Adapting RVR simulation techniques for residual connectedness network reliability models

Héctor Cancela   María E. Urquhart

cancela@fing.edu.uy   urquhart@fing.edu.uy

Investigación Operativa - InCo - Pedeciba Informática

Facultad de Ingeniería, Universidad de la República

Montevideo, Uruguay

Abstract

The RVR (recursive variance reduction) simulation technique has been used with success for the evaluation of the $\mathcal{K}$-*terminal reliability* measure of networks where only links can fail.

In this paper, we show how this technique can be adapted for computing the $\mathcal{K}$-*terminal residual connectedness reliability* measure in the case of networks where nodes can fail. We prove that RVR simulation of residual connectedness reliability has a lower variance than standard Monte Carlo simulation, leading to better estimates.

We study the worst case computational complexity of the RVR method, and we discuss the influence of the nodes failure probability in the algorithm performance, which makes it more efficient and specially suited for very reliable netwoks.

Key words — Simulation, Monte Carlo methods, residual connectedness network reliability, recursive variance reduction.

1   Introduction

Consider an undirected communication network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$ consisting of a set of nodes $\mathcal{V}$, a set of connecting links $\mathcal{E}$ and a set of terminals $\mathcal{K}$ (a fixed subset of the node-set). We suppose that links are perfect, but that nodes can fail, and are assigned an independent probability of failure q_u(called node unreliability).

The success of communication between all surviving terminals (all the nodes in $\mathcal{K}$ which are in operating state) is a random event, which has probability $R(\mathcal{G})$. The problem of evaluating $R(\mathcal{G})$ or its complement, $Q(\mathcal{G}) = 1 - R(\mathcal{G})$, is usually called the $\mathcal{K}$-*terminal residual connectedness reliability* problem.

In graph terms, $R(\mathcal{G})$ is the probability that there is at least a path between any pair of operational nodes in $\mathcal{K}$ with all its nodes and links working. The particular case $\mathcal{K} = \mathcal{V}$ is called the *network residual connectedness reliability* problem [BSS91].

The exact evaluation of $R(\mathcal{G})$ was shown to be *NP*-hard [PB83] even in the special cases where $\mathcal{K} = V$, and where all nodes have the same failure probability. An alternative is the use of Monte Carlo simulation, which can give an estimate of the reliability measure for larger

networks. Unfortunately, the standard Monte Carlo method is not efficient when the network is very reliable, needing many iterations (then, much computing time) to obtain accurate estimates.

A number of variance reduction techniques (such as antithetic variates, stratifying and importance sampling) have been proposed in different contexts to improve the efficiency of the standard Monte Carlo method. References in this topic are [JR92], [Lom94], [Ros94], [CE96a], [Ros96], [Fis97]. To the best of our knowledge, none of them has been applied to the $\mathcal{K}$-*terminal residual connectedness reliability* problem. One possible reason is the non-monotone character of this reliability measure, which (at least for some of the most well known techniques) makes it difficult to prove that there will be any accuracy improvement at all.

For other network reliability models (in particular, source-terminal and $\mathcal{K}$-terminal reliability in networks where nodes are perfect and only links can fail), a number of works [CE95, CE98, CE96b] have shown that a variance reduction simulation technique known as RVR (Recursive Variance Reduction) has the potential for efficient evaluation of these measures. In this paper we study the application of this technique to the $\mathcal{K}$-*terminal residual connectedness reliability* problem.

The paper is organized as follows. The rest of this section presents some general notation and preliminary definitions. Section 2 recalls the standard Monte Carlo method. Section 3 proposes an RVR method adapted to the $\mathcal{K}$-*terminal residual connectedness reliability* problem. Section 4 presents the implementation and complexity analysis of the method. Finally, the conclusions of this work are presented in Section 5.

Notation (General)

$\mathcal{G}$	$(\mathcal{V}, \mathcal{E}, \mathcal{K})$: an undirected network topology
$\mathcal{V}$	$\{u_1, \ldots, u_n\}$: the node-set of $\mathcal{G}$
$\mathcal{E}$	$\{l_1, \ldots, l_m\}$: the link-set of $\mathcal{G}$
$\mathcal{K}$	*target set* of $\mathcal{G}$, $\mathcal{K} \subseteq \mathcal{V}$; the nodes of $\mathcal{K}$ are the *terminals* of $\mathcal{G}$
m,n,k	the number of [links, nodes, terminals] of $\mathcal{G}$
$|A|$	cardinality of the set A
$\mathbf{1}_{(.)}$	indicator function: $\mathbf{1}_{True} = 1$, $\mathbf{1}_{False} = 0$
x_u	$\mathbf{1}_{(\text{node } u \text{ is up})}$: state of node u
r_u	$\Pr\{x_u = 1\}$: operating probability of node u
q_u	$\Pr\{x_u = 0\} = 1 - r_u$: failure probability of node u
X	$(x_{u_1}, \ldots, x_{u_n})$: random network-state vector
$\mathcal{G}_X$	network with node-set and terminals set derived from $\mathcal{E}$ and $\mathcal{K}$ respectively by removing all failed nodes in X
$\Phi_{\mathcal{G}}$	structure function associated with the $\mathcal{K}-terminal$ *residual connectedness reliability*: $\Phi_{\mathcal{G}}(X) = \mathbf{1}_{(\mathcal{G}_X \text{ is } \mathcal{K}\text{-connected})}$ for $X \in \{0,1\}^n$
$Y(\mathcal{G})$	$\Phi_{\mathcal{G}}(X)$: random state of the network $\mathcal{G}$
$R(\mathcal{G})$	$\mathrm{E}\{\Phi_{\mathcal{G}}(X_{\mathcal{G}})\}$: $\mathcal{K}$-*terminal residual connectedness reliability* of $\mathcal{G}$
$Q(\mathcal{G})$	$1 - R(\mathcal{G})$: $\mathcal{K}$-*terminal residual connectedness unreliability* of $\mathcal{G}$

