Hop Limits in Network Design: What is the Impact on Optimal Network Cost?

Sebastian Orlowski and Roland Wessäly
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Email: {orlowski,wessaely}@zib.de

***Abstract*— We investigate the impact of hop-limited routing paths on the total cost of a telecommunication network. For different survivability settings (no survivability, link and path restoration), the optimal network cost without restrictions on the admissible path set is compared to the results obtained with two strategies to impose hop limits on routing paths.**

Based on optimal solutions for 10 real-world based problem instances, we show that hop limits may lead to an unpredictable raise in total network cost – even with large hop limits. The total network cost with a hop limit of 7 hops for all demands can be up to 25% higher than without restrictions on the admissible path set. With our second strategy, which imposes demand-dependent hop limits based on the shortest hop count, we obtain similar results. This indicates that column generation techniques should be applied to deal with all admissible paths.

***Index Terms*— survivable network design, hop limits, routing, restoration, branch-and-cut algorithm, mixed-integer programming**

I. Introduction

A. Motivation

In the design of telecommunication networks, many issues have to be taken into account. Among others, there may be constraints on the admissible routing paths like a maximum number of allowed hops (the number of links in a path). Such hop limits can have technological reasons (such as degradation of signal quality or too high transmission delay when the number of hops increases), or they can merely be imposed to simplify the planning process. This raises two questions:

1) Do hop limits really simplify the planning process?
2) How much does the overall network cost increase when hop limits are imposed?

On ten real-world based problem instances, this paper compares the optimal network cost with different kinds of admissible path sets: without any restrictions, with demand-independent hop limits, and with demand-dependent hop limits. Each of these path sets is tested with different survivability settings (no survivability, link or path restoration), and with different hop limits (where appropriate). It turns out that the quality of the solutions obtained with a hop-limited path set compared to the optimal solutions obtained by allowing all routing paths is rather unpredictable. Thus, whenever the used technology allows it, it is advisable to allow all routing paths and to employ column generation techniques. However, if hop limits are used, they should be demand-dependent rather than globally fixed.

This work was supported by the DFG Research Center "Mathematics for key technologies" (FZT86), Berlin, `www.fzt86.de`.

B. Literature

In many publications on network design in which a routing has to be determined, a mathematical model with a path flow formulation (also called arc-path formulation) has been used to tackle the arising multicommodity flow subproblems. Compared to an edge flow (also called node-arc) formulation, such a path flow formulation has the main advantage that restrictions on the admissible path set can be more easily modeled. In addition, such a formulation has a relatively small number of constraints, but unfortunately at the expense of an exponential number of path variables.

To cope with the large number of variables and to reduce calculation times, some authors apply column generation techniques [1]–[4], while other authors feed a fixed, precalculated set of routing paths into an LP or MIP solver. For the latter approach, several ways of defining the admissible path set have been presented. Common variants are

1) a global hop limit which is the same for all demands [5], [6],
2) a demand-dependent hop limit, which is the length of a shortest hop path for a given demand plus some fixed additional number of hops [7], and
3) a demand-dependent hop limit, which is iteratively raised until a specified number of paths has been found for each demand [8].

We now briefly present the problems investigated in these papers, together with the used solution approach.

Murakami [1] uses a path flow formulation for comparing the cost of path restoration with stub release and link restoration under a single link failure scenario, either with a given working path routing or with joint optimization of continuous working and spare capacities. The author applies column generation for both working and restoration paths using a (quadratic) shortest path algorithm.

Dahl and Stoer [2] formulate the problem of installing discrete working and spare link capacities for the survivability models reservation and diversification with so-called metric inequalities [9]. These inequalities are generated at runtime using a path flow formulation which is solved using column generation for routing paths in all operating states.

Poppe and Demeester [3] use a similar model to formulate the problem of installing continuous spare capacities for link

and path restoration based on a given working path routing. Column generation is used to identify missing restoration paths.

Wessäly [4] determines discrete working and spare capacities using a path flow formulation for reservation, diversification, or path restoration. This path flow LP is solved by generating working and restoration paths only when needed.

