A Hybrid Genetic Algorithm for Routing Optimization in IP Networks Utilizing Bandwidth and Delay Metrics

Anton Riedl
Institute of Communication Networks, Munich University of Technology
Munich, Germany
E-mail: Anton.Riedl@ei.tum.de

Abstract **– In this paper we study routing optimization in the context of IP traffic engineering, which relies on conventional, destination-based routing protocols. We introduce the different concepts of routing optimization and discuss their implications for traffic engineering. The specific focus is on routing technologies that utilize multiple metric types – in our case delay and bandwidth metrics – in order to derive the shortest paths towards every destination in the network. A novel hybrid genetic algorithm is presented, which allows the computation of an optimized set of link metrics, considering single as well as dual-metric protocols. Finally, the benefits of two metric types for traffic engineering are demonstrated.**

Keywords – IP Traffic engineering, genetic algorithm, bandwidth-delay sensitive routing, destination-based routing, OSPF, EIGRP

I. Introduction

Routing optimization is an important method of Internet traffic engineering, which is applied on a medium-term time basis (at least some hours) and to a rather coarse level of flow granularity (e.g., aggregation of all traffic flows between specific ingress/egress nodes). In cases where increasing traffic loads or temporary traffic variations cause localized link congestions, routing optimization can be carried out to resolve – or at least alleviate – potential network performance problems. The idea is to adjust routing to current load situations and, thus, better utilize available network resources, leading to improved Quality of Service (QoS). The routing optimization problem can be formulated as follows: Given a dimensioned network and a traffic demand matrix, we would like to find a routing solution, which optimizes a certain network QoS measure. As QoS measure several expressions are conceivable. Most definitions found in the literature are based on link utilization values, as these correlate with packet delay and packet loss within routers. A common objective of routing optimization, which we also adopt in this work, is the minimization of the maximum link utilization in the network. This measure is quite intuitive and also very simple to determine throughout the optimization process.

In this paper we consider routing optimization based on conventional interior gateway protocols. These protocols rely on link metrics in order to derive the shortest paths and to determine the outgoing interfaces towards all destinations. As a consequence, it is possible to optimize routing by setting the metric values appropriately. We specifically focus on routing protocols, which take into account two types of link metrics – one for delay and one for bandwidth. However, in the context of routing optimization the two metrics do not really have any physically relevant meaning. Instead, they are used purely as generic means for the sake of traffic engineering.

The remainder of the paper is organized as follows: In section II existing routing concepts are presented and their implications for routing optimization are discussed. We show how shortest paths are determined when delay and bandwidth metrics are considered and illustrate the advantage of dual-metric protocols over their single-metric counterparts. In section III a hybrid genetic algorithm for the computation of optimized routing schemes is introduced. Computation results and performance issues are presented in section IV. Section V concludes the paper.

II. Routing Optimization Concepts in IP Networks

A. Destination-Based vs. Source/Flow-Based Routing

Two fundamentally different routing concepts exist, which strongly influence the optimization procedure and the achievable results: destination-based routing and source- or flow-based routing. Conventional routing protocols such as OSPF [1], EIGRP [2], or IS-IS [3], follow the next-hop destination-based routing paradigm. Within each router the forwarding decision for an IP packet is based solely on the destination address specified in the packet header. No information about the origin or any other context of the packet is taken into account. As a consequence, this routing procedure is simple and quite efficient. However, it imposes limitations on routing optimization, as illustrated in Figure 1.

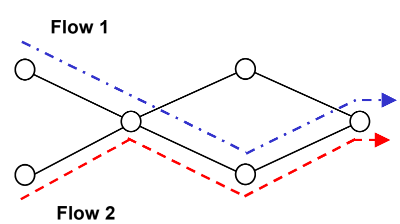

Figure 1 Limitations of destination-based routing

Whenever two traffic flows with the same destination cross each other's way they are merged and sent out over the same interface. This might cause traffic overload on some links, while other links are still only lightly utilized.

To overcome these limitations, new flow-based routing technologies such as Multiprotocol Label Switching (MPLS) [4] have been developed. MPLS makes it possible to establish an overlay routing structure within the IP network – independently of the used routing protocol – and to set up explicit paths for individual traffic flows. Each MPLS-routed IP packet is marked with a special label, which the routers along the way consider for forwarding decisions. Thus, the routing pattern does not depend on the underlying routing protocol, but rather on the label and forwarding information that is stored in the MPLS routers. Instead of determining the path of a packet based on the destination address in a hop-by-hop manner, the path is now fixed by the router, which marks the packet with the appropriate label. This introduces a high degree of "routing freedom" as any desired routing pattern can be obtained.

