A. Next-Hop Destination Based Routing: OSPF

With OSPF [1], a router's packet-forwarding decision is based solely on the destination address specified in the packet header. No further information about the packet such as its origin or any other context is taken into account. In order to determine the routes to all reachable destinations, OSPF routers exchange link state information and, thus, learn about the network topology. Based on link metric values, which are contained in the link state information messages, the shortest path is computed to every node in the network. For destinations outside the network the path to the respective border router is determined. An optional feature of OSPF routing is load sharing across equal-cost multi-paths (ECMP). Whenever a router can reach a destination node via several paths with equal metric sums, it splits the traffic evenly across the corresponding outgoing interfaces (in case ECMP is activated).

As a typical interior gateway protocol OSPF was mainly developed to provide connectivity among network nodes and to reroute traffic in case of sporadic node or link failures. Since frequent link metric changes lead to unstable routing behavior, OSPF by itself is not capable of quickly adapting to short-term load variations. Nevertheless, it can be used to alleviate medium to long-term traffic congestion problems. By setting the link metric values appropriately, it is possible to adjust the routes and react to the current load situation, thus, improving network QoS. However, the destination-based forwarding property of OSPF induces certain path characteristics, which impose limitations on routing optimization. Figure 1a) demonstrates this for two traffic flows (with a flow we denote the aggregation of all IP packets with identical ingress/egress nodes). Whenever two traffic flows, which are destined to the same egress node, cross each other's way, they are merged. From thereon they traverse the same links, possibly causing congestion while other links are still only lightly utilized. As can easily be seen in the picture, this situation cannot be prevented by changing the link metrics.

Another deficiency of OSPF-based traffic engineering lies in the transient behavior while changing the routing pattern from one metric setting to another. After metric modifications are detected, OSPF routers distribute the new link state information and then recomputed the shortest paths. During this transition phase inconsistencies might arise, affecting active connections, which need to be rerouted (e.g., through increased packet loss or packet reordering). Therefore, it is advisable to consider the number of rerouted flows or the amount of affected traffic, when performing routing adaptation on basis of OSPF. As will be shown later, there exists a tradeoff between achievable QoS enhancement and amount of rerouted traffic.



a) OSPF b) MPLS

Figure 1 OSPF and MPLS routing

B. Connection-Oriented Source Routing: MPLS

Multiprotocol Label Switching (MPLS) [2] was developed to overcome the limitations of conventional routing protocols. MPLS allows the specification of explicit routes through the network, so-called label switched paths (LSPs). At the ingress nodes incoming IP packets are classified and labels are attached to their headers. Routers within the network base their forwarding decisions only on these labels. This way, the routes through the network do not depend on the underlying routing protocol (such as OSPF), but rather on the classification and labeling process at the ingress nodes and the respective forwarding information stored in the label switching routers along the way. Figure 1b) illustrates the advantage of MPLS over OSPF. Instead of having to transmit both flows along the same path, it is now possible to set up two mostly disjoint LSPs and, thus, better utilize available network resources.

In our context, MPLS-based traffic engineering aims at finding an LSP design such that network QoS is optimized, i.e., the maximum link utilization in the network is minimized. Several papers, which deal with this topic, have been published recently. One category of possible approaches is based on the formulation of mathematical optimization models [3]. In cases where there is no limit to the number of LSPs and load sharing is possible across all LSPs, the problem resembles the well-known linear multicommodity flow problem. For all other cases, the continuous problem becomes a mixed-integer one. As these problems are often too complex to be solved analytically, heuristics need to be applied [4][5].

