CIRRELT

Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

Network Reliability Evaluation and Optimization: Methods, Algorithms and Software Tools

Mohamed-Larbi Rebaiaia
Daoud Ait-Kadi

December 2013

CIRRELT-2013-79

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

Network Reliability Evaluation and Optimization: Methods, Algorithms and Software Tools

Mohamed-Larbi Rebaiaia*, Daoud Ait-Kadi

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Mechanical Engineering, 1065, avenue de la Médecine, Université Laval, Québec, Canada G1V 0A6

Abstract. Networks reliability analysis consists of determining the probability of correct operations of a system. Generally, the network is modeled as a connected graph and formally approached as a stochastic coherent binary system (SCBS). It represents the functioning of a system such as telecommunication and transportation networks, or simply hardware/software devices. Real networks model could be very larges and complexes which provoke the reliability evaluation to become intractable with traditional algorithms, so it is needed to invent more efficient techniques. Approaches used for determining the reliability of a system involve exact and approximate techniques. This paper introduces the problem of evaluating and optimizing the reliability of networks. It presents the most important methods, algorithms and software tools, and an interesting review of the literature. This work is based on our experiment to conducting projects in reliability engineering, essentially in telecommunication and transport networks.

Keywords. Reliability engineering techniques, networks, Birnbaum importance measure.

Acknowledgements. This work was partially supported by the National and Integrated Radiocommunition Network (Réseau national intégré de radiocommunication (RENIR)) under the supervision of Quebec Government, and the Natural Sciences and Engineering Research Council of Canada (NSERC). The authors would like to thank them and also to thank the FORAC and CIRRELT teams for putting at our disposal the necessary computers to carry out our research.

---

* Corresponding author: Mohamed-Larbi.Rebaiaia@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2013

1. Introduction

Nowadays, sensitive systems are rarely free of defaults. They are critical so they must be robust, reliable, secure, extensible, open, and must survive to crash. Network stochastic models have become widely used for mathematically representing complex systems such as telecommunication, transport, electricity services and manufacturing just to name few of them. In many such networks, the physical problem has source and sink nodes, and a network of links connecting all the nodes.

Initially reliability systems specification and evaluation were designed in cases where failure of such systems could have caused massive damage or loss of human life. Examples include special shuttle (e.g. crash of Challenger in 1986), aircraft systems, nuclear reactor control systems, and defense and control commands. Now, it has been generalized to major sensitive industry facilities as well as products sold to the general public costumers.

The performance of such networks can be measured by evaluating the reliability index, which represents the probability that a network operates. At the beginning of the lifecycle of a product development, the reliability can be used as a part of system design procedure. The performance of a design can formally be studied by changing the topology of a network by adding, removing or replacing some components for augmenting its reliability or for satisfying certain specifications. Particular solution for designing systems is to use stochastic networks. A stochastic network G models a physical system in which each edge and/or each node can fail statistically independently with a number representing the non-failure probability such that failures of any network elements do not influences each other. In real world network, sites correspond to nodes and links to uni- or bi-directional edges. The network reliability analysis problem consists of measuring the global probability value given failure/operation probabilities for edges/nodes. The classical network reliability problem supposes the existence of a surviving subgraph which continues to be connected with the rest of the network enunciated as the probability $Pr\{$there exist operating paths from a node s to each node in a subset $\boldsymbol{K}\}$ or more simply the probability $Pr\{$there exist at least an operating path from a source node s to a sink node t in the network$\}$ where $\boldsymbol{K}$ is a subset of nodes belonging to the network. A typical example for the case of a national public radiocommunication network is to answer in case a critical situation likes a big fire: “what is the probability that any connection links between regional police department and fire stations falls/up”. To compute the reliability of a network, most of reliability methods proceed into two phases. In the first phase a Boolean expression is obtained based on its topology; while in the second phase a numerical value is measured using any one of the algorithms presented in the literature. In case the measured reliability approaches the value 1, it is concluded that the system is extremely reliable, otherwise, when it tends to 0, the system is very bad and we should expect it to fail regularly down.

The IEC 60050-191 defines the international vocabulary of the "reliability" of a system; as "the probability of performing its function all over a time interval [0, t]". Formally, reliability is the conditional probability at a given confidence level that a system will perform its intended function properly without failure and satisfy specified performance requirements during a given time interval [0, t] when used in the manner and for the purpose intended while operating under the specified application and operation environment stress levels (Johnson and Malek, 1988). In addition, when time dependent calculations are performed, only availability, noted A(t), or unavailability, noted U(t) calculations are possible without approximations. Availability could be another expression of the reliability under certain conditions. In the literature, authors have given some significance to the term availability. "Instantaneous availability, A(t), is the probability that a system is performing properly at time t and is equal to reliability for non-repairable systems and steady-state availability, which is the probability that a system will be operational at any random point of time and is expressed as the expected fraction of time a system is operational during the period it is required to be operational are such availabilities taken from a general framework of availability definitions, say Johnson and Malek (1988). "Maintainability is the probability of successfully performing and completing a specified corrective maintenance action within a prescribed period of time at a desired confidence level with specified manpower, skill levels, test equipment, technical data, operating and maintenance documentation, and maintenance support organizations and facilities, and under specific environmental conditions" as defined by Kececioglu (1986) in Johnson and Malek (1988).

Availability calculation of networks gives a more precise solution and is used as a performance indicator (e.g. wireless communication system). It could be an important aspect of many service level agreements (SLA) and can be a unique selling point in the service description operators. Network availability value represents the percentage of time that the communication between two points of the network should be possible. However, it does not indicate whether the quality of this communication is satisfactory for a particular service (e.g. surfing the web, or Voice-over-IP), or if it has been affected by factors such as delay or loss. A network error can lead to congestion, causing quality to deteriorate to a level where the service is no longer possible, despite the fact that the connection between any two services still connected. Service availability indicates the quality required to provide particular service users finals. The service availability also depends on the requirements of the service in question; the results for Voice-over-IP (VoP) for example will differ from those for surfing the web. Dynamic routing in IP, MPLS and ATM, among others, is a complicating factor again. Defects in a section of a network can cause worst services and reduced quality of service in other sections, which affects the reliability of the system infrastructure and the availability of the service. The value of the availability of a network is as important as knowing its reliability. They are intrinsically linked and can be combined for determining the reliability of the whole system. In a radiocommunication network for example, there are two major subsystems; the microwave system and the transmitter/receiver system. Each one of them has its infrastructure. Between any two antennas (transmitter/receiver) the monitoring microwave system can detect the signal fading and thus it determine its duration, and at each elapsed time, the system records the sum of the signal loss which is used to evaluate the availability of the entire microwave transmission/reception system. The performance of the transmission/reception system is determined using the combination of the reliability of the hardware and the software parts of the network.

Network reliability analysis problem has been the subject of many scientific productions. The literature contains numerous of such methods (Rebaiaia, 2011; Hardi et. al., 2007; Kuo et al., Locks and Wilson, 1992, El-Ghanim, 1999). They consists of evaluating the 2-terminal reliability of networks noted R_{s-t} (s : source; t: target), $\boldsymbol{K}$ and all-terminal. General theory, has discussed extensively two techniques; exact and approximate methods (Locks, 1980; Corine, 1993). They are considered as enumerative techniques such as Abraham (Abraham, 1979) and Heidtmann (1989) algorithms, inclusion/exclusion (Lin et al, 1976) formulation, network decomposition (Rosenthal and Frisque, 1977), techniques using Boolean algebra (Fratta and Montanari, 1973), set theory (Aggarwal et al., 1975), and other methods employing the concept of minimal pathsets/cutsets (MPS/MCS). Determining MCS for example is essential to evaluate the reliability indices and to investigate the different scenarios to find the redundant components which could be added to improve the load point reliability. Enumerating all MCS may be a preferable way if the number of paths is too huge to be practically enumerated than the number of cuts (Appendix A presents MCS-based procedure for determining the set of MCS (see. Rebaiaia and Ait-Kadi, 2012)). Examples of this kind of preferences is the 2x100 lattice which has 299 paths and just 10000 cuts (Kuo et al., 2007; Rebaiaia and Ait-Kadi, 2013), and complete network with 10 nodes from which it can be generated 109601 minimal paths and 256 cuts. In existing algorithms (Jasmon and Kai, 1985; Yan et al., 1994; Rebaiaia et al., 2012), minimal paths are deduced from the graph using simple and systematic recursive algorithms that guarantee the generated paths set to be minimal. The enumeration of MCS is more problematic because they need advanced mathematics, set theory and matrices manipulation.

Enumeration appears to be the most computationally efficient. An initiative of solution has been proposed in Shier and Whited, 1985. In the paper of Locks and Wilson, 1992, the authors presented a method for generating MPS directly from MCS, or vice-versa for s-coherent systems.

It starts with the inversion of the reliability expression accomplished by a recursive method combining a 2-step application of De-Morgan's theorems. Yan et al., 1994 presented a recursive labelling algorithm for determining all MCS in a directed network, using an approach adapted from dynamic programming algorithms. The algorithm produces all MCS, and uses comparison logic to eliminate any redundant cutsets (Rebaiaia, 2011). This algorithm is an enumeration technique used to improve the computational efficiency and space requirements of the algorithm. Jasmon and Kai, 1985 uses an algorithm which proceeds by deducting first, the link cutsets from node cutsets and, second the basic minimal paths using network decomposition. So, in addition to the enumeration of cutsets directly, it is possible to obtain them from the inversion of minimal paths (Rebaiaia and Ait-Kadi, 2012). In such topic, one of the best algorithms is due to Al-Ghanim, 1999. It is based on a heuristic programming algorithm to generate all MPS. The algorithm proceeds by creating a path, then iterates back from an explored node in the current path using unexplored nodes until to reach the source node. Recently, Yeh, 2007 presented a simple algorithm for finding all MPS between the source and the sink nodes. It is based on the universal generating function. More recently, Rebaiaia and Ait-Kadi (2012) proposed an elegant and fast algorithm to enumerate MPS using a modified DFS technique. The procedure uses each discovered path to generate new MPS from sub-paths. The above procedure is repeated until all MPS are found. The algorithm didn’t at all produce any redundant MPS. More, they extended their work with theoretical proofs and the usage of sophisticated techniques for dynamic data structure manipulation of complex networks.

A large number of real systems cannot be easily modeled using ordinary techniques. This is do because they cannot be separated into small independent subsystems and generated a hug of event states representation. These events are not "memory less" and event times may differ by several orders of magnitude conducting to prohibitively large information to be handled.

Computing such measure has been proved to be NP-hard. Rosenthal (1974; 1977), was the first to show that the recognition problem of determining if a network contains reliable Steiner Tree of given cardinality is NP-Complete. In practical cases, all known methods for calculating the connection probability for two nodes (network reliability algorithm) take time exponential in the size of the network, it belong to the class of NP-complex problems, and in the theory they have been classified as #P-complete (read. numbered P-complete) by Valiant (Valian, 1979) and demonstrated with preciseness in (Provan and Ball, 1983). Following these works, Badlaender and Wolle (2004), consider a list of such problems and prove their #P-completeness.

When addressing the problem of the reliability network evaluation for a given system under a known topological structure, one is quickly confronted with two issues requiring answers beforehand, which are: (1) is the reliability of all components available? (2) What are the appropriate tools for computing the reliability in accordance to network dimensions?

This paper uses a stochastic network model in which each link/node can be either of two states: operative or failed with a probability P. A link can be directed or undirected. The state of a link/node is a random event that is statistically independent of the state of any other link/node.

This work try to present the state of the art of network reliability evaluation and optimization in which overview some methods, algorithms and tools are surveyed, a literature review and the impact of these reliability methods to resolve big reliability problems is discussed.

The following sections are introduced as follows:

Section 2 introduces some important definitions. Section 3 proposes a concise state-of-the-art. A review of the literature is presented in section 4. Section 5 introduces some of the well-known tools and software tools for computing the reliability. At last, section 6 ends this paper and Appendices detail well-known methods and algorithms.

2. BACKGROUND PRELIMINARIES

2.1 Definitions

2.1.1. Stochastic graph

A stochastic graph $G = (V, E)$ is a finite set V of nodes and a finite set E of incidence relations on the nodes called edges. The edges are considered as transferring a commodity between nodes with a probability p. They may be directed or undirected and are weighted by their existence probabilities. The graph in such case, models a physical network, which represents a linked set of components providing services. In this work, other terms are used to define stochastic graphs such as reliability model or simply network. They give exactly the same meaning.

2.1.2. Subgraph and partial graph

A subgraph of a given graph $G = (V, E)$ is a graph $G' = (V', E')$ such that $V' \subset V$ and $E' = (V' \times V') \cap E$.

A partial graph of a given graph $G = (V, E)$ is a graph $G'' = (V, E'')$ such that $E'' \subset E$.

2.1.3. Graph state, associated probability and associated partial graph

As each element of the network during system operations may be up (operating) or be down (failure), thus a state Boolean cardinality $|element\ states| = |(up, down)| = 2$. It is also well-recognized in Graphs theory that there are 2^{n+m}possible states for a network, with $n = |V|$ and $m = |E|$.

2.1.4. Network Reliability

Different notions of network reliability have been introduced in the literature, i.e, deterministic and stochastic (Frank and Friech, 1971), (Hwang, Tillman and Lee, 1981). Networks can be defined as a physical or organisational infrastructure that can be modeled as a graph composed of nodes and links (directed or undirected) in which each edge has associated value corresponding to the probability that such component is functioning or not. Because each edge and each node can fail with a probability value, the reliability of a network $G = (V, E)$ is defined as the probability that the system (network) will perform its intended function without failure over a given period of time and under specific conditions.