Herzberg and Bye [5] is the only paper known to us in which the cost effect of hop limits is investigated to some extent. The authors consider the spare capacity assignment problem for a given working path routing under a single link failure scenario with link restoration. A path flow formulation with continuous capacities is presented. On one small but well connected test instance (11 nodes, 23 links) with integer capacities (obtained by solving a linear program, rounding the capacities up to the next integer and trying to heuristically lower some of the capacities again), the effect of hop limits is tested by enumerating all paths up to a given number of hops, which varies between 3 and 7. The results show that on the investigated test instance, restricting the restoration paths to a length of at most five hops is enough to obtain optimal solutions, which the authors state to having observed on other test instances as well.

Xiong and Mason [6] use a path flow formulation for path restoration without stub release and for link restoration under a single link failure scenario. A set of working and restoration paths is precalculated, which contains at most 40 paths per demand. A path length restriction of 6 and 10 hops is imposed for the two small and the two larger test instances, respectively.

Iraschko, MacGregor, and Grover [7] compare the cost of link and path restoration (with or without stub release) for single link failures both with a predefined shortest path routing and with joint working and spare capacity optimization. Capacities and flow variables in the path flow model are allowed to take any integer value. For each demand, the authors enumerate all paths up to a given hop limit, which is the length of a shortest hop path for this demand plus a fixed number of additional hops, complemented by a small set of link disjoint paths to guarantee a solution. This leads to a large path set and thus probably to good solutions, although no good lower bounds are given for the test networks. The authors report on very long calculation times even on small test instances as soon as working and spare capacities are optimized together (9 hours for link restoration, 2.7 days for path restoration without stub release on an instance with 10 nodes, 22 links, and 45 demands).

Doucette and Grover [8] compare several protection and restoration mechanisms for networks of varying density. The authors use a path flow formulation with arbitrary integer capacities and a predetermined path set. All paths up to an iteratively adapted hop limit are enumerated until at least 5, 10, or 20 paths have been found for each demand (the exact number depends on the considered problem).

C. Discussion of the different approaches

A column generation approach can be used to obtain optimal solutions whenever the pricing problem (i.e., the problem of identifying missing path variables) is exactly solvable in polynomial time. For instance, this is the case when restrictions on the admissible path set are absent or take the form of hop limits, which are often used settings. On the contrary, when the pricing problem is $\mathcal{NP}$-hard (e.g., for path restoration with stub release [10]), the column generation approach does no longer yield guaranteed optimal solutions, but it can still be used as a good heuristic.

The main advantage of a predefined path set is the fact that nearly arbitrary wild path set restrictions can be incorporated in the model. In practice, the predefined path set often consists simply of all paths up to a given number of hops (which may be demand-dependent or not). On the other hand, as the set of possible paths per demand is often too large to be completely enumerated, this approach usually yields heuristic solutions.

Although popular belief states that this strategy often leads to near-optimal results, we know of no explicit study of the cost effects of hop limits which is, like the one presented in this paper, based on a modular link capacity model, integrated optimization of working and failure routings, optimal solutions, and a reasonably large set of test instances. However, these ingredients are indispensable for a fair comparison of the different approaches under realistic planning conditions. This is a gap which we intend to fill with this work.

As far as calculation times are concerned, these are often comparable with a predefined path set and with column generation, according to several of the authors mentioned above. For sparse networks and small hop limits, enumerating all admissible paths is quickly done and leads to small calculation times. However, as the number of paths increases exponentially with density and hop limits, available memory and calculation time can be easily exceeded. This is the point where column generation comes in, since the calculation times with this approach increase much more slowly with the number of allowed hops than by enumerating all paths.

This paper is structured as follows: after a brief sketch of our mathematical model and algorithmic approach in Section II, we present and discuss our computational tests and results in Section III. Eventually, we conclude with Section IV.

II. Model and Algorithm

We investigate network design problems dealing with an integrated planning of

- a topology,
- modular link capacities,
- a routing during normal operation, and
- a routing in all single link failure states.

The used model is derived from the mixed-integer linear programming formulation described in [11], which also models hardware requirements imposed by network elements and interface cards. These extensions, as well as the possibility to respect existing parts of a network (as opposed to greenfield

planning) are actually implemented in our network planning tool DISCNET [12] but have been omitted in this paper.

