Edith Cowan University

Research Online

2005

Establishing Physical Survivability of Large Networks using Properties of Two-Connected Graphs

Daryoush Habibi
Edith Cowan University

Hoang N. Nguyen
Edith Cowan University

Quoc Viet Phung
Edith Cowan University

Kung-meng Lo
Edith Cowan University

Follow this and additional works at: https://ro.ecu.edu.au/ecuworks

Part of the Engineering Commons

Establishing Physical Survivability of Large Networks using Properties of Two-Connected Graphs

Daryoush Habibi, Hoang N. Nguyen, Quoc V. Phung, and Kung-meng Lo
COMMUNICATIONS RESEARCH GROUP
SCHOOL OF ENGINEERING AND MATHEMATICS
EDITH COWAN UNIVERSITY
JOONDALUP WA 6027 AUSTRALIA
Email: d.habibi@ecu.edu.au
Voice: +61 8 6304 5787 Fax: +61 8 6304 5811

Abstract— **Establishing the physical survivability of large networks is not a trivial task. Some techniques for assessing physical survivability such as the** *cutset* **method can not deal with large size networks [1], [2]. A fast technique for finding** *biconnected components* **of a graph and testing the network for node-/link-bridges, presented in [3], does not provide any further information, such as identifying the fundamental cycles within the network, which would significantly benefit the next phase of network design for protection using such techniques as shared backup path protection (SBPP), p-cycle, or ring protection [3]. This paper presents an alternative technique, based on graph theory, for evaluating the physical survivability of networks. This technique can deal with network sizes of many thousand nodes, with computational times which are comparable with the** *biconnected components* **method, whilst providing more information about the susceptibility of a network to individual link and node failures in preparation for the next phase of network protection design.**

I. INTRODUCTION

Design of survivable communication networks has been a challenging problem. Without establishing network survivability, there can be severe consequences when a physical link fails. Network failures which may be caused by dig-ups, vehicle crashes, human errors, malfunctional systems, fire, rodents, sabotage, natural disasters (eg. floods, earthquakes, lightning storms), and some other factors, have occurred quite frequently and sometimes with unpredictable consequences. To tackle these, survivability measures can be implemented at the service layer, the logical layer, the system layer, and the physical layer. The physical layer is the base resource infrastructure of the network, and to be able to protect it, we need to ensure that the physical topology of the network has sufficient link and node diversity. Without this, protection at higher layers will not be feasible. With the implementation of Dense Wavelength Division Multiplexing (DWDM) in the optical backbone of metropolitan and long-haul networks, greater flexibility is achieved in providing alternate routes for lightpath connections. However, the survivability problem at the physical layer remains the same. In fact, it becomes even more critical, because each link of a backbone network carries huge amounts of traffic and the failure of an optical component, such as a fiber cut or a node failure, may cause a very serious problem in terms of loss of data and profit. For instance, the direct voice-calling revenue loss from failure of major trunk group is frequently quoted at $100,000/minute or more. Therefore, network survivability is a critical and imperative issue in telecommunication networks today, particularly in optical networks.

A physical topology is considered to be survivable if it can cope with any single failure of network components by rerouting those connections affected by the failure through alternative paths. Clearly, this requires some resource redundancies in the network. Using the graph theory terminologies, a survivable physical network must be a two-connected graph, or a biconnected graph [3]. The Menger's theorem [4] gives the necessary and sufficient condition for survivability of networks at the physical layer, using the connectivity between network's cutsets. However, the computational complexity of this model grows exponentially with the size of the network, since a network with N nodes would yield $2^N - 2$ cutsets. Therefore, the cutset technique cannot efficiently deal with even moderate size networks of say 40 nodes, and larger networks are out of computational reach of this technique. Testing for survivability of large networks can be done using using a technique called *biconnected components* of a graph introduced by W. D. Grover [3]. This technique can determine vulnerable links and nodes of the network. However, verifying network survivability is just the first step in network planning, after which we need to apply appropriate protection routing schemes using such techniques as SBPP, p-cycle, or ring protection [5], [6], [7]. It is therefore very helpful if the algorithm used for determining the physical survivability of the network can also provide additional information which is of benefit to protection design.

