PEDECIBA Informática
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay

Reporte Técnico RT 08-18

A new simulation method based on the RVR principle for the rare event *K*- network reliability problem.

Héctor Cancela **Mohamed El Khadiri** **Gerardo Rubino**

2008

A new simulation method based on the RVR principle for the rare event
K- network reliability problem.
Cancela, Héctor; El Khadiri, Mohamed; Rubino, Gerardo
ISSN 0797-6410
Reporte Técnico RT 08-18
PEDECIBA
Instituto de Computación – Facultad de Ingeniería
Universidad de la República

Montevideo, Uruguay, 2008

A NEW SIMULATION METHOD BASED ON THE RVR PRINCIPLE FOR THE RARE EVENT $\mathcal{K}$-NETWORK RELIABILITY PROBLEM

HÉCTOR CANCELA

University of the Republic of Uruguay
J. Herrera y Reissig 565, Montevideo, Uruguay

cancela@fing.edu.uy

MOHAMED EL KHADIRI

Department Gestion Logistique & Transport, IUT of Saint-Nazaire, University of Nantes
58, rue Michel Ange, Saint-Nazaire Cedex, FRANCE

mohamed.el-khadiri@univ-nantes.fr

GERARDO RUBINO

INRIA, Campus de Beaulieu, 35042 Rennes Cedex, France

gerardo.rubino@inria.fr

In this paper we consider the evaluation of a well known $\mathcal{K}$-*network unreliability* parameter by means of a new RVR Monte-Carlo method. It is based on seres–parellel reductions and a conditioning procedure using pathsets and cutsets for recursively changing the original problem into the unreliability problem for a smaller network. We illustrate by experimental results that the proposed method has good behavior in rare event cases and offers significant speed-ups over other state-of-the art variance-reduction techniques.

1 Introduction

We consider an undirected $\mathcal{K}$-connected communication network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$ where $\mathcal{V}$ is the node-set, $\mathcal{E}$ is the link-set and $\mathcal{K} \subset \mathcal{V}$ is the terminal-set (also called the target-set). Nodes do not fail, but each link can be either operational or failed. The probability of failure of link $l \in \mathcal{E}$ is $q_l = 1 - r_l$ (r_l is its reliability). With respect to failures, links behave independently from each other.

At an instant of interest, the operational links define a partial graph of $\mathcal{G}$ and we are concerned by the evaluation of the $\mathcal{K}$-*network unreliability* parameter $Q(\mathcal{G}) = 1 - R(\mathcal{G})$ which is the expectation of the binary random network state $Y(\mathcal{G})$ with value 1 if the random partial graph is not $\mathcal{K}$-connected and 0 otherwise. The number $R(\mathcal{G})$ is the $\mathcal{K}$-network reliability.

Satyanarayana and Wood [8] show that exact values of $R(\mathcal{G})$ can be obtained by a linear-time algorithm for *sp-reducible* topologies. Unfortunately, in the general case the problem is shown to be NP-hard. Consequently, Monte-Carlo methods are used for evaluating networks with large size.

For a fixed sample size N, the standard Monte Carlo (SMC) estimator $\widehat{Y}(\mathcal{G})$ is a sample mean based on N independent and identically distributed r.v. with the same distribution function as $Y(\mathcal{G})$. As it is pointed by Elperin, Gertsbakh and Lomonosov [5], the main drawback of the SMC estimator is the unbounded growth of the relative length of the ε–level confidence interval when the network becomes more and more reliable (that is, when $Q(\mathcal{G})$ approaches 0). Unbiased variance-reduction estimators have been proposed to reduce this effect. Indeed, with equal sample size N, their expectation is equal to $Q(\mathcal{G})$ and their variances are smaller than the SMC estimator, leading to smaller relative length of ε–level confidence interval than the SMC one. The reader can see, for instance, the works by Cancela and El Khadiri [1], by Elperin et al. [5], by Hui, Bean and Kraetzl [6] and by Ross [7].