2.1.5. Paths, chains, connected graph, cuts, minimal paths, minimal cuts

1- A path is a chain $\mu = (x_1, \ldots, x_q)$ in which the terminal endpoint of arc μ_i is the initial endpoint of arc μ_{i+1} for all $i < q$. Hence, we often write $\mu = (x_1, x_{k+1})$, where *k* is the number of edges and *k* is considered as the length of the path (chain).

2- A graph *G* is said to be *connected* if between any two nodes $x, y \in V$ there exists a chain $\mu = (x, y)$.

3- A path *P* in a graph *G* is said to be a *1*-path if any two nodes $x, y \in V$ they are linked by only one edge. Any path in a graph *G* which is a *1*-path is said to be a branch.

4- Minpath (MP): A subset of a path with minimal number of elements that still make the system functioning.

5- Minpath set (MPS): The set of all minpaths of a network.

6- A cut *C* is a set of elements such that if all of them are true then the system is failed.

7- Mincut: A subset of a cut with minimal number of components that still make the system fail.

8- Mincut set (MCS): The set of all mincuts.

Let x_i be the state component and x the state vector, they can define what follows:

- $x_i(t) = \begin{cases} 1 & if \text{ the component is up at time } t \\ 0 & otherwise \end{cases}$
- $x = (x_1, x_2, \ldots, x_m)$ a state vector of the system S of order *m* such that $x \in \Omega_i = \{0,1\}^m$ the state space of the system.

The system is then represented by its structure function $\Phi\colon \Omega \to \{0,1\}$ defined as follows :

$$\Phi(x) = \begin{cases} 1 & if \text{ the system is up when the state vector is } x \\ 0 & if \text{ the system is down when the state vector is } x \end{cases}$$

After specifying the structure function Φ, a probabilistic structure is defined. The usual framework is to assume that the state of the i^{th} component is a random binary variable and that the state vectors are independent..

If a system contains P minpath set $P_1, P_2, \dots, P_P$ and C mincut set $C_1, C_2, \dots, C_C$ its structure function can be represented by :

$$\Phi(x) = \max_{1 \le j \le P} \min_{i \in P_j} x_i = \min_{1 \le j \le C} \max_{i \in C_j} x_i$$

The basic expression of the reliability R of a network G is presented in the following form :

$$R(G) = Probability(G \text{ non} - \text{defaulting on the time interval } [0, t[) = Pr\{\Phi(X) = 1\}$$

$$= E\{\Phi(X)\} = \sum_{X \in \Omega_i} \Phi(X) Pr\{X = x\}$$

where $E\{\Phi(X)\}$ is the mathematical expectation, and $p_i = Pr\{X_i = 1\}$ and $q_i = Pr\{X_i = 0\} = 1 - p_i$.

Note that p_i is called elementary reliability. We can observe that this is a static problem, because time is not explicitly used in the analysis.

However in case the network where nodes are also prone to failure it is possible to replace nodes by arcs and thus we return to a network without imperfect nodes. In such case the reliability is expressed by:

$$R(G) = \prod_{v_j \in K} p_i \,.\, R(G')$$

where G' is the graph G with perfect terminal nodes and ***K*** is the set of perfect nodes of the network.

3. State-of-the-Art

The primary network reliability consists of evaluating three measures for probabilistic networks. They are:

- *Two*- terminal. Probability that communication is enabled between a source *s* and a destination *t*,
- *K*-terminal. Probability that every node in *K* (a subset of nodes) can communicate with every other node in *K*, and, It consists of computing the reliability measure as : Pr{there exist operating paths from ***s*** to node in *K*},

- *All*-terminal. Probability that every node can communicate with every other nodes in the network.

Several methods to solve the problem of reliability evaluation have been proposed in the literature. There are those that have been implemented and those who remained in the state of simple theories. One major class of exact methods is based on topological methods such as those based on the factoring theorem, the reduction or decomposition (Rosenthal, 1977), (Satyanarayana, 1982), (Wood, 1985). They are more recent and more effective. The reduction operations reduce the size of the network while preserving reliability. They are used to evaluate the reliability of particular networks such as series-parallel graphs in polynomial time (Corinne Lucet, 1993). Factoring methods proceeds by decomposing the network into smaller networks from which the network reliability is deduced by composing small parts reliabilities. For the most complex networks, reduction and decomposition must be combined with the factoring methods to give a very effective tool. A second major class is called enumeration methods. They are two-fold: state enumeration methods and cut/path set enumeration methods. The most basic state based method is complete state enumeration, requiring the generation of 2^m states of a network with m arcs. Cut/path set require enumerating all minimal cut/path set in the graph. These methods involve a substantial research because they are classics in reliability of systems. In these methods, minimal cuts/paths are first determined, and the calculation of reliability is based on them and consists of making the corresponding events disjoint. Three well-known algorithms are used for making them disjoint. They are: Abraham (Abraham, 1979), Heidtmann (Heidtmann, 1989) and recursive-exclusive-exclusion (IE) also known as Poincaré's theorem algorithms (Riordan, 1985) or simply inclusive-exclusive method. Other similar techniques have been published in many papers, just to name few, they are: GKG (Veeraraghavan, (1988); Veeraraghavan, and Trivedi, 1991), and CAREL (Soh and Rai (1991)). It is simple to note that the reliability evaluation problem is NP-complete, the generation of an exact solution is very problematic (Bal, 1986;Valian, 1979).

The desire for fast computation with great accuracy have led to a varied of clever techniques for estimating networks reliability (Colbourn and Harms, 1985). There are two main investigation areas: the estimation of reliability by Monte Carlo sampling techniques, and the bounding of reliability. In the first, simulation consists of generating independent samples and estimating the unknown parameter corresponding to the reliability by an unbiased estimator along with the confidence intervals for the estimate. The relevance of this estimate is related to the number of samples, and their generation. If this number is high, the cost of simulation methods approach or exceed those of the exact methods. In the second, bounding methods attempt to produce absolute upper and lower bounds on the reliability measures from the algebraic structure of the problem.

Other intuitive methods use cutsets derivatives instead of pathsets, because in any networks of *m* edges and *n* nodes, the order of the number of cutsets is 2^{n-2} and the order of the number of paths is 2^{m-n+2}, and for a class of networks having nodes of average degree greater than four, as $m > 2n$ and $2^{m-n+2} > 2^{n-2}$ thus such networks have a larger number of paths than cutsets (Rubino, 1988). Example of such networks are fully dense complete networks that for n = 10, the number of minimal paths is equal to 109601 and minimal cuts is equal to 256 (Rebaiaia and Ait Kadi, 2011). Also for the 2x100 lattice network, minimal paths number is equal to 2^{99} and minimal cutsets is equal to 10000. Yet, the use of cutsets is much more advantageous than pathsets for the case of dense networks.

Recently, Binary Decision Diagrams (BDD) and their multiple version have been developed as a new formalisms to minimize the size of networks reliability expressions and for evaluating the reliability of networks. They have been overused in different domains, such as hardware/software systems validation and verification mathematical models. They are called verifiers or simply, model-checking. Coudert & Madre (1992) and Rauzy (1993) are the first to introduce BDDs for evaluating networks reliability and since that date a lot of algorithms based on BDDs have been implemented in different tools. Unhappily, realistic tools still away from giving solutions in a reasonable amount of duration-time because the problem is NP-hard (Bal 1980, 1986). So, it still open and can be announced as: "What is the efficient method that can generates a solution for evaluating and thus optimizing the reliability in case of large networks?" The problem is very complex and any method cannot by itself give adequate solutions.

In the following, we propose several algorithms to cover this problematic. They are presented in the following literature review:

4. Literature Review

4.1. Minimal Paths set (MPS), Minimal Cuts set (MCS), Sum of Disjoint Products (SDP)

Several algorithms have been developed to enumerate MPS/MCS, most of them require advanced mathematics or can only be applied to either direct or undirect graphs and alternative solutions have been proposed by different authors (Soh and Rai, 1993; Jasmon and Kai, 1985; Yeh, 2009; Al-Ghanim, 1999). Some are specific to the determination of MCS (Patvardhan and Prasad, 1996, Lin *et al.*,2003) and others to MPS (Yeh, 2007; Jasmon and Kai, 1985). Some MCS methods are highly related to the MPS because they are derived from them. Shier and Whited (1985), have proposed a technique for generating the minimal cuts from the minimal paths, or vice-versa. The process is a recursive 2-stage expansion based upon De Morgan's theorems and Quine-type minimization. Rebaiaia (2011) has designed an elegant algorithm that generates MCS from BDD.

Awosope and Akinbulire (1991), present a simple method based on input-reduction programming technique that automates the deduction of MPS/MCS. The method has been applied to a power-system structure in the form of power-arms (termination bus bars, branch and protective devices) is the only initial input data needed. The authors argue that the results obtained, in terms of minimal paths and minimal cut-set are similar of those of the literature. In Fotuhi *et al*. (2004), a method called "Path Tracing Algorithm" has been introduced. It can handle both simple and complex networks, and considers both directional and bi-directional branches. As a demonstrating proof the authors explained the procedure using a bridge-network and illustrated by application to a more tedious system. Sandkar et al (1991) propose an algorithm to obtain all path sets that give the required flow at the DC terminal in a power system. By multiplying these path sets, using Boolean algebra, all minimal cut sets that do not transmit the required flow are obtained. From these minimal cut sets, the expression for the probability of failure of transmission of required flow at the DC terminal can be obtained.

Buzacott noted in (Buzacott, 1980) that it is important to well determine the order of the cuts before using the disjoint products version of the cut-based methods. He proposed that the usual approach is to order them in a descending order in terms of the number of arcs in the cut. In Veeraraghavan and Trivedi (1991), authors describe an efficient boolean algebraic algorithm to compute the probability of a union of non-disjoint sets. The algorithm uses the concept of multiple variable inversions.

The paper of Mishra and Chaturvedi (2008) presents an algorithm to enumerate global and 2-terminal cutsets for directed networks. Several benchmark networks have been used to evaluate directed networks with sum-of-disjoint-product (SDP) based multi-variable inversion (MVI) technique without any requirement of complex mathematics or graph-theory concepts.

Author in Yeh (2009) introduces a simple algorithm for finding all MPs before calculating the binary-state network reliability between the source and the sink nodes called one-to-one reliability. It is based on the universal generating function method (UGFM) and a generalized composition operator. The computational complexity of the proposed algorithm is also detailed and an example illustrates the generation of all minimal paths.

Lin and Donaghey (1993), describe an approach using Monte Carlo simulation to generate minimal path sets by tracing through the system from the input to the output components of the reliability diagram in a random manner. The frequencies of distribution of the minimal cut sets are determined during the simulation. Also, the paper of Malinowski (2010) presents a new efficient method of enumerating all minimal MPSs connecting selected nodes in a mesh-structured network. This task is fulfilled in two steps. In the first step, the algorithm tries to find all loop-free paths. In the second step, a recursive procedure gradually merges the paths belonging to different paths sets. The authors argue the efficiency of this method by a series of tests. The problem of this method is due to the backtracking procedure used to deduce all spanning trees. It has shown that it grows exponentially.

The paper (Balan, 2003) extends the work of Abraham and Heidtmann by introducing a new preprocessing strategy which works well for SDP algorithms with single-variable inversion (SVI). The authors have observed that optimal preprocessing for SVI-SDP can be different from optimal preprocessing for SDP algorithms which use multiple-variable inversion; one reason for this is that MVI-SDP algorithms handle disjoint minpaths much more effectively than SVI-SDP algorithms do. Both kinds of SDP algorithms profit from prior reduction of elements and of subsystems ---which are in parallel or in series. In Soh and Rai (1991), experimental results are presented showing the number of disjoint products and computer time involved in generating sum of disjoint product (SDP) terms. The authors have considered 19 benchmark networks containing paths (cuts) varying from 4 (4) to 780 (7376). Several SDP reviewed techniques are generalized into three propositions to find their inherent merits and demerits. An efficient SDP technique is, then, utilized to run input files of paths/cuts preprocessed using (1) cardinality, (2) lexicographic, and (3) Hamming distance ordering methods and their combinations.

In the study of Yeh (2005), a new algorithm based on some intuitive properties that characterize the structure of MPs, and the relationships between MPs and subpaths are developed to improved SDP techniques. The proposed algorithm is not only easier to understand and implement, but is also better than the existing best-known SDP based algorithm. It is based on disjointed algebra and BDD algorithm. Yufang (2010) introduces an improved and simplified algorithm used to solute disjointed MPS. According to the different path length of MPS, he proposes two procedures to disjoint MPSs. He processed for the MP whose length is n-1, keep the original arcs unchanged and add the inversion of those arcs which are not included in the network and get the disjointed result; disjoin the left MP set based on BDD algorithm and realize it through programming. It is shown that this method is efficient and accurate. It provides a new approach for reliability analysis of large scale network system. The application of Binary Decision Diagrams (BDDs) as an efficient approach for the minimization of Disjoint Sums-of-Products (DSOPs) is discussed in Fey and Drechler (2002). The authors tell that the use of BDDs has the advantage of an implicit representation of terms. Due to this scheme, the algorithm is faster than techniques working on explicit representations and the application to large circuits that could not be handled so far becomes possible. They showed that the results with respect to the size of the resulting DSOP are as good or better as those of the other techniques. Locks (1987) describes a minimizing version of the Abraham SDP algorithm, called the Abraham-Locks-Revised (ALR) method, as an improved technique for obtaining a disjoint system-reliability formula. The principal changes are: (1) Boolean minimization and rapid inversion are substituted for time-consuming search operations of the inner loop. (2) Paths and terms are ordered both according to size and alphanumerically. ALR reduces the computing cost and data processing effort required to generate the disjoint system formula compared to the seminal Abraham paper, and obtains a shorter formula than any other known SDP method. Very substantial savings are achieved in processing large paths of complex networks.

