Analysis of Link Reversal Routing Algorithms for Mobile Ad Hoc Networks

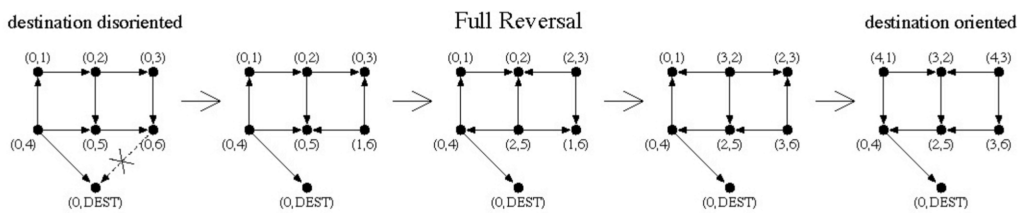
Nicolas Born
nborn@student.ethz.ch

Seminar of Distributed Systems WS 04/05

Report

1. Summary

Link reversal routing algorithms were introduced by Gafni and Bertsekas in [1]. If you have converted an ad hoc network to a destination oriented graph, the full reversal algorithm is pretty simple. It works by using one simple rule. Whenever a node becomes a sink, it reverses all of its incoming links. An execution is illustrated in the figure below.



Paper [2] by Busch, Surapaneni and Tirthapura (the one I read) proves lower and upper bounds for deterministic link reversal routing algorithms, especially for the full and partial reversal routing algorithms. They derive the following results:

- The full reversal algorithm requires $O(n^2)$ work and time (n: number of nodes, which have lost routes to the destination).
- The partial reversal algorithm requires $O(n \cdot h^* + n^2)$ work and time (h^*: nonnegative integer).
- For every deterministic link reversal algorithm, there are initial states which require $\Omega(n^2)$.

2. Discussion

In my opinion, the most surprising result is that the full reversal algorithm is optimal in the worst case while the partial reversal algorithm is not (since h^* may grow arbitrarily large). So we can say that the full reversal algorithm outperforms the partial reversal algorithm in the worst case. But we expect that the latter needs less work in the average case. It would be interesting to prove this. And have variations of the partial reversal algorithm (as it is stated in [1]) which are better in the worst case been discussed?

The good news is that the time and work until stabilization only depends on the number of bad nodes. The bad news: there exists an inherent lower bound on efficiency of link reversal algorithms.

Now that we know the bounds for link reversal routing algorithms, we are able to compare them to other known routing algorithms. There are many other algorithms which are more efficient than link reversal routing algorithms. For instance, compared to algorithms based on shortest path that store the informations in tables, link reversal routing algorithms' only advantage is that they don't need to update at every link failure. If a link gets lost for some reason (maybe the destination moved out the range of some node), there might be still another directed path to the destination from every node and so the graph could still be destination oriented. Algorithms based on shortest path need to update in this scenario too.

There are practical non-deterministic algorithms that are based on deterministic link reversal algorithms (like TORA [3]). Are there any known upper bounds for these algorithms? Are they better in the worst-case?

The model they use in [2] is somehow limited since they always consider connected graphs only, because our algorithms wouldn't work in bipartite graphs anymore. This isn't very realistic because no node failures are tolerated in this case. For the sake of simplicity, they assume that there only exists one destination. They write that if there are several destinations, one has to create a destination oriented graph for each destination. Unfortunately, they don't prove this in their paper.

The original paper was written over 20 years ago, in 1981. I was surprised that link reversal routing algorithms have the same age as I do. I don't know why it took so long to analyze these algorithms. But now we know that there are better algorithms out there for sure. Although, I guess that link reversal routing algorithms are still useful as basis for more sophisticated algorithms.

3. References

[1] E.M. Gafni and D.P. Bertsekas. Distributed algorithms for generating loop-free routes in networks with frequently changing topology. IEEE trans. On commun., COMM-29:11-18, 1981.

[2] C. Busch, Srikanth Surapaneni and Srikanta Tirthapura. Analysis of link reversal routing algorithms for mobile ad hoc networks. SPAA ,03.

[3] V. D. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In IEEE Infocom '97 - 16th Conference on Computer Communications. IEEE, 1997.