While the proposed methodologies have been developed for pure MPLS networks (i.e., all traffic is carried by LSPs), we investigate scenarios where MPLS is used as a complement to OSPF routing. Assuming that traffic is initially routed by OSPF, we take a few flows off of the shortest paths and route them along LSPs. In order to reduce administrational effort, the number of LSPs should be kept as small as possible. Taking for example the network scenarios in Figure 1, it would be sufficient to establish only LSP 1 and leave flow 2 to be routed by OSPF. (Note: Since MPLS signaling messages (e.g., LDP [6]) are sent as native IP packets, all routers in the network need to be able to differentiate between native IP packets and label switched packets. Therefore, no special technology is necessary in order to mix OSPF and MPLS routing.)

III. OPTIMIZATION MODELS AND SOLUTION PROCEDURES

A. OSPF Link Weight Adaptation

Finding link metrics, which minimize the maximum utilization, is NP-hard. Although the problem can be formulated as a mixed-integer program [7], it is usually too complex to be solved even for medium-size networks. As a consequence, several heuristics have been proposed in the literature. Most of the methods rely on so-called meta-heuristics such as simulated annealing, genetic algorithms, or tabu search [8][9][10][11]. In all cases, a specific routing solution is represented by its respective metric vector. Through variation of the metric vector and repeated evaluation of the resulting routing pattern, the overall solution space is searched for good solutions. The actual search strategy depends on the applied heuristic.

Simulated Annealing Algorithm OSPF-SA

Our algorithm is based on simulated annealing where the metric vector is gradually modified in order to drive the overall objective towards an optimum. The procedure is depicted in Figure 2. After each modification, the new routing together with the resulting link utilization values is computed. Furthermore, the amount of rerouted traffic (in comparison to the original routing configuration) is determined. The metric values are integers in the range of 1 to M_{max}. Although OSPF supports very large metric values (up to 65535), the parameter M_{max} does not have to be very large for optimization purposes. In our procedure we use $M_{max} = 20$.

In order to apply the optimization procedure for routing adaptation purposes, we specify a combined performance criterion. Instead of considering only the maximum link utilization u_{max} as the optimization objective, we take the weighted sum $w_{umax} \cdot u_{max} + w_{flow} \cdot n_{rerouted}$ where $n_{rerouted}$ is the number of flows, which are affected by the adaptation process.

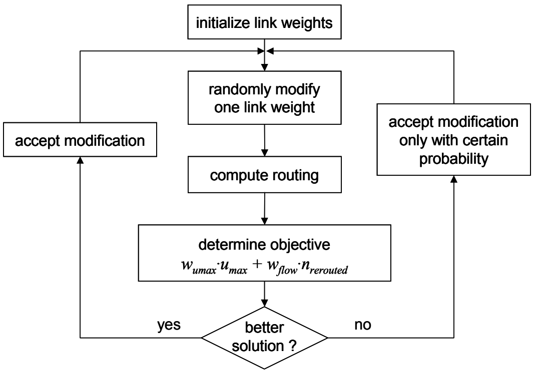


Figure 2 Simulated Annealing Algorithm

Varying the weights w_{umax} and w_{flow} lets us explore the tradeoff between achievable network QoS and the number of rerouted flows. Figure 3 illustrates such a tradeoff investigation for a sample network with 40 nodes and 150 links. The network is loaded with 500 flows, which in case of hop-based shortest-path routing produce a maximum link utilization of 1.03. By optimizing the OSPF metrics this value can be reduced to 0.43. However, the corresponding routing adaptation process would affect 130 flows ($w_{flow} = 0$). Already a slight increase of w_{flow} (i.e., x-axis close to 0 but not exactly 0) leads to a significant reduction of the number of rerouted flows (from 130 to 55) while the maximum utilization is still tolerable (0.49). With increasing weight w_{flow} the number of affected flows is decreased even further, however, only at the expense of rising maximum utilization values.