The given networks consist of a set of nodes and potential (undirected) links between these nodes. For each of these links, some set of *link designs* (e.g., STM-N or OC-N capacities) may be specified, out of which at most one may be installed. Every link design installable on a given link has a capacity and a cost value assigned to it. The latter can be composed of a fixed installation cost and length-dependent cost. The final topology consists exactly of those links for which a link design is chosen by the algorithm; the other links are omitted.

In addition to the topology and capacity planning input, a *survivability* concept is given, which are in our case no survivability, link or path restoration. These routing conditions are described in our model using a path flow formulation. With link or path restoration, full restoration of all single link failures is assumed.

The problem is solved with DISCNET [12], which uses a branch-and-cut framework based on an LP relaxation containing link design variables but no routing variables. The path flow formulation of the routing constraints is used to test whether given link capacities allow a feasible routing, and if not, to generate generalizations [4], [10] of metric inequalities [9] which are added to the LP relaxation to cut off the infeasible solution. To solve the path flow formulation, we use column generation for working and restoration paths. A more detailed description of this approach, further employed cutting planes, and the methods used to identify missing paths can be found in [4], [10].

III. Computational Results

In this section, we describe our computational tests to evaluate the effect of two kinds of hop limits for working paths on the overall network cost. After a short presentation of our 10 real-world based instances stemming from SDH-, WDM-, and leased line planning problems, we show and discuss our results.

We present two test series, each for no survivability, link or path restoration. In the first series, a fixed global hop limit between 3 and 7 is imposed on all working paths. In the second series, a demand-dependent hop limit is imposed on the working paths: the length of a routing path is restricted by the number of hops in a shortest hop path for the corresponding demand, plus some fixed number (which is the same for all demands). The latter parameter varies between 1 and 4. In both series, the hop limits are only imposed on working paths; the length of restoration paths is unlimited.

For each test instance, Table I shows the number of nodes, potential links, and demands, respectively, together with the average node degree $\bar{d} = 2|E|/|V|$ and the number of available link designs (#ld), which is the same for all links of an instance.

In all figures presented in this paper, 100 corresponds to the *optimal* network cost when *all* routing paths are allowed, regardless of their length.

TABLE I
CHARACTERISTICS OF THE TEST INSTANCES

Name	$\|V\|$	$\|E\|$	$\|D\|$	$\bar{d}$	#ld
g1	10	25	29	5.0	2
g2	12	18	27	3.0	4
g3	15	21	13	2.8	3
g4	15	22	105	2.9	7
g5	18	21	62	2.3	9
g6	18	27	62	3.0	9
g7	20	28	119	2.8	6
g8	14	21	91	3.0	5
g9	24	27	72	2.3	7
g10	24	30	101	2.5	12

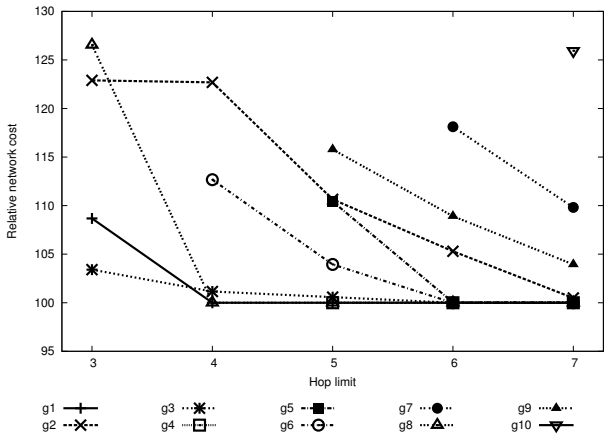

Fig. 1. No survivability, fixed hop limit

A. Fixed hop limit

Figure 1 shows the optimal network cost for each of our instances when a hop limit between 3 and 7 is imposed on all paths. As can be seen, three instances (g1, g3, and g8) have near-optimal solutions even with a global hop limit of 4, while other instances (g7, g9, and g10) need some very long paths (with at least 6, 5 and 7 hops, respectively) to allow a solution at all. The figure shows that the cost of instance g10 is about 25% higher with hop limit 7 than with all possible routing paths allowed.

