A METHOD FOR MAXIMIZING COMMUNICATIONS NETWORK RELIABILITY

by

S. Singhal and G.L. Thompson

DRO-70-26-84

December, 1984

(W. P. No. 3-84-85}

(Management Sciences Research Report No. 509)

A Method for Maximizing Communications Network Reliability

21 September 1984

Sharad Singhcti

and

Gerald L Thompson

This report **was prepared as part of** the activities of the Management Science Research Group, Carnegie-Mellon University, under Contract No. N00014-82-K-0329 NR 047-048 with the U.S. Office of Naval Research. Reproduction in whole or part is permitted for any purpose of the U.S. Government.

Management Science Research Group
Graduate School of Industrial Administration
Carnegie-Mellon University
Pittsburgh, Pa 15213

Abstract

The network reliability problem has been well covered in the literature. The idea of reliability found in most literature consists of defining a numerical parameter called *overall network reliability* and then suggesting methods for computing that parameter for a given network. We take this analysis one step further by providing an algorithm for *determining* a network which will have near maximum reliability under certain given constraints.

Given n terminals and one central computer, the problem is to construct a network which links each terminal to the central computer subject to the following conditions : (1) each link must be economically feasible ; (2) the minimum number of links should be used ; and (3) the reliability should be maximized. We argue that the network satisfying condition (2) is a spanning arborescence of the network defined by condition (1). A definition for the reliability of an arborescence is given. Since the problem of maximizing reliability is NP complete, a heuristic algorithm is developed which provides a good solution for the arborescence having maximum reliability. Computational experience for networks consisting of up to 900 terminals is given.

1. Introduction

The problem of maximizing the reliability of a communications network has widespread application especially due to the advent of computer communications networks. For example, we might be interested in knowing the probability of CMU being able to communicate with Harvard on the TELENET network, and also the *overall reliability* of the TELENET network. We refer to each city on the network as a *node.* As mentioned in [26], conceptually the task of determining the overall reliability is a simple one. All one must do is

1. Consider each of the possible states of the system (a state is defined by listing the successful and failed components).

2. Identify which of the states result in successful system operation (i.e every node in the network is able to communicate with every other node).

3. Add together the probability of occurrence of each successful state.

The final sum is the overall system reliability i.e. the probability of the system being in successful state. Clearly, this approach is computationally infeasible for large systems because an n node communication network will have 2^n states. In current literature, several manageable definitions for overall reliability have been proposed [11, 18]. A comprehensive survey of network reliability problems is given in [8]. Most network reliability problems are. in the worst case, NP-hard [12].

Network reliability problems are more difficult than many standard combinatorial optimization problems because the correctness of a solution to a reliability problem cannot be verified in polynomial time. However, there are in fact linear and polynomial time algorithms available for network reliability problems having special structure.

A probabilistic network consists of a set of nodes and links that fail with some known joint probability distribution. In addition to a communications network, other practical examples of a probabilistic network are electric power systems, water aqueducts and transportation systems. For a probabilistic graph G, reliability models have been classified as *Rooted Problems* or *Unrooted Problems* by Satyanarayna as follows [41].

Rooted Problems

- *Source to Terminal Reliability:* Probability that a specified vertex in G can send communication to another specified vertex. This problem is studied in [9].

- *Source to all terminal (SAT) Reliability:* Probability that a specified vertex in G can send communication to all other vertices.

- *Source to K-terminal (SKT) reliability:* Probability that a specified vertex in G can send communication to a set of K specified vertices.

Unrooted Problems

- *2-terminai reliability:* Probability that a specified *vertex* pair in G can communicate.

- *Overall Reliability:* Probability that all vertex pairs in G can communicate.

- *K-terminal reliability:* Probability that among a set of K specified vertices in G, all vertex pairs can communicate.

The rooted problems are meaningful only for directed graphs. For an undirected network, each link can communicate both ways, and the three rooted problems are equivalent to each of the three corresponding unrooted problems.

The most widely studied problem in literature is the source to terminal reliability for which a review of known methods is given in [30, 39]. The problem that is the focus of our attention in this chapter is the source to all terminal reliability which is the same as the overall reliability for an undirected network. This problem was solved in [40] using the concept of a *t-graph* which is a fundamentally different approach from ours. We assume that the final communications network will be *an* arborescence and try to find the specific arborescence which will have the maximum reliability. In [40] it is assumed that the communications network has already been designed and the objective is only to calculate the overall reliability by decomposing the network into rooted trees. Combinatorial properties of various definitions of reliability are discussed in [21, 22, 23]. The other relevant references for the reliability problem are [31, 34, 37]. An exhaustive bibliography of the literature published on complex system reliability (i.e. power network, communications network etc.) evaluation is given in [24, 44].