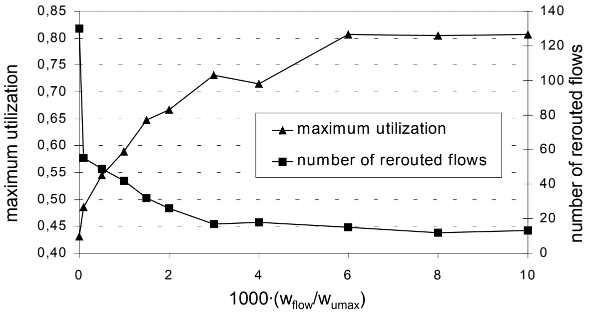


Figure 3 OSPF routing adaptation tradeoff

B. Complementary LSP Optimization

To find an optimal set of complementary LSPs we choose a linear programming approach with two models that can be applied either individually or in combination. The network is modeled as a directed graph $\mathcal{G}$ = ($\mathcal{V}$, A) with node set $\mathcal{V} = \{v_1, v_2, ..., v_N\}$ and edge set $A = \{a_1, a_2, .., a_L\}$. $\mathcal{V}$ contains N vertices representing routers in the network, all of which are capable of label switching as well as OSPF routing. The nodes are also identified by the index set $V = \{1, 2, ..., N\}$. Edge set A comprises L unidirectional links, for which we use notation (i,j) if v_i is the respective source node and v_j the target node. A capacity value c_{ij} is associated with every link (i,j). A number of traffic demands exist, which need to be routed through the network, starting at a flow's ingress node and terminating at the egress node. The individual flows are identified by an index set F. Each such flow f, with $f \in F$, has an effective bit rate of D_f. The ingress and egress nodes of flow f are denoted by $I(f)$ and $E(f)$, respectively. On basis of a given routing pattern (e.g., determined by OSPF for a certain link metric vector), the initial route of each flow can be determined. Thus, it is possible to specify the amount of traffic d_{ij}^f, which flow f originally contributes to the total load on link (i,j) before LSP optimization is performed.

Flow-based LSP Optimization Model (F-LSP)

We can now specify a fundamental optimization model, which is a simple mixed-integer extension of the well-known multicommodity flow problem.

$$\sum_{j:(i,j)\in A} x_{ij}^f - \sum_{j:(j,i)\in A} x_{ji}^f = \begin{cases} y_f, & for\,all\ i \in V, f \in F : i = I(f) \\ -y_f, & for\,all\ i \in V, f \in F : i = E(f) \\ 0, & otherwise \end{cases} \tag{1}$$

$$\sum_{f \in F} y_f \le MaxFlow \tag{2}$$

$$\sum_{f \in F} x_{ij}^f \cdot D_f + \sum_{f \in F} (1 - y_f) \cdot d_{ij}^f \le u_{max} \cdot c_{ij}, \quad for\,all\ (i,j) \in A \tag{3}$$

$$0 \le x_{ij}^f \le 1, \quad for\,all\ (i,j) \in A, f \in F \tag{4}$$

$$y_f \in \{0,1\}, \quad for\,all\ f \in F \tag{5}$$

$$z = \min(u_{max}) \tag{6}$$

Variable x_{ij}^f specifies the share of flow f, which is carried by MPLS and traverses link (i,j). The binary variable y_f indicates whether flow f is actually switched by MPLS ($y_f = 1$) or whether it is routed along the original OSPF path ($y_f = 0$). On basis of y_f, (1) establishes the necessary balance equations for a flow at its source, destination, and intermediate nodes. With (2) the maximum number of label-switched ingress/egress flows ("I/E flows") is limited to *MaxFlow*. In order to formulate our objective of minimizing the maximum utilization in the network (6), the LHS of (3) specifies the total amount of traffic, which traverses link (i,j). It consists of all LSP-carried traffic plus the portion of the traffic still routed by OSPF. For all links, this value has to be smaller than variable u_{max} times the link capacity.

Solving the optimization problem leaves us with possibly several LSPs between an ingress and an egress node since x_{ij}^f can be fractional. If only a certain number of LSPs per I/E flow are allowed, further binary indicator variables need to be introduced. However, this increases the complexity of the model drastically, making it inadequate for larger networks. Therefore, we suggest the following path-based approach to reduce the number of effective LSPs in the network.