In a similar way, Figures 2 and 3 show the corresponding results for link and path restoration, respectively.

Comparing Figure 1 with Figures 2 and 3, it can be seen that the cost effect of hop limits is much greater without survivability than with restoration. This is probably due to the fact that only working paths are length-bounded in our calculations, but not the restoration paths.

However, considering all three figures, there are some instances (like g1 and g3) for which an optimal solution is always obtained with relatively short paths, while other instances (like g7, g9 and g10) always need very long paths to achieve a low network cost or to allow a feasible solution at all. Altogether, it is not clear which properties of the network are responsible for the corresponding behavior; neither network density, nor capacity granularity (number of link designs), nor

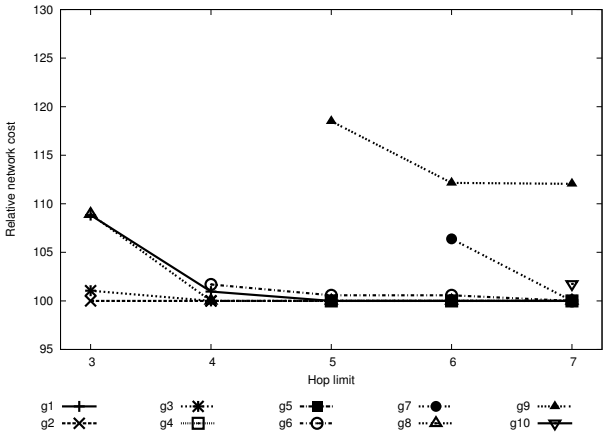

Fig. 2. Link restoration, fixed hop limit

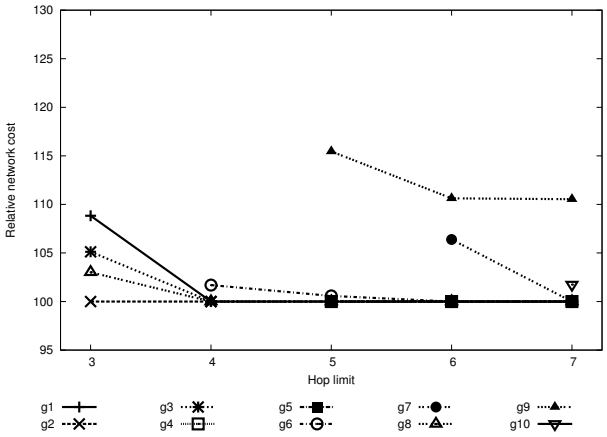

Fig. 3. Path restoration, fixed hop limit

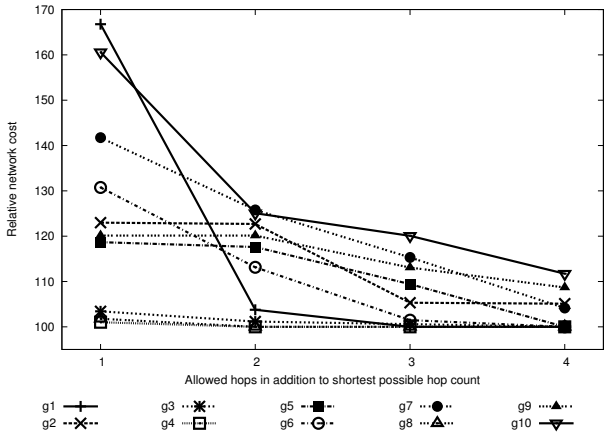

Fig. 4. No survivability, demand-dependent hop limit

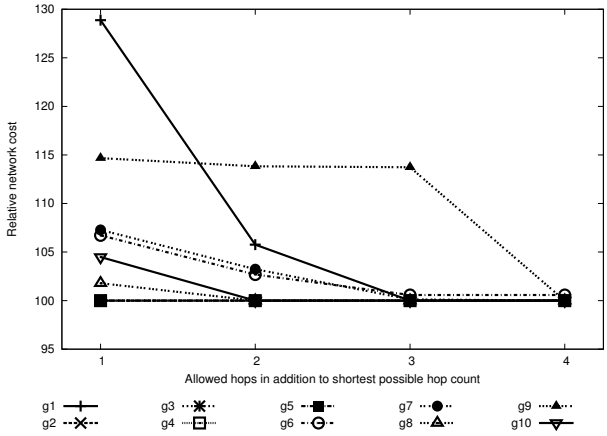

Fig. 5. Link restoration, demand-dependent hop limit

the number of demands seem to be good indicators, at least for our test instances.