More recently, Rebaiaia and Ait-Kadi (2012), propose an elegant and fast algorithm to enumerate MPSs using a modified DFS technique (Tarjan (1972)). The procedure uses each discovered path to generate new MPS from sub-paths. The above procedure is repeated until all MPSs are found. The algorithm didn't at all produce any redundant MPS. More, they extended their work with theoretical proofs and the usage of sophisticated techniques for dynamic data structures manipulation of complex networks.

4.2. Reduction and factoring based methods

Factoring algorithm decompositions have been proved to be effective in case the networks are irreducible. They proceed on perfect and imperfect networks (Rebaiaia and Ait-Kadi 2013; Rebaiaia *et al*., 2009; Theologou and Carlier, 1991; Simard; 1996). They can be applied for direct (Wang and Zhang, 1997) and undirected network (Choi and Jun, 1994). They have been more worked intensively for the undirected graphs (Satyanarayana and Wood, 1985; Wood, 1982; 1986; Satynarayana, 1980, 1982; Satynarayana and Wood, 1985, and Choi and Jun, 1995).

Moskovitz (1958) is one of the pioneers who used factoring theorem for undirected networks. It was introduced informally first for minimizing electronic circuits by Moore and Shannon (1956). The principle of factoring theorem is exactly the same as those of Moore and Shannon and the well-known Bayes theorem. All these theorems are a version of the probability total theorem, also known as the conditional probability theorem. Readers are invited to consult books on probability theory.

A number of other papers reviews factoring theorem in the beginning of 1970s (Misra ,1970; Murchland, 1973; Rosenthal, 1974, 1977; Nakazawa, 1976. The application of factoring theorem has been cited in (Ball, 1986; Valian, 1977; Chang, 1981; Satyanarayana and Chang, 1983, and Johnson, 1982) as a worst case computational complexity and the optimality of classes are NP-hard. To reduce the complexity of this theorem, Satyanarayana (1985) proposes a unified formula for analysis some category of reliability networks also based on Inclusion-Exclusion Poincare's theorem. The idea is to derive a formula from Poincaré reliability expression that involves the non-cancelling terms. He established two theorems for that. In the first one, he announces that the domination of a cyclic *K*-graph is always zero and in second, that, the domination of an acyclic *K*-graph with *m* links and *n* vertices is $(-1)^{m-n+1}$. He gives a demonstrating example using 4-nodes bridge network, and details all the *K*-trees of the network. Another similar work (Satyanarayana and Wood (1985)) introduces a new scheme to derive polygon-to chains by reduction on the structure of the network. They proposed seven polygons-to chains reductions (Rebaiaia and Ait-Kaci, 2009, 2011, 2012, 2013). A polygon-to chain reductions is a successive application of factoring theorem on polygons, which must exist as substructures in the graph. Chang (1981), and Satyanarayana and Chang (1983) use a graph invariant $D(G_K)$, called domination, to analyze the complexity of computing *K*-terminal reliability using a factoring algorithm with series and parallel reductions. $D(G_K)$ is equivalent to the number of certain rooted acyclic orientations of G (see Tarjan, 1972). Johnson (1982) discusses other relationships. Satyanarayana (1980) introduces the concept of minimum domination noted mathematically by $L(G) = \min D(G)$ $(D(G)\text{: the degree of the graphe})$ and shows that the lack of the factoring theorem is that directed networks can only be handled in a limited way ((Agrawal, 1974; Nakazawa, 1976). Until now, it has discussed the ideal case where network are perfect, in other words, networks subjected to only edges' failure. The problem is so difficult and become more complex if one supposes that also nodes could fail randomly. Such networks are called imperfect networks. Unhappily just few works have been dedicated to the problem. One of the precursors of doing research in this direction is Theologou and Carlier (1991). They have showed using a clever artefact that it is possible to apply factoring theorem with some minor modifications. The problem still opened until a demonstration done by Simard under the supervision of Simard and Ait-Kadi (1996) concerning the good way to reduce polygon-to chains in case that both nodes may fail as well as edges. In Rebaiaia *et al*. (2011) and Rebaiaia and Ait-Kadi (2013), a table similar to those of Satyanarayana and Wood (1985) has been established with all the transformation formulas for seven polygons-to chain reductions, and theorem and demonstration formulated. Resende (1986) discusses the design and implementation of PolyChain, a program for reliability evaluation of undirected networks of a special structure using polygon-to-chain reductions. The author presents a small problem and tested it, illustrating the code's output. Theologou (1990) in her dissertation proposes another representation of data structures to optimize the execution time and space memory.

4.3. Binary Decision Diagrams Methods

Bryant (1986) was the first to use the work of Akers (Akers, 1978) on the application of binary decision diagrams for symbolic verification of integrated circuits. BDDs have been investigated and implemented first by Bryant (1986, 1992). The problem with BDD representation despite their effectiveness is that, their exponential growing size due to a wrong order declaration between variables. Ruddell (1993) first used an algorithm based on dynamic programming techniques to reduce the size of the BDD and Bollig *et al.* (1996) demonstrate that improving the variable ordering of OBDD is NP-Complete. Coudert and Madre (1992) and Rauzy (1993) applied first, BDDs for evaluating networks reliability. Kuo et al. (1999) used a methodology to evaluate the terminal-pair reliability, based on edge expansion diagrams using OBDD. The algorithm proceeds by traversing the network with diagram-based edge expansion and the reliability is obtained by directly evaluating it recursively on this OBDD. A simple and systematic recursive algorithm has been introduced by *Lin et al.* (2003) that guarantees the generated MCS with ease. This algorithm is combined with OBDD to calculate the reliability of networks. Chang *et al.* (2004) propose an efficient approach based on OBDD to evaluate the reliability of a non-repairable systems and the availability of a repairable system with imperfect fault-coverage mechanisms. Various approaches have been used for the analysis of multi-state systems; examples the BDD-based method (Xing and Dugan, 2002-a), (Tangand and Dugan, 2006). (Xing and Dugan, 2002-b). Zang et al. (1999), also proposed efficient approaches based on multi-state BDD (MBDD), and Phased-missions systems analysed using BDD.

Large techniques for designing, modeling and computing the reliability of systems are presented in the following table. Perhaps the most important reliability solution is composed by a set of techniques taken from this table.

Reliability engineering techniques (source : http://www.barringer1.com/nov07prb.htm)				
Accelerated Life Testing (ALT)	Design Review	Highly Accelerated Life Test (HALT)	Monte Carlo	Reliability Engineering
Availability	Effectiveness	Highly Accelerated Stress Screen (HASS)	Normal Distribution	Reliability Growth
Bathtub Curves	Electronic Components	Life Cycle Cost	Overall Equipment Effectiveness (OEE)	Reliability Policies
Block Diagram Models	Environmental Stress Screen (ESS)	Life Units	Pareto Distribution	Reliability Testing
Capability	Events/Incidents	Load-Strength	Poisson Distribution	Simultaneous Testing
Configuration Control	Exponential	Lognormal	Probability Plots	Software Reliability
Contract For Reliability	Failure	Maintainability	Process Reliability	Sudden Death Testing
Cost Of Unreliability	Failure Forecast	Maintenance	Quality Function Deployment QFD	Total Productive Maintenance (TPM)
Critical Items List	Failure Rates	Maintenance Engineering	Reliability	Weibayes Estimates
Data	Fault Tree Analysis	Management's Role	Reliability Audits	Weibull Analysis
Decision Trees	FMEA	Mean Time	RBDs	Weibull Corrective Action
Dependability	FRACAS Systems	Mechanical Component Interactions	Reliability-Centered Maintenance	Weibull Database

4.4. Reliability optimization

Many papers have been published to solve the reliability optimization problem. A reliability optimization problem is a mathematical model formulated for optimizing the reliability of a system under a set of constraints. One early model cited in the literature is called "the redundancy" problem. It consists of optimizing the reliability of a series-Parallel system. Such system has several stages in series and each stage has parallel redundancy to improve the reliability (Woodhouse (1972), Misra (1972), Coit and Smith (1996)., Kuo *et al.* (2000), Misra (1991)). "Network designs can be based on multiple choices of redundant configurations, and different available components which can be used to form links. The reliability of a network system can be improved through redundancy allocation, or for a fixed network topology, by selection of highly reliable links between node pairs, yet with limited overall budgets and other constraints as well. The choice of a preferred network system design requires the estimation of its reliability" say Marsegurra et al., 2005.

There are several approaches that provide solutions to such problems. For example, Fyffe *et al*. (1968) use dynamic programming while limiting the problem by considering a single type of component available for each subsystem. Misra (1972) shows a new way to transform the model of constrained optimization to a saddle point problem by using Lagrange multipliers. Newton's method is used to solve the nonlinear algebraic equations and uses the Maximum principle to arrive at the optimal decision. Similar solutions have been published in Moskowitz (1958) and Misra (1970). Osaki (1972) consider a standby – redundant model of two units and applied their technique to maintain the system with high reliability using the preventive maintenance policy. Chern shows that the problem of redundancy allocation is NP-hard (Chern, 1992).To tackle the problem, others models have been used as genetic programming (Coit and Smith, 1996), Zizler and Thiele (1999), Zitzler (1999) and Srinivas and Deb (1993), heuristics (Coit and Konak, 2006), (Coit and Wattanapongsakorn, 2004) and Ant Colony (Kuo *et al.*, 2000). For a useful bibliography the reader is referred to Kuo *et al.* (2000). Recently, Marsegurra et al. (2005) formulate the redundancy problem as a multiple optimization approach which maximizes the network reliability and minimizing its associated variance when component types with uncertain reliability and redundancy levels are the decision variables. Coit *et al.* (2006) used multi-objective programming for the series-parallel systems (figure 10) and makes some transformations to translate the problem that is initially non-linear to a linear model whose solution is accessible using CPLEX tool. Also Coit and Konak (2006) present a multi-criteria approach for optimizing the system where the components reliability are estimated with uncertainty. The problem is to maximize the estimated reliability of the system while minimizing the variance associated with it. Another interesting work was published by Ha *et al.* (2006). It solves the problem of optimizing the redundancy by applying heuristics, called as tree-heuristic. This heuristic allows multiple local optimal solutions. Rebaiaia and Ait-Kadi (2010) proposed a solution based on BDDs representation and composition. This approach proceeds first by generating the BDD corresponding to the objective function. A second BDD representing one constraint is associated to the first one. A third BDD is then generated by the composition of earlier BDDs. The algorithm iterates until to cover all the constraints. From the last BDD a solution is generated and the reliability is computed.

One other alternative solution to optimizing the reliability of networks is precisely to act on components which can bring some improvements to the reliability, practically like the redundancy allocation process but the solution is presented differently. It is known that some components are more important than others to the functioning of the system in term of their contribution to the whole system. They are termed measure by component importance or redundancy importance or simply structural importance interchangeable with Birnbaum's importance index (Birnbaum, 1969). There are other importance indices that have been intensively studied in the literature as Vesel-Fussel importance measure (Meng, 1996) , (Vasely, 1970) and so. Despite the fact that they are interesting source of information; the problem with such importance indices is they cannot be determined automatically in case the network is complex.

5. Software Tools for reliability evaluation

Several companies have produced software packages for evaluating and optimizing reliability of ordinary/sensitive systems. The objective was to either to use the software for their purposes or to attempt to sell it. One of the interesting tools that has been commercialized last years is MechRel designed by the Carderock Division of the Naval Surface Warfare Center (CDNSWC) and was sponsored by the Office of Naval Technology under the Logistics Exploratory Development Program. The design evaluation procedures supported by this tool, consider failure rate data, materiel properties, operating environment and critical failure modes at the component part level. This software tool can be downloaded free of charge by visiting the CDNSWC website. The program procedures participate to evaluate designs for reliability in the early stages of development, to provide management emphasis on reliability with standardized evaluation procedure, to provide an early estimate of potential spare parts requirements, to quantity critical failure modes, to evaluate the reliability for performing trade off studies and selecting an optimum design concept and to design accelerated testing for verification of the reliability performance.

Rockwell International Corporation developed in 1965 a FORTRAN software package with the acronym SCOPE (System for Computing Operational Probability Equations). SCOPE provides an exact system reliability function of the component reliabilities. It is based on either the minimal paths or else the minimal cuts. This function is derived by the method of inclusion-exclusion technique as discussed previously. The SCOPE software uses mathematical theory given in Locks (1971), including some comparisons to the Esary-Proschan bounds (Esary and Proschan, 1963) that can be used to build effective transformation that can be improved by only splitting edges until the approximation graph becomes series-parallel. Burris (1972) improves SCOPE and changes the name to MAPS (Method for the Analysis of the Probabilities of Systems). The implementation incorporates a modularity feature and coded in PL/1, which can easily uses binary digit manipulation. A further extension of MAPS is SPARCS Simulation Program for Assessing the Reliabilities of Complex Systems. It was programmed by Cooley (1976). SPARC implementation in addition to MAPS version uses Monte-Carlo combined with Bayesian techniques to assess the reliability and the MTBF (mean time between failures) of a complex system. A more efficient version of SPARCS, called SPARCS-2 has been prepared by Lee (1977).

David et al. (2004), implemented an interesting algorithm within SATURN, for managing and evaluating the reliability of transport systems. The algorithm contributes to optimizing the degree of the quality of service, the travel demand and the degree of stability of any network which is referred to as the ability of network the expected goals under different circumstances (e.g. variation in demand over different days). It is supposed that the network maintain at least one possible travel path under environmental factors.