The aim of our work is to propose a new efficient estimator that belongs to the variance-reduction family. It is a recursive method which uses at each call:

- a series–parallel reduction procedure which reduces the network size and preserves the (un)reliability;
- a conditioning process which transforms the evaluation of the network into the evaluation of smaller ones where series–parallel reductions may appear;
- an appropriate random selection of one of the resulting networks for continuing the recursive process.

This process terminates when it is called on trivial cases (*sp-reducible* networks or those with reliability equal to 0 or to 1). This scheme has been exploited by Cancela and El Khadiri to build recursive variance reduction (RVR) estimator [1] where the conditioning step is based on a $\mathcal{K}$-cutset of the network. As illustrated by Cancela and El Khadiri in [2], this estimator behaves very well in the source-terminal case. Unfortunately, its performance decreases when the size of $\mathcal{K}$ increases. In particular, it becomes non-competitive when it is involved for the evaluation of the all-terminal network reliability parameter, as demonstrated by examples in Section 4.

In this paper, we show how to use both a $\mathcal{K}$-pathset and a $\mathcal{K}$-cutset in the conditioning step. We illustrate on several configurations that this idea leads to a general behavior improving upon the previous RVR estimator and other, state-of-the-art, methods.

The paper is organized as follows. Section 2 is devoted to notation and definitions. Section 3 discusses how to exploit series–parallel reductions and both a $\mathcal{K}$-pathset and a $\mathcal{K}$-cutset to define a new recursive variance reduction estimator. Section 4 is devoted to some numerical illustrations and comparisons to other published methods. Some conclusions and possible improvements appear in the last section.

2 Notation and definitions

Let us begin this section by model definitions and some notation.

- $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$: an undirected network topology;
- $\mathcal{V}$: the node-set of $\mathcal{G}$;
- $\mathcal{E}$: the link-set of $\mathcal{G}$;
- $\mathcal{K} \subseteq \mathcal{V}$: the set of *terminal* nodes;
- L : event "link l is up";
- $\overline{E}$: complement of event E;
- $r_l = \Pr\{L\}$: reliability of link l;
- $q_l = \Pr\left\{\overline{L}\right\} = 1 - r_l$: unreliability of link l;
- $\mathbf{1}_E$: indicator function of the event E;
- $X(\mathcal{G}) = (\mathbf{1}_L)_{l\in\mathcal{E}}$: random network-state vector;
- $|A|$: cardinality of the set A;
- $\mathcal{G}_X$: subnetwork of $\mathcal{G}$ derived from $\mathcal{G}$ by removing all failed links in $X \in \{0,1\}^{|\mathcal{E}|}$;
- $Y(\mathcal{G})$: $\mathbf{1}_{(\mathcal{G}_{X(\mathcal{G})} \text{ is not } \mathcal{K}\text{-connected})}$: random state of the network $\mathcal{G}$;
- $Q(\mathcal{G}) = \mathrm{E}\{Y(\mathcal{G})\} = 1 - R(\mathcal{G})$: $\mathcal{K}$*-terminal unreliability* parameter of $\mathcal{G}$;
- $R(\mathcal{G}) = \mathrm{E}\{1 - Y(\mathcal{G})\}$: $\mathcal{K}$*-terminal reliability* of $\mathcal{G}$.

The RVR method presented in this paper use the following notation and definitions.

- For a subset $\mathcal{E}'$ of $\mathcal{E}$, the subnetwork $\mathcal{G}' = (\mathcal{V}, \mathcal{E}', \mathcal{K})$ of $\mathcal{G}$ is $\mathcal{K}$-connected if there is at least one path in $\mathcal{G}'$ between every pair of nodes in $\mathcal{K}$;

- A subset C of $\mathcal{E}$ is a $\mathcal{K}$-cutset of $\mathcal{G}$ if $\mathcal{G}' = (\mathcal{V}, \mathcal{E} - C, \mathcal{K})$ is not $\mathcal{K}$-connected;

