European Journal of Operational Research 238 (2014) 810–823

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Decision Support

Fast computation of bounds for two-terminal network reliability

Stefano Sebastio [b,*], Kishor S. Trivedi [a], Dazhi Wang [a], Xiaoyan Yin [a]

[a] Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA
[b] IMT Institute for Advanced Studies, Lucca, Italy

ARTICLE INFO

Article history:
Received 27 November 2013
Accepted 27 April 2014
Available online 9 May 2014

Keywords:
Binary Decision Diagram (BDD)
Bounds computation
Large network reliability
Reliability graph
Reliability prediction

ABSTRACT

In this paper, an algorithm for the fast computation of network reliability bounds is proposed. The evaluation of the network reliability is an intractable problem for very large networks, and hence approximate solutions based on reliability bounds have assumed importance. The proposed bounds computation algorithm is based on an efficient BDD representation of the reliability graph model and a novel search technique to find important minpaths/mincuts to quickly reduce the gap between the reliability upper and lower bounds. Furthermore, our algorithm allows the control of the gap between the two bounds by controlling the overall execution time. Therefore, a trade-off between prediction accuracy and computational resources can be easily made in our approach. The numerical results are presented for large real example reliability graphs to show the efficacy of our approach.

1. Introduction

Large and complex networks are used in telecommunications, aerospace, transportation, power, banking and many such sectors. In the design of such networks, their reliability ($R(t)$) is an important attribute. To support reliability prediction of such networks, many algorithms have been developed. Reliability graph (Colbourn, 1987; Sahner, Trivedi, & Puliafito, 2012; Shooman, 2002) is one of the commonly used non-state-space models for network reliability analysis. Network reliability analysis problem is commonly referred to as the reliability graph analysis problem, the s-t connectedness problem or the two-terminal reliability analysis problem (Colbourn, 1987; Shier, 1991). In the two-terminal network problem the interest lies in computing the probability of communication between a fixed pair of nodes. As long as there is a communication path among that pair of nodes the network is considered operative. Many algorithms for the solution of this problem are known but none of them scale up to the real networks that tend to be very large. To deal with the otherwise computationally intractable problem of the reliability computation of large-scale networks, reliability bounds are often sought. This paper contributes with a new bounding algorithm. Note that we provide upper and lower bounds on network unreliability. Clearly by subtracting from 1, these can be converted into lower and upper bounds respectively, of network reliability.

Exact solution methods for the network reliability problem can be divided into two classes: the factoring/decomposition methods (Misra, 1970; Page & Perry, 1989; Satyanarayana & Chang, 1983; Wood, 1985) and the minpaths/mincuts enumeration methods. In the factoring/decomposition method, the basic idea is to condition on an edge in the reliability graph and break the model into two disjoint cases. In one disjoint case, it is assumed that the edge is up and a new graph is generated by merging the two nodes of the edge. In the other disjoint case, it is assumed that the edge is down and a new graph is generated by removing the edge. This factoring/decomposition method is recursively applied on each of the two subgraphs until the resulting graphs have a series–parallel structure. In the minpaths/mincuts enumeration method, a Boolean expression is constructed based on all minpaths (or mincuts) and evaluated. In the reliability graph theory (Colbourn, 1987) a path is defined as a set of edges (components) so that if these edges are all up, the system is up; it is minimal if it has no proper subpaths. While a cut is defined as a set of edges so that if these edges are all down, the system is down; it is minimal if it has no proper subcuts. Based on the Boolean expression evaluation technique, the approaches can be further divided into inclusion/exclusion methods (Kim, Case, & Ghare, 1972; Lin, Leon, & Huang, 1976; Satyanarayana & Prabhakar, 1978), Sum of Disjoint Products (SDP) methods (Hariri & Raghavendra, 1987; Heidtmann, 1989; Jane & Yuan, 2001; Luo & Trivedi, 1998; Rai, Veeraraghavan, & Trivedi, 1995; Soh & Rai, 1993; Veeraraghavan & Trivedi, 1990) and Binary Decision Diagram (BDD) methods (Chang, Lin, Chen, & Kuo, 2003; Kuo, Lu, & Yeh, 1999, 2007; Xing, 2004; Zang, Sun, & Trivedi, 2000). The BDD methods are more recent and in most

* Corresponding author. Tel.: +39 3286730377.
E-mail address: stefano.sebastio@imtlucca.it (S. Sebastio).

http://dx.doi.org/10.1016/j.ejor.2014.04.035
0377-2217/

cases are more efficient for Boolean expression manipulation (Bryant, 1986). SDP methods were popular until recently and clearly outperform inclusion/exclusion methods (Locks, 1982).