In this paper, based on graph theory, we introduce a new method for examining the physical survivability of networks using properties of 2-connected graphs. This technique also determines all simple distinct cycles on the network which is

useful for the protection design. The rest of this paper is organized as follows: Sec. II provides the theoretical background, including some basic definitions, properties of 2-connected graphs, and the theory of cutsets. It also outlines the necessary condition for a physical network to be survivable. Sec. III outlines our proposed method for verifying network survivability using the properties of 2-connected graphs. Sec. IV presents the results and finally Sec. V provides our conclusions for the work presented in this paper.
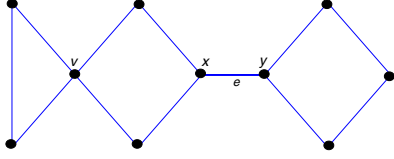

Fig. 1.   A network with a link bridge (e) and a node-bridge (v)

II. SURVIVABILITY VERIFICATION FRAMEWORK

In this section we will first outline some notations and definitions related to graph theory. We will then look at some theorems and techniques for establishing the physical survivability of the network.

A. Definitions

The following definitions are adopted from [8] and [9].

- *Graph*: A graph G is a pair of sets V and E satisfying $E \subseteq [V]^2$, where V is a set of vertices (or nodes) and E is the set of edges (or links) connecting two distinct vertices in V.
- *Connected graph*: a non-empty graph G is *connected* if any two of its vertices are linked by a path in G, and is k-connected if any two of its vertices can be joined by k independent paths.
- *Subgraph*: a graph $G'(V', E')$ is called a subgraph of the graph $G(V, E)$, denoted by $G' \subseteq G$, if $V' \subseteq V$ and $E' \subseteq E$.
- *Component*: a maximum connected subgraph of G is called a *component* of G.
- *Cutvertex and Bridge*: As illustrated in Fig. 1, a vertex $v \in V$ of graph $G(V, E)$ is called a *cutvertex* (or node-bridge) if it separates two other vertices of the same component. An edge $e \in E$ is called *cutedge* (or link-bridge) if it is the only means of connecting its end vertices.
- *H-path*: Given a graph H, a path P is called H-path if P is non-trivial, and meets the graph H exactly at its end vertices.
- *Block*: a maximal connected sub-graph without a cutvertex is called a block. A block of a graph G will either be a maximal 2-connected sub-graph, a bridge, or an isolated vertex. Conversely, every such sub-graph is a block. Different blocks of G overlap on at most one

vertex, which is then a cut vertex of G. Thus, every edge of G lies in a unique block, and G is the union of its blocks. This is demonstrated in Fig. 2.
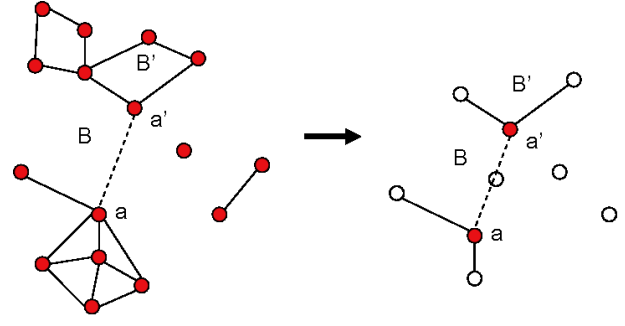

Fig. 2.   A graph and its blocks, adopted from [8]

Based on the above definitions, we shall now describe two existing techniques for determining the physical survivability of networks.

B. Survivability via Cutsets

In the cutset technique, a cut divides the graph representing the network into two subgraphs, referred to as a cutset, and the size of the cutset is defined as the number of edges connecting these two subgraphs. If for every possible cutset, there are two or more links between the two subgraphs of the cutset, then the network is survivable. Menger's theorem [4], [8], given below, determines the connectivity of a network by examining its cutsets.

Theorem 2.1: A topology with the set of vertices (nodes) N and the set of edges (links) E is 2-connected if and only if every non-trivial cut $\langle S, N - S \rangle$ has a corresponding cutset of size greater than or equal to 2.

Network survivability can be verified using Menger's theorem. However, as discussed earlier, the complexity of the algorithm increases exponentially with the number of nodes and it cannot deal with large networks.

C. Survivability via 2-Connected Graphs

From Theorem 2.1, it can be deduced that the cutsets of a cycle always have a size of 2. Furthermore, a 2-connected graph can be easily constructed from simple cycles. The following proposition implies a method for constructing such graph [8].
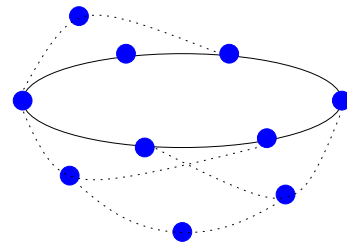

Fig. 3.   2-connected graphs

Proposition 1: A graph is 2-connected if and only if it can be constructed from a cycle by successively adding H-paths to graph H already constructed.