2

Definitions & Nomenclature

- A network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$ is $\mathcal{K}$−connected if there is at least one path in $\mathcal{G}$ between every pair of nodes in $\mathcal{K}$ (all the nodes in $\mathcal{K}$ are connected in $\mathcal{G}$).

- For a given node u in $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$, $\mathcal{G} - u$ denotes the network with node-set $\mathcal{V} - \{u\}$ and link-set derived from $\mathcal{E}$ by removing all links incident to u. The target set of $\mathcal{G} - u$ is equal to $\mathcal{K} - \{u\}$.

- For a given node u in $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$, $\mathcal{G}/u$ denotes the network derived from $\mathcal{G}$ by setting the operating probability of node u, $r_u = 1$.

2 Standard Monte Carlo Method

The unbiased crude Monte Carlo estimator of the reliability parameter $R(\mathcal{G})$ is a sample mean $\widehat{Y(\mathcal{G})}$. More precisely,

$$\widehat{Y(\mathcal{G})} = \frac{1}{N} \sum_{i=1}^{N} Y^{(i)} = \frac{1}{N} \sum_{i=1}^{N} \Phi_{\mathcal{G}}(X^{(i)}) \tag{1}$$

where $X^{(1)}, \ldots, X^{(N)}$ constitute a random sample of X. The variance of this estimator is

$$\mathrm{Var}\left\{\widehat{Y(\mathcal{G})}\right\} = \mathrm{Var}\left\{Y(\mathcal{G})\right\}/N = R(\mathcal{G})Q(\mathcal{G})/N \tag{2}$$

and it is estimated by the unbiased estimator

$$\widehat{V} = \widehat{Y(\mathcal{G})}(1 - \widehat{Y(\mathcal{G})})/(N-1) = \frac{1}{N(N-1)} \sum_{i=1}^{N} (Y^{(i)} - \widehat{Y(\mathcal{G})})^2. \tag{3}$$

The simulation algorithm consists of repeating independently N times the following experiment. A sample of each variable x_u is taken in order to form a sample of vector state X. The Depth First Search procedure is called to decide if all remaining working nodes in $\mathcal{K}$ are connected in the resulting subnetwork of $\mathcal{G}$. The estimation of $R(\mathcal{G})$ is the frequency of subnetworks where remaining working nodes in $\mathcal{K}$ are connected. The algorithm can be expressed as follows:

1. Initialization : $\widehat{Y} = 0$.
2. For each experiment $n = 1, \ldots, N$ do
   2.1 For each link $u = 1, \ldots, n$ do
       sample $\mathcal{U}$ from $Uniform(0,1)$;
       If $(\mathcal{U} \in [0, q_l[)$ Then $x_u = 0$ Else $x_u = 1$.
   2.2 Evaluate structure function $\Phi(X)$ and add $\Phi(X)$ to $\widehat{Y}$.
3. Compute the estimate of $R(\mathcal{G})$: $\widehat{Y} = \widehat{Y}/N$.
4. Compute the estimate of $\mathrm{Var}\left\{\widehat{Y}\right\}$: $\widehat{V} = \widehat{Y}(1 - \widehat{Y})/(N-1)$.

The well known drawback of the crude Monte Carlo method is the large sample size required to have accurate estimates when the network is highly reliable. Variance reduction techniques (such as antithetic variates, stratifying and importance sampling) have been proposed to counteract this problem. To the best of our knowledge, none of them has been applied to the $\mathcal{K}$-*terminal residual connectedness reliability* parameter. In next section, we will present the application of one particular variance reduction technique to the $\mathcal{K}$-terminal residual connectedness reliability problem, and prove that it leads to improved accuracy.

3   Recursive Variance Reduction for $\mathcal{K}$-terminal residual connectedness reliability estimation

As we saw in the precedent section, the standard ("crude") Monte Carlo method employs the random variable $Y(\mathcal{G}) = \Phi_{\mathcal{G}}(X_{\mathcal{G}})$ to estimate the residual connectedness reliability of a network $\mathcal{G}$. The aim of variance-reduction methods is to construct random variables having same expectation as $Y(\mathcal{G})$ and smaller variance, since a sample mean based on such a random variable is a more accurate estimator than the standard Monte Carlo one.