SHARPE (Sahner R.A., K.S. Trivedi, 1987; Sahner R.A., K.S. Trivedi, 1993) can both solve fault-tree models. SHARPE also solves reliability block diagrams and networks reliability. It can provide semi-symbolic expressions for the reliability function. With SHARPE and its associated algorithm SI-IARPE, they solve Markov and semi-Markov reward models for their steady-state, transient and cumulative behavior. Specification is textual, but abilities include solving the model for many different parameters using "loop" specifications (Sahner R.A., K.S. Trivedi, 1993).

One another efficient program named SYREL for symbolic Reliability Algorithm was first designed by Hariri and Raghavendra (1987) and implemented on VAX 11/750. SYREL can analysis large networks with modest memory and time requirement. The algorithm incorporates conditional probability, set theory and Boolean algebra. SYREL has been incorporated in many other general tools in Reliability engineering software. In Resend (1986), an efficient algorithm was programed in FORTRAN. The software implements techniques based on polygon-to-chain reduction for evaluating the reliability of K-terminal networks as discussed previously. The power of this implementation is that it uses efficient data structures allowing fast reliability computation of large undirected networks (Resende, 1986). The author presents in this article several large reducible networks up to 35000 edges as discussed in the literature and shows the performance capabilities of the programed code. The program code is similar to the algorithm of Satyanarayana and Wood (1982) and Wood (1985, 1986). Theologou and Carlier (1991) and Theologou (1990) improved the implementation of Resende (1986) using a more sophisticated techniques for manipulating data structure and algorithm memory representation. They call such tool RES. According to authors, RES usually used simple and polygon-to chain reductions, biconnected decomposition each time a separable graph is recognized by the procedure. When network nodes are imperfect, RES call another program named RES_{vf}. For augmenting the performance of their implementation, they uses and tested two strategies, an optimal and empirical one for selecting the node with the minimal sum of endpoints degrees. Authors argue that their factoring implementation gives better results compared with others programs taken from literature. Hardy et al. (2006) proposed a new series of algorithms implemented in a tool. They detailed the functioning of their tool and showed that it is intended to solve larger real networks. They also testify that their algorithm can solve the optimization of reliability network. Choi and Jun (1994) present a software base-factoring algorithm which includes a pivoting edge selection strategy. The algorithm employs pivoting edge selection based on the minimal path and applied it for partial polygon-to chain reductions. This tool is composed by a series of algorithm's implementation called FAPSA (Flow augmenting path search algorithm). They are numbered from 1 to 5 as FAPSA-1 to FAPSA-5. These algorithms are specialized. For example FAPSA-3 uses the partial type-1 polygon-to chain with series and parallel reductions and FAPSA-5 uses the polygon-to chain reduction in addition to the reductions made by FAPSA-4. Berkeley Reliability Toolkit (Tu et al. (1993)) is expected to serve as the engine of "design-for-reliability. It works with a circuit simulator in order to simulate reliability for actual circuits; it can act as an interactive tool for design.

The following table gives the components of four modern well-known commercial software tools.

Reliasoft http://www.reliasoft.com	Isograph http://www.isograph-software.com	WEIBULL Weibull.com	Reliability Analytics Toolkit http://reliabilityanalytics.com
The Synthesis® Platform Integration to Empower the Reliability Organization	Availability simulation		Redundancy Calculators
Weibull++® Reliability Life Data Analysis	Attack Tree+	Reliability Life Data Analysis	Reliability Growth Planning
ALTA® Accelerated Life Testing Analysis	Life Cycle Cost analysis	Accelerated Life Testing Analysis	Spares Calculators
DOE++® Experiment Design and Analysis	Weibull Analysis	Experiment Design and Analysis	Weibull Analysis
RGA® Reliability growth analysis	RCM SAP Interface	Reliability Growth Analysis	Reliability Growth Analysis
λPredict® Standard Based Reliability Prediction	RCM MAXIMO Interface	FMEA and FMECA Analysis	Normal Analysis
MPC® MSG-3 Maintenance Program Creation	RCM Ellipse Interface		Environmental Impact Analysis
Orion eAPI® Web-Based Asset Mmanagement	Report Designer		Quality Function Deployment QFD
BlockSim® System Analysis with RBDs or Fault Trees	Ebtreprise System	System Analysis with RBDs or Fault Trees	Reliability Modeling and Analysis
RENO® Probabilistic Event Related Analysis	Hazop+2013	Role	Reliability Prediction
RCM++® Reliability Centered Maintenance	Reliability Centered Maintenance	Reliability Centered Maintenance	Reliability-Centered Maintenance
RBI® Risk based inspection analysis	NAP	Mechanical Component Interactions	

6. CONCLUSION

This paper attempts to give an overview of system reliability analysis with evaluation and optimization aspects. It presents some methods and algorithms on exact and approximation solutions; it remains many others. The content discusses both theoretical and practical bases of several widely used techniques. We presented a concise state-of-the art and a detailed literature review in which it has enumerated the most interesting papers and subjects of network reliability since the first works. Finally, to close the contents of this paper, we discuss well-known software tools related to network reliability evaluation and optimization and compared by presenting the software components of four modern commercialized tools.

Appendices

Appendix A- Enumeration methods

A.1. State-Space Enumeration

State-space enumeration method is the most direct brute force approach for computing the reliability of a network. It proceeds simply by determining the whole set of state vectors, checking for each one if the network is operational or not. The whole set of state vectors represents all the combinations where each of the *m* edges can be good or bad, resulting in 2^m combinations. Each of these combinations is considered as an event E_i. These events are all mutually exclusive (disjoint) and the reliability expression is simply the probability of the union of the subset of events that contain a path between *s* and *t* which is expressed as follows:

$$\mathrm{R}_{\mathrm{s,t}}(S) = \Pr(E_1 \cup E_2 \cup \ldots \cup E_m) = \Pr(E_1) + \Pr(E_2) + \cdots + P(E_m) \tag{1}$$

where $E_1 \cup E_2 = \emptyset \ \forall i, \forall j, i \neq j$

Suppose we want to use state enumeration method to evaluate the reliability between node *a* and node *c* of the network presented in figure 1. First, we adopt some conventional terms. Let the term *good* means that there is at least one path from *a* to *c* for the given combination of good and failed edges. The term *bad*, on the other hand, means that there are no paths from *a* to *c* for the given combination of good and failed edges. The result-good or bad is determined by inspection of the graph, they are reported in table 1.

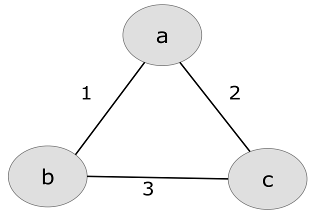


Figure 1. A simple network

1- No failure: $E_1 = 1\ 2\ 3$ Good	# Combination: $\binom{3}{0} = \frac{3!}{0!3!} = 1$
2- One failure: $E_2 = 1'\ 2\ 3$; $E_3 = 1\ 2'\ 3$; $E_4 = 1\ 2\ 3'$ Good Good Good	# Combination: $\binom{3}{1} = \frac{3!}{1!2!} = 3$
3- Two failures: $E_5 = 1'\ 2'\ 3$; $E_6 = 1\ 2'\ 3'$; $E_7 = 1'\ 2\ 3'$ Bad Bad Good	# Combination: $\binom{3}{2} = \frac{3!}{2!1!} = 3$
4- Three failures: $E_8 = 1'\ 2'\ 3'$ Bad	# Combination: $\binom{3}{3} = \frac{3!}{3!0!} = 1$

The reliability is deduced from the addition of the good events. It is as follows :

$$R_{a,c}(G) = [p_1 p_2 p_3] + [q_1 p_2 p_3 + p_1 q_2 p_3 + p_1 p_2 q_3] + [q_1 p_2 q_3] = p_2 + p_1 p_3 - p_1 p_2 p_3$$

It is simple to verify this equality, it suffices to use the relation that $q_i = 1 - p_i$ for $i = 1, 2, 3$.

A.1.1 Path enumeration – Cut enumeration

This method is executed in two steps. First step consists of enumerating MPS or MCS. In second step the reliability evaluation needs the development of the symbolic expression in terms of the probability of various components being operational/non-operational. If MPS/MCS are mutually exclusive, the probability of the union of *m* events (corresponding to components state; working/failed) can be written if MPS = $\{P_1, P_2, \dots, P_m\}$ and MCS = $\{C_1, C_2, \dots, C_m\}$ where $\bar{C_j}$ represents the event "the components of the j[th] minimal cut are not functioning", thus

$$\begin{aligned} \mathrm{R(G)} &= \Pr(P_1 \cup \dots \cup P_m) = \Pr(P_1) + \Pr(P_2) + \dots + P(P_m) \,/ E_1 \cup E_2 = \emptyset\ \forall i, \forall j, i \neq j \qquad (2) \\ &= 1 - (\Pr(\bar{C_1}) + \Pr(\bar{C_2}) + \dots + \Pr(\bar{C_n})) \end{aligned}$$

For the same example presented earlier (figure 1.1.), the algorithm generates two minpaths,

$Path_1$: {1,3}; $Path_2$: {2}.

The structure function is equal to:

$$\Phi(X) = 1 - (1 - x_1 x_3)(1 - x_2) = x_2 + x_1 x_3 - x_1 x_2 x_3$$

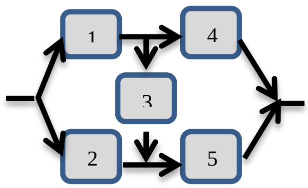
and the reliability is :

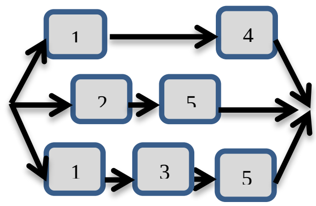
$R_{a,c}(G) = E\{\Phi(X)\} = p_2 + p_1p_3 - p_1p_2p_3$

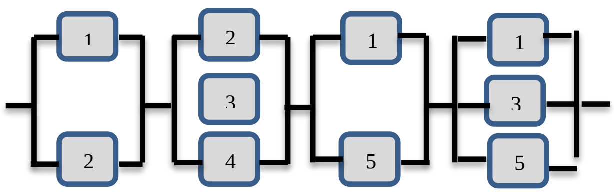
If $p_1 = 0.9$; $p_2 = 0.9$; $p_3 = 0.9$, then $R_{a,c}(G) = 0.981$

Note that state enumeration and path enumeration methods give the same expression the same value of any network reliability.

The following example is known in the literature as "Bridge network" (first graph), from which we generate the sets of MPS (second graph) and MCS (third graph). Note that the reliability value generated from graph 2 and graph 3 are equals.







MPS = {P_1 = {x_1, x_4}; P_2 = {x_2, x_5} and P_3 = {x_1, x_3, x_5}}.

MCS = {C_1 = {x_1, x_2}; and C_2 = {x_1, x_5}; C_3 = {x_2, x_3, x_4}; C_4 = {x_4, x_5}}.

A.1.2 Sum of Disjoint Product Methods

A.1.2.1 Introduction

The starting point of an analysis is Boolean polynomial expression **B** for system success as a logical sum of the MPS. Dually, the polynomial for failure is a sum of MCS.

Let the system have *m* MPS $P_1, P_2, \dots, P_m$, each minpath is a term of the minimized for of **B**. Due to the s-coherence properties (Barlow and Proschan, 1975), every MPS has all 1-valued variable. The minimal expression for success is:

$$\Phi(X) = P_1 + P_2 + \cdots + P_m = P_1 + \bar{P}_1P_2 + \bar{P}_1\bar{P}_2P_3 + \cdots + \bar{P}_1\bar{P}_2\bar{P}_3 \dots P_m \qquad (3)$$

The statement is simple to be proved by induction on the Boolean terms due to the following Boolean equality:

$$x + y = x + \bar{x}y$$

A.1.2.2. Inclusion-exclusion formula

The inclusion-inclusion formula also called Poincaré-formula or Poincaré theorem can be used for generating directly the expression of the reliability. It is as follows:

$$E\{\Phi(G)\} = \sum_{1\le i\le m} E(P_i) - \sum_{1\le i_1<i_2\le m} E\left(P_{i_1}.P_{i_2}\right) + \cdots + (-1)^{m+1}.E(P_1.P_2.P_3\cdots P_m) \qquad (4)$$

The structure function relative to the early network is: $(G) = P_1 + P_2$: P_1 and P_2 are the paths. Thus, the reliability is,

$$R(G) = E\{\Phi(G)\} = E\{P_1 + P_2\} = E\{P_1\} + E\{P_2\} - E\{P_1.P_2\} = E\{1{,}3\} + E\{2\} - E\{\{1{,}3\}.\{2\}\}$$
$$= p_2 + p_1.p_3 - p_1p_2p_3$$

Note that Poincaré formula generates the same expression as earlier.
It is adequate for the calculus to derive from Poincaré formula a recursive function. It is as follows:

$$R_j = \Pr\left\{\bigcup_{i=1}^{j-1} S_i\right\} = R_j + P_j - \Pr\left\{S_j \bigcap \left(\bigcup_{i=1}^{j-1} S_i\right)\right\} \qquad (5)$$

R_j : is the term of the reliability at j^{th} step.

A.1.2.3 Recursive Disjoint Product (Abraham method)

Recursive disjoint product has been introduced by Abraham (Abraham, 1979). It accomplishes the same objective as Poincaré method but results in a different form of the probability polynomial (Locks, 1980). Given the list of MPSs corresponding to a network, the algorithm builds recursively the expression of the reliability by accumulating the probability of the MPS P, one MPS at a time. The recursion formula of Abraham is:

$$R_j = R_j + \Pr\left\{S_j \bigcap \left(\bigcup_{i=1}^{j-1} S_i\right)\right\} = R_j + \Pr\{S_j \cap \bar{S}_1 \cdots \cap \bar{S}_{j-1}\}$$

For the network in figure 1.1, the method of Abraham is used as follows:

$Path_1$: {1,3} = x_1x_3; $Path_2$: {2}=x_2

The first term corresponds to the first path : Term 1 : x_1x_3;

Outer loop 1:

Term 2: $\bar{x}_1x_2$;

Term 3: $x_1\bar{x}_3x_2$;

Stop.

