Improving Bounds on Network Reliability: Some Examples

Timothy B. Brecht

Department of Computational Science
University of Saskatchewan
Saskatoon, Saskatchewan, S7N 0W0
CANADA

Charles J. Colbourn

Computer Communications Networks Group
Department of Computer Science
University of Waterloo
Waterloo, Ontario, N2L 3G1
CANADA

ABSTRACT

A popular measure of the reliability of a computer network in an environment of statistically independent link failures is the probability that the network remains connected. Computing the probability that k specified nodes of a network are connected (k-terminal reliability) is a well known computationally difficult problem. The probability that two nodes are connected (2-terminal reliability) and the probability that all nodes are connected (all-terminal reliability) are two widely studied cases of k-terminal reliability. Each of these problems is NP-hard, and thus efficiently computable reliability bounds are of significant interest. A variety of methods are known for efficiently computing 2-terminal and all-terminal bounds; however, few results apply to the k-terminal problem. We develop a simple strategy to obtain bounds on k-terminal reliability and demonstrate improvements on the previous best bounds for 2-terminal, k-terminal and all-terminal reliability.

1. Introduction

A communications network is often modelled using a *probabilistic graph* G, containing the set V of n nodes, representing communication sites in the network and the set E of e edges, representing the communication links between the sites. This model assumes that the nodes of a network are perfectly reliable (that is, they do not fail) and that the failure of links is random and therefore statistically independent. A common measure of the reliability of a network is the probability that it remains operational subject to the random failure of its links.

A network is considered to be *operational* if all of the specified nodes of the network are connected. The probability that k specified nodes of the network are connected is termed *k-terminal reliability*. Two cases of k-terminal reliability that have been given much attention are *2-terminal reliability* (the probability that two specified nodes of the network are connected) and *all-terminal reliability* (the probability that all nodes of the network are connected).

The computation of exact values for each of these reliability problems has been shown to be #P-complete [1], [2]. Therefore the computation of upper and lower bounds on the actual reliability of the network are of significant interest. Since computations of exact values of reliability are likely to require an exponential amount of time for general networks, we study bounding methods that can be computed efficiently (in polynomial time).

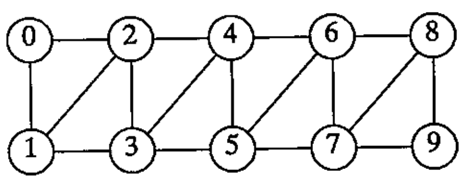
2. The Bounding Strategy

The approach we use to obtain improved reliability bounds is to use any method for computing 2-terminal bounds to compute an upper bound $U(x,y)$ and a lower bound $L(x,y)$, for all pairs of nodes (x,y) in the network. This yields a completely connected network in which each edge $e = (x,y)$ has a success probability $P(x,y) = P_e$ between $L(x,y) = L_e$ and $U(x,y) = U_e$.

For example, in **Table 1** we show the values $1 - L(x,y)$, computed for the 10 node ladder (shown in **Figure 1)** using the Brecht-Colbourn [3] edge disjoint path method for computing lower bounds. The table was computed by assuming that each individual edge e, in the network is operational with probability $p_e = 0.90$. This table will be referred to when discussing examples of how each of the 2-terminal, all-terminal and k-terminal bounds are computed.

$1 - L(x,y)$ for 10 node ladder $p = 0.90$.										
	0	1	2	3	4	5	6	7	8	9
0	1.0	0.019	0.019	0.023	0.026	0.029	0.032	0.036	0.039	0.055
1	0.019	1.0	0.004	0.007	0.010	0.013	0.017	0.020	0.023	0.039
2	0.019	0.004	1.0	0.004	0.007	0.010	0.013	0.017	0.020	0.036
3	0.023	0.007	0.004	1.0	0.004	0.007	0.010	0.013	0.017	0.032
4	0.026	0.010	0.007	0.004	1.0	0.004	0.007	0.010	0.013	0.029
5	0.029	0.013	0.010	0.007	0.004	1.0	0.004	0.007	0.010	0.026
6	0.032	0.017	0.013	0.010	0.007	0.004	1.0	0.004	0.007	0.023
7	0.036	0.020	0.017	0.013	0.010	0.007	0.004	1.0	0.004	0.019
8	0.039	0.023	0.020	0.017	0.013	0.010	0.007	0.004	1.0	0.019
9	0.055	0.039	0.036	0.032	0.029	0.026	0.023	0.019	0.019	1.0