We now define a new random variable, $Z(\mathcal{G})$, as follows:

$$
Z(\mathcal{G}) = \begin{cases} 0 & \text{if } \mathcal{G} \text{ is not } \mathcal{K}\text{-connected;} \\ 1 & \text{if } \mathcal{G} \text{ is } \mathcal{K}\text{-connected and all nodes are perfect;} \\ r_u & \text{if } \mathcal{K} = \{u\} \text{ is a singleton set;} \\ R_C + (1 - R_C)Z(\mathcal{G}_V) & \text{otherwise.} \end{cases} \tag{4}
$$

where

C $\{u_1, u_2, \ldots, u_{|C|}\}$: all non-perfect nodes of $\mathcal{G}$ ($C = \{u/u \in \mathcal{V}, r_u < 1\}$)

A_C the event "all nodes in C are in working state"

R_C $\Pr\{A_C\} = \prod_{i=1}^{|C|} r_{u_i}$

B_i the event "all nodes in $\{u_1, \ldots, u_{i-1}\}$ are working and u_i is failed"

     $\Pr\{B_i\} = (1 - r_{u_i}) \prod_{j=1}^{i-1} r_{u_j}$

$\mathcal{G}_i$ $(\mathcal{G}/u_1/\ldots/u_{i-1}) - u_i$ for $1 \leq i \leq |C|$

V discrete random variable independent from random variables $Z(\mathcal{G}_i)$, with distribution

$$
\Pr\{V = v\} = \Pr\{B_v\}/(1 - R_C) = (1 - r_{u_v}) \prod_{j=1}^{v-1} r_{u_j}/(1 - R_C), 1 \leq v \leq |C|.
$$

The random variable $Z(\mathcal{G})$ applies a recursive partitioning of the network state sampling space, based on the states of the nodes in the set C of all non-perfect nodes of $\mathcal{G}$. To define the variable we make use of $|C|$ networks $\mathcal{G}_i$, $1 \leq i \leq |C|$. Each $\mathcal{G}_i$ is deduced from $\mathcal{G}$ by deleting a node and marking as perfect $(i - 1)$ nodes; the value of $Z(\mathcal{G})$ is the sum of a constant plus the value of $Z()$ applied to one (randomly chosen, according to distribution V) network $\mathcal{G}_i$.

4

We must now prove that $Z(\mathcal{G})$ has same expectation as the standard Monte Carlo estimate $Y(\mathcal{G})$ (then, it can be used to estimate the residual connectedness reliability) and smaller variance (it is more accurate than standard Monte Carlo).

We begin by proving a recursive formula for residual connectedness reliability (this formula could also be used for exact evaluation of this parameter).

Lemma 3.1 *For any $\mathcal{K}$-connected network $\mathcal{G}$ which has at least one non-perfect node, then*

$$R(\mathcal{G}) = R_C + \sum_{i=1}^{|C|} \Pr\{B_i\} R(\mathcal{G}_i) \tag{5}$$

where C, R_C, B_i, and $\mathcal{G}_i$ are as defined above.

Proof . We first observe that, as by hypothesis network $\mathcal{G}$ has at least one non-perfect node, set C is non-empty. Then the the set $\{A_C\} \cup \{B_i, u_i \in C\}$ is a partition of the network states sample space (as either all nodes in C are up, or at least one of them is failed). By applying the total probability theorem, we have

$$
\begin{aligned}
R(\mathcal{G}) \;=\; & \Pr\{\text{the graph obtained removing failed nodes from } \mathcal{G} \text{ is } \mathcal{K}\text{-connected}\} = \\
=\; & \Pr\{\text{the graph obtained removing failed nodes from } \mathcal{G} \text{ is } \mathcal{K}\text{-connected}|A_C\} \Pr\{A_C\} \\
+\; & \sum_{i=1}^{|C|} \Pr\{\text{the graph obtained removing failed nodes from } \mathcal{G} \text{ is } \mathcal{K}\text{-connected}|B_i\} \Pr\{B_i\}.
\end{aligned}
$$

But as C contains all non-perfect nodes of $\mathcal{G}$, and A_C corresponds to the case where all nodes in C are up, then the network resulting from $\mathcal{G}$ after removing all failed nodes is the same $\mathcal{G}$. As by hipothesis $\mathcal{G}$ is $\mathcal{K}$-connected then

$$\Pr\{\text{the graph obtained removing failed nodes from } \mathcal{G} \text{ is } \mathcal{K}\text{-connected}|A_C\} = 1.$$

By definition $\mathcal{G}_i = (\mathcal{G}/u_1/\ldots/u_{i-1}) - u_i$, that is to say $\mathcal{G}_i$ is the network obtained from $\mathcal{G}$ by setting nodes $u_1, \ldots, u_{i-1}$ as perfect, and deleting node u_i. Then, as B_i is the event "all nodes in $\{u_1, \ldots, u_{i-1}\}$ are working and u_i is failed",

$$
\begin{aligned}
\Pr\{\text{the graph obtained removing failed nodes from } \mathcal{G} \text{ is } \mathcal{K}\text{-connected}|B_i\} \;=\; \\
= \Pr\{\mathcal{G}_i \text{ is } \mathcal{K}\text{-connected}\} \;=\; R(\mathcal{G}_i).
\end{aligned}
$$

By substituting these values in the equation obtained by applying the total probability theorem, we complete the proof of this property.