Proof: Cleary, every graph constructed as proposed is 2-connected. Conversely, let G be a 2-connected graph, then G contains a cycle, and a subgraph H is constructible, as evident in Fig. 3. Any edge $x, y \in E(G) \backslash E(H)$ with $x, y \in H$ defines a H-path. Then, H is an induced sub-graph of G. If $H \neq G$, then by the connectedness of G, there is an edge vw with $v \in G - H$ and $w \in H$. As G is 2-connected, $G - w$ has a $v - H$ path P. Then wvP is a H-path in G, and $H \cup wvP$ is a constructible sub-graph of G. ∎

III. The Proposed Technique

Assume that G' and G'' are two blocks of graph G. From Proposition 1, we can deduce the connectivity of graph G depending on the relation between G' and G'', as described below.

1) If G' and G'' have at least 2 common vertices, then G is a 2-connected graph with no cutvertex (i.e. node bridge) or cutedge (i.e. link bridge).
2) If G' and G'' only have one common vertex, then G is a 2-connected graph with a cutvertex which is the common vertex.
3) If G' and G'' are separated by a cutedge, then G is not a 2-connected graph, and the cutedge cannot be protected.
4) If G' and G'' have no common links or nodes, then G is not a 2-connected graph, and therefore it is not survivable.

Based on the above discussion, we can use the relationship between networks' cycles or 2-connected graphs to verify the survivability of its physical topology. An undirected graph is thus seen as the combination of all the fundamental cycles. Using Alg. 1, these fundamental cycles can be found from a spanning tree $\{V, T\}$ of a graph $G = \{V, E\}$ (eg. the spanning tree highlighted by thick lines in Fig. 4).

Algorithm 1 *Finding cycle*

Input : A tree T and an edge e whose end-nodes is in T;
Output: A cycle P formed by T and e;
  init
  $(s, d) \leftarrow$ end-nodes of e;
  $queue \leftarrow [node.s, node.P]$; $check \leftarrow 0$;
  while $check == 0 \& queue \neq \emptyset$ **do**
    $[v] \leftarrow head(queue)$; $queue \leftarrow queue - \{head(queue)\}$;
    if $v.s == d$ **then**
      $check = 1$; $P \leftarrow v.P$
    else
      for all v_k is neighbour of $v.s$; **do**
        $node.s \leftarrow v_k$; $node.P \leftarrow P \cup v_k$;
        push $node$ into $queue$;
      end for
    end if
  end while

However, it is not easy to find all of the fundamental cycles in the graph. For instance, in Fig. 4, the edges represented with thin lines are not part of the spanning tree (shown by thick lines). If any of these edges are added to the tree, it will form a unique cycle, but such cycle is not necessarily a fundamental cycle (eg. consider adding edge b-c). Any set of cycles found from the spanning tree can be used to verify the survivability of the topology from which it is generated. An algorithm for finding a set of cycles through spanning tree of a graph is represented in Alg. 1. An efficient method for finding fundamental cycles of a graph, referred to as $Paton's$ algorithm, is outlined in [10]. Further discussion of this topic is outside the scope of this paper.
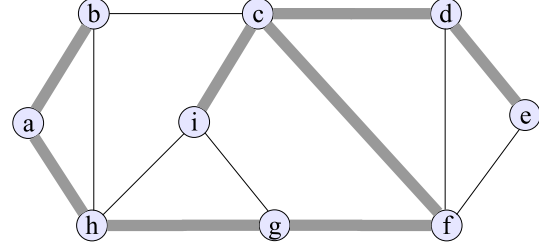

Fig. 4. Spanning Tree on a arbitrary graph

If a graph is 2-connected, then each vertex of the graph will be at least on one of the cycles resulting from Alg. 1. Hence, such set of cycles is sufficient to verify the survivability of the graph. Next, we introduce an algorithm, represented by Alg. 2, which not only verifies the survivability of a graph, but also identifies the vulnerabilities of the graph, eg. cutvertices and/or cutedges if they exist.

IV. Simulation

In this section we shall first demonstrate the inviability of the cutset technique for establishing the survivability of large networks. It should be noted that in the cutset technique, the number of subnetworks that must be considered for a network of N nodes is $2^N - 2$. Therefore, the computational time increases logographically with the size of the network. Table I gives the actual and estimated computational times achieved using a Pentium 4, 2.8 GHz processor. It should be noted that the computational times for networks of 40 nodes and higher are extrapolated from the computational time of the smaller networks, based on the increase in the computational complexity as a function of the number of network nodes. We can clearly see from the results of Table I that the cutset technique is not suitable for large networks.