However, for complex networked systems, such as aircraft current return networks, the above approaches are not adequate and the exact solution for reliability is not feasible. As shown in (Provan & Ball, 1983), the exact computation of two-terminal reliability is NP-hard (Ball, 1986). This has led to the use of Monte Carlo methods (Botev, L'Ecuyer, Rubino, Simard, & Tuffin, 2013; Cancela & El Khadiri, 2003; Cancela, LâEcuyer, Lee, Rubino, & Tuffin, 2010, 2012, 2013; Easton & Wong, 1980; Elperin, Gertsbakh, & Lomonosov, 1991; Fishman, 1986; Hui, Bean, Kraetzl, & Kroese, 2003; Kumamoto, Tanaka, Inoue, & Henley, 1980; Manzi, Labbe, Latouche, & Maffioli, 2001; Zenklusen & Laumanns, 2011) and computation of reliability upper and lower bounds. Recently, in order to reduce the number of samples required by Monte Carlo techniques, approximation approaches based on Machine Learning have been proposed (Rocco & Muselli, 2007; Srivaree-ratana & Smith, 1998; Yeh, Lin, Chung, & Chih, 2010). The inclusion–exclusion (IE) method (Hudson & Kapur, 1985) provides bounds on system reliability when minpaths are used and on system unreliability when mincuts are used. Esary-Proschan (EP) method (Esary & Proschan, 1970; Natvig & Eide, 1987) transforms graph to parallel combination of all minpaths to obtain reliability upper bound and a series combination of all mincuts to obtain the reliability lower bound. However, these methods need to enumerate all minpaths and all mincuts to compute the bounds, and these mincuts and minpaths grow exponentially with the size of the network and are hence infeasible for large scale networks. Most of the other reliability bounds computation techniques are based on finding a certain subset of minpaths/mincuts or series–parallel subgraphs (Elmallah & AboElFotoh, 2006; Galtier, Laugier, & Pons, 2005; Shanthikumar, 1988). Specifically, some of the computation techniques are based on the reliability polynomials (Ball & Provan, 1983; Colbourn, 1987), which is also referred to as subgraph counting method. By assuming all arcs in the network have the same probabilities of operation, system reliability will be simplified to be a polynomial expression of the arc operation probabilities; based on this polynomial the reliability bounds are computed. However, such a method is limited to the constraint that all the arcs have the same operation probabilities. Another well-known technique is Edge-Packing method (Aboelfotoh & Colbourn, 1989; Colbourn, 1988), also known as edge-disjoint method. A subset of disjoint minpaths and mincuts are selected to compute the lower bound and upper bound of system reliability respectively. Some techniques are based on Sum of Disjoint Products (SDP) method as in (Beichelt & Spross, 1989; Jane, Shen, & Laih, 2009). A few recent papers have developed decomposition techniques based on Binary Decision Diagram (BDD) (Niu & Shao, 2011) where the quality of the approximation can be improved according to a predetermined accuracy value. Graphs with special structures are also considered in (Cheng & Ibe, 1992; Feo & Johnson, 1990; Soh, Rai, & Trahan, 1994).

One problem associated with known bounds computation techniques is that they are not able to improve the reliability upper bound and lower bound by specifying the execution time. For example, for a specific reliability graph, the outputs of some bounds computation algorithms are fixed. If the bounds are not tight enough, they cannot be improved by a longer execution of the bounds computation algorithms. Many bounds computation algorithms can improve the reliability bounds by varying predetermined parameters for higher accuracy. However, this requires users know explicit definitions regarding accuracy parameters. In addition, the execution time can vary significantly for different input network models given the same accuracy, which makes it difficult to anticipate the real execution time by the users. Methods that exploit the Monte Carlo approach do allow the user to control the total computational effort but only in an indirect manner by choice of the sample size. In this paper, we propose an efficient algorithm using BDD representation for the reliability bounds calculations, which allows users to explicitly specify the execution time that is to be used. The advantage of such an approach is that it can not only search for important minpaths/mincuts that reduce the gap between the reliability upper and lower bounds, but also keep improving the bounds given longer execution time. The BDD representation of the bounds makes the minpath/mincut selection and bounds computation very fast and efficient. Heuristics are first used to find important minpaths/mincuts that can greatly increase/decrease the reliability lower/upper bound, then an exhaustive search is utilized to enumerate all minpaths/mincuts and compute their contribution to the current reliability lower/upper bound.

The main contribution of this paper is thus twofold: first, a novel bounds computation algorithm for two-terminal network reliability (or unreliability) problem is proposed. Second, this algorithm is designed to yield higher accuracy given more execution time.

This paper is organized as follows. Section 2 presents the mathematical background for the unreliability upper and lower bounds computation. Section 3.1 provides detailed procedures for unreliability upper bound computation including heuristic minpath search, exhaustive minpath search and minpath selection. Similarly, Section 3.2 describes detailed procedures for unreliability lower bound computation including heuristic mincut search, exhaustive mincut search and mincut selection. Numerical results of the proposed bounds computation approach are presented for several example graphs in Section 4 including a large real example reliability graph which contains over 4 trillion minpaths. Section 5 concludes this paper.

2. Basics of bounds computation

In reliability graph theory (Colbourn, 1987; Shooman, 2002) the two terminal reliability is defined as the probability that two nodes (s, t), respectively the source s and the destination t (also called terminal or sink), are able to communicate. The source node s does not have any incoming edges and the destination t does not have any outgoing edges. The graph model $G = (V, E)$ consists of a set V of nodes (also called vertices), a set E of directed edges and a relation of incidence that associates with each edge two nodes. Each edge is assigned a failure probability or its complement known as its reliability.

The reliability of a system (or a network) represented by a reliability graph can be expressed in terms of paths or cuts. A path is defined as a set of edges (components) so that if these edges are all up, the system is *up*; it is minimal (*minpath*) if it has no proper subpaths. While a cut is defined as a set of edges whose removal brings the system *down*; it is minimal (*mincut*) if it has no proper subcuts.

A system structure function ϕ is an indicator function defined on the status of all minpaths or mincuts (or equivalently on the status of the edges), whose output value is 1 if the system is up and 0 if the system is down.

2.1. Unreliability upper bound computation

The unreliability upper bound is computed by taking a subset of all the minpaths in the reliability graph. Assume the status of edge i is represented by Boolean variable e_i, where

$$e_i = \begin{cases} 1, & \text{edge } i \text{ is up} \\ 0, & \text{edge } i \text{ is down} \end{cases}$$