$\square$

We now use the previous lemma to show that random variable $Z(\mathcal{G})$ has expectation equal to the residual connectedness reliability parameter.

Proposition 3.2 *For any network $\mathcal{G}$, let $Z(\mathcal{G})$ be the random variable defined in Equation 4. Then $Z(\mathcal{G})$ verifies*

$$\mathrm{E}\{Z(\mathcal{G})\} = R(\mathcal{G}) \tag{6}$$

Proof . We will proceed by induction on the number of nodes of the network.

(a) Boundary conditions:

Consider a network G which has only one node ($\mathcal{V} = \{u\}$). Then either $\mathcal{K} = \emptyset$, or $\mathcal{K} = \{u\}$.

If $\mathcal{K} = \emptyset$, the network is not $\mathcal{K}$-connected, by definition its reliability $R(G)$ is 0. Examining Definition 4 we see that $Z(G)$ is a constant, equal to 0. Consequently $Z(G)$ has expectation $R(G)$.

If $\mathcal{K} = \{u\}$, the network works if and only if u is up, which happens with probability r_u. Then the value of $R(G)$ is r_u; and by Definition 4 $Z(G)$ is a constant, equal to r_u. Consequently $Z(G)$ has expectation $R(G)$.

(b) Inductive step:

Suppose that for all networks G with number of nodes strictly smaller than n_0 the expectation of $Z(G)$ is equal to $R(G)$.

We want to show that the same holds for all networks with number of nodes $n = n_0$.

Hypothesis: For all G with number of nodes $n < n_0$,

$$\mathrm{E}\left\{Z(G)\right\} = \mathrm{E}\left\{Y(G)\right\} = R(G).$$

Thesis: For all G with number of nodes $n = n_0$,

$$\mathrm{E}\left\{Z(G)\right\} = \mathrm{E}\left\{Y(G)\right\} = R(G).$$

Proof . Let G be any network with n_0 nodes. Then one of the following four cases holds:

(i) If G is not $\mathcal{K}$-connected; by definition $R(G) = 0$, and $Z(G)$ is constant and equal to 0. Then $\mathrm{E}\left\{Z(G)\right\} = R(G)$.

(ii) If $\mathcal{G}$ is $\mathcal{K}$-connected and all nodes are perfect; by definition $R(G) = 1$, and $Z(G)$ is constant and equal to 1. Then $\mathrm{E}\left\{Z(G)\right\} = R(G)$.

(iii) If $\mathcal{K} = \{u\}$ is a singleton set; by $Z(G)$ is constant and equal to r_u, and $R(G) = r_u$. Then $\mathrm{E}\left\{Z(G)\right\} = R(G)$.

(iv) Otherwise, $\mathcal{G}$ is $\mathcal{K}$-connected, there is at least one non-perfect node ($|C| \geq 1$), and $|\mathcal{K}| > 1$. Then
$$Z(G) = R_C + (1 - R_C)Z(G_V) \tag{7}$$

where R_C, G_i and V are as defined above.

Then

$$\mathrm{E}\left\{Z(G)\right\} = R_C + (1 - R_C)\mathrm{E}\left\{Z(G_V)\right\} =$$

6

$$\begin{aligned}
&= R_C + (1 - R_C) \sum_{i=1}^{|C|} \mathrm{E}\left\{Z(G_i)\right\} \Pr\left\{V = i\right\} = \\
&= R_C + (1 - R_C) \sum_{i=1}^{|C|} \mathrm{E}\left\{Z(G_i)\right\} \Pr\left\{B_i\right\} / (1 - R_C) = \\
&= R_C + \sum_{i=1}^{|C|} \mathrm{E}\left\{Z(G_i)\right\} \Pr\left\{B_i\right\}.
\end{aligned}$$

Since all graphs G_i $G_i = (G/u_1/\ldots/u_{i-1}) - u_i$ have one less node than G, the inductive hypothesis holds giving $\mathrm{E}\left\{Z(G_i)\right\} = R(G_i)$. Substituting in the precedent equation, we have

$$\mathrm{E}\left\{Z(G)\right\} = R_C + \sum_{i=1}^{|C|} R(G_i)\Pr\left\{B_i\right\}. \tag{8}$$

Finally, applying Lemma 3.1, we arrive to

$$\mathrm{E}\left\{Z(G)\right\} = R(G). \tag{9}$$

This completes (b). □

From (a) and (b), it follows that for all networks G,

$$\mathrm{E}\left\{Z(G)\right\} = R(G).$$

□

In the next proposition we study the variance of $Z(\mathcal{G})$.