Before presenting the performance results of our proposed technique, we shall give an example of how our approach works over an arbitrary physical topology G as shown in Fig. 5(a), with the set of nodes V and edges E.

Since this topology is an unconnected topology, the first step results in a tree T, being a subgraph of G, and an unconnected node 13, as shown in Fig. 5(b). T has a set of nodes V_T and edges E_T, where $V_T = V - \{13\}$, and $E_T = E - \{(2, 4), (2, 5), (8, 9), (11, 12)\}$. The spanning tree

Algorithm 2 Survivability verification

Input : $cycles$ = list of cycles;
  m = number of cycles; E = set of network links
Output: Nodebridges, 2-connected subgraphs, linkbridge
  init
  $deep = 1; back = 1;$
  $currentCycle \leftarrow firstcycle$
  while $newcycles \neq cycles$ **do**
    for $i = size(back) : m - 1$ **do**
      $|loc| = cycles(i+1) \bigcap currentCycle$
      case $|loc| > 1$: joint at more than 2 vertices
      $newcycle = \bigcup currentCycle, cycles(i+1)$
      $cycles = \{cycles \setminus \{cycles(1), cycles(i+1)\}\}$
      $cycles = \bigcup\{cycles\{cycles, newcycles\}\}$
      $currentCycle \leftarrow firstcycle$
      case $|loc| = 1$: possible cut vertex at loc
      push i into $queue$;
      $possibBridge \leftarrow loc$ % possible cut vertex at loc
      push $indices(end - 1)$ into $back$
      push $(i + 1)$ into $deep$
      if $i = m - 1$ & $length(deep) > 1$ **then**
        $Nodebridges = \bigcup\{Nodebridges, possibBridge(last)\}$
        2-connected= $\bigcup$ {2-connected, cycles(deep(last))}
      end if
    end for
  end while
  for $i = 1$ to $|E|$ **do**
    if E(i) intersect two distinct blocks (2-connected or single node subgraphs) **then**
      $linkbridges = \bigcup\{linkbridges, Ei\}$
    end if
  end for

TABLE I

ACTUAL AND ESTIMATED COMPUTATIONAL TIMES FOR ESTABLISHING SURVIVABILITY USING CUTSET TECHNIQUE

No. of Nodes	Time (s)
10	0.728
20	825
25	26350
40	1.0737×10^9 (34 years)
80	1.1806×10^{21} (3.7×10^{11} centuries)
200	1.5693×10^{57}

(a) An arbitrary physical topology

(b) The spanning tree
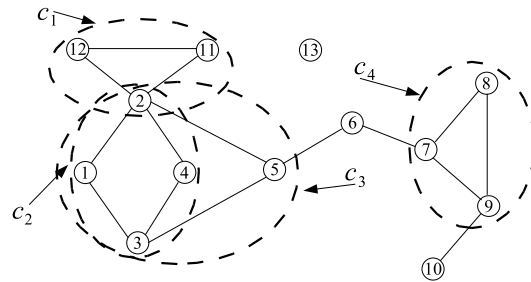
(c) Survivable-base results

Fig. 5. *An illustrative example*

can be determined using Prim's algorithm or Kruskal's algorithm [11].
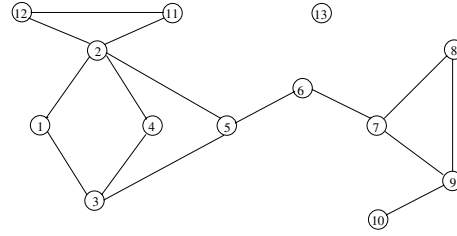
Next, a set of cycles is found using Alg. 1. In our example, this consists of 4 cycles $\{c_1, c_2, c_3, c_4\}$ as shown in Fig. 5(c).