Table 1



Note:

$s = 0, t = 9$
$K = \{0,1,8,9\}$

Figure 1

These newly computed probabilities are not statistically independent, because edges used to compute the bounds for one pair of nodes i,j can be reused to compute a lower bounds for another pair of nodes k,l. Furthermore, no information is known about the dependencies and none can be assumed. Therefore, many of the popular techniques for computing bounds do not apply (since they assume that edge failures are statistically independent).

However, Zemel [4] and Assous [5] employ a linear programming formulation due to Hailperin [6] to produce the following bounds which can be applied to such a model.

A best possible upper bound is given by:

$$\beta = \min \left[1, \min_{C \in K^*} \sum_{e \in C} U_e \right]$$

where K^* is the set of all minimal cutsets of G.

A best possible lower bound is given by:

$$\alpha = \max \left[0, 1 - \min_{S \in P^*} \sum_{e \in S} (1 - L_e) \right]$$

where P^* is the set of all minimal pathsets of G.

Using these equations, an upper bound may be obtained by finding a minimum weight cutset using the U_e values as weights and a lower bound may be obtained by finding a minimum weight pathset using the values $(1 - L_e)$ as weights.

These bounds will typically produce improved upper bounds for lower values of p_e, and improved lower bounds for higher values of p_e. The examples outlined here demonstrate how the existing lower bounds can be improved for higher values of p_e. This is especially useful in light of studies [7] which have determined the typical reliability of a link in the ARPANET to be 0.98.

3. Computing 2-Terminal Bounds

In computing 2-terminal reliability we compute the probability that two specified nodes in the network can communicate. The specified nodes are often referred to as the *source* node s and the *target* node t. In this case a minimal pathset is a minimum weight s,t-path and a minimal cutset is a minimum weight s,t-cut. Both problems have efficient solutions [8] and hence an upper and lower bound can be computed efficiently.

To compute the new lower bound for the 10 node ladder we simply determine a minimum weight s,t-path using the edge weights shown in Table 1. This computation yields the edges (0,2), (2,3), (3,5), (5,7) and (7,9). The weights on the edges are summed to give the value 0.054678 and the resulting bound is $max(0,1 - 0.054678) = 0.945322$.

In Table 2 we compare this new lower bound to the Kruskal-Katona subgraph counting method developed by Van Slyke and Frank [9], using a theorem of Kruskal [10] and Katona [11]. The bound is also compared with the Brecht-Colbourn edge disjoint path method [3] and the exact measure of the graph's reliability. The exact measure, as well as all of the other exact measures shown in later sections, are computed using a linear time algorithm developed by Wald and Colbourn [12] for computing k-terminal reliability of partial 2-trees.

2-Terminal Bounds (10 node ladder)				
p	KK	B-C	2t->2t	exact
0.75	0.324913	0.418296	0.583862	0.789632
0.85	0.564144	0.690536	0.865406	0.933385
0.90	0.727901	0.832302	0.945322	0.973290
0.92	0.798671	0.883775	0.966567	0.983718
0.94	0.868407	0.929193	0.982157	0.991315
0.96	0.931706	0.965913	0.992537	0.996356
0.97	0.958415	0.980044	0.995942	0.998012
0.98	0.979970	0.990769	0.998262	0.999144
0.99	0.994567	0.997598	0.999582	0.999793

Table 2

The table shows that the technique of using 2-terminal bounds to obtain new bounds can be used effectively to compute bounds which can be substantially better than the previous best bounds. One possible explanation for the improvement obtained in this manner, is that it exploits the local structure of the graph to improve the overall bound.

4. Computing All-Terminal Bounds

A bound on all-terminal reliability is a bound on the probability that all nodes in the network can communicate. In this case a minimal pathset is a minimum weight spanning tree and a minimal cutset is a minimum weight network cut. Both of these problems can be solved efficiently [8] and hence the upper and lower bound can be computed efficiently.