Note that all terminals of a network G can communicate with each other if and only if there is at least one spanning tree of G with all its branches operative. This notion is used in [6] to compute the reliability of a network. One other method which deserves special mention due to its simplicity is given in [10]. It defines system success as the case when all terminals of G are able to communicate with each other and gives an algorithm for finding mutually disjoint success branches. The reliability expression of each branch can be written directly using a set of rules. Then the reliability of the whole network is the sum of the reliability of each success branch. Other papers [1, 2, 3, 5, 17, 32] interpret network reliability differently and provide methods for evaluating a different kind of reliability.

Several approaches for evaluating the reliability of a network have been proposed. Methods relying on the Markov approach [13, 15} suffer from having too many system states. The network approach [16] requires the explicit manual derivation of several complex formulae and hence is not suitable for computer application. There are some algorithms based upon systematic network reduction [19] which apply star delta type of transformations to the network thereby making the network less complicated. To reduce the size of a communications network, it is also possible to define conditions under which a link from a network can be removed without affecting its overall reliability [38]. Some reasonably fast methods based on fault tree analysis also exist [29].

C^st benefit analysis is another important facet of the problems in network reliability. The effect of redundancy in a communications network on maintenance and outage costs is discussed in [33]. Some authors [25] prefer to use the more global term *quality measure* of a network and characterize the quality of a network by its reliability, maintenance costs, safety and other related attributes.

It is also very important to consider capacity constrained reliability problems. Network Flow type of problems such as communications system having fixed channel capacities of its links are considered in [7]. If some multiprocessors can connect only a specified number of terminals then we get the degree constrained problem which is considered in this paper.

As mentioned earlier, the general reliability problem is NP complete. For this reason, heuristics play a very important role in the reliability literature. The literature has many papers discussing the philosophy of reliability heuristics [28], comparisons of heuristics [36] based on speed and accuracy, notion of local optimality [27], and details of heuristic algorithms [35, 4].

2. Review of some Concepts and Definitions

Unless otherwise specified, by a graph we mean a connected undirected graph. These definitions are the same as those given in [43] and have been reproduced here for the sake of completeness. The proofs for all lemmas and theorems in this section are given in [42].

Definition 1: A connected graph which has no cycles is called a *tree.*

Definition 2: A *rooted tree* is a *tree* in which an arbitrary (but fixed) node is given the name *root node.* If each edge of a rooted tree is replaced by a directed arc pointing towards the root node, it is called an *arborescence.*

Note that there is a unique arborescence corresponding to any rooted tree.

Definition 3: For any graph G = (V,A), the rooted tree T = (V^^ is called a *spanning rooted tree* of G if A_T is a subset of A. When each edge of the spanning rooted tree is replaced by an arc directed towards the root node, we get a *spanning arborescence* of G.

Definition 4: For a given spanning arborescence of G = (V, A), if i, j € V and (i,j) G A_T, then the *predecessor node of i* is said to be j or symbolically P('> = j-

One can interpret predecessor of i to be like the "father" of i in the sense of a family tree.

Definition 5: Nodes i and j are called the *end nodes* of arc (i,j).

Definition 6: A node i 6 V is said to be a *junction node* of a rooted tree T = (VrA^J if it is the end node of at least three arcs belonging to A_T.

Definition 7: A node i 6 V is said to be a *beginning node* if it is the end node of exactly one arc € A_T,

Definition & Let i and j be two nodes of an arborescence T = (V, A_T). If there is no directed path in T from node i to j or from node j to i, then the arcs (i,j) and (j,i) are called *cross arcs* with respect to T. If there is a directed path from i to j and (i,j) p A $_T$ then (i,j) is called an *up arc.*

Deflnition 9: For each i € V we define the *successor set* V_i = {h : p(h)-- ih Thus Y_i is the set of nodes which are immediately below i, i.e. the set of nodes for which i is the predecessor node.

Definition 10: The *successor function* of i represented by s(i), for i G V is defined inductively as follows :

$$s(i) = 1 + \sum_{h \,€\, ¥} s(h)$$

In other words, s(i) « 1 + number of nodes in T below i.

Definition 11: The *ramification index* or Ram of a spanning arborescence of a communications network is defined as

$$Ram = \frac{n<n+1)}{2} - \overset{n}{\underset{j.i}{TT Z}}— s(j)$$

where n+1 is the root node of the arborescence.

Definition 12: Given an arborescence T = (V,A_T) and an arc (i,j) ∉ A_T, a unique undirected cycle is formed when (i,j) is added to A_T. The node on this cycle having the largest successor function is called the *maxima/ cycle node* created by (i,j).