- A subset C of $\mathcal{E}$ is a $\mathcal{K}$-pathset of $\mathcal{G}$ if the subnetwork $\mathcal{G}' = (\mathcal{V}, C, \mathcal{K})$ is $\mathcal{K}$-connected;

- Two links are in series if they are adjacent and they have only one common node with degree two and not belonging to $\mathcal{K}$. A series reduction consists of replacing two links l, l' in series by a single link between the non-shared extreme nodes. The operating probability of the new link is $r_l r_{l'}$;

- Two links are in parallel if they have the same extremities. A parallel reduction consists of replacing two parallel links l, l' by a single link between the same nodes. The operating probability of the new link is $r_l + r_{l'} - r_l r_{l'}$;

- $\tilde{\mathcal{G}}$ denotes the network resulting from applying all possible series and parallel reductions to the network $\mathcal{G}$;

- A $\mathcal{K}$-tree of $\mathcal{G}$ is a tree of $\mathcal{G}$ whose leaves belong to $\mathcal{K}$;

- A network $\mathcal{G}$ is *sp-reducible* if it is either a $\mathcal{K}$-tree or it can be reduced to a $\mathcal{K}$-tree by successive series and parallel reductions ($\tilde{\mathcal{G}}$ is a $\mathcal{K}$-tree);

- For a given link l in $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$, $\mathcal{G} - l$ denotes the network with node-set $\mathcal{V}$ and link-set derived from $\mathcal{E}$ by removing link l from $\mathcal{E}$. The terminal set of $\mathcal{G} - l$ is the same as the terminal set of $\mathcal{G}$;

- For a given link l in $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{K})$, $\mathcal{G} * l$ denotes the network derived from $\mathcal{G}$ by contracting link $l = \{u, v\} \in \mathcal{E}$ (eliminating l and merging its extremities u and v into a new node w).

The terminal set of $\mathcal{G} * l$ is equal to $\mathcal{K} - \{u, v\} \cup \{w\}$ if u and v belong to $\mathcal{K}$, to $\mathcal{K} - \{u\} \cup \{w\}$ if $u \in \mathcal{K}$ and $v \notin \mathcal{K}$, to $\mathcal{K} - \{v\} \cup \{w\}$ if $u \notin \mathcal{K}$ and $v \in \mathcal{K}$, and to $\mathcal{K}$ otherwise.

3 A new recursive variance reduction method

The series–parallel reduction procedure, performed in time linear on the network size, allows to evaluate exactly *sp-reducible* networks or to reduce the size of networks containing parallel or series links, without changing the network unreliability, that is $Q(\mathcal{G}) = Q(\tilde{\mathcal{G}})$. The reader can find more details about these reductions in the work by Satyanarayana and Wood [8]. Assume that $0 < Q(\mathcal{G}) < 1$. If $\mathcal{G}$ is *sp-reducible* then $Q(\mathcal{G}) = Q(\tilde{\mathcal{G}}) = 1 - \prod_{l \in \tilde{\mathcal{G}}} r_l$. Otherwise, the following proposition allows to write the network unreliability of $\mathcal{G}$ as a function of the unreliabilities of smaller networks where series and parallel links may appear. These networks are obtained by contracting and deleting links of a fixed $\mathcal{K}$-cutset and a fixed $\mathcal{K}$-pathset in $\tilde{\mathcal{G}}$.

Proposition 3.1 *For any network $\mathcal{G}$ such that $0 < Q(\mathcal{G}) < 1$ and for any pair $(C_{\tilde{\mathcal{G}}}, P_{\tilde{\mathcal{G}}})$ of $\mathcal{K}$-cutset and $\mathcal{K}$-pathset in $\tilde{\mathcal{G}}$, we have:*

$$Q(\mathcal{G}) = q_0(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|} Q(\tilde{\mathcal{G}}_i) q_i(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|} Q(\tilde{\mathcal{G}}'_i) p_i(P_{\tilde{\mathcal{G}}}) \quad (1)$$