Path-based LSP Optimization Model (P-LSP)

For every I/E flow several LSP alternatives are specified. Let $P_f = \{1, 2, ...\}$ be the index set of path options for flow f. Then, set L_{fp} contains the edges along path p, $p \in P_f$, from node $I(f)$ to node $E(f)$. With this, we have:

$$\sum_{p\in P_f} a_p^f = y_f, \quad for\,all\ f \in F \tag{7}$$

$$a_p^f \le b_p^f, \quad for\,all\ f \in F, p \in P_f \tag{8}$$

$$\sum_{p\in P_f} b_p^f \le MaxLSP, \quad for\,all\ f \in F \tag{9}$$

$$\sum_{f\in F} y_f \le MaxFlow \tag{10}$$

$$\sum_{f\in F}\ \sum_{p\in P_f:(i,j)\in L_{fp}} a_p^f \cdot D_f + \sum_{f\in F}(1-y_f)\cdot d_{ij}^f \le u_{max}\cdot c_{ij}, \quad for\,all\,(i,j)\in A \tag{11}$$

$$0 \le a_p^f \le 1, \quad for\,all\ f \in F, p \in P_f \tag{12}$$

$$y_f \in \{0,1\}, \quad for\,all\ f \in F \tag{13}$$

$$b_p^f \in \{0,1\}, \quad for\,all\ f \in F, p \in P_f \tag{14}$$

$$z = \min(u_{max}) \tag{15}$$

Equations (7) specify, that whenever I/E flow f is label-switched, the traffic is distributed across the possible LSPs. Having the binary "LSP-in-use" variable b_p^f, inequalities (8) and (9) keep the number of activated LSPs per I/E flow below a certain threshold *MaxLSP*. Similar to (3), (11) gives the lower bound for the maximum utilization variable, which again is minimized (15).

Optimization Procedure

In a first step, F-LSP is solved. The optimization process returns the set of I/E flows, which need to be considered for label switching in order to minimize the maximum link utilization. With a simple heuristic, the necessary LSPs can be derived from the flow variables x_{ij}^f. Since at this point, the number of LSPs for some I/E flows might exceed a desired maximum *MaxLSP*, the result of F-LSP can be fed into P-LSP, which will end up with a specified number of active LSPs per I/E flow.

Optimization model P-LSP can of course be used without first having to solve F-LSP. Whenever a set of possible LSP alternatives is given for all I/E flows (e.g., k-shortest paths), it can be applied directly.

IV. RESULTS AND DISCUSSION

The optimization procedures were applied to several network scenarios of different size under various load conditions. The number of nodes (edges) was in the range of 11 (48) to 50 (170) with 100 to 1000 flows. The results presented below were obtained for a hierarchical network with 40 nodes, 150 links, and 500 flows (same as in section III.A). The flows' bit rates were randomly distributed and all of the same order.

We consider three approaches to optimized routing adaptation: First, traffic flows are only adjusted by modifying link metrics ("pure OSPF adaptation"). Second, OSPF metrics are left unchanged and only LSPs are introduced in order to improve network QoS. Third, a combination of both is applied.

Routing all flows according to hop-based OSPF, causes a maximum link utilization of 1.03. On the other hand, by putting all flows under the control of MPLS without limiting the number of LSPs, a u_{max} of 0.26 can be achieved (lower bound). However, naively solving the corresponding linear multicommodity flow problem and deriving the necessary LSPs from the flow variables would leave us with a total of 597 LSPs. As we can see from the following graphs, a much smaller number is already sufficient if MPLS is only used as a complement to OSPF.