The configuration of the spanning tree resulting from the first step allows us to conclude that G is an unconnected topology. However, further analysis of the physical topology can be performed in the second step, through the survivable-base algorithm, Alg. 2. The input of the second step is the spanning tree T of Fig. 5(b), and the output is shown in Fig. 5(c). Note that topology G contains 3 maximal survivable-bases, namely as $S_1 = \{c_1\}$, $S_2 = \{c_2, c_3\}$, and $S_3 = \{c_4\}$. S_1 and S_2 share nodes 2 in graph G, hence node 2 is a *cutvertex* (or node-bridge). There are 3 link-bridges which are $(5 - 6), (6 - 7)$ and $(9 - 10)$. Nodes $6, 10, 13$ which are not part of any 2-connected block are referred to as single nodes.
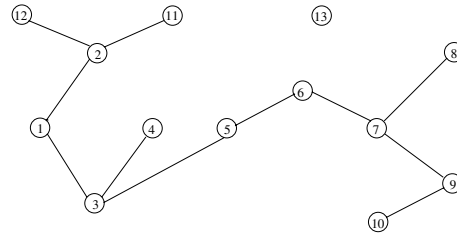
To demonstrate the computational efficiency of our algorithm, we randomly generate networks with various sizes. Fig. 6 represents the simulated results of 10 randomly generated networks with the number of nodes varying from 15 to 375 nodes. The graph shows that the verification time increases almost linearly with increasing the number of network nodes. The simulation results have shown that 2-connected graph theorem can be used to identify the weak nodes/links of a given
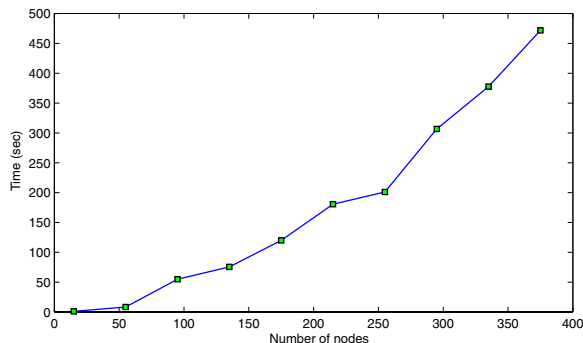
Fig. 6. *Simulation times resulting from the proposed technique, averaged over 10 randomly generated networks with an average nodal degree of 3*

large size network much faster than some other techniques such as 'cutset'. Furthermore, it also provides information about all distinct cycles in the network, useful for the next phase of network planning, which cannot be provided by the *biconnected components* technique.

V. CONCLUSION

In this paper we have presented a new approach for evaluating the physical survivability of large networks. The computational efficiency of this approach, when dealing with large networks, is comparable to the *biconnected components* approach in [3]. However, our technique is also capable of providing all the distinct fundamental cycles of the network, if required. It also identifies all node-bridges and link-bridges

of the network, not previously considered in the literature. The work presented in this paper forms a good basis for further developments towards design and optimization of survivable networks.

REFERENCES

[1] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks. part i-protection," in *Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM'99*, vol. 2, (New York, NY, USA), pp. 744–751, 1999.
[2] W. N. Grover and J. Doucette, "Topological design of survivable mesh-based transport networks," *Annals of Operations Research*, vol. 106, pp. 79–125, 2001.
[3] W. D. Grover, *Mesh-based Survivable Networks: Options and Strategies for Optical, MPLS, SONET/SDH, and ATM networking*. Prentice Hall PTR, 2004.
[4] E. Modiano and A. Narula-Tam, "Survivable routing of logical topologies in WDM networks," in *IEEE INFOCOM2001*, 2001.
[5] Q. V. Phung, D. Habibi, H. N. Nguyen, and K.-M. Lo, "An efficient approach to optimal wavelength routing in WDM optical networks," in *The 12th IEEE International Conference on Networks*, vol. 2, (Singapore), pp. 600–604, 2004.
[6] Q. V. Phung, D. Habibi, H. N. Nguyen, and K. Lo, "Capacity utilization versus congestion levels in wavelength routing for large scale WDM networks," *To appear in the Proceedings of the 11th Asia-Pacific Conference on Communications*, 2005.
[7] Q. V. Phung, D. Habibi, H. N. Nguyen, and K. Lo, "K pairs of disjoint paths algorithm for protection in WDM optical networks," *To appear in Proceedings of the 11th Asia-Pacific Conference on Communications*, 2005.
[8] R. Diestel, *Graph Theory*. New York: Springer-Verlag, 2000.
[9] L. Foulds, *Graph Theoy Applications*. Springer-Verlag, 1992.
[10] P. Keith, "An algorithm for finding a fundamental set of cycles of a graph," *Communication of the ACM*, vol. 12, no. 9, 1969.
[11] E. Horowitz, S. Sahni, and S. Rajasekaran, *Computer Algorithms*. Computer Science Press, 1998.