$\Phi(X) = x_1x_3 + \bar{x}_1x_2 + \ x_1\bar{x}_3x_2;$

Simple checking :

The substitution of the complemented variable in $\Phi(X)$, gives :

$\Phi(X) = x_1x_3 + \bar{x}_1x_2 + \ x_1\bar{x}_3x_2 = x_1x_3 + (1 - x_1)x_2 + \ x_1(1 - x_3)x_2 = x_2 + x_1.x_3 - \ x_1x_2x_3$

and the reliability is :

$R(G) = p_2 + p_1.p_3 - p_1p_2p_3$

Proof of the equivalence of Recursive disjoint products and Recursive inclusion-exclusion:

$$P_j = \Pr\{S_j\} = Pr\left\{S_j \bigcap \left(\bigcup_{i=1}^{j-1} S_i\right)\right\} + Pr\left\{S_j \bigcap \left(\overline{\bigcup_{\iota=1}^{j-1} S_\iota}\right)\right\}$$

Thus

$$P_j - Pr\left\{S_j \bigcap \left(\bigcup_{i=1}^{j-1} S_i\right)\right\} = Pr\left\{S_j \bigcap \left(\overline{\bigcup_{\iota=1}^{j-1} S_\iota}\right)\right\}$$

The accumulated probability is exactly the sum of the probability of the terms.

A.1.3. **Disjoint Products** (Heidtmann method, 1989)

This algorithm is a modification of the algorithm given earlier by Abraham. The inversion use multiple-variable. It is simpler and more efficient than Abraham's. On the example of figure 1, it proceeds as follows:

First term is: x_1x_3 ;

Second term is: $\overline{x_1x_3}x_2$

And the structure function is

$\Phi(X) = x_1x_3 + \overline{x_1x_3}x_2 = x_1x_3 + (1 - x_1x_3)x_2 = x_2 + x_1.x_3 - x_1x_2x_3$

A.2. Reduction and factorisation

A.2.1. Reduction rules

In order to reduce the size of network which leads to minimizing the computing cost of the network reliability, it is needed to apply some reduction techniques. The idea behind the reduction is to transform each graph partition into a simplified form, while preserving its reliability. They are (reductions) similar to those of the factoring theorem, which consist of the replacement of a particular structure (e.g. a polygon) embedded in the graph within the abstraction of the rest of the graph. The demonstration of such procedure uses the following reduction rules which are resumed in what follows:

Let e_a = *(u, v)* and e_b = *(u, w)* be two series edges in G_K such that degree(v) = 2 and $v \notin K$. Applying reduction procedure leads to obtain the sub-graph *G'* by replacing e_a and e_b with a single edge e_c = *(u, w)* and the corresponding reliability is computed by $p_c = p_a p_b$, and it defines Ω = 1 and *K'* = *K*.

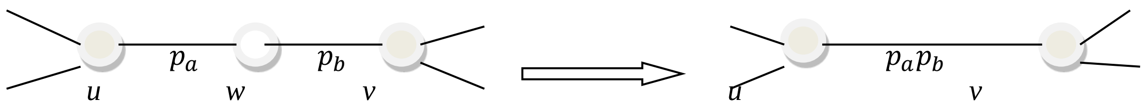


Figure 2. Series reduction

Parallel reduction. Let e_a = *(u, v)* and e_b = *(u, v)* be two parallel edges in G_K (the network graph) and suppose that p_i = 1 - q_i (*i* = *a* or *b*). A parallel reduction obtains *G'* by replacing e_a and e_b with single edge e_c = *(u, v)* with reliability $p_c = (1 - q_a q_b)$, and it defines Ω = 1 and *K'* = *K*. We note that Ω is a multiplicative operator derived from $R(G_K) = \Omega.R(G'_{K'})$

.

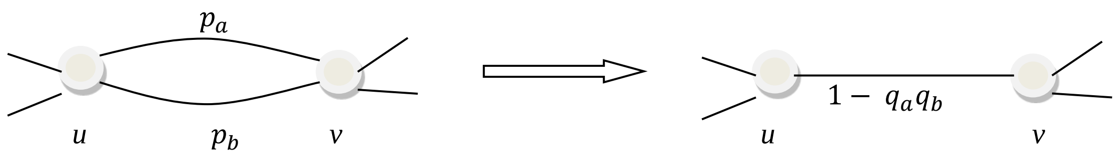


Figure 3. Parallel reduction

Let e_a = *(u, v)* and e_b = *(u, w)* be two series edges in G_K such that degree(v) = 2 and *u, v,* $w \in K$. A degree-two reduction obtains G' by replacing e_a and e_b with single edge e_c = *(u, w)* with reliability $p_c = \frac{p_a p_b}{1 - q_a q_b}$, and it defines Ω = 1 and $K' = K - v$.

Figure 4. Degree two reduction

For directed networks, some additional reduction rules have been introduced (see. Deo and Medidi, 1992). They can be presented as follows:

1. All edges going into the source and all edges going out of the sink can be removed. These edges do not lie on any simple source-to-sink path and are thus irrelevant.
2. Every vertex, except the source and the sink, with 0 in-degree or 0 out-degree can be removed.

3. If a single edge is directed into or out of a vertex, its anti-parallel edge can be removed. Since any simple path through this vertex has to use this single edge, the anti-parallel edge is irrelevant.
4. If there is a single edge out of the source or into the sink, then this edge can be contracted. To get the reliability of the original network, the reliability of the reduced network is multiplied by the success probability of the contracted edge.
5. Series edges can be reduced as shown in figure l.2. (Exceptionally the edges are directed from *u* to *v*)
6. Parallel edges can be reduced as shown in figure l.3. (edges are directed from *u* to *v*).
7. Generalized series reduction, analogous to the elimination of degree-2 vertex in undirected networks, can be performed as shown in the following figure.

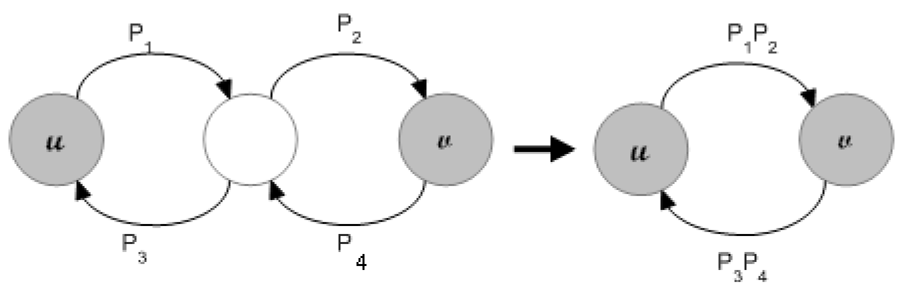


Figure 5. Generalized series reduction

Delta-to star reduction. Delta-to star reduction consists of replacing a topological delta structure by a star structure. Nodes of the delta structure must all be *K*-nodes. The added node is not a *K*-node and its probability is computed as follows:

$p_x = \frac{\alpha}{\alpha+\beta_1}$, $\quad p_y = \frac{\alpha}{\alpha+\beta_2}$, $p_z = \frac{\alpha}{\alpha+\beta_3}$ and $\quad p_{u_0} = \frac{(\alpha+\beta_1)(\alpha+\beta_2)(\alpha+\beta_3)}{\alpha^2}$

with $\alpha = p_a p_b + p_a p_c + p_b p_c - 2 p_a p_b p_c$ and $\beta_1 = q_a q_b p_c$, $\beta_2 = q_a p_b p_c$, $\beta_3 = p_a q_b q_c$

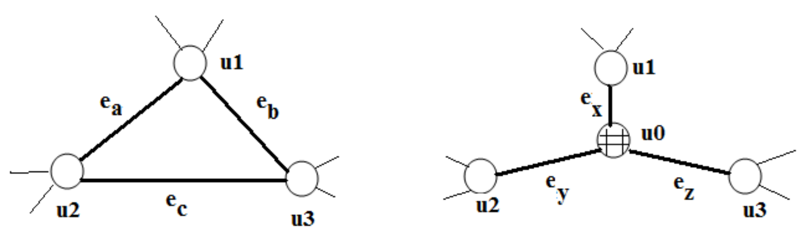


Figure 6. Delta-to star reduction

A.2.2. Generalized factoring theorem

The factoring theorem consists of pivoting on every edge e_i in the graph and decomposing the original problem with respect to two possible states of edge e_i:

$$R(S) = p_i R(S|e_i\, works) + q_i R(S|e_i\, fails)$$

where $R(S)$ is the reliability of the system *S* and $R(S|e_i works)$ is the reliability of the system *S* when the edge e_i is in operation and $R(S|e_i\, fails)$ is the reliability of the system *S* when the edge e_i is not in operation and each probability p_i is asserted to the edge *i* and $q_i = (1 - p_i)$ the opposite of p_i. The earlier equation can be recursively applied to the induced graph, until the generated subgraph contains just one edge. Generally, the factoring theorem and reduction rules work together for obtaining the minimal reliability expression possible. Figure 1.6, gives an overview of the application of factoring decomposition using the graph in figure 1.1. In this example, edge 1 has been picked for generating two state-subgraph at left where it has been supposed that edge 1 is working, and subgraph at right the opposite case (edge 1 fails).

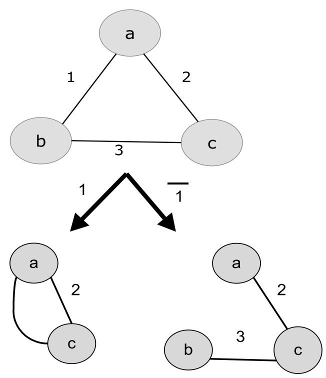


Figure 7. First step performing the factoring decomposition.

One interesting algorithm combines three procedures; a parallel, reduce and partition procedure has been published in Deo and Medidi (1992). This algorithm employs techniques introduced by Wood (1986) and Satyanarayana and Chang (1983) and a parallel algorithm for scheduling the computing the reliability of subgraphs. A formal description of the parallel, Reduced and partition parts are in the following procedure PAR_REDUCE&PARTITION as called by Deo and Medidi (1992). It is as follows:

```
procedure PAR_REDUCE&PARTITION (G:network with s and t; multiplier: real)
          ***this procedure adds (multiplier - Rel (G) ) to the global variable R***
begin
   Reduce G, using reduction rules, if applicable;
         ***if R-Rule 4 is applied, multiplier gets modified***
   PATH3(G, found, path, length);
         ***returns the shortest s-t path in G***
         ***if s=t, this returns length = 0 and found = true***
     if found then
         for i := 1 to length do
           edge e:= path[i];
           pe: = success probability of edge e;
           G1 := G-e;
              if a free processor is available then
```

```
                create parallel task PAR_REDUCE&PARTITION (Gl, multiplier.qe);
           ***the free processor executes this task con- currently***
             else
                PAR_REDUCE&PARTITION (G1 , multiplier.qe);
            end if
         G := G*e;
         multiplier : = multiplier.pe;
       end for
   lock R; R := R + multiplier; unlock R;
          ***Variable R is the terminal-pair reliability of the input network***
 end if
 end { PAR-REDUCE&PARTITION}
```

The data structures used in sequential Reduce_Partition (see Deo and Medidi (1992)) are also used in implementing Parallel_Reduced&Partition. The overhead in the parallel task generation calls provided by parallel commands is avoided by using a queue where the free processors are busy waiting for parallel tasks. The variable *R*, representing the terminal-pair reliability of the input network, is shared by all processors and is referenced only in a critical section (Deo and Medidi (1992)).

A.3. Binary Decision Diagram (BDD)

Binary Decision Diagrams (BDDs for short) is a clever and simple representation of Boolean expressions. It can be considered as a flexible and dynamic notation of directed acyclic graphs (DAG for short). The idea behind BDDs is their transformation from a DAGs to a more reduced one called ROBDD for Reduced Binary Decision Diagrams (Bryant (1986)). They have received a lot of attention in different fields like computational logics, hardware/software verification and in VLSI design. The implementation and manipulation of BDD algorithms is composed by three procedures introduced in Bryant (1986): *restrict*, *apply* and If-Then-Else (ITE).

The representation and the simplification of a Boolean expression proceeds in 4-steps:

- Construct the binary decision tree (BDT) associated with the graph formula.
- Transform the BDT to a BDD by applying the following rules by :

  a- Merging equivalent leaves of a binary decision tree.

  b- Merging isomorphic nodes.

  c- Elimination of redundant tests.

Figure 8 gives an overview of BDDs representation and the transformation from a DAG to a ROBDD.