Proposition 3.3 *For any network $\mathcal{G}$, let $Z(\mathcal{G})$ be the random variable defined in Definition 4. Then $Z(\mathcal{G})$ verifies*

$$\mathrm{Var}\left\{Z(\mathcal{G})\right\} \leq R(\mathcal{G})Q(\mathcal{G}) = \mathrm{Var}\left\{Y(\mathcal{G})\right\}. \tag{10}$$

Proof. We will proceed by first observing that if either $\mathcal{G}$ is not $\mathcal{K}$-connected; $\mathcal{G}$ is $\mathcal{K}$-connected and all nodes are perfect; or $\mathcal{K} = \{u\}$ is a singleton set, then by definition $Z(\mathcal{G})$ is a constant, and consequently $\mathrm{Var}\left\{Z(\mathcal{G})\right\} = 0 \leq \mathrm{Var}\left\{Y(\mathcal{G})\right\}$.

Now we can concentrate on the remaining case, when $Z(\mathcal{G}) = R_C + (1 - R_C)Z(\mathcal{G}_V)$. Then (applying elemental properties of the variance function)

$$\mathrm{Var}\left\{Z(\mathcal{G})\right\} = \mathrm{Var}\left\{R_C + (1 - R_C)Z(\mathcal{G}_V)\right\} = (1 - R_C)^2 \mathrm{Var}\left\{Z(\mathcal{G}_V)\right\}. \tag{11}$$

We will make use of the fact that for all G, $0 \leq Z(G) \leq 1$ (this property is proved in Proposition A.1 in Appendix A). Then $0 \leq Z(\mathcal{G}_i) \leq 1$ for all i, which implies that $0 \leq Z(\mathcal{G}_V) \leq 1$. If any random variable W verifies $a \leq W \leq b$, a well-know property states that $\mathrm{Var}\left\{W\right\} \leq (b - \mathrm{E}\left\{W\right\})(\mathrm{E}\left\{W\right\} - a)$. Then

$$\mathrm{Var}\left\{Z(\mathcal{G}_V)\right\} \leq (1 - \mathrm{E}\left\{Z(\mathcal{G}_V)\right\})(\mathrm{E}\left\{Z(\mathcal{G}_V)\right\}). \tag{12}$$

As by Property 3.2

$$\mathrm{E}\left\{Z(\mathcal{G})\right\} = R_C + (1 - R_C)\mathrm{E}\left\{Z(\mathcal{G}_V)\right\} = R(\mathcal{G}),$$

then

$$\mathrm{E}\left\{Z(\mathcal{G}_V)\right\} = (R(\mathcal{G}) - R_C)/(1 - R_C);$$

substituting in Equation 12 we find

$$
\begin{aligned}
\mathrm{Var}\left\{Z(\mathcal{G}_V)\right\} &\leq (1 - (R(\mathcal{G}) - R_C)/(1 - R_C))(R(\mathcal{G}) - R_C)/(1 - R_C) \\
&= (1 - R(\mathcal{G}))(R(\mathcal{G}) - R_C)/(1 - R_C)^2.
\end{aligned}
$$

Now we substitute $\mathrm{Var}\left\{Z(\mathcal{G}_V)\right\}$ in Equation 11, and we find

$$
\begin{aligned}
\mathrm{Var}\left\{Z(\mathcal{G})\right\} &\leq 1 - R_C)^2(1 - R(\mathcal{G}))(R(\mathcal{G}) - R_C)/(1 - R_C)^2 \\
&= (1 - R(\mathcal{G}))(R(\mathcal{G}) - R_C).
\end{aligned}
$$

As $R_C \geq 0$, then $R(\mathcal{G}) - R_C \leq R(\mathcal{G})$, and

$$\mathrm{Var}\left\{Z(\mathcal{G})\right\} \leq (1 - R(\mathcal{G}))(R(\mathcal{G}) - R_C) \leq (1 - R(\mathcal{G}))R(\mathcal{G}) = \mathrm{Var}\left\{Y(\mathcal{G})\right\}. \qquad (13)$$

This completes the proof. □

4 Implementation and complexity analysis

This section presents an algorithmic description of the proposed method (procedure RVR). When this procedure is called with parameter $\mathcal{G}$, it returns a pseudo-random trial of the random variable $Z(\mathcal{G})$ defined in Equation 4 .