Theorem 13: For a given incoming arc, the change in ramification index of a **rooted tree depends** on **the** outgoing arc. If the g'th arc between i and the maximal cycle node created by the incoming arc (\.\) is taken out then this change in ramification index is given by :

$$A R(g) * 2 \bullet \sum_{i*1}^{9-1} s(i) - (f - e \bullet 2g - 1) \bullet s(g)$$

where

k — **The** maximal cycle node created by (i,j).

e — = Number of arcs from node i to node k.

f — = Number of arcs from node j to node k.

g — * **The** number of arcs on the directed path in the arborescence starting at arc(i.j) and ending at the outgoing **arc (ends** inclusive).

s(t) — » The successor function of the fth node on the path from node i to node k. t=1 corresponds to node i.

Theorem 14: If an arc (i,j) is to be brought into the solution, then the outgoing arc which gives the maximum decrease in the ramification index has the maximal cycle node as one of its end nodes.

3. Statement of the Computer Network Reliability Problem

We construct the graph of a communications network by carrying out the following steps. Consider each terminal as a node of the graph. Connect two nodes if it is economically feasible to directly link the corresponding terminals. Now add another node representing the central computer and connect it to the nodes of ai! terminals with which it can be directly linked. This gives us a graph which we call a communications network which is formally defined below.

Definition 15: A *Communications Network* is the graph G = {V,A} where

V = Set of all terminals = {1,2,....,n}.

A = Set of all permissible links between pairs of terminals and between the terminal and the central computer.

The reliability of a communications network should be inversely proportional to the number of terminals rendered inoperative due to the breakdown of one or more links. The objective is to find a network consisting of minimum number of permissible links and having maximum reliability in this sense. A connected graph with minimum number of links directed towards the central computer is an arborescence so that we will concentrate on finding an arborescence with maximum reliability.

Note that for an arborescence with root node n+1 corresponding to the central computer, the number of terminals rendered inoperative due to the failure of the link from terminal j to its predecessor node (Definition 4) is simply equal to s(j) (Definition 10). Therefore the number of terminals rendered inoperative due to the breakdown of an "average" link is

$$NOP = \frac{1}{n}\left[\sum_{j=1}^{n} s(j)\right] \qquad (1)$$

Theorem 16: The Ram and NOP of an arborescence are inversely related to each other.

Proof: From Equation (1) and Definition 11 above we can deduce that

$$NOP = \frac{n+1}{2} - \frac{Ram}{n}$$

Hence for a higher ramification index, the value of NOP will be lower so that the reliability would be higher. ■

4. Formal Definition of Network Reliability

Consider a spanning arborescence of the communications network.

Let d(j) = Number of links between terminal j and the central computer.

p = Probability of a particular link being in working condition. It is assumed that all links have the same probability of working and that they have independent probabilities of being in working condition.

P(j) = Probability that terminal j can communicate with the central

computer.

Theorem 17: The probability that terminal j can communicate with the central computer is given by

$$R(j) \bullet p^{*3}. \qquad (2)$$

Proof: The proof follows from the fact that the **probability of each link operating is independent of the others and there are d(j) links on the path from terminal j to the central computer. Since each link must be working in order for the computer and terminal j to communicate. Equation (2) must hold.** •

Definition 1& The reliability Rei of an arborescence having n nodes is defined as the geometric mean of R(j) for each terminal in the arborescence, i.e.

n
TTT 1 1/n

Re. »‡i l p]**

or equivalently,

Hei « **p** VoJ

Before we prove any properties of Rei we must first prove the following theorem.

Theorem 19: For any arborescence on n nodes, if s(j) is the successor function of node j as defined in Definition 10 and d(ji is as defined above, we must have

$$\sum_{j=1}^{n} s(j) = \sum_{j=1}^{n} d(j) \qquad (4)$$

Proof: We will **prove** this **assertion by** the method of finite induction. The statement is obvious for **an** arborescence having two nodes, one of which is the root node. Now assume that the theorem is true for arborescence having up to k nodes. Consider an arborescence having k+1 nodes. Delete an arbitrary beginning node x (see Definition 7). The resulting arborescence has k nodes for which Equation (4) holds. Now add x back to this arborescence. The successor function of each node on the path from the oeginning node *zo* the root nca-i increases *by* one sinpi> irg that rhe increase in the sum of all successor functions is equal to d(x). Thus both sides of Equation (4) increase by the same amount and the proof is complete. •

Theorem 20: The reliability Rel of an arborescence is directly related to the ramification index.