B. Single-Metric vs. Dual-Metric Routing

In the case of destination-based routing protocols a router determines an outgoing interface based on metric values, which quantitatively describe the distance to a destination node. Most commonly, single additive metrics are assigned to every link and a shortest-path algorithm is used to determine the preferred path from each node to every other node in the network ("single-metric routing"). While link metrics often have physically relevant meanings such as "delay" or "cost", they can also be used in a generic way purely for the sake of routing optimization. By setting appropriate link metric values, one can indirectly influence and, thus, optimize the routing scheme.

In addition to single-metric protocols, routing schemes exist, which allow more than one metric taken into account when computing the length of a path towards a destination node ("multiple-metric routing") [5][6]. One example is Cisco's routing protocol EIGRP, which incorporates four metric types. However, only two of them are used by default: one additive metric ("delay") and one concave metric ("bandwidth"). In the following, we will focus on these two metric types and show how they can be used for routing optimization. The distance to a destination node is now computed by the normalized metric formula

$$M = \frac{1}{\min_i (bw_i)} + \sum_i d_i = \max_i (icm_i) + \sum_i d_i \,.$$

Parameter bw_i denotes the bandwidth of link i, while d_i refers to its static delay value. Thus, a router takes the sum of all delay values towards the destination node and adds a bandwidth component, which is the inverse of the smallest bandwidth along the path ("bottleneck"). From all possible path options it selects the one with smallest path metric M. For further considerations we will refer to the bandwidth component as "inverse capacity metric" *icm* and take the maximum along the path instead of the reciprocal value of the bandwidth minimum. Figure 2 illustrates the concept of bandwidth-delay sensitive routing. The distance metric M associated with path A-B-C-E is 7 (delay sum of 3 plus bandwidth component of 4), while path A-D-E has only an overall metric of 5. As a consequence, router A selects router D as its next-hop neighbor towards E, and flow 1 is routed over A-D-E.

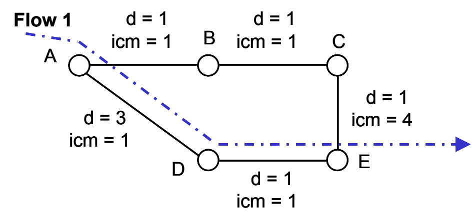

Figure 2 Dual-metric routing

Depending on the values for *d* and *icm*, emphasis is either put on small overall delay (i.e., "short" paths), on high throughput, or on a mixture of both. In case metrics *d* are substantially larger than metrics *icm*, the overall path metric is mainly determined by delay values. Only when there are several alternatives with equal smallest delay sum, the bandwidth component really matters. The router then selects the outgoing interface with the largest possible throughput ("widest-shortest path"). In the opposite case (*icm* >> *d*), high throughput paths are preferred, and delay is mainly used to break ties ("shortest-widest path"). Whenever the two link metrics are of the same order, no clear preference is given to either one of them. Link metrics are then used in their most generic form as means of routing optimization without any physically relevant significance.

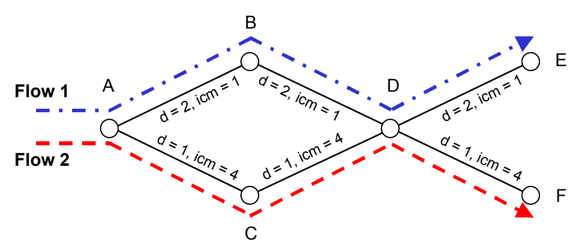

Figure 3 Fish-pattern routing with two metrics

Routing optimization based on dual-metric protocols has the potential to achieve better results than its single-metric counterpart. Since the second metric introduces some more flexibility, a greater variety of routing patterns can be realized. Note that it is always possible for dual-metric protocols to produce single-metric routing patterns. Simply setting all *icm* values to the same constant *c* would result in a single-metric shortest-path computation with metric function *(M-c)*. The benefit of dual-metric protocols can be demonstrated by means of the network scenario in Figure 3. Assume we have two traffic flows with different destinations, whose paths have several nodes in common. Let A be the first node where the two flows come together and D be the last common node on their way. While single-metric shortest-path routing would merge the flows at node A and send both of them either via B or via C, dual-metric routing protocols can achieve the flow pattern given in the figure. For flow 1, the chosen path has a total metric of 7, while the link metrics along the route via C would sum up to 8. For flow 2 the situation is different and the total metrics of the upper and the lower path are 9 and 7, respectively. The trick is to use the inverse capacity metric to make one path option appear more costly for one traffic flow, while for the other flow a larger *icm* value has no extra effect (since it already experiences high *icm* values on further links, which the two flows do not share). However, the actual gain in QoS, which EIGRP can achieve over OSPF, depends greatly on the network topology and the traffic demands.