<div align="center">Procedure RVR($\mathcal{G}$)</div>

Input: network $\mathcal{G}$
Output: a random sample of random variable $Z(\mathcal{G})$

1. Check end recursion condition:
    1.1. Check if $\mathcal{G}$ is not $\mathcal{K}$-connected, if true return(0).
    1.2. Check if $\mathcal{G}$ is $\mathcal{K}$-connected and all nodes are perfect, if true return(1).
    1.3. Check if the network has only one terminal: If $\mathcal{K} = \{u\}$ return(r_u).
2. Find C: $C = \{u_1, \ldots, u_{|C|}\}$ the set of all non-perfect nodes.
3. Compute the probability that all components in C are operating: $R_C = \prod_{i=1}^{|C|} r_{u_i}$.
4. Generate a trial v of V (with distribution $\Pr\left\{V = i\right\} = \Pr\left\{B_i\right\}/(1 - R_C)$, $1 \leq i \leq |C|$).
5. Construct the corresponding network : $\mathcal{G}_v = (\mathcal{G}/u_1/u_2/\ldots/u_{v-1}) - u_v$.
6. Recursive step: return($R_C + (1 - R_C) \times \mathrm{RVR}(\mathcal{G}_v)$).

We look now at the worst-case complexity of procedure RVR. Steps 1.1 and 1.2 can be accomplished by a single DFS in the network (complexity $O(|\mathcal{E}|)$). Steps 2, 3, 4 and 5 take $O(|C|)$ operations; as $C \subseteq \mathcal{V}$, these steps have complexity $O(|\mathcal{V}|)$. The recursive step, 6, takes time constant plus the cost of the operations in the recursive calls. As the recursion involves a single graph, $\mathcal{G}_v = (\mathcal{G}/u_1/u_2/\ldots/u_{v-1}) - u_v$, it results that the computations involved in the recursion process can be represented by a linear computational structure. Its root corresponds to the network $\mathcal{G}$ under study, each internal node corresponds to a recursive call (task 6) and the last node presents a network that can be exactly evaluated (conditions 1.1, 1.2 and 1.3 of the above procedure). The recursion ends if $|\mathcal{K}| \leq |\mathcal{V}|$ is 1, or if $|C| = 0$. Because at each recursive step the number of nodes $|\mathcal{V}|$ of the network resulting from task 5 is diminished by 1, and the size of C is diminished by v (which is greater or equal to 1), it results that the size of the related linear structure is bounded by $|\mathcal{V}|$. The total complexity to generate a trial of $Z(\mathcal{G})$ is then $O((|\mathcal{V}| + |\mathcal{E}|) \times |\mathcal{V}|)$; if we want to express it only in terms of the size of the node set, the worst case complexity per trial is $O(|\mathcal{V}|^3)$.

In the previous analysis, we saw that the depth of recursion depended on $|C|$ and on the value of pseudo-random variable v. It is also interesting to look at the "expected" complexity, in the sense of taking into account the randomness of v. When all nodes have identical failure probabilities $q_u = q$, it can be shown that the depth of recursion increases monotonely with q (Appendix B). When $q \to 0$, the depth of recursion is bounded above by $H_{|\mathcal{V}|}$, the partial sum of the first $|\mathcal{V}|$ terms of the armonic series (this is a tight bound, in the sense that for some networks it is the exact value of the recursion depth). This series has asymptotic expansion $H_n = \ln(n) + \gamma + 1/2n - 1/12n^2 + O(1/n^4)$ ([SF96], page 169); the total complexity is then $O(\log(\mathcal{V})|\mathcal{V}|^2)$. In the case where $q \to 1$, the recursion depth is bounded above by $\mathcal{V}$ (also this value is a tight bound), and the total complexity is then $O(|\mathcal{V}|^3)$. These results show that the procedure will be more efficient for very reliable networks, when q is very close to 0.

5 Conclusions

In this paper we have presented the *$\mathcal{K}$-terminal residual connectedness reliability* measure, and we have proposed a new variance reduction method, called RVR (recursive variance reduction), which can be used to estimate this measure. We have shown that RVR is more accurate than standard ("crude") Monte Carlo simulation, leading to better estimates with less replications. We have also analyzed the worst case complexity of an implementation of the RVR procedure, and discussed the influence of the nodes failure probability in the algorithm performance.

As future work, it would be interesting to analyze the impact of the size of set C in the algorithm performance.

Acknowledgment

References

[BSS91] F. Boesch, A. Satyanarayana, and C. Suffel. On residual connectedness network reliability. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science, volume 5*, pages 51–59. 1991.

[CE95]   H. Cancela and M. El Khadiri. A recursive variance-reduction algorithm for estimating communication-network reliability. *IEEE Transactions on Reliability*, 44(4):595–602, December 1995.

[CE96a] H. Cancela and M. El Khadiri. An improvement to the total hazard method for system reliability simulation. *Probability in the Engineering and Informational Sciences*, 10(2):187–196, 1996.

[CE96b] H. Cancela and M. El Khadiri. A simulation algorithm for source-terminal communication network reliability. In *Proceedings of the 29th Annual Simulation Symposium*, pages 155–161, New Orleans, Louisiana, April 1996. IEEE Computer Society Press.

[CE98]   H. Cancela and M. El Khadiri. Series-parallel reductions in Monte Carlo network reliability evaluation. *IEEE Trans. Rel.*, 47(2):159–164, June 1998.