Proof. From Equations (3) and (4) it follows that

$$Rel = p^{(1/n)\sum s(j)}$$

Using the ramification index Definition 11 we can write

$$Rel = \exp\left[\log(p)\,\frac{1}{n}\left(\frac{n(n+1)}{2} - Ram\right)\right] \qquad (5)$$

which can be rewritten as

$$Ram = \frac{n(n+1)}{2} + \frac{n}{(-\log(p))}\log(Rel) \qquad (6)$$

Since $p < 1$ we have $\log(p) < 0$ implying that Ram and Rel are directly related. •

We are now ready to prove several useful properties of Rel.

Definition 21: A *shortest path arborescence* is an arborescence in which there are the minimum possible number of links between any node and the root node.

It is well known that the problem of finding the shortest path arborescence can be solved in polynomial time [20, 14].

Theorem 22: For any communications network, the shortest path spanning arborescence with the central computer as the root node has the maximum reliability.

Proof: According to Equation (3), reliability is maximized when the sum of all d(j) is minimized. For the shortest path arborescence each individual d(j) is minimized and therefore it must have the maximum reliability. •

Corollary 23: For any communications network, the upper bound on the reliability is p. This upper bound may not be achievable for a general network.

Proof: The minimum value possible for each d(j) is d(j) = 1 for j=1,...,n and this is achieved when each terminal is connected directly to the central computer. In this case from Equation (3) we get that Rel = p. For a complete graph it is always possible to find an arborescence with Rel *** p but this may not be possible for a general communications network. ***

Corollary 24: For any communications network, the lower bound on the reliability is

$$\mathrm{Rel} = p^{(n+1)/2} \tag{7}$$

which corresponds to a Hamiltonian Path with the central computer as the last node on the path.

Proof: For a Hamiltonian Path Ram * 0 [43] so it has the least possible reliability. According to Definition 11,

$$\mathrm{Ram} = 0 \Rightarrow \sum_{j*1}^{n} s(j) = \frac{n(n+1)}{2} \tag{8}$$

The assertion follows by substituting Equations (4) and (8) in Equation (3). •

5. Maximizing Reliability Subject to Degree Constraints

Finding **the** maximum reliability spanning arborescence of a communications **network** is **an easy problem** since all **we need** to do is to find a shortest path spanning arborescence. However a more practical problem is to find a spanning arborescence **subject to some kind of degree constraint for each vertex. In** real life problems **the links coming into a node are connected to a multiplexer located** at the **node, and the multiplexer cannot take more than a predetermined number of links. In this section we discuss the effect the degree constraint has on the general reliability problem.**

For simplicity we assume that each terminal or node has the **same degree constraint. However the whole** discussion can be easily generalized to different **degree constraint for different terminals.**

Definition 25: The *degree constraint* of each node, denoted by U. is the :"£/.imurn *vv.p.:.* •- •;• links :iiut can co:fie •.:.; *z.v.* rerminsi.

Theorem 26: A nontrivial degree constrained reliability problem must have

$1 £ U £ D_{max} - 1$

where D_{max} is the maximum degree of any node in the communications network.

Proof: If U is less than 1 then there is no solution. If U is D_{max} or more then from Theorem 22 in this chapter, the shortest path arborescence is the optimal answer which cart be found in polynomial time. Note that for U - 1 we get the Hamiltonian Path problem. •

6. Applications to Reliability of Directed Graphs

Most of the concepts of reliability discussed so far are defined for arborescences. Since we can also find a spanning arborescence for a directed graph, we can extend the validity of the definitions and theorems to the case of directed g/aphs. In particular note that Theorem 22 is true for the directed case also. However, for a directed graph the value of Rei would depend upon which node is chosen as the root node. Furthermore Rei will have to be interpreted as the probability of every node being able to communicate with the root node and this will *not* be the same as the overall reliability of the network.

We consider a special class of directed graphs, the Directed Rectangular Lattice Graphs (DRLG) [42]. Figure 1 shows a 4 X 4 DRLG . Note that node 1 has been split into two nodes 1 and 17. For a DRLG we have D_{max} - 2 so that from Theorem 26 the only nontrivial degree constraint would be U = 1 which corresponds to a Hamiltonian Path. We now show that the case U^{s} 1 can be solved in polynomial time. In [42] we had mentioned without proof that for a DRLG a Hamiltonian Path can be found in polynomial time. Theorem 27 serves as a proof for that statement also.

Theorem 27: If x (#1) is a beginning node of a spanning arborescence of a DRLG then there exists an incoming cross arc (x,j) (See Definition 8) which leads to a new spanning arborescence having one less beginning node and one less junction node.

Proof: Every node in a DRLG has indegree of either one or two. By inspection of Figure 1, if (i,j) is the only arc coming into node j then it must also be the only arc coming out of node i and therefore must also ?!wavs bp ⁴? ?^v ᶜp?nning arhorescpnee. Thus everv beginning node must

have exactly two arcs coming into it. Since D_{max} = 2, every junction node of any spanning arborescence must have exactly two arcs of the arborescence coming into it.