The application of a minimum weight spanning tree algorithm using the weights in Table 1 produces the following set of edges: (1,2), (3,2), (4,3), (5,4), (6,5), (7,6), (8,7), (0,1) and (9,7). The weight of the minimum spanning tree is 0.063270 to produce an all-terminal bound of $max(0,1 - 0.063270) = 0.936730$.

The lower bound result obtained in this fashion is compared in Table 3 with the Ball-Provan [13] method for computing all-terminal bounds and the exact value of the the graphs all-terminal reliability.

All-Terminal Bounds (10 node ladder)			
p	BP	2t->all	exact
0.75	0.576232	0.446289	0.769946
0.85	0.831994	0.835893	0.927912
0.90	0.930207	0.936730	0.971522
0.92	0.958374	0.962212	0.982786
0.94	0.979065	0.980341	0.990911
0.96	0.992209	0.992007	0.996234
0.97	0.996144	0.995721	0.997960
0.98	0.998553	0.998196	0.999129
0.99	0.999714	0.999574	0.999791

Table 3

The improvements gained in the all-terminal case are of particular interest, as this is the first time (to our knowledge) that 2-terminal bounds have been applied to all-terminal bounds. The improvements were also quite unexpected in the all-terminal case since most of the work done on efficiently computable reliability bounds has concentrated on all-terminal bounds.

5. Computing K-Terminal Bounds

When computing k-terminal reliability, k nodes are specified as target nodes. These nodes are required to be able to communicate with each other in order for the network to be considered operational. In this case a minimal pathset is a minimum weight Steiner tree and a minimal cutset is a minimum weight Steiner cut. The problem of computing a minimum weight Steiner cut can be solved efficiently using network flows [8]. Therefore an upper bound can be computed efficiently. Unfortunately, the problem of computing a minimum weight Steiner tree is NP-Hard [14]. However, a heuristic method for computing an approximation to a minimum weight Steiner tree may be used to obtain a lower bound.

Let w be the weight of a minimum weight Steiner tree. The k-terminal reliability of the graph is $\geq max(0, 1-w)$. Let w' be the weight of any Steiner tree (not necessarily minimum weight). Since $w' \geq w$, $max(0, 1-w') \leq max(0, 1-w)$. Therefore $max(0, 1-w')$ is a lower bound on k-terminal reliability.

The accuracy of the lower bound depends upon the accuracy with which the Steiner tree can be computed. We take two approaches here. The first, chosen for its simplicity, involves finding a minimum spanning tree on the k target nodes to form a Steiner tree. The second approach is based on the dual ascent approach due to Wong [15], with some modifications to take into account the triangle inequalities which are formed as a result of computing the best 2-terminal bound between all pairs of nodes in the graph.

In the example graph (10 node ladder) we use the four extreme (corner) nodes of the graph as target nodes. The minimum weight spanning tree on the target nodes uses the edges (0,1), (8,1) and (9,8) for a total weight of 0.061212 to give the bound $max(0,1 - 0.061212) = 0.938788$. Wong's method uses the edges (0,1), (1,7), (7,8) and (7,9) to form a Steiner tree with weight 0.061212, for a bound of $max(0,1 - 0.061212) = 0.938788$.

It is difficult to compare the results obtained for k-terminal lower bounds as no powerful results are known. Therefore we compare our results against all-terminal bounds, since an all-terminal lower bound is also a k-terminal lower bound. The exact value is also included for comparison. (Note that the all-terminal bounds could also be used and compared with the 2-terminal bounds.)

K-Terminal Bounds (10 node ladder)					
p	BP	2t->all	spanning	Wong's	exact
0.75	0.576232	0.446289	0.509094	0.447571	0.769946
0.85	0.831994	0.835893	0.845510	0.815436	0.927912
0.90	0.930207	0.936730	0.938788	0.938788	0.971522
0.92	0.958374	0.962212	0.963082	0.963082	0.982786
0.94	0.979065	0.980341	0.980625	0.980625	0.990911
0.96	0.992209	0.992007	0.992065	0.992065	0.996234
0.97	0.996144	0.995721	0.995739	0.995739	0.997960
0.98	0.998553	0.998196	0.998200	0.998200	0.999129
0.99	0.999714	0.999574	0.999575	0.999551	0.999791

Table 4

The results in Table 4 demonstrate that this method of computing k-terminal lower bounds can produce bounds that are quite good. The table also shows that although quite simple, the method of heuristically computing a Steiner tree by determining a minimum spanning tree on the target nodes can in some instances produce a better bound than the more sophisticated Wong's algorithm. However, in most of our test cases, Wong's algorithm produces better results.