C. Equal-Cost Multi-Path ECMP

Another possible feature of routing protocols, which influences the optimization process, is load sharing. In destination-based routing protocols this capability is often implemented in form of the "equal-cost multi-path" concept. Whenever a router can reach a destination node via several paths with equal metric sums, it splits up the traffic evenly across all corresponding outgoing interfaces. Note, if dual-metric routing protocols are used, "equal cost" does not necessarily mean that all metric components are the same on all load-sharing paths. While metric combinations of several paths might be equal, their delay and bandwidth portions could be quite different.

D. Routing Optimization Classes

Based on the concepts discussed in the preceding paragraphs various categories of IP routing optimization can be identified. Source and flow-based routing protocols ("MPLS") certainly provide the greatest potential for routing optimization. Assuming that there are no special restrictions concerning the routes through the network, we can try to find a global optimum by solving a mixed-integer program for the multicommodity flow problem (MC) or by applying an appropriate heuristic. However, at this point we focus on destination-based routing protocols. Therefore, we consider it sufficient to obtain a lower bound of the routing optimization problem by solving the continuous version of MC by linear programming (LP).

The category of destination-based routing optimization is further divided into single-metric ("OSPF") and multiple-metric ("EIGRP") models, each having a version with and without load-sharing ("ECMP"). Most of the literature about this type of IP routing optimization deals with OSPF as the underlying protocol [7][8][9][10]. A mixed-integer programming formulation of the routing optimization process for bandwidth-delay sensitive protocols has been presented at [11]. However, since it is not possible to solve the mixed-integer program for medium-size and large networks, appropriate heuristics are necessary.

III. Genetic Algorithm for Routing Optimization

A. Genetic Algorithm Basics

Genetic algorithms (GA) are based on the idea of natural selection. It is suggested that an individual's strength to survive in the world is determined by its gene structure and that over many generations only "good" genes prevail, whereas "bad" ones are rejected. Furthermore, it is expected that bringing together individuals with good gene combinations produces again good or even better ones.

Genetic algorithms apply this principle to optimization problems by representing possible solution alternatives through appropriate gene strings and performing operations of "natural selection" on these strings [12]. At first, a random set of strings is generated ("Generation 0"). Then, the three basic operators "reproduction", "crossover", and "mutation" are carried out repeatedly until some termination criterion is reached. During each iteration ("generation") the existing strings are transformed into solutions and their quality ("fitness") is evaluated. Figure 4 illustrates the general procedure of genetic algorithms.

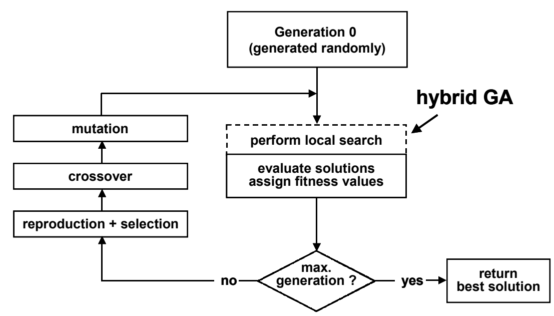

Figure 4 Genetic algorithm flow chart

The reproduction process creates a new generation. Starting from an existing generation, strings are reproduced with a probability proportional to the quality of the corresponding solution. Strings, which represent solutions with good properties, have a higher chance to survive than strings depicting solution points with bad characteristics ("survival of the fittest"). The crossover operator chooses pairs of strings, breaks up their gene sequence at random places, and exchanges the genetic information. Finally, the mutation operator introduces new genetic material by randomly selecting and changing single genes. Mutation is important to partially shift the overall search process to new locations within the solution space. Otherwise, the search process would converge to a local optimum without having the chance to consider any further points.

As indicated in Figure 4, we extend the pure genetic algorithm framework and perform a local search heuristic before evaluating the solutions (therefore, "hybrid GA"). In the literature, this combination of genetic algorithms with a local search process is also referred to as "memetic algorithm".