In Figure 1 if x is a beginning node then arcs (w,x) and (y,x) are not in the spanning arborescence so that (y,z) and (w,z) *must* be in the arborescence implying that z is a junction node. If we remove (w,z), the arborescence splits into two distinct parts. Let the part containing node w be called C_w and the other part which contains node z be called C_z.

(a) If x $\notin$ C_w, then add (w,x) to connect C_w and C_z which gives an arborescence having one less beginning node and one less junction node.

(b) If x 6 C , then restore (w,z) and interchange the roles of w and y. Case (a) now applies.

■

Corollary 28: If an arborescence of a DRLG has JN junction nodes (and **number of beginning nodes BN = JN+1) then we can construct a Hamiltonian Path in exactly JN (=BN-1) steps.**

7. An Algorithm for Solving the Degree Constrained Network Reliability Problem

When solving a degree constrained reliability problem we first find a shortest path spanning arborescence of the network. If this arborescence happens to satisfy the degree constraints also then the problem is solved. However, if no shortest **path spanning arborescence satisfying the degree** constraint is **found** then **we pivot to minimize Ram index until the degree constraint *is* satisfied for the first time. Note that if we reach Ram = 0 then we get a Hamiltonian Path which of** course **satisfies the most stringent possible degree** constraints. **Once we** find a **feasible arborescence we can again start to maximize Ram taking care that no degree constraint is violated.**

The Algorithm

Consider an undirected graph G ^{s} (V,A) where V is the set **of** vertices and A is the set of edges of G. Denote by T = ($V_r A_l$j a spanning arborescence of G. Note that A. is a subset of A.

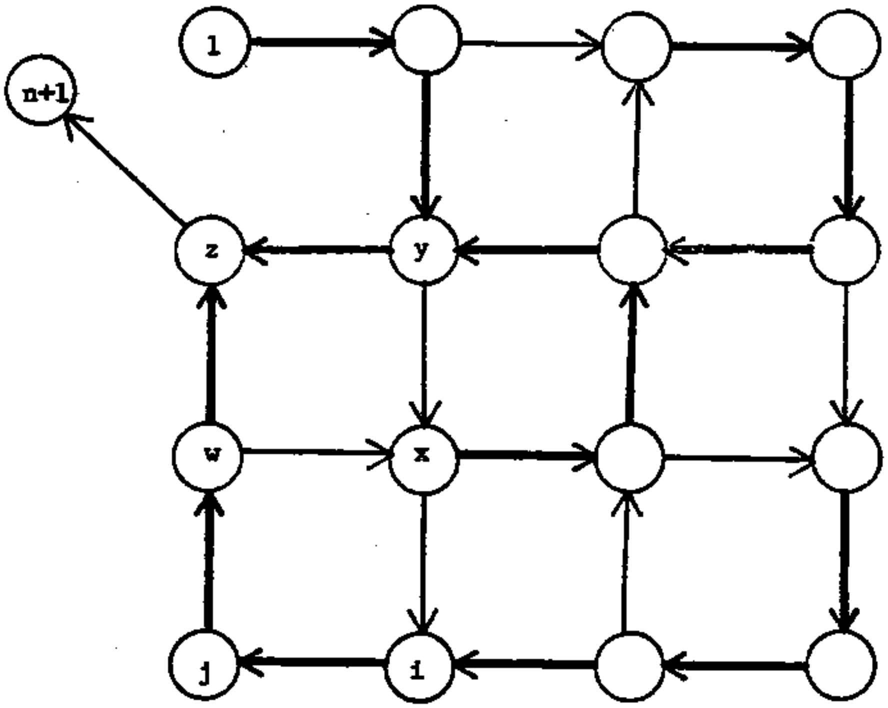

Figure 1: A 4 X 4 DRLG with spanning arborescence having darkened edges. A beginning node x, and corresponding junction z node is labeled.

- **Step 0 :** ***Find Shortest Path Arborescence.*** **Find a shortest path arborescence T using any of the standard methods. If this arborescence satisfies the degree constraint then STOP — the optimal answer is obtained.**

- **Step 1 :** ***Calculate the successor function and ramification index of*** **T. Using Definition 10 calculate s(i) for all i € V. Then from Definition 11 calculate Ramfl").**

- **Step 2 :** ***Check the Degree Constraint.*** **If T satisfies the degree constraints then go to step 6 else go to step 3.**

- **Step 3 :** ***Find an incoming arc to decrease Ram.*** **For each (undirected) arc (i,j) 6 A - A_T calculate using Theorem 13, the maximum possible decrease in ramification index if that arc is brought into A_T in the direction from i to j or from j to i. Let ARam be the maximum decrease ir, ramification ndex (e,f) the corresponding incoming arc and [illegible] the maximal cycle -ode**

thus created Let the new ramification index be Ram_{new} = Ram - ARam.