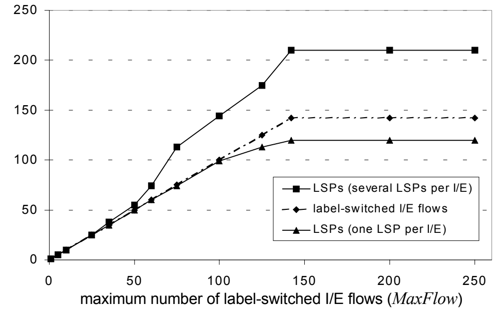


Figure 4 Number of MPLS flows and LSPs over MaxFlow

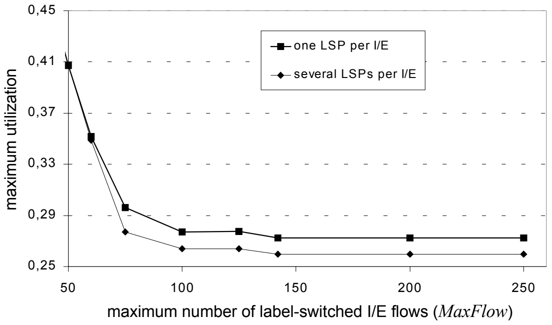


Figure 5 Maximum utilization over MaxFlow

Figure 4 and Figure 5 illustrate the process of complementary LSP optimization. Starting from pure OSPF routing (i.e., *MaxFlow* = 0), the number *MaxFlow* of allowed MPLS flows is slowly increased. By solving F-LSP models we obtain the optimum sets of label-switched I/E flows with possibly several LSPs per flow. To restrict the number of LSPs per I/E flow to one, a subsequent P-LSP optimization is carried out, taking all LSPs from the F-LSP model as path alternatives. The graph in Figure 4 shows the actual number of I/E flows and LSPs for various values of *MaxFlow* with an optimality gap of up to 2%. In Figure 5 the respective maximum utilization values are plotted. Following observations are noteworthy:

- significant u_{max} reductions are already achieved for relatively small numbers of label-switched I/E flows and LSPs, e.g., u_{max} is about 0.41 (0.29) for 50 (75) LSPs .
- the optimum u_{max} is reached for 142 label-switched I/E flows with a total of 210 LSPs.
- applying P-LSP after F-LSP optimization might greatly reduce the total number of LSPs while still achieving very good network QoS, e.g., for $MaxFlow = 142$, the number of LSPs decreases from 210 to 120 while u_{max} is only increased by 0.01.

All mixed-integer programs were solved with CPLEX 7.5. On an Intel Pentium 4 computer (2.4 GHz, 2 GByte RAM) the processor time for solving F-LSP and subsequent P-LSP problems was in the order of a few minutes. As indicated in Section III.B, F-LSP problems with LSP-restrictions could not be solved for larger *MaxFlow* values (time limit: 20 h).