where

- *l_1 is a a link belonging to both $C_{\tilde{\mathcal{G}}}$ and $P_{\tilde{\mathcal{G}}}$ sets (by definition of $\mathcal{K}$-pathset and $\mathcal{K}$-cutset, such a link necessarily exists);*
- *$\{l_1, l_2, \ldots, l_{|C_{\tilde{\mathcal{G}}}|}\}$ is the set of links of $C_{\tilde{\mathcal{G}}}$;*
- *$E_0 = \overline{L_1}.\overline{L_2} \ldots .\overline{L_{|C_{\tilde{\mathcal{G}}}|}}$ is the event "all links of $C_{\tilde{\mathcal{G}}} = \{l_1, l_2, \ldots, l_{|C_{\tilde{\mathcal{G}}}|}\}$ are failed";*
- *$q_0(C_{\tilde{\mathcal{G}}}) = \prod_{l \in C_{\tilde{\mathcal{G}}}} q_l$ is the probability of E_0;*
- *$E_1 = L_1$ is the event "link l_1 is operational";*

- $q_1(C_{\tilde{\mathcal{G}}}) = r_{l_1}$ *is the probability of* E_1*;*

- *for each* i*,* $2 \le i \le |C_{\tilde{\mathcal{G}}}|$*;*

  - $E_i = \overline{L_1}.\overline{L_2}\ldots.\overline{L_{i-1}}L_i$ *is the event "all links in* $\{l_1, l_2, \ldots, l_{i-1}\}$ *are failed and* l_i *is operational";*

  - $q_i(C_{\tilde{\mathcal{G}}}) = q_{l_1}q_{l_2}\ldots q_{l_{i-1}}r_{l_i}$ *is the probability of* E_i*;*

  - $\tilde{\mathcal{G}}_i = \left(\tilde{\mathcal{G}} - l_1 - l_2 - \ldots - l_{i-1}\right) * l_i$ *is the network obtained from* $\tilde{\mathcal{G}}$ *by deleting all links in* $\{l_1, l_2, \ldots, l_{i-1}\}$ *and contracting* l_i *;*

- $\{l_1, l_2^{'}, \ldots, l_{|P_{\tilde{\mathcal{G}}}|}^{'}\}$ *is the set of links of* $P_{\tilde{\mathcal{G}}}$*;*

- $E_0^{'} = L_1.L_2^{'}\ldots.L_{|P_{\tilde{\mathcal{G}}}|}^{'}$ *is the event "all links of* $P_{\tilde{\mathcal{G}}}$ *are operational";*

- $p_0(P_{\tilde{\mathcal{G}}}) = \prod_{l \,\in\, P_{\tilde{\mathcal{G}}}} r_l$ *is the probability of* $E_0^{'}$*;*

- $E_1^{'} = \overline{L_1}$ *is the event "link* l_1 *is failed";*

- $p_1(P_{\tilde{\mathcal{G}}}) = r_{l_1} = 1 - q_{l_1}$ *is the probability of* $E_1^{'}$*;*

- *for each* i*,* $2 \le i \le |P_{\tilde{\mathcal{G}}}|$*;*

  - $E_i^{'} = L_1.L_2^{'}\ldots.L_{i-1}^{'}\overline{L_i^{'}}$ *is the event "all links in* $\{l_1, l_2^{'}, \ldots, l_{i-1}^{'}\}$ *are operational and* $l_i^{'}$ *is failed";*

  - $p_i(C_{\tilde{\mathcal{G}}}) = r_{l_1}r_{l_2^{'}}\ldots r_{l_{i-1}^{'}}q_{l_i^{'}}$ *is the probability of* $E_i^{'}$*;*

  - $\tilde{\mathcal{G}}_i^{'} = \left(\tilde{\mathcal{G}} * l_1 * l_2^{'} * \ldots * l_{i-1}^{'}\right) - l_i^{'}$ *is the network obtained from* $\tilde{\mathcal{G}}$ *by contracting all links in* $\{l_1, l_2^{'}, \ldots, l_{i-1}^{'}\}$ *and deleting* $l_i^{'}$*.*