- Step 4 : *Check possible cases.* If

Ram_{new} ≥ Ram STOP - Ram cannot be decreased further, tf another trial is desired go back to Step 0 and generate a new, different spanning arborescence.

If **Ram_{new}** < **Ram** Go to next step.

- Step 5 : *Updating.* Let (h,k) be the outgoing arc given by Theorem 14. Let ^ = A_T + {(e,f)} - {(h,k)}, Ram = Ram_{R0W}. Update the successor functions. If the resulting arborescence satisfies the degree constraints go to the next step else go to step 3.

- Step 6 : *Find an incoming arc to increase Ram.* For each (undirected) arc (i,j) G A - A_T and whose introduction to the arborescence does not violate any degree constraint, calculate the ramification index if that arc is brought into A_y using Theorem 13 in the direction from i to j or from j to i. Let **Ram_{new}** be the maximum of these ramification indexes and let (e,f) be the corresponding arc.

- Step 7 : *Check the possible cases.* If no suitable incoming arc is found then the current arborescence is a "good" solution. Else go to next step. .

- Step 8 : *Perform Pivoting.* Bring the cross arc (e,f) having ramification index Ram_{new} into the solution by setting A_T = A_T - **{(f,k;** : (f,k) G A_T} U {(e,f)}. Let Ram = Ram_{new}. Update the successor function. Go to step 6.

8. Interpretation of Computational Results

We coded the algorithm described in Section 7 in FORTRAN and implemented it on DECSYSTEM-20 at Carnegie-Mellon University. The results obtained are shown in Figures 2 and 3. In each of these tables the columns have the following interpretations :

- Column 1 is the number of nodes in the generated graph.

- Column 2 is the maximum possible reliability of any arborescence of the npnerated graph as cornputed using Equarion (5^V a?-':^^? -^at 3=0.99.

Number of Nodes excluding root node = m
Number of Arcs in the Graph « 2 * m * log(m)
All the values are averaged over 4 to 7 problems of each size
The reliabilities were calculated using Equation 22 with p»0.99
Indegree of Every node £ 3

(1) m	(2) Max Rel	(3) Obtained Rel	(4) Ratio(%)	(5) # Pivots	(6) CPU Seconds
100	0.9717	0.9247	95.2	122	1.3
200	0.9703	0.9129	94.1	233	3.9
300	0.9693	0.8972	92.6	337	7.6
500	0.9681	0.8838	91.3	573	17.9
700	0.9679	0.8749	90.4	796	29.4
900	0.9664	0.8641	89.4	1001	42.3

Figure 2: Computational results for maximum indegree * 3

Number of Nodes excluding root node = m
Number of Arcs in the Graph = 2 * m * log(m)
All the values are averaged over 4 to 7 problems of each size
The reliabilities were calculated using Equation 22 with p=0.99
Indegree of Svery node £ 2

(1) m	(2) Max Rel	(3) Obtained Rel	(4) Ratio(%).	(5) # Pivots	(6) CPU Seconds
100	0.9719	0.9046	93.0	114	0.9
200	0.9702	0.8875	91.5	223	2.9
300	0.9689	0.8735	90.2	348	7.9
500	0.9683	0.8585	88.7	564	15.6
700	0.9667	0.8359	86.5	800	35.0
900	0.9667	0.8324	86.1	1030	58.0

Figure 3: Computational results for maximum indegree = 2

Theorem 22 explains how such an arborescence may be found. The value in Column 2 corresponds to the *exact* solution.

- Column 3 is the actual reliability found for a spanning arborescence of the generated graph which satisfies the stated degree constraint. This is a *heuristic* solution, since there may exist a spanning arborescence which also satisfies the degree constraint and has higher reliability.

- Column 4 is the ratio of Columns 3 and 2. This ratio is consistently better than 86% which indicates that the heuristic solution is a relatively good one. The ratio decreases for larger problems suggesting that solutions obtained for larger problems are not as good as those obtained for smaller ones. As one would expect intuitively, the ratio is higher when the degree constraint is more relaxed.

- Columns 5 and 6 give the average number of pivots and the CPU time taken for each value of m.

There are several interesting observations that can be made concerning the results given in Figures 2 and 3. Note that it was always possible to obtain a reliability very close to the maximum possible for the unconstrained problem, while still satisfying the degree constraints. In all the cases tested the computational times required were acceptable and vary approximately linearly with the number of nodes.