[Fis97] G.S. Fishman. *Monte Carlo: Concepts, algorithms and applications*. Springer, 1997.

[JR92]   C.H. Jun and S.M. Ross. System reliability by simulation: Random hazards versus importance sampling. *Probability in the Engineering and Informational Sciences*, 6, 1992.

[Lom94] M. Lomonosov. On Monte Carlo estimates in network reliability. *Probability in the Engineering and Informational Sciences*, 8:245–264, 1994.

[PB83]   J.S. Provan and M.O. Ball. The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM Journal on Computing*, 12:777–788, 1983.

[Ros94] S.M. Ross. A new simulation estimator of system reliability. *Journal of Applied Mathematics and Stochastic Analysis*, 7(3), 1994.

[Ros96] S.M. Ross. *Simulation*. Academic Press, 1996.

[SF96]   Robert Sedgewick and Philippe Flajolet. *An introduction to the analysis of algorithms*. Addison-Wesley, 1996.

A   Appendix A

We prove here a property of random variable $Z()$ which is used in the proof of Propostion 3.3.

Proposition A.1 *For any network $\mathcal{G}$, let $Z(\mathcal{G})$ be the random variable defined in 4 . Then $Z(\mathcal{G})$ verifies*

$$0 \le Z(\mathcal{G}) \le 1. \tag{14}$$

Proof .   We will proceed by induction on the number of nodes of the network.

(a) Boundary conditions:
Consider a network G which has only one node ($\mathcal{V} = \{u\}$). Then either $\mathcal{K} = \emptyset$, or $\mathcal{K} = \{u\}$. In the first case, $Z(G)$ is a constant equal to 0; in the second one, it is a constante equal to r_u (which is the operating probability of node u). Then in both cases, $0 \le Z(\mathcal{G}) \le 1$.

(b) Inductive step:
Suppose that for all networks G with number of nodes strictly smaller than n_0, the inequalities $0 \le Z(\mathcal{G}) \le 1$ are verified. We want to show that the same holds for all networks with number of nodes $n = n_0$.

Hypothesis: For all G with number of nodes $n < n_0$,

$$0 \le Z(G) \le 1.$$

Thesis: For all G with number of nodes $n = n_0$,

$$0 \le Z(G) \le 1.$$

Proof .   Let G be any network with n_0 nodes. Then one of the following four cases holds:

(i) If G is not $\mathcal{K}$-connected; by definition $R(G) = 0$, and $Z(G)$ is constant and equal to 0. Then $0 \le Z(\mathcal{G}) \le 1$.

(ii) If $\mathcal{G}$ is $\mathcal{K}$-connected and all nodes are perfect; by definition $R(G) = 1$, and $Z(G)$ is constant and equal to 1. Then $0 \le Z(\mathcal{G}) \le 1$.

(iii) If $\mathcal{K} = \{u\}$ is a singleton set; by $Z(G)$ is constant and equal to r_u, and $R(G) = r_u$. Then $0 \le Z(\mathcal{G}) \le 1$.

(iv) Otherwise, $\mathcal{G}$ is $\mathcal{K}$-connected, there is at least one non-perfect node ($|C| \ge 1$), and $|\mathcal{K}| > 1$. Then

$$Z(G) = R_C + (1 - R_C)Z(G_V) \tag{15}$$

where R_C, G_i and V are as defined above. Since all graphs $G_i = (G/u_1/ \ldots /u_{i-1}) - u_i$ have exactly one less node than G, the inductive hypothesis holds giving $0 \le Z(G_i) \le 1$, which implies that $0 \le Z(G_V) \le 1$. Then

$$Z(G) = R_C + (1 - R_C)Z(G_V) \ge R_C + (1 - R_C)0 = R_C \ge 0 \tag{16}$$

11

and

$$Z(G) = R_C + (1 - R_C)Z(G_V) \leq R_C + (1 - R_C)1 = R_C + (1 - R_C) = 1. \qquad (17)$$

This completes (b). □

From (a) and (b), it follows that for all networks G,

$$0 \leq Z(G) \leq 1.$$

□

B   Appendix B

We give in this appendix an outline of the analysis of the mean recursion depth for procedure RVR, for networks such that all nodes are either perfect (reliability 1) or can fail with probability q (reliability $r = 1 - q$).

We define $P(G)$ as the mean recursion depth for procedure RVR applied to network G.