6. Conclusions

One of the main contributions of this method for computing bounds on network reliability is to bridge the gap between the 2-terminal, all-terminal and k-terminal problems. We have shown that this method can be used to improve upon existing bounds especially if they were combined with other existing bounds using the linear programming technique described by Colbourn and Harms [16]. For more details on how to further improve these bounds see [17]. As well, this technique for computing new bounds from existing bounds may prove useful as a means of computing k-terminal bounds (for which no other bounds are known).

Acknowledgements

We thank Steve Lui for providing us with an implementation of Wong's algorithm and Daryl Harms for supplying us with an implementation of the Ball-Provan all-terminal bounds as well as giving us a base implementation upon which to build the Kruskal-Katona 2-terminal bounds. We also thank Nancy

Ross for help with bounds with statistical dependence. Research of the second author is supported by NSERC Canada under grant A0579.

References

[1] Valiant, L.G., "The Complexity of Enumeration and Reliability Problems", *SIAM Journal on Computing*, *Vol.* 8, pp. 410 - 421, 1979.

[2] Provan, J.S. and M.O. Ball, "The Complexity of Counting Cuts and of Computing the Probability that a Graph is Connected", *SIAM Journal on Computing*, *Vol.* 12, *No.* 4, pp. 777-788, November, 1983.

[3] Brecht, T.B. and C.J. Colbourn, "Lower Bounds on Two-Terminal Network Reliability", CCNG Report E-127, Computer Communications Network Group, University of Waterloo, Waterloo, Ontario, March, 1985.

[4] Zemel, E., "Polynomial Algorithms for Estimating Network Reliability", *Networks*, *Vol.* 12, pp. 439 - 452, 1982.

[5] Assous, J.Y., "Bounds for Terminal Reliability", Temple University, Philadelphia, Pennsylvania, preprint, 1984.

[6] Hailperin, T., "Best Possible Inequalities for the Probability of a Logical Function of Events", *American Mathematics Monthly*, *Vol.* 72, pp. 343-359, 1965.

[7] Frank, H. and W. Chou, "Network Properties of the ARPA Computer Network", *Networks*, *Vol.* 4, pp. 213 - 239, 1974.

[8] Lawler, E., **Combinatorial Optimization: Networks and Matroids**, Holt, Rinehart and Winston, San Francisco, California, 1976.

[9] Van Slyke, R. and H. Frank, "Network Reliability Analysis: Part I", *Networks*, *Vol.* 1, pp. 279-290, 1972.

[10] Kruskal, J.B., "The Number of Simplices in a Complex", **Mathematical Optimization Techniques**, ed. R. Bellman, University of California Press, Berkeley, California, pp. 251-278, 1963.

[11] Katona, G., "A Theorem on Finite Sets", **Theory of Graphs (Proceedings of Tihany Colloquium, September, 1966)**, ed. P. Erdös and G. Katona, Academic Press, New York, New York, pp. 187-207, 1966.

[12] Wald, J.A. and C.J. Colbourn, "Steiner Trees in Probabilistic Networks", *Microelectronics and Reliability*, *Vol.* 23, *No.* 5, pp. 837 - 840, 1983.

[13] Ball, M.O. and J.S. Provan, "Bounds on the Reliability Polynomial for Shellable Independence Systems", *SIAM Journal on Algebraic and Discrete Methods*, *Vol.* 3, pp. 166-181, 1982.

[14] Garey, M.R. and D.S. Johnson, **Computers and Intractability**, Freeman Press, 1979.

[15] Wong, R.T., "A Dual Ascent Approach for Steiner Tree Problems on a Directed Graph", *Mathematical Programming*, *Vol.* 28, pp. 271-287, 1984.

[16] Colbourn, C.J. and D.D. Harms, "Bounding All-Terminal Reliability in Computer Networks", Technical Report E-123, Computer Communications Network Group, University of Waterloo, Waterloo, Ontario, 1985.

[17] Brecht, T.B. and C.J. Colbourn, *Improving Reliability Bounds in Computer Networks*, (submitted for publication).