B. *Implementation of the Algorithm*

String Representation

A crucial point of every genetic algorithm is the string representation of possible solution alternatives. After every crossover and mutation step, one has to be able to turn the resulting strings into valid solutions. Our approach is analog to the ones presented in [9][10][13]. We enumerate all links in the network and associate link weights with each of them. A specific gene string contains the weights of all links in the order of their enumeration. For the single-metric routing case, we have one string representing a routing scenario (i.e., one metric associated with every link), while in the bandwidth-delay case a genetic representation of a certain solution requires two strings (one for *d* and one for *icm*). From the metric strings, a specific solution can be deduced by applying the respective shortest-path computation either with one or with two metrics. Link weights are integers ranging from 1 to a maximum value. In practice, routing protocols allow quite large maximum values (e.g., in case of OSPF up to 65535). However, for traffic engineering and routing optimization purposes the metric values can be kept much smaller.

Fitness Function and Power Scaling

As we would like to minimize the maximum link utilization in the network, we choose the inverse of this value as our fitness function. This way, routing solutions with smaller maximum link utilizations receive higher fitness values and, thus, have a higher chance to be reproduced when setting up a new generation. In order to influence the reproduction process we apply power scaling to the fitness function:

$$fitness = \left(\frac{1}{\max\limits_{i \in links} (utilization_i)} \right)^p \qquad p > 0 \cdot$$

With $p < 1$ we can achieve that fitness values of bad solutions are increased relatively to the best ones, thus, avoiding that they die out too fast and that the optimization process converges too early. For $p > 1$, the gap between good and bad solutions is increased, forcing the process to converge faster.

C. *Local Search Heuristic*

In order to improve the performance and the speed of the genetic algorithm, a local search process precedes the evaluation step shown in Figure 4. Before evaluating each routing solution, simple heuristics are used to divert traffic from the link with the highest utilization. This is done repeatedly until no further improvement can be achieved. As the search heuristic is deterministic, a certain metric string always results in the same routing solution. Thus, it is suitable for the use within the genetic algorithm framework.

Single-Metric Heuristic

In case of shortest-path routing, traffic can be diverted from a link by increasing its metric. Therefore, we increment the metric of the highest-utilized link and reroute all traffic flows. If this step leads to a higher utilization value in the network it is reverted and the local optimization heuristic ends. Otherwise, the procedure is repeated for the link, which now shows highest utilization. This search heuristic is analog to the one in [10].

Dual-Metric Heuristic

For routing protocols with bandwidth and delay metrics there are some more possibilities of diverting traffic from individual links. We can increase either the delay metric or the inverse capacity metric. Furthermore, combinations are possible that might also result in a change of the path structure: the delay metric can be increased while at the same time the *icm* metric is decreased, or vice versa. Therefore, the search heuristic iteratively applies these metric modifications to the highest-utilized link. Each metric change is accepted if it does not lead to an increase of the maximum utilization. If no further improvements can be obtained, the heuristic stops.

IV. Results and Algorithm Evaluation

A. *Comparison of GA and Hybrid GA*

At first, we would like to demonstrate the benefits of the local search heuristic. In the graph of Figure 5 the best maximum utilization value of each generation is shown over time for a sample run of the pure genetic algorithm GA and of the hybrid version. The total time is about 10 minutes on a Linux PC. For GA, the population size was set to 2000, while the hybrid algorithm works with a population size of 500. Since GA is less complex, it proceeds faster from generation to generation even though its population size is four times larger. In 10 minutes it has produced about 150 generations, while the hybrid version only reaches about 100 generations. However, from the beginning on, the maximum utilization values of the hybrid algorithm lie below the values of pure GA. The lower bound corresponds to the linear multicommodity problem, which can be solved by any standard LP-solver such as CPLEX.

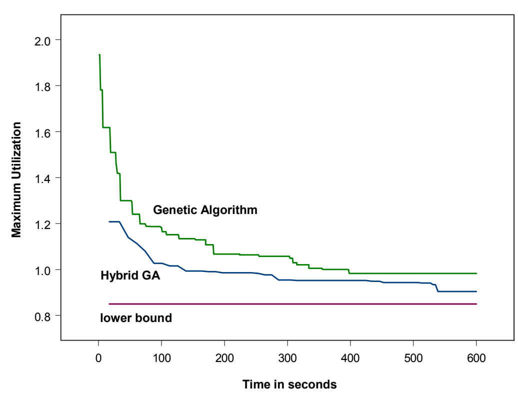

Figure 5 Comparison of GA algorithms

B. *Comparison of Routing Optimization Approaches*

The hybrid GA was applied to several network architectures. Table I gives the parameters of four typical scenarios. Scenario 1 is a hierarchical network with 39 access nodes, each connected to one of 11 backbone nodes. Traffic enters the network only at three backbone sites. From there traffic relations exist with each of the access nodes. This scenario represents a network where the majority of traffic comes from external sources through peering points or from servers, which are located at a few central sites. Scenario 2 comprises only the backbone of scenario 1. All access nodes are collapsed into the respective backbone node and the traffic flows are aggregated accordingly. Scenarios 3 and 4 are networks of different size without distinct hierarchy levels and without any specific traffic hot spots. All traffic flows have been generated randomly between any two nodes of the network. For each of the four scenarios the maximum link utilization is given, which results from standard OSPF routing with all link metrics equal to 1.