The minimal path length which is needed to allow a solution at all can significantly change with different survivability requirements: instance g2 needs paths of length 7 to obtain an optimal solution when no survivability is required, while with link or path restoration, working paths of length 3 are sufficient.

In summary, the quantitative behavior of total network cost as a function of the hop limit is quite unpredictable. For obtaining near-optimal solutions with high probability, one would have to enumerate all paths up to at least 8 or 9 links. Especially in dense networks, this soon leads to a very large path set which is no longer manageable.

B. Demand-dependent hop limit

As an alternative, we now set the hop limit for each demand individually, as a function of the length of a shortest hop path for this demand. In addition to this shortest hop length, an admissible path is allowed to employ k further links, where k has the same value for all demands. Figures 4, 5, and 6 show the results with $k = 1, \ldots, 4$, for the same survivability settings as above (note the different scale on the cost axis in Figure 4). Again, 100 corresponds to the *optimal* network cost when *all* routing paths are allowed.

Figures 5 and 6 show that with link or path restoration, 4 hops in addition to the shortest path length are sufficient to achieve optimality in all of our test instances, and 3 additional hops are sufficient in all instances but one.

On the contrary, this approach does not yield equally good results without survivability, as indicated by Figure 4. With 4 additional hops, the cost for two out of the ten instances is at least 10% above the optimal solution, and for four instances, the difference exceeds 5%. With only 3 additional hops, the situation is even much worse.

Investigating the solutions obtained without survivability more closely, we have noticed that

- demands with a high demand value are most often routed on short paths,
- wherever the demand-dependent hop limits have a strong effect, see instances g7, g9, and g10, about 10–20% of all demands are routed on relatively long paths, i.e., with 8 links or more. For other instances, very few demands are

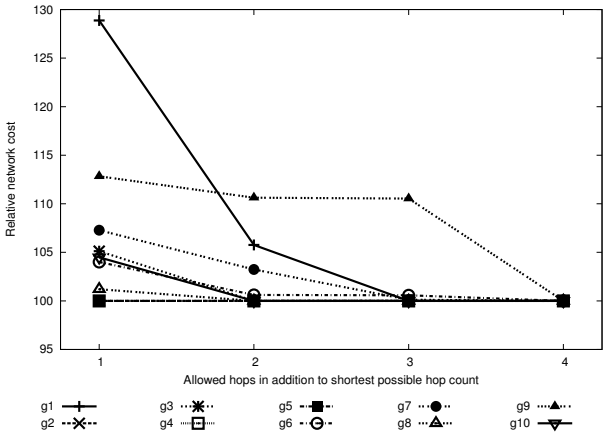

Fig. 6. Path restoration, demand-dependent hop limit

routed on long paths. Unfortunately, whether an instance needs long paths for many demands or not can only be seen after calculating an optimal solution.

Summarizing our results, we have not found any suitable criterion which could tell us in advance whether hop limits would have a major impact on the overall network cost for a given instance. The results with a predefined path set are more or less unpredictable, which strongly encourages the use of column generation techniques. However, if predefined paths are used, the choice of paths should be done on a per-demand basis and not by setting a global hop limit. A good compromise for restoration is the use of a predefined path set with demand specific hop limits for the working paths, combined with column generation for the restoration paths, as described above. This holds for link restoration as well as for path restoration; as can be seen from the figures, the behavior of the problem instances is nearly the same for both restoration concepts.