We did not run problems having a more relaxed degree constraints because clearly they would have yielded better solutions and run faster. When degree constraint is 1 then we are simply looking for a Hamiltonian Path which has a unique reliability given by Equation (7), so these were not run either.

If the reliability found in Column 3 of Figures 2 and 3 is deemed to be unsatisfactory then there are several methods of improving the results. We can increase the value of p by using better quality links between terminals. Usually this is possible only due to a technological breakthrough. Or we can add extra links to the communications network to get a better spanning arborescence. If the network consists of all possible links then the maximum reliability spanning arborescence will have reliability p as given by Corollary 23.

9. Conclusions

The literature on network reliability is extensive and several different definitions for network reliability have been suggested. We investigated the problem of maximizing the reliability of a communications network formed when several terminals had to communicate with one central computer. It was shown that an arborescence with high ramification index also has a high reliability. We proved that a shortest path spanning arborescence has the highest possible reliability among all spanning arborescences for both a directed and an undirected graph.

A heuristic program was developed for finding the maximum reliability of an arborescence for which the indegree of each node was constrained by an upper bound. Computational results were obtained which demonstrated that the degree constrained maximum reliability is an exponential problem in the worst case but is easy to solve approximately in practice. It was found that the computation time varies linearly with the number of nodes.

References

1. Abraham. J.A. "An improved algorithm for network reliability/' *IEEE Transactions on Reliability R-28* (Apr 19791 58-61.

2. Aggarwal, K.K., Gupta, J.S.. and Misra, K.B. "A simple method for Reliability evaluation of a communications system." *IEEE Transactions on Communications COM-23* (May 1975), 563-565.

3. Aggarwal, KJC. Misra, K.B., and Gupta, J.S. "A fast algorithm for Reliability evaluation." *IEEE Transactions on Reliability R-24* (Apr 1975), 83-85.

4. Aggarwal, K.K., Gupta, J.S., Misra, K.B. "A new heuristic criterion for solving a redundancy optimization problem." *IEEE Transactions on Reliability R-24* (1975), 86-87.

5. Aggarwal, K. K., and Rai, S. "An Efficient Method for Reliability Evaluation of a General Network." *IEEE Transactions on Reliability R-27* (Aug 1978), 206-209.

6. Aggarwal, K. K., and Rai, Suresh. "Reliability Evaluation in Computer Communications Network." *IEEE Transactions on Reliability R-30.* 1 (Apr 1981), 32-35*

7. Aggarwal, K.K., Chopra, Y.C., and Bajwa, J.S. "Capacity considerations in Reliability analysis of communication systems." *IEEE Transactions on Reliability R-31.* 2 (Jun 1982), 177-181.

8. Agrawal, Avinash, Barlow, Richard E. "A Survey of Network Reliability and Domination Theory." *Operations Research 32.* 3 (1984), 478-492.

9. Agrawal, Avinash, Satyanarayna, A. "An O(|Ej) Time Algorithm for Computing the Reliability of a Class of Directed Networks." *Operations Research 32.* 3 (1984), 493-515.

10. Ahmad, Hasanuddin S. "A Simple Technique for computing Network Reliability." *IEEE Transactions on Reliability R-31.* 1 (Apr 1982), 41-44.

11. Ball, M.O. "Computing Network Reliability." *Operations Research 27* (Jul-Aug 1979), 823-838.

12. Ball, Michael O. "Complexity of Network Reliability Computations." *Networks 10* (1980), 153-165.

13. Billinton, R., and Bollinger, K.E. "Transmission system Reliability evaluation using Markov Process." *IEEE Transactions on Power Apparatus and Systems PAS-87* (Feb 1968), 538-547.

14. Bondy J. A. and Murty U. S. R~ *Graph Theory with Applications.* North Holland, 1978.

15. Buzacott, J.A. "Markov approach to finding failure time of repairable systems." *IEEE Transactions on Reliability R-19* (Nov 1970), 128-134.

16. Buzacott, J.A. "Network approaches to finding the Reliability of repairable systems." *IEEE Transactions on Reliability R-19* (Nov 1970), 140-146.

17. Fralta, L. and Montanari, U. G. "A recursive method based on case analysis for computing network terminal Reliability." *IEEE Transactions on Communications CQM-26* Aug 1978), 1166--; 177.

18. Frank, K, and Frisch, LT. "Analysis and design of survivable networks." *IEEE Transactions on Communications Technology COM-18* (Oct 1970), 501-519.

19. Gadani, J.P., and Misra, K.B. "A network reduction and Transformation algorithm for assessing system effectiveness indexes." *IEEE Transactions on Reliability R-30,* 1 (Apr 1981), 48-57.