TABLE I
NETWORK PARAMETERS

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
nodes/links	50/63	11/24	20/30	50/85
number of flows	117	33	200	400
characteristics	hierarchical network	only backbone	flat network	flat network
max. utilization	0.573	0.573	0.55	0.785

Table II summarizes the optimization results for the considered topologies. As expected, dual-metric routing protocols ("EIGRP") are able to achieve lower utilization values than single-metric routing protocols ("OSPF"). Furthermore, load sharing – although quite limited for destination-based routing – allows additional QoS improvement. However, the achievable performance gain depends on the network topology and the load situation. In our scenarios the QoS advantage for EIGRP over OSPF ranges from only about 0.3% (scenario 3) up to roughly 17% (scenario 1). The gain, which can be obtained by activating ECMP, also varies from about 2% to 15%.

TABLE II
OPTIMIZED MAXIMUM UTILIZATION VALUES

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Lower Bound	0.33	0.33	0.347	0.338
OSPF	0.406	0.406	0.364	0.384
OSPF ECMP	0.353	0.353	0.355	0.361
EIGRP	0.347	0.383	0.363	0,365
EIGRP ECMP	0.334	0.347	0.351	0.350

The strength of EIGRP optimization becomes most apparent when comparing the results of scenario 1 and scenario 2. Whenever the metric values of the access links are considered for the route computation process, i.e., the access links are part of the relevant network topology (scenario 1), one can best take advantage of the bandwidth metrics. In these cases we have many flow relations as indicated in Figure 3 (assume that nodes A, B, C, and D are backbone nodes, while E and F are access nodes). While OSPF would aggregate all traffic flows that go through the same final backbone node (irrespective of the actual target node behind this backbone node), EIGRP is able to select different routes and, thus, possibly distribute the traffic in a better way. This particular advantage is given up if routing is performed only on basis of the backbone network (scenario 2), leading to a utilization increase of about 10%.

V. CONCLUSION

In this paper we have discussed different possibilities of routing optimization as a means of IP traffic engineering. For the optimized metric setting problem we have presented a hybrid genetic algorithm, which considers simple-metric as well as dual-metric destination-based routing protocols. In addition to the typical functionality of genetic algorithms, our version includes a simple search heuristic, which speeds up the optimization process.

The results obtained for various network scenarios demonstrate that routing optimization in general can greatly improve network performance. Furthermore, the use of two metric types can be beneficial, enhancing network QoS even more. Although the actual gain depends on the topology and the load situation, it could still make a significant difference, especially when traffic is approaching some undesirable threshold.

REFERENCES

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

[2] "Enhanced Interior Gateway Routing Protocol," Cisco White Paper EIGRP, http://www.cisco.com/warp/public/103/eigrp-toc.html

[3] R. Callon, "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments," IETF RFC 1195, December 1990

[4] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture," *IETF RFC 3031*, January 2001

[5] Z. Wang, J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications", IEEE Journal of Selected Areas in Communications, Vol. 14, No. 7, pp. 1228-1234, 1996

[6] S. Chen, K. Nahrstedt, "An overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions," IEEE Network Magazine, 12(6):64-79, November/December 1998

[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," to appear in J. of Combinatorial Optimization

[10] M.G.C. Resende et al., "A memetic algorithm for the weight setting problem in OSPF routing," presented at 6th INFORMS Telecommunications Conference, Boca Raton, Florida, March 2002

[11] A. Riedl, D.A. Schupke, "A Flow-Based Approach for IP Traffic Engineering Utilizing Routing Protocols With Multiple Metric Types," presented at 6th INFORMS Telecommunications Conference, Boca Raton, Florida, March 2002

[12] D.E. Goldberg, "Genetic Algorithms in Search, Optimization & Machine Learning," Addison-Wesley, Massachusetts, 1989

[13] A. Riedl, "A Versatile Genetic Algorithm for Network Planning," in Proc. of EUNICE'98, Munich, September 1998