In our calculations for this paper, we simulated a hop-limited path set by using a hop-limited variant [4] of Dijkstra's shortest path algorithm for identifying missing routing paths during the column generation procedure. The calculation times often raised monotonically with the hop limit but rather in a linear way than exponentially as it is the case when all paths up to a given length are enumerated. This property makes the column generation approach well scalable. Even without any hop limit, most of the instances could be solved to optimality within the range of seconds or minutes on a Linux machine with 1 GB of RAM and a 1.7 MHz processor.

IV. Conclusion

In this paper, we have studied the impact of varying hop limits on the overall network cost in the planning of a telecommunication network. Based on our mixed-integer linear programming model, we have solved 10 problem instances to optimality using a branch-and-cut framework and column generation for working and restoration paths. For each of these instances, we have considered three different admissible path sets and three survivability concepts (no survivability, link and path restoration).

Based on these provably optimal solutions, we have compared the overall network cost without restrictions on the admissible paths to the network cost with hop-limited paths, either with demand-dependent or with demand-independent hop limits.

Concerning the two questions from the introduction, we found out that hop limits may sometimes lead to good solutions and relatively short calculation times, but may as well make a problem infeasible or cause the total network cost to increase by an unpredictable amount. With a fixed hop limit of 7 hops (for which it is hard to enumerate all paths), the total network cost has been up to 25% higher than without hop limits. With a demand-dependent hop limit (shortest hop count plus some additional fixed number of hops), the situation is a little better, but the increase in network cost can still be about 15% with 4 additional hops.

Thus, whenever the used technology allows it, one should use either column generation or, as the second best choice, demand-dependent hop limits, or a combination of both, but no globally fixed hop limit for all demands.

References

[1] K. Murakami, *Survivable Network Management for High-Speed ATM Networks*. PhD thesis, Carnegie Mellon University, September 1995.

[2] G. Dahl and M. Stoer, "A cutting plane algorithm for multicommodity survivable network design problems," *INFORMS Journal of Computing*, vol. 10, no. 1, pp. 1–11, 1998.

[3] F. Poppe and P. Demeester, "Economic allocation of spare capacity in mesh-restorable networks: Models and algorithms," in *Proceedings of the 6th International Conference on Telecommunication Systems (ICTSM 1998), Modeling and Analysis*, Nashville, USA, pp. 77–86, March 1998.

[4] R. Wessäly, *Dimensioning Survivable Capacitated NETworks*. PhD thesis, Technische Universität Berlin, April 2000.

[5] M. Herzberg and S. Bye, "An optimal spare-capacity assignment model for survivable networks with hop limits," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM'94)*, pp. 1601–1606, September 1994.

[6] Y. Xiong and L. Mason, "Restoration strategies and spare capacity requirements in self-healing ATM networks," in *Proceedings of the 16th IEEE Infocom 1997*, Kobe, Japan, pp. 353–360, 1997.

[7] R. Iraschko, M. MacGregor, and W. Grover, "Optimal capacity placement for path restoration in stm or atm mesh survivable networks," *IEEE Transactions on Networking*, vol. 6, pp. 325–336, June 1998.

[8] J. Doucette and W. Grover, "Comparison of mesh protection and restoration schemes and the dependency on graph connectivity," in *Proceedings of the Third International Workshop on the Design of Reliable Communication Networks (DRCN 2001)*, Budapest, Hungary, pp. 121–128, October 2001.

[9] M. Iri, "On an extension of the maximum-flow minimum-cut theorem to multicommodity flows," *Journal of the Operations Research Society of Japan*, vol. 13, no. 3, pp. 129–135, 1971.

[10] S. Orlowski, "Local and global restoration of node and link failures in telecommunication networks," diploma thesis, Technische Universität Berlin, February 2003.

[11] S. Orlowski and R. Wessäly, "Comparing restoration concepts using optimal network configurations with integrated hardware and routing decisions," in *Proceedings of the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN 2003)*, Banff, Alberta, Canada, October 2003. to appear, available as ZIB Report SC-03-09 at `http://www.zib.de`.

[12] atesio GmbH, Rubensstr. 126, D-12157 Berlin, Germany, DISCNET, 2000–2003. Information available at `http://www.atesio.de`.