Proof. Since both $(E_i)_{0\le i\le |C_{\tilde{\mathcal{G}}}|}$ and $(E_i^{'})_{0\le i\le |P_{\tilde{\mathcal{G}}}|}$ constitute partitions of the state space, the events $(\overline{L_1}.E_i)_{0\le i\le |C_{\tilde{\mathcal{G}}}|}$ and $(L_1.E_i^{'})_{0\le i\le |P_{\tilde{\mathcal{G}}}|}$ are exhaustive and mutually exclusive. As $\overline{L_1}.E_1 = \emptyset$,

$\overline{L_1}.E_i = E_i$, for $i \neq 1$, $L_1.E'_1 = \emptyset$ and $\overline{L_1}.E'_i = E'_i$, for $i \neq 1$, we deduce that $(E_i)_{2 \le i \le |C_{\tilde{\mathcal{G}}}|}$ and $(E'_i)_{2 \le i \le |P_{\tilde{\mathcal{G}}}|}$ form a set of exhaustive and mutually exclusive events. The total expectation theorem leads to

$$\begin{aligned} Q(\mathcal{G}) = \mathrm{E}\left\{Y(\tilde{\mathcal{G}})\right\} &= \mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E_0(C_{\tilde{\mathcal{G}}})\right\}\Pr\left\{E_0(C_{\tilde{\mathcal{G}}})\right\} + \sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|}\mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E_i(C_{\tilde{\mathcal{G}}})\right\}\Pr\left\{E_i(C_{\tilde{\mathcal{G}}})\right\} \\ &+ \mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E'_0(P_{\tilde{\mathcal{G}}})\right\}\Pr\left\{E'_0(P_{\tilde{\mathcal{G}}})\right\} + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|}\mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E'_i(P_{\tilde{\mathcal{G}}})\right\}\Pr\left\{E'_i(P_{\tilde{\mathcal{G}}})\right\}. \end{aligned}$$

When all links of a $\mathcal{K}$-cutset are failed, the network is always not $\mathcal{K}$-connected and when all links of a $\mathcal{K}$-pathset are operational, the network is always $\mathcal{K}$-connected. It results that $\mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E_0(C_{\tilde{\mathcal{G}}})\right\} = 1$ and $\mathrm{E}\left\{Y(\tilde{\mathcal{G}})/E'_0(P_{\tilde{\mathcal{G}}})\right\} = 0$. Based on these results and above notation and definitions, the above equality becomes

$$Q(\mathcal{G}) = q_0(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|} Q(\tilde{\mathcal{G}}_i) q_i(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|} Q(\tilde{\mathcal{G}}'_i) p_i(P_{\tilde{\mathcal{G}}})$$

which corresponds to Expression (1). □

Transformation (1) can be used to compute exactly $Q(\mathcal{G})$ as follows :

$$Q(\mathcal{G}) = \begin{cases} 1 & \text{if } \mathcal{G} \text{ is not } \mathcal{K}\text{–connected;} \\ 0 & \text{if } \mathcal{K} \text{ is a singleton set;} \\ 1 - \prod_{l \in \tilde{\mathcal{G}}} r_l & \text{if } \mathcal{G} \text{ is } \textit{sp-reducible}; \\ q_0(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|} q_i(C_{\tilde{\mathcal{G}}}) Q(\tilde{\mathcal{G}}_i) + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|} p_i(P_{\tilde{\mathcal{G}}}) Q(\tilde{\mathcal{G}}'_i) & \text{otherwise} \end{cases} \quad (2)$$