Proposition B.1 *Let G be a network with node-set $\mathcal{V}$, edge-set $\mathcal{E}$, and terminals set $\mathcal{K}$, and let H be the complete graph with the same node-set and terminals set as G. Then $P(G) \leq P(H)$.*

Proof .   If the procedure RVR stops for network H, it must stop for network G (as if either conditions 1.1, 1.2 or 1.3 are true for a complete graph, they will be true for a subgraph); and the network H_V will be the complete graph corresponding to G_V, so that the same relation will hold for successive recursion steps. Then, the recursion depth for H will be greater or equal than for G. □

Proposition B.2 *Let G be the complete graph with node-set $\mathcal{V}$ and terminals set $\mathcal{K}$, and let H be the complete graph with the same node-set and terminals set equal to $\mathcal{V}$. Then $P(G) \leq P(H)$.*

Proof .   If the procedure RVR stops for network H, it must stop for network G (condition 1.1 never holds in a complete graph; if either conditions 1.2 or 1.3 are true for a given graph and terminals set, they will be true for the same graph with a terminals set included in the previous one); and the same relation will hold between networks H_V and G_V, and correspondingly for successive recursion steps. Then, the recursion depth for H will be greater or equal than for G. □

Proposition B.3 *Let $G_{n,p}$ be the complete graph with node-set $\mathcal{V}$ of size $n > 1$ and terminals set $\mathcal{K} = \mathcal{V}$, with $p < n$ perfect nodes and such that for all $u \in C$, $r_u = r$. Then*

$$P(G_{n,p}) = 1 + \left(\sum_{v=1}^{n-p} qr^{v-1}P(G_{n-1,p+v-1}) \right) / (1 - r^{n-p}). \qquad (18)$$

12

Proof .   This formula results directly from the definition of $Z()$ (equation 4), and from observing that $(G_{n,p}/u_1/\ldots/u_{i-1}) - u_i$ is equal to $G_{n-1,p+v-1}$ (the complete graph with one less node, and with $v-1$ more perfect nodes). $\qquad\square$

Corollary B.4 *Let $G_{n,p}$ be the complete graph with node-set $\mathcal{V}$ of size $n > 1$ and terminals set $\mathcal{K} = \mathcal{V}$, with $p < n$ perfect nodes and such that for all $u \in C$, $r_u = r$. Then*

$$P(G_{n,p}) = P(G_{n-p,0}).\tag{19}$$

Proof . This corollary results from formula 18, and can be proved by complete induction on n. $\qquad\square$

Proposition B.5 *Let $G_n = G_{n,0}$ be the complete graph with node-set $\mathcal{V}$ of size $n > 1$ and terminals set $\mathcal{K} = \mathcal{V}$, with 0 perfect nodes and such that for all $u \in C = \mathcal{V}$, $r_u = r$. Then*

$$P(G_n) = \sum_{k=1}^{n} \frac{1}{\sum_{i=0}^{k-1} r^i}.\tag{20}$$

Proof . From equations 18 and 19, we have

$$
\begin{aligned}
P(G_n) &= 1 + \left(\sum_{v=1}^{n} qr^{v-1}P(G_{n-v})\right)/(1-r^n) = \\
&= 1 + \left(qP(G_{n-1}) + r\sum_{v=2}^{n} qr^{v-2}P(G_{n-v})\right)/(1-r^{n-p}) = \\
&= 1 + \left(qP(G_{n-1}) + r(P(G_{n-1})-1)(1-r^{n-1})\right)/(1-r^n) = \\
&= 1 + (qP(G_{n-1}) + rP(G_{n-1}) - r^nP(G_{n-1}) - r + r^n))/(1-r^n) = \\
&= 1 + (P(G_{n-1}) - r^nP(G_{n-1}) - r + r^n))/(1-r^n) = \\
&= (1 - r^n - r + r^n)/(1-r^n) + P(G_{n-1})(1-r^n)/(1-r^n) = \\
&= (1-r)/(1-r^n) + P(G_{n-1}).
\end{aligned}
$$

Then
$$P(G_n) = \sum_{k=1}^{n}(1-r)/(1-r^k) = \sum_{k=1}^{n} \frac{1}{\sum_{i=0}^{k-1} r^i}\tag{21}$$

which completes the proof. From this formula, we can see that the mean depth $P()$ for complete graphs decreases monotonely with the value of r (i.e, the mean recursion depth will be smaller for more reliable networks).

When $r \to 1$, then $\sum_{i=0}^{k-1} r^i \to k$, and $P(G) \to \sum_{k=1}^{n} \frac{1}{k} = H_n$, the sum of the first n terms of the harmonic numbers series.

When $r \to 0$, then $\sum_{i=0}^{k-1} r^i \to 1$, and $P(G) \to \sum_{k=1}^{n} 1 = n$.

$\qquad\square$ rom formula

$$P(G_n) = \sum_{k=1}^{n} \frac{1}{\sum_{i=0}^{k-1} r^i} \tag{22}$$

we can see that the mean depth $P()$ for complete graphs decreases monotonely with the value of r (i.e, the mean recursion depth will be smaller for more reliable networks).

When $r \to 1$, then $\sum_{i=0}^{k-1} r^i \to k$, and $P(G) \to \sum_{k=1}^{n} \frac{1}{k} = H_n$, the sum of the first n terms of the harmonic numbers series.

When $r \to 0$, then $\sum_{i=0}^{k-1} r^i \to 1$, and $P(G) \to \sum_{k=1}^{n} 1 = n$.