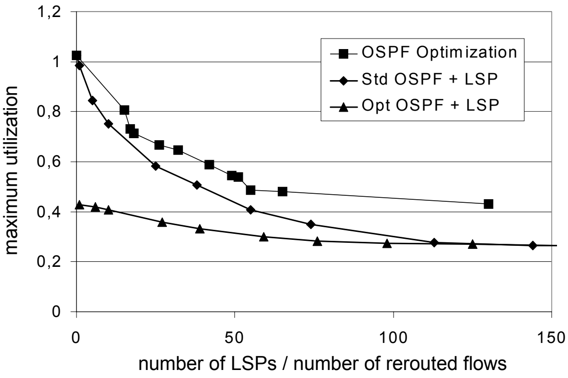


Figure 6 Comparison of route adaptation strategies

In Figure 6 the performance of the different routing adaptation strategies is compared. We again assume that all traffic is initially routed according to hop-based OSPF ($u_{max} = 1.03$). First, routing is optimized only based on OSPF. Using our simulated annealing algorithm and varying the objective weights, we can obtain several metric settings, which reflect the tradeoff between network QoS and rerouted flows (upper curve in Figure 6). The best u_{max}, which can be achieved by OSPF optimization, is 0.43. The corresponding route adaptation process affects 130 flows. In a second approach, we leave OSPF routing unchanged and introduce a number of complementary LSPs (middle curve). This procedure has already been discussed in the preceding paragraphs. The third method is basically the same as approach 2, only that now OSPF routing is optimized before being complemented by MPLS (lower curve). A network operator could use such a graph to select an appropriate route adaptation procedure that would fit the individual needs. Assuming that a u_{max} of 0.6 would be considered sufficient, it is probably not advisable to install MPLS equipment and to set up LSPs. A slight modification of OSPF link metrics would be just as good. If u_{max} should be decreased to around 0.4, the decision is made a lot harder. The operator could stick with only OSPF and optimize the link metrics. However, as this would not leave much room for further optimization (e.g., in case of additional traffic growth), it would be worthwhile considering complementary LSPs. Depending on whether OSPF is optimized or not, a total of about 10, respectively 55 LSPs are necessary.

V. CONCLUSION

In this paper we have proposed various procedures for the optimization of routing adaptation in IP networks. It has been shown that routing adaptation based on OSPF and complementary MPLS can achieve great QoS improvements without having to install a large number of LSPs. In some cases it might even be good enough to apply only partial OSPF-based routing adaptation, where some of the obtainable QoS enhancement is given up in favor of a lower number of rerouted traffic flows. An algorithm, which lets an operator investigate this tradeoff, has been presented. For cases, where pure OSPF routing optimization is not sufficient, two mixed-integer programming models have been specified. They can be used to compute the optimal set of LSPs, which need to be installed in the network. In future work, the proposed methods will be extended in order to take into account multiple traffic classes with different QoS requirements.

REFERENCES

[1] J. Moy, “OSPF Version 2,” IETF RFC 2328, April 1998.

[2] E. Rosen, A. Viswanathan, R. Callon, “Multiprotocol Label Switching Architecture,” IETF RFC 3031, January 2001.

[3] T. Carpenter, K.R. Krishnan, D. Shallcross, “Enhancements to Traffic Engineering for Multi Protocol Label Switching,” Proceedings of 17th International Teletraffic Congress (ITC), Salvador da Bahia, December 2001.

[4] Y. Wang, Z. Wang, “Explicit Routing Algorithms for Internet Traffic Engineering,” Proceedings of 8th International Conference on Computer Communications and Networks (ICCCN), 1999.

[5] S. Schnitter, G. Haßlinger, “Heuristic Solutions to the LSP-Design for MPLS Traffic Engineering,” Proceedings of 10th International Telecommunication Network Strategy and Planning Symposium (NETWORKS), Munich, June 2002.

[6] L. Andersson et al., “LDP Specification,” IETF RFC 3036, January 2001.

[7] A. Bley, M. Grötschel, R. Wessäly, “Design of Broadband Virtual Private Networks,” Model and Heuristics for the B-WiN. Technical report, Preprint SC 98-13, Konrad-Zuse_Zentrum für Informationstechnik, Berlin, 1998.

[8] B. Fortz, M. Thorup, “Internet traffic engineering by optimizing OSPF weights,” Proceedings of INFOCOM 2000, Tel-Aviv, Israel, March 2000.

[9] M. Ericsson, M.G.C. Resende, P.M. Pardalos, “A Genetic Algorithm For The Weight Setting Problem in OSPF Routing,” J. of Combinatorial Optimization, 6:299-333, 2002.

[10] E. Mulyana, U. Killat, “A Hybrid Genetic Algorithm Approach for OSPF Weight Setting Problem,” Proceedings of the 2nd Polish-German Teletraffic Symposium PGTS 2002, Gdansk, Poland, 2002

[11] A. Riedl, “A Hybrid Genetic Algorithm for Routing Optimization in IP Networks Utilizing Bandwidth and Delay Metrics,” Proceedings of IEEE Workshop on IP Operations and Management (IPOM), Dallas, USA, 2002.