20. Garfinkel, Robert S. and Nemhauser, George L. *Integer Programming.* John Wiley and Sons, 1972.

21. Hagstorm, Jane Nichols. *Combinatoric Tools for Computing Network Reliability.* Ph.D. Th., University of California, Berkeley, 1980.

22. Hagstorm, Jane Nichols. "Using the Decomposition tree of a network in Reliability Computations." *IEEE Transactions on Reliability R-32* (1983), 71-78.

23. Hagstorm, Jane Nichols. Combinatorial Properties for Directed Network Reliability with a Path Length criterion. Tech. Rept. 82-14, College of Business Administration, University of Illinois, &Box 4348, ChBox 4348, Ch, 1983.

24. Inoue, K., **and** Henley, EJ. Computer aided Reliability and safety analysis of complex systems. IFAC Sixth Triennial World Congress, Aug, 1975. Part 1110

25. Jedrzejowicz, Piotr. "Allocation of resources to maximize quality measures of a system." *IEEE Transactions on Reliability R-31.* 1 (Apr 1982).

26. Jensen, P.A., and Bellmore, M. "An algorithm to determine the Reliability of a complex system." *IEEE Transactions on Reliability R-18* (Nov 1969), 169-174.

27. Kohda, Takehisa, and Inoue, Koichi. "A Reliability optimization method for complex systems with the criterion of local optimality." *IEEE Transactions on Reliability R-31.* 1 (Apr 1982), 109-111.

28. Kuo, W., Hwang, C.Lf Tillman, F.A. "A note on heuristic methods in optimal system reliability." *IEEE Transactions on Reliability R-27* (Dec 1978), 320-324.

29. Lee, Keun K. "A Compilation technique for exact system reliability." *IEEE Transactions on Reliability R-30.* 3 (Aug 1981), 284-288.

30. Lin, P.M., and Leon, B.J., and Huang, T.C. "A new algorithm for symbolic system reliability analysis." *IEEE Transactions on Reliability R-25* (Apr 1976), 2-15.

31. Mirchandani, Pitu B. "Shortest distance and Reliability of Probabilistic Networks." *Computers and Operations Research 3* (1976), 347-355.

32. Misra, K. B., and Rao, T. S. M. "Reliability Analysis of Redundant Network using flow graph." *IEEE Transactions on Reliability R-19* (Feb 1970), 19-24.

33. Moreira de Souza, J., and Landrault, C. "Benefit Analysis of Concurrent Redundancy techniques." *IEEE Transactions on Reliability R-30.* 1 (Apr 1981), 67-70.

34. Murchland, J. D. Fundamental concepts and relations for Reliability Analysis of multi-state systems. In *Reliability and Fault Tree Analysis,* Barlow, R. E., Fussel, J. B., Singapurwaila, N. D., Eds., SIAM, Philadelphia, 1975, pp. 581-618.

35. Nakagawa, Y.. Nakashima, K. "A heuristic method for determining optimal reliability allocation." *IEEE Transactions on Reliability R-26* (Aug 1977), 156-161.

36. Nakagawa, Yuji, and Miyazaki, Satoshi. "An experimental comparison of the heuristic methods for solving system Reliability optimization problems." *IEEE Transactions on Reliability R-30,* 2 (Jun 1981).

37. Nakazawa, Hayao. "Bayesian Decomposition Method for Computing the Reliability of an oriented network/' *IEEE Transactions on Reliability R-25* (1976), 77-80.

38. Nakazawa, Hayan. "Decomposition method for computing the Reliability of complex networks." *IEEE Transactions on Reliability R-30.* 3 (Aug 1981), 289-292.

39. Satyanarayna, A., and Prabhakar, A. "New topological formula and rapid algorithm for reliability analysis of complex networks." *IEEE Transactions on Reliability R-27* (Jun 1978), 82-100.

40. Satyanarayna, A., and Hagstorm, J.N. "A new algorithm for the reliability analysis of multi-terminal networks." *IEEE Transactions on Reliability R-30* (Oct 1981), 325-334.

41. Satyanarayna, A. "A unified formula for analysis of some network reliability problems." *IEEE Transactions on Reliability R-37.* 1 (Apr 1982), 23-32.

42. Thompson, G.L, and Singhal, S. "A Successful Algorithm for Solving Directed Hamiltonian Path Problems." *Operations Research Letters 3.* 1 (Apr 1984), 35-42.

43. Thompson, G.L., Singhal, S. A Successful Algorithm for the Undirected Hamiltonian Path Problem. Working Paper 497, GSIA, Carnegie-Mellon University, Pittsburgh, PA, August, 1984. Accepted for publication in Discrete Mathematics

44. Wilkov, R. S. "Analysis and Design of Reliable Computer Network." *IEEE Transactions on Communications COM-20* (Jun 1972), 660-678.