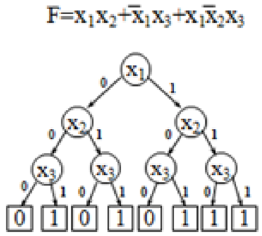
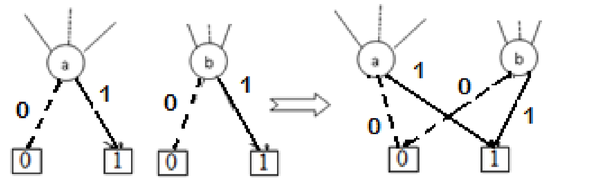
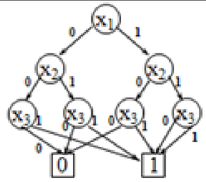
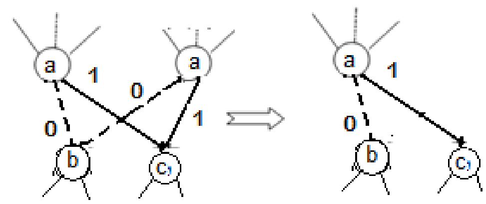
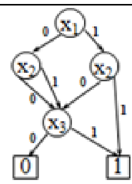
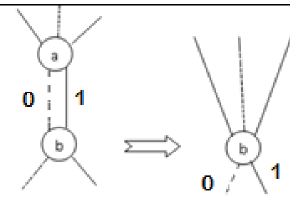
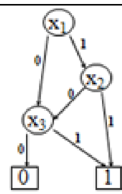
Steps	BDD at the beginning of reduction steps	Reduction rules
Step 1		
Step 2		
Step 3		
		

Figure 8. A logic expression and its transformation from a DAG until Normal form (ROBDD).

Network reliability is calculated using the following relation and the application of the algorithm is presented in figure 9.

$$P(F) = P\big(ite(x, F_1, F_2)\big) = P(x.F_1) + P(\bar{x}.F_2) = (1-p_x)P(F_1) + p_xP(F_2)$$
$$= P(F_1) + p_x(P(F_2) - P(F_1))$$

```
Algorithm Reliability_Evaluation(F, G)
    if ( F == 0)
        return 1    //***  Boolean value 1 (one)  ***//
      else  if  (F == 1)
            return 0    //***  Boolean value 0 (zero)  ***//
            else  if (computed-table has entry {F, P_F})
                return  P_F
              else
              P_F = Prob(F1)  +  P(x) * (Prob(F2)-  Prob(F1)  )
              end
        end
      Insert_computed_table ({F, P_F})
      return  P_F
    end
  end
```

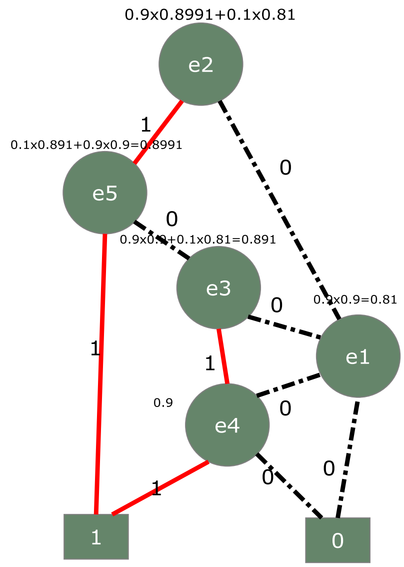


Figure 9. Solution given by the earlier algorithm

The following procedure works for generating MCSs using a depth first search algorithm and data information's taken from specialized matrix as shown in Rebaiaia and Ait-Kadi (2012).

procedure *Generation_of MCS* (G:network with s and t)

begin

Place the squared node on top of a stack 1 //* records DFS visits to ROBDD nodes *//.

Place the squared node on top of a stack 2 //* records cut's nodes.

Place on the top of the stack 1 all the ascending nodes of the top variable in the stack.

Place the node top of the stack 1 on top of the stack 2, if the edge (link) is dotted.

Continue until the variable reach the root node.

If so, a cut has been found. Write the content of the stack 2 as a line of a matrix.

Remove top variable from stack 1 and from stack 2.

Continue the procedure until stack 1 is empty.

Apply the filtering process by removing all the redundant paths (cuts) using the matrix of paths (cuts)

Display MSC Matrix

end_procedure

A filtering procedure removes redundant cuts from the set of all the cuts. It proceeds as follows:

1- Sort the matrix CS in an ascending order according to the size of each vector (number of variables);
2- Take the first vector and compare it with each of the others vectors;
3- If the members of the intersection are equal to the first vector then remove the actual vector from CS matrix;
4- Iterate using the others vectors of the matrix.

The following pseudo-description shows the processing of the filtering procedure as explained bellow:

```
procedure filtering(CS, MCS)

n = length(CS); /* size of matrix vector */
   m = size(CS);  /* size of matrix vector
  for i = 1,m-1
      v(k) = CS(i,k) (k=1,...,n)(CS(i,k)≠0)
     for j = i+1,m
         w(k) = v(k) ∩ CS(j,k) (k=1,...,n)(CS(i,k)≠0)
         if w(k) = v(k) (CS(j,k) is a redundant vector)
            remove vector  CS(j,k);
        end_if
    end_for
 end_for
   MPS = CS
       display MPS
end_procedure
```

Appendice B. Reliability optimization

Modern systems are by nature very complex. To remain competitive, the guarantee of high system reliability at low cost is essential. Computing system reliability is usually not sufficient because it would also provide mechanisms to optimize the reliability taking into account budgetary constraints and parameters which could vary in real-time. Several solutions have been published in the literature (see Kuo *et al.*, 2000) (Rebaiaia and Ait-Kadi, 2010). Thus, one solution for improving the reliability of a system is to add identical components which can be chosen as design alternatives or by giving more capacity to those that already exist. This model reflects precisely the problem of redundancy allocation. Another method consists of using Binary Decision Diagrams which provides a solution to compute the reliability of a system. The algorithm generates the BDD corresponding to constraints and objective function. The manipulation using APPLY procedure (already cited) consists of composing all the constraints with the objective function. The last BDD will represent the solution of the problem. Another alternative solution to optimizing the reliability of networks is precisely to act on the most important component which can bring some improvements to the reliability. The method is called: the Birnbaum's importance index (Birnbaum, 1969). The solution is simple to be evaluated; it suffices to differentiate the expression of the reliability for each pivoting component. The choice is then fixed on the component which has the highest reliability value. The process is iterated until the expected network reliability is approached.

The discussed three models are presented in what follows:

B.1 Series-parallel configuration

Consider the following redundancy allocation problem in m-stage series system and where in each stage (subsystem) there is a number of components in parallel. The problem can be presented as follows:

$$\max_x R(x) = \prod_{j=1}^{m} R(x_j) = R(x_1) \times R(x_2) \times \ldots \times R(x_m)$$

Subject to:

$$\left|\begin{array}{l} \sum_i \sum_j c_{ij}\, x_{ij} \leq C \\ \sum_i \sum_j w_{ij}\, x_{ij} \leq W \\ \qquad\qquad x_{ij} \in N \\ \text{where} \\ R(x_1) = 1 - \prod_{j=1}^{m1}(1 - r_{1j})^{x_{1j}}, \\ R(x_2) = 1 - \prod_{j=1}^{m2}(1 - r_{2j})^{x_{2j}}, \\ \qquad\vdots \\ R(x_s) = 1 - \prod_{j=1}^{ms}(1 - r_{1j})^{x_{sj}} \end{array}\right.$$

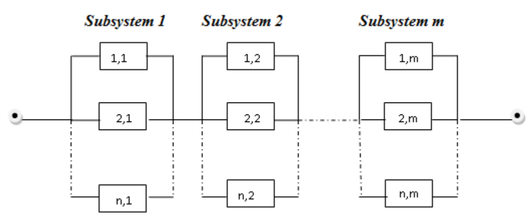


Figure 10. A series-parallel system

It can be noted that this problem is non-linear, so it becomes quickly intractable. This is due to the size of the system which grows with the number of components in each subsystem. Thus, making it linear can creates a way to get a solution without fear of the size of the model. The processing method is simple and intuitive (see Coit and Konak, 2006 for details) it is explained in what follows:

Maximizing the problem is equivalent to minimizing a separable function by introducing the logarithm of the original problem and this is due to the fact that the function is monotonic and positive. The transformation steps are as follows:

$$\max_x\left[1 - \prod_{j=1}^{m}(1 - r_{ij})^{x_{ij}}\right] = \min_x\left[\prod_{j=1}^{m1}(1 - r_{ij})^{x_{ij}}\right]$$

The generalization for all the terms of the objective function and by considering the properties of the logarithm function the following expression is obtained:

$$\max_{\mathrm{x}} \mathrm{R(x)} = \min_{x} ln\left\{\left[\prod_{j=1}^{m1}(1-r_{1j})^{x_{1j}}\right]\times\left[\prod_{j=1}^{m2}(1-r_{2j})^{x_{2j}}\right]\times\cdots\times\left[\prod_{j=1}^{ms}(1-r_{sj})^{x_{sj}}\right]\right\} =$$
$$\min_{x}\left\{\left[\sum_{j=1}^{m1} x_{1j} ln(1-r_{1j})\right]+\left[\sum_{j=1}^{m2} x_{2j} ln(1-r_{2j})\right]+\cdots+\left[\sum_{j=1}^{ms} x_{sj} ln(1-r_{sj})\right]\right\}$$

If we suppose that $y_{ij} = -ln(1-r_{ij})$, the problem of minimization is transformed again to a maximization one, but which can be solved more easily because we get a function composed of separable linear terms as shown in what follow:

$$\max_{x}\left\{\left[\sum_{j=1}^{m1} x_{1j}y_{1j}\right]+\left[\sum_{j=1}^{m2} x_{2j}y_{2j}\right]+\cdots+\left[\sum_{j=1}^{ms} x_{sj}y_{sj}\right]\right\} = \max_{x}\sum_{i=1}^{s}\sum_{j=1}^{m_i} y_{ij}\, x_{ij}$$

Finally the problem becomes :

$$\max_{x}\sum_{i=1}^{s}\sum_{j=1}^{m_i} y_{ij}\, x_{ij}$$

Subject to :

$$\left|\begin{array}{l} \sum_i \sum_j c_{ij}\, x_{ij} \le C \\ \sum_i \sum_j w_{ij}\, x_{ij} \le W \\ \sum_{j=1}^{m_i} y_{ij}x_{ij} > -\ln\left(1-R_i^{min}\right), i = 1,\cdots,s \\ x_{ij} \in N \end{array}\right.$$

Solution to such problem is now affordable; it suffices to use any optimizing tool such as CPLEX or Gurobi.

B.2. Binary decision Diagrams optimization

Binary decision diagrams (BDDs) have been discussed earlier. The following example explains how to apply the BDDs to represent a solution relative to the optimization problem. First, we represent the constraints domain, the objective function and both of them.

Suppose a mathematical model composed by a constraint and an objective function presented as follows:

$$2x_0 + 3x_1 + 5x_2 + 5x_3 \ge 7$$

$$2x_0 + 3x_1 + 4x_2 + 6x_3$$

The corresponding ROBDD of the constraint and the function are depicted using the graphic representation as shown in figure 11.

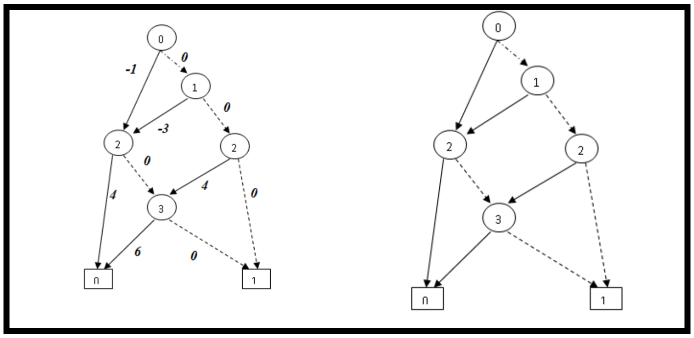


Figure 11. Reduced BDD for the constraint $2x_0 + 3x_1 + 5x_2 + 5x_3 \geq 7$ (left) and for the objective function $2x_0 - 3x_1 + 4x_2 + 6x_3$ (right).

We can add others constraints without any problem and augmenting the dimension of the system of constraints because it suffices to apply the procedure ITE using the logical AND to the BDD structure of the constraints to tackle the problem. The conjunction of the BDD's gives a new BDD which represents the system. For example: Suppose that $x_0 + x_1 + x_2 + x_3 \leq 2$ is another constraint. The new BDD built from the conjunction of $2x_0 + 3x_1 + 5x_2 + 5x_3 \geq 7$ and $x_0 + x_1 + x_2 + x_3 \leq 2$ is represented in figure 1.12.

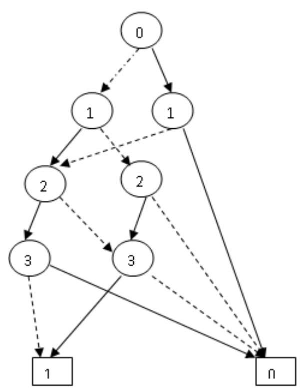


Figure 12: Composition of BDD's

Example : The bridge network

Suppose that the corresponding individual reliabilities of each component and the cost are given by the following two vectors:

$R = (0.7, 0.9, 0.8, 0.65, 0.7)$ and $C = (4, 5, 4, 3, 3)$

The intersection of the BDDs generates a new BDD and from which it is shown that the reliability of the system grows from 0,72 if $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 1, 1, 0)$ to 0.99 if $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 0, 0, 3)$.

Appendice C. Reliability Importance Measures

The Birnbaum reliability importance measure of a component i, denoted by

$$I_B(i; \mathbf{p}) = \frac{\partial h(p)}{\partial p_i} = h(1_i, \mathbf{p}) - h(0_i, \mathbf{p})$$

where $(\cdot_i\,, \mathbf{p}) = (p_1, \cdots, p_{i-1}, \cdot, p_{i+1}, \cdots, p_n)$

Thus, the Birnbaum structural importance measure is defined as follows:

$$I_B(i) = I_B\left(i; (\frac{1}{2}, \cdots, \frac{1}{2})\right) = \frac{1}{2^{n-1}} |\{(1\cdot_i\,, x) : \phi(0_i, x)\}| > \phi(\cdot_i\,, x)$$

Where | . | denotes the cardinality of a set.

The Birnbaum measure may be interpreted as the probability of the system being in a functioning state with the i^{th} component being the critical one (the failure of this component which is assumed functioning coincides with failure of the system) (Xie and Shen, 1989). It can be remarked that Birnbaum's measure is independent of the i^{th} component.

The following elements will show what type of action to improve the system reliability.

C.1. Definition (from Xie and Shen, 1989)

The Δ-importance of the i^{th} component, $I_\Delta^{(i)}$, is defined as the increase of the system reliability due to an improvement of the i^{th} component. That is

$$I_\Delta^{(i)} = \Delta_i = h(p_i', \mathbf{p}) - h(\mathbf{p})$$

where p_i' is the reliability for the improved component and $h(p_i', \mathbf{p})$ is the system reliability after the improvement on the i^{th} component.

Using the above equation it can be easily demonstrated that:

$$\Delta_i = (p_i' - p_i).\, I_B^{(i)}$$

It expected that Δ-importance measure is another equivalent model of Birnbaum important measure.

C.2. Reliable improvement

Another quality of improvement is to fix the new value of the probability of the i^{th} component by a reliable one such as $p_i' = 1$. Thus,

$$\Delta_i = (1 - p_i).I_B^{(i)} = q_i.I_B^{(i)}$$

C.3. Active redundancy

As discussed in section 1, redundancy provision is one effective solution which can improve the reliability of a system. However this type of improvement action can be identified using the earlier Δ-importance measure and such as the ith component is organized in parallel with another identical. In such case we get:

$$p_i' - p_i = 1 - (1 - p_i)(1 - p_i) - p_i = 2p_i - p_i^2 - p_i = p_i q_i$$

Then the improvement action gives :

$$\Delta_i = p_i q_i.I_B^{(i)}$$

Example:

Call back the undirected bridge network. Let :

$x_1, x_2, x_3, x_4,$ and x_5 are state variables and $p_1 = 0.8, p_2 = 0.3, p_3 = 0.6, p_4 = 0.7,$ and $p_5 = 0.5$.

The function structure is

$$\phi(x) = x_1x_3 + x_2x_4 + x_1x_4x_5 + x_2x_3x_5 - x_1x_2x_3x_4 - x_1x_3x_4x_5 - x_1x_2x_3x_5 - x_1x_2x_4x_5 - x_2x_3x_4x_5 + 2x_1x_2x_3x_4x_5$$

From the earlier equation we can compute what follows:

1- Birnbaum structural importance:

$I_B(1) = 0{,}545;\ I_B(2) = 0.270; I_B(3) = 0.168; I_B(4) = 0.445;\ I_B(5) = 0.250$

2- Δ-importance measure:

$$\Delta_1 = 0.109;\ \Delta_2 = 0.189;\ \Delta_3 = 0.084;\ \Delta_4 = 0.178;\ \Delta_5 = 0.075$$

C.4. A Birnbaum BDD-based algorithm for optimizing network reliability and constraints cost

The following algorithm combines Birnbaum importance measure and BDD structure representation for optimizing the design of the network reliability with minimal cost that satisfy a set of constraints. The objective function and the constraints are stated as follows:

$$Minimize \;\; C(G') = \sum_{i=1}^{m} c_i x_i$$
$$subject\ to: R(G') \geq R_0$$

Where:

- $x_i (x_i \in \{0,1\})$ is equal to 1 if edge e_i exists in the solution and 0 otherwise.
- c_i is the cost of a link $e_i \in E$.
- $G' = (V, E')$ is a partial graph of G such that $E' = e_i \in E / x_i = 1$.
- $C(G')$ is the total cost of network G'.
- $R(G')$ is the network all-terminal reliability.
- R_0*:* The minimum reliability requirement.

Algorithm *ND_Improvement*(G, BDD Φ, R_0) [Hardy et al., 2006]

Begin

Inputs:

G = (V, E, p, c); BDD Φ encoding the network reliability of G; R_0: minimal all-terminal network reliability required

Output:

Partial network G' (with edge set $E' \subseteq E$) represents the best solution found; total cost C best associated.

Step 1 (Initialization)

(a) $E' \subseteq E$

(b) Compute $R_{max} = R(G, p)$ by applying Algorithm (Reliability_Evaluation).

if ($R_{max} < R_0$) then the problem has no solution

(c) $C_{init} = \sum_i c_i$ $(i = 1, \ldots, |E|$

(d) $C_{best} = C_{init}$

Step 2

(a) Order links in E' according to their network importance measure.

(b) Select link e_i which has the minimum measure such that $R(G'/e_i)$ is down) $\geq R_0$

(c) if such link does not exist then go to Step 3 else

$* \; p_i = 0$

$* \; E' = É \backslash e_i$

$* \; C_{best} = C_{best} - c_i$

$*$ go to Step 2

Step 3

For each edge c_i in E':

For each edge $e_{i'}$ in $E \backslash E'$ such that $c_{i'} < c_i$:

if $R(G' \backslash e_i$ down & $e_{i'}$, up) $\geq R_0$ and $R(G' \backslash e_i$ down & $e_{i'}$, up) $\leq R(G')$:

$*$ restore $p_{i'}$ from its original value

$* \; p_i = 0$

$* \; E' = É \backslash e_i \cup \{e_{i'}\}$

$* \; C_{best} = C_{best} - c_i + c_{i'}$

end

endfor

References

Abraham J. A., 1979, An improved algorithm for network reliability, *IEEE Transactions on Reliability*, pp. 58–61.

Aggarwal K. K., et al., 1975, A fast algorithm for reliability evaluation, *IEEE Transactions Reliability*, R-24.

Agrawal A., 1982, Reliability analysis of rooted directed networks, PhD Thesis, Dept. of IEOR, University of California, Berkeley.

Akers S. B., 1978, Binary Decision Diagrams, *IEEE Transactions on Computers*, C-27(6), pp. 509–516.

Al-Ghanim AM., 1999, A heuristic technique for generating minimal path and cutsets of a general network, *Computers & Industrial Engineering*, 36: 45–55.

Andersen H. R., 1999, An Introduction to Binary Decision Diagrams, *Lecture Notes, IT University of Copenhagen*.

Balan A. O, 2003, An Enhanced Approach To Network Reliability Using Boolean Algebra, Honors Thesis, Department of Computer Science and Mathematics, Lafayette College.

Ball M. 0, 1986, An overview of the complexity of network reliability computations, *IEEE Transactions on Reliability*, R-36, 1986.

Barlow R. E., Proschan F., 1975, Statistical Theory of Reliability, Holt, Rinehart and Winston.

Birnbaum Z.W, 1969, On the importance of different components in a multicomponent systems, Multivariate analysis II, *Academic Press*, pp. 581-592.

Bodlaender, H. L. and Wolle, T., 2004, A note on the complexity of network reliability problems, technical report UU-CS-2004-001, Utrecht university.

Bollig B., Ingo W., 1996, Improving the Variable Ordering of OBDDs Is NP-Complete, *IEEE Transactions on Computers*, 45(9), pp. 993–1002.

Bryant. R. E., 1986, Graph-Based algorithms for Boolean function manipulation, *IEEE Transactions on Computers*, 35(8), 677–691.

Bryant. R. E., 1992, Symbolic Boolean Manipulation with Ordered Binary Decision Diagrams, *ACM Computing Surveys*, 24(3), pp. 293–318.

Burris J. L., 1972, Model for the Analysis of the Probability qf Systems, MBA Research Report, Department of Administrative Sciences, Oklahoma State University.

Buzacott J. A., 1980, A recursive algorithm for finding reliability measures related to the connection of nodes in a graph, *Networks*, 10, pp. 311-327.

Chang M K., 1981, A graph theoretic appraisal of the complexity of network reliability algorithms, PhD Thesis, Dept. of IEOR, University of California Berkeley.

Chang Y-R, Amari S. V., and Kuo S-Y,2004, Computing System Failure Frequencies and Reliability Importance Measures Using OBDD, *IEEE Transactions on Computers*, 53 (1), pp. 54-68.

Chern, M. S., 1992, On the computational complexity of reliability redundancy allocation in a series system, *Operation Research Letter*, 11, 309-315.

Choi, M. S. and Jun, C. H., 1995, Some variants of polygon-to-chain reductions in evaluating reliability of undirected network, *Microelectronics and Reliability*, 35(1), pp. 1-11.

Ciardo, G., J. Muppala, and K. Trivedi, 1992, Analyzing Concurrent and Fault-Tolerant Software using Stochastic Reward Nets," *Journal of Parallel and Distributed Computing*, 15, pp. 255-269.

Coit A. W and Konak A., 2006, Multiple weighted objectives heuristic for the redundancy allocation problem, *IEEE Transactions on Reliability*, 55(3), 551-558.

Coit A. W and Smith A. E., 1996, Reliability optimization for series-parallel systems using a genetic algorithm, *IEEE Transactions on Reliability*, 45(2), 254-260, 1996.

Coit A. W., Jin T. and Wattanapongsakorn, 2004, System optimization considering component reliability estimation uncertainty: a multi-criteria approach, *IEEE Transactions on Reliability*, 53(3), 369-380.

Colbourn C.J and Harms D.D., 1985, Bounding all-terminal reliability in computer networks, *Computer Communications Network Group*, University of Pengasinan, Technical Report E-123.

Cooley J. W., 1976, Simulation Program for Assessing the Reliability of Complex Systems, Ph.D. Dissertation, Oklahoma State University.

Corinne Lucet,1993, Méthode de décomposition pour l'évaluation de la fiabilité des réseaux, Thèse de doctorat, Université de technologie de Compiègne, France.

Coudert O. and J. C. Madre, 1992, Implicit and incremental computation of primes and essential primes of Boolean functions, in Proc. of the 29th ACM/IEEE Design Automation Conference, *IEEE Computer Society Press*, pp. 36–39.

Walting D., Sumalee A., Connors R., Balijeâlli C., (2004), Advanced methods for evaluating network reliability, Second International Symposium on Transportation Network Reliability (INSTR 2004), Christchurch, New Zealand.

Deo, N. and Medidi, M., 1992, Parallel algorithms for terminal-pair reliability, *IEEE Transactions on Reliability*, 41(2), pp. 201-209.

Dotson, W. P. and Gobien, J. O., 1979, A new analysis technique for probabilistic graphs, *IEEE Transactions Circuits and Systems*, 26, pp. 855-865.

Esary J. and Proschan F., 1963, Coherent structures of nonidentical components, Technometrics, 5, pp. 191-209.

Fonseca C. M. and P. J. Fleming, 1995, An overview of evolutionary algorithms in multiobjective optimization, *Evol. Comput*., vol. 3, no. 1, pp. 1–16.

Frank H. and Friech I.T., 1971, Communications, Transmission, and transportation Networks, Addison-Wesley.

Fratta, L. and Montanari, U., 1973, A boolean algebra method for computing the terminal reliability in a communication network, IEEE *Transactions on Circuit Theory*, CT-20, pp 203-211.

Fyffe DE, Hines WW, Lee NK, 1968, System reliability allocation and a computational algorithm, *IEEE Transactions on Reliability*, R-17(2):64-69.

Garey M. R., Johnson D. S., 1979, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Co.

Ha C. and Kuo W., 2006, Multi-Path heuristic for redundancy allocation: The tree heuristic, *IEEE Transactions on Reliability*, 5537-43.

Hardy. G., C. Lucet, and N. Limnios., 2006, A BDD-Based Heuristic Algorithm for Design of Reliable Networks with Minimal Cost, J. Cao et al. (Eds.: MSN 2006), LNCS 4325, pp. 244-255.

Hardy. G., C. Lucet, and N. Limnios., 2007, K-Terminal Network Reliability Measures With Binary Decision Diagrams, *IEEE Transactions on Reliability*, 56, 3.

Hariri S. and C. S. Raghavendra, 1987, "SYREL: a symbolic reliability algorithm based on path and cutset methods," *IEEE Trans. Comput*. C-36, pp.1224–1232.

Raghavendra C. S., V. K. Prasanna Kumar, and S. Hariri, "Reliability analysis in distributed systems," IEEE Trans. Comput. 37, 352–358 (1988).

Heidtmann K.D., 1989, Smaller Sums of Disjoint Products by Subproduct Inversion, *IEEE Transactions on Reliability*, 38, pp. 305-311.

Hopcroft J. E., and Tarjan R. E., 1973, Dividing a graph into triconnected components, *SIAM, J. Computing*, 2, pp 135-158.

Hwang C. L., Tillman F.A and Lee M.H., System reliability evaluation : techniques for Complex/Large systems- a review, *IEEE. Transactions of Reliability*. R-30, pp. 416-423.

Institute of Electrical and Electronics Engineers, 1990, IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries, New York, NY,

Jasmon, G. B. & O. S. Kai., 1985, A New Technique in Minimal Path and Cutset Evaluation, *IEEE Transactions on Reliability*, Vol. R-34, no. 2, pp. 136-143.

Johnson R., 1982, "Some Combinatorial Aspects of Network Reliability", PhD. Thesis, Dept. of IEOR, University of California, Berkeley, 1982.

Johnson A. M. and Malek M., 1988, Survey of software tools for evaluating reliability, availability, and serviceability, *ACM Computing Surveys*, 20 (4).

Kuo S-Y, LU S-K andYeh F-M., 1999, Determining pair-reliability on edge expansion diagram using OBDD, *IEEE Transactions on Reliability*, 48 (3), pp. 235-246.

Kuo, W., V. R. Prasad, F.A. Tillman, and C.L. Hwang, 2000, Optimal Reliability Design: Fundamentals and Applications, *Cambridge University Press*, New York, 2000.

Lee. C. Y., 1959, Representation of Switching Circuits by Binary-Decision Programs, *Bell Systems Technical Journal,* 38, pp. 985–999.

Lee. Keun Kuk, 1977, Rejection Methods for Generating Random Deviates and Their Application in System Reliability, MS Dissertation, Oklahoma State University.

Lin P. M et al., (1976), A new algorithm for symbolic system reliability analysis, *IEEE Transactions of Reliability*, R-25, 1976.

Lin, John Y., and Donaghey, Charles E. (1993). Monte Carlo simulation to determine minimal cut sets and system reliability. In Proceedings of the 1993 annual reliability and maintainability symposium, Atlanta, *GA, USA*.

Lin, H. Y., Kuo, S. Y. and Yeh, F. M, 2003, Minimal cutset enumeration and network reliability evaluation by recursive merge and BDD, in Proc. of the Eighth IEEE International Symposium on Computers and Communication, 2, pp. 1341-1346.

Liu, H. H., Yang, W. T. and Liu, C. C, 1993, An improved minimizing algorithm for the summation of disjoint products by Shannon's expansion, *Microelectronic Reliability*, 33(4), pp. 599-613.

Locks, M. O. and Wilson, J. M, 1992, Note on disjoint products algorithms, *IEEE Transactions on Reliability*, 41(1), pp. 81-84.

Locks, M. O, 1980, Recursive Disjoint Products, Inclusion-Exclusion, and Min-Cut Approximations, *IEEE Transactions on Reliability*, R-29, pp. 368-371.

Locks, M. O, 1978, System reliability analysis: A tutorial, Microelectron. Reliab., Vol. 18, pp. 335- 345.

Locks M. O., (1971), Exact Minimal-State System Reliability Analysis, Proceedings: Computer Science and Statistics, Fifth A. Syrup. on the Interface Western Periodicals Company, 13000 Raymer Street, North Hollywood, California 91605, pp. 180-192.

Marseguerra M., Zio E., Podofillini L., and Coit D. W., 2005, *IEEE Transactions on Reliability*, 54(2), pp. 243-253.

Meng F.C., 1996, Comparing the importance of system components by some structural characteristics, *IEEE Transactions on Reliability*, 45(1), pp. 59-65.

Misra K.B. and Sharma U., 1991, An effective approach for multiple criteria redundancy optimization problem, *Microelectronics and reliability*, 31(2/3), 303-321.

Misra K. B., 1970, An algorithm for the reliability evaluation of redundant networks, *IEEE Trans. Reliability*, R-19, 1970, pp146-151.

Moore E. F., Shannon C. E.,1956, Reliable circuits using less reliable relays, J. Franklin Institute, 262, pp 191-208.

Moskowitz, Fred, 1958, The analysis of redundancy networks, *AIEE Transactions Commun. Electron.*, vol 39, pp 627-632.

Murchland J., 1973, Calculating the probability that a graph is disconnected, Working Paper No. 182, 1973.

Nakazawa, H., 1976, Bayesian decomposition method for computing reliability of an oriented network, *IEEE Tranactions on Reliability*, R-25, pp 77-80.

NASA's COSMIC (The University of Georgia, Barrow Hall, Athens, GA 30601), Exact Minimal Path and Minimal-Cut Techniques for Determining System Reliability, Program MFS-16499; System for Computing Operational Probability Equations (SCOPE), Version I (for IBM 7094), Program MFS-16410, Version II (for IBM 360), Program MFS-24035; Apportionment Prediction (APRDCT), Program MFS-24034.

Osaki S., 1972, Reliability Analysis of a two-uni standby-redundant system with preventive maintenance, *IEEE Transactions on Reliability*, R-21(1), pp. 21-29.

Page, L. B. and Perry, J. E., 1988, A Practical Implementation of the factoring theorem for network reliability, *IEEE Transactions on Reliability*, 37(3), pp. 259-267.

Page, L. B. and Perry, J. E., 1989, Reliability of directed network using the factoring theorem, IEEE Transactions on Reliability, 38(5), pp. 556-562.

Provan J. S. and M. O. Ball, 1983, The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM J. Comput*. 12(4), pp. 777-788.

Rebaiaia M-L, 2011, A contribution to the evaluation and optimization of network reliability, Ph.D thesis, Department of Mechanical Engineering, Laval University, Canada.

Rebaiaia M-L, Ait-Kadi D., 2011, Ait-Kadi Daoud, An efficient Algorithm for Enumerating Minimal PathSets in Communication Networks, Networked Digital Technologies, Communications in Computer and Information Science, , Volume 293, Part 8, 444-458, CCIS 293, pages 444-458, Springer, 2012.

Rebaiaia Mohamed-Larbi, Ait-Kadi Daoud, 2012, Algorithms for Generating Minimal Cutsets in R-Networks, IFAC-PapersOnLine, Editor: Dolgui Alexandre & Borangui Theodor, Information Control Problems in Manufacturing, Volume # 14, Part# 1, IFAC-PapersOnLine.net, Elsevier, ISSN: 1474-6670.

Rebaiaia M-L, Ait-Kadi D, (2010), Model based Binary Decision Diagrams for Complex Networks Reliability Optimization, IFAC-PapersOnLine, Editor: Milik, Adam, Hrynkiewicz, Edward, Programmable Devices and Embedded Systems, Volume # 10, Part# 1, IFAC-PapersOnLine.net, Elsevier, ISSN: pp. 1474-6670.

Rebaiaia M-L, Ait-Kadi D., Merlano A., 2009, "A Practical Algorithm for Network Reliability Evaluation Based on the Factoring Theorem – A case study of a generic radio communication system, Journal of Quality, Vol 16, No. 5.

Rebaiaia Mohamed-Larbi, Ait-Kadi Daoud, 2012, Algorithms for Generating Minimal Cutsets in R-Networks, IFAC-PapersOnLine, Editor: Dolgui Alexandre & Borangui Theodor, Information Control Problems in Manufacturing, Volume # 14, Part# 1, IFAC-PapersOnLine.net, Elsevier, ISSN: 1474-6670.

Rebaiaia Mohamed-Larbi, Ait-Kadi Daoud, 2013, A new Algorithm for Generating Minimal Cutsets in R-Networks, AASRI Procedia, SciencesDirect, Elsevier, Vol 5, pp 67-76.

Resende, M. G. C., 1986, A program for reliability evaluation of undirected networks via polygon-to-chain reductions, *IEEE Transactions on Reliability*, 35(1), pp. 24-29.

Riordan J., 1958, An introduction to combinatorial analysis, New York, Wiley, 1958.

Rosenthal A., 1974, Computing reliability of complex systems, PhD Thesis, University of California, Berkeley, 1974.

Rosenthal A., 1977, A computer scientist looks at reliability calculations, Reliability and Fault Tree Analysis, eds. R. E. Barlow, J. B. Fussel, N. D. Singpurwalla, *SIAM J.,* pp. 133-152.

Rosenthal A.,1977, Computing the reliability of complex networks, *SIAM J. Appl. Math*. 32, pp. 384-393.

Rosenthal A. and Frisque. D, 1977, Transformations for simplifying network reliability calculations, *Networks*, ed. John Wiley & Sons, Inc, 7, pp. 97-109.

Rubino G., 1988, Network reliability evaluation, in State-of-the-art in performance modeling and simulation, K. Bagchi and J. Walrand editors, Gordon and Breach Books.

Rudell, R., 1993, Dynamic variable ordering for ordered binary decision diagrams, in *Proc. of the IEEE/ACM international conference on Computer-aided design*, 42-47.

Sahner R.A., K.S. Trivedi, 1987, Reliability Modelling using SHARPE, *IEEE Transactions on Reliability*, 36(2), pp.186-193.

Sahner R.A., K.S. Trivedi, 1993, A Software Tool for Learning About Stochastic Models', *IEEE Transactions on Education*, 36(1).

Satyanarayana A. and Chang M. K.,1983, Network reliability and the factoring theorem, *Networks*, 13, pp. 107-120.

Satyanarayana A., 1980, Minimal domination of graphs, Reliability Seminar Talk, Operations Research Center, University of California, Berkeley.

Satyanarayana A., Prabhakar A., 1980, Polygon-to-chain reductions and network reliability, *IEEE Tranactions on Reliability*, Report ORC 82-4, 1982.

Wood R. K., 1982, Polygon-to-chain reductions and extensions for reliability evaluation of undirected networks, PhD thesis, Dept. of Industrial Engineering and Operations Research, University of California, Berkeley, 1982.

Satyanarayana A., Prabhakar A., 1980, New topological formula and rapid algorithm for reliability analysis of complex systems, *IEEE Transactions on Reliability*, R-27, pp. 82-100.

Satyanarayana, A. and Wood R.K., 1985, A linear-Time Algorithm For Computing K-Terminal Reliability In series-Parallel networks, *SIAM J. Comput*., Vol. 14 (4), November, 1985.

Satyanarayana, A., 1982, A unified formula for analysis of some network reliability problems, *IEEE Transactions on Reliability*, R-31, pp. 23-32.

Shier D. R., D.E. Whited, 1985, Algorithms for generating minimal cutsets by inversion, *IEEE Transactions on Reliability*, R-34, pp. 314- 319.

Simard, C., 1996, Contribution Au Problème D'évaluation De La Fiabilité Des Réseaux, Mémoire de Maîtrise Sciences, Laval University, Québec, Canada.

Soh S. and Rai, S., 1991, CAREL: Computer Aided reliability evaluator for distribution computing networks, *IEEE Transactions of Parallel Distribution and Systems*, 2, pp. 199-213.

[20] Srinivas N. and K. Deb, "Multiobjective function optimization using nondominated sorting genetic algorithms," *Evol. Comput*., vol. 2, no. 3, pp. 221–248, 1995.

Tang Z. and J. B. Dugan, 2006, BDD-based reliability analysis of phased-mission systems with multimode failures, *IEEE Tranactions on Reliability*, 55(2), pp. 350–360,.

Tarjan, R, 1972, Depth-first search and linear graph algorithms, *SIAM Journal of Computing*, 1(2), pp. 146-160.

Theolougou, O. R. and Carlier, J. G., 1991, Factoring and reduction for networks with imperfect vertices, *IEEE Transactions on Reliability*, 40(2), pp. 210-217.

Theolougou, O. R., 1990, Contribution à l'évaluation des réseaux, thèse de doctorat, and Carlier, Université de Technologie de Compiègne, France.

Tu R. H., Rosenbaum E. and Quader K., 1993, Berkley reliability tools—BERT, *IEEE Transactions on computer-aided design of integrated circuits and systems*, 12 (10), pp. 1524-1534.

Tutte W. T., 1966, Connectivity in Graphs, University of Toronto Press, Toronto, 1966.

Valiant, L. G., 1979, The Complexity of enumerating and reliability problems, *SIAM Journal of Computing*, 8(3), pp. 410-421.

Vasely W.E, 1970, A time dependent methodology for fault tree evaluation, *Nuclear Engineering Design*, vol. 13, pp. 337-360.

Veeraraghavan M, and Trivedi K.S, 1991, An improved algorithm for symbolic reliability analysis of networks, *IEEE Transactions of Reliability*, R-40, pp 347-358.

Veeraraghavan, M, 1988, Modeling and evaluation of fault-tolerant multiple processor systems, PhD Thesis, Department of Electrical Engineering, Duke University.

Wang W. And Zhang H., 1997, The methods of Reduction in Networks Reliability Computing, *Microelectron Reliability*, 37 (3), pp-461-465.

Wood R. K, 1982, Polygon-to-Chain reductions and extensions for computing K-terminal reliability in an undirected network, ORC 82-12, Operations Research Center, University of California, Berkeley.

Wood R. K., 1982, Polygon-to-chain reductions and extensions for reliability evaluation of undirected networks, PhD thesis, Dept. of Industrial Engineering and Operations Research, University of California, Berkeley, 1982.

Wood R. K, 1986, Factoring Algorithms for Computing K-Terminal Network Reliability, *IEEE Transactions on Reliability*, R-35, NO. 3.

Wood R. K., 1985, "A factoring algorithm using polygon-to-chain reductions for computing K-terminal network reliability", *Networks*, 15, 1985, pp 173-190.

Woodhouse C., 1972, Optimal Redundancy Allocation by Dynamic Programming, IEEE Transactions Reliability, pp. 60-62.

Xie M. and Shen K., 1989, On ranking of system components with respect to different improvement actions, Microelectrom. Reliab., 29(2), pp. 159-164.

Xing L. and Dugan J. B., 2002-1, Analysis of generalized phased-mission systems reliability, performance and sensitivity, *IEEE Transactions Reliability*, vol. 51, no. 2, pp. 199–211.

Xing L. and J. B. Dugan, 2002-2, Generalized imperfect coverage phased-mission analysis, in Proc. of the Annual Reliability and Maintainability Symposium, pp. 112–119.

Yan L, Taha H, Landers L.A., 1994, Recursive Approach for Enumerating Minimal Cutsets in a Network. *IEEE Transactions on Reliability*, 43, pp. 383-388.

Yeh W-C. A Simple Heuristic Algorithm for Generating All Minimal Paths. IEEE Transactions on Reliability, 2007, 56: 488-494.

Zang X., H. Sun, and K. S. Trivedi, A BDD-based algorithm for reliability analysis of phased-mission systems, *IEEE Tranactions on Reliability*, 48, no. 1, pp. 50–60.

Zitzler E. and L. Thiele, 1999, “Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach,” *IEEE Trans. Evol. Comput*., vol. 3, no. 4, pp. 257–271.

Zitzler E.,1999, Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications, Swiss Federal Institute of Technology (ETH) Zurich, Dec. 1999. TIK-Schriftenreihe Nr. 30, Diss ETH no. 13398, Shaker Verlag, Germany, ISBN 3-8265-6831-1.