The computations involved by the recursive function (2) can be represented by a tree structure with leaves correspond to calls with *sp-reducible* networks, or networks having unreliability equal to 0 or equal to 1. Because the exact evaluation problem is in the NP–hard class, we exploit Expression (2) for deriving the recursive random variable

$$Z(G) \begin{cases} 1 & \text{if } \mathcal{G} \text{ is not } \mathcal{K}\text{–connected;} \\ 0 & \text{if } \mathcal{K} \text{ is a singleton set;} \\ 1 - \prod_{l \in \tilde{\mathcal{G}}} r_l & \text{if } \mathcal{G} \text{ is } \textit{sp-reducible}; \\ q_0(C_{\tilde{\mathcal{G}}}) + \alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}) \left(\sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|} \mathbf{1}_{(\mathcal{U} \in J_i(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))} Z(\tilde{\mathcal{G}}_i) + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|} \mathbf{1}_{(\mathcal{U} \in J'_i(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))} Z(\tilde{\mathcal{G}}'_i) \right) & \text{otherwise} \end{cases} \quad (3)$$

where

- $\alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}) = \sum_{i=2}^{|C_{\tilde{\mathcal{G}}}|} q_i(C_{\tilde{\mathcal{G}}}) + \sum_{i=2}^{|P_{\tilde{\mathcal{G}}}|} p_i(P_{\tilde{\mathcal{G}}})$ is the probability of the event $(\bigcup_{i=2}^{|C_{\tilde{\mathcal{G}}}|} E_i) \bigcup (\bigcup_{i=2}^{|P_{\tilde{\mathcal{G}}}|} E_i')$;

- $(J_i(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))_{2 \leq i \leq |C_{\tilde{\mathcal{G}}}|}$ and $(J_i'(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))_{2 \leq i \leq |P_{\tilde{\mathcal{G}}}|}$ form a partition of $[0,1]$ (the length of $J_i(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})$ is $q_i(C_{\tilde{\mathcal{G}}})/\alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})$ and the length of $J_i'(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})$ is $p_i(P_{\tilde{\mathcal{G}}})/\alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})$);

- $\mathcal{U}$ is a random variable with uniform distribution on $[0,1]$, s-independent of all random variables used in the recursive process.

and we propose to use the sample mean $\widehat{Z}(\mathcal{G})$,

$$\widehat{Z}(\mathcal{G}) = \sum_{i=1}^{N} Z^{(i)}(\mathcal{G})/N, \tag{4}$$

based on N s-independent samples of $Z(\mathcal{G})$ for estimating $Q(\mathcal{G})$.

Remark 3.2 *The above ideas have been used by Cancela and El Khadiri in [1] to build a similar recursive r.v. $F(\mathcal{G})$ which transforms the problem into identical problems related to networks obtained by contracting and deleting links of a fixed $\mathcal{K}$-cutset of the network, instead of using both $\mathcal{K}$-cutset and $\mathcal{K}$-pathset as above. They proved that*

$$\mathrm{E}\{F(\mathcal{G})\} = Q(\mathcal{G}) \tag{5}$$

and

$$\mathrm{Var}\{F(\mathcal{G})\} \leq (Q(\mathcal{G}) - q_0(C_{\tilde{\mathcal{G}}}'))R(\mathcal{G}) \leq Q(\mathcal{G})R(\mathcal{G}) = \mathrm{Var}\{Y(\mathcal{G})\} \tag{6}$$

where $C_{\tilde{\mathcal{G}}}'$ is the $\mathcal{K}$-cutset used at the first call to the recursive process and $q_0(C_{\tilde{\mathcal{G}}}') = \prod_{l \in C_{\tilde{\mathcal{G}}}'} q_l$ is the probability of the event "all links of $C_{\tilde{\mathcal{G}}}'$ are failed". The proof procedure employed for that case in [1] can also be used to show that the r.v. $Z(\mathcal{G})$ defined in (3) verifies

$$\mathrm{E}\{Z(\mathcal{G})\} = Q(\mathcal{G}) \tag{7}$$

and

$$\mathrm{Var}\left\{Z(\mathcal{G})\right\} \leq \left(Q(\mathcal{G}) - q_0(C_{\tilde{\mathcal{G}}})\right)\left(R(\mathcal{G}) - p_0(P_{\tilde{\mathcal{G}}})\right) \leq Q(\mathcal{G})R(\mathcal{G}) = \mathrm{Var}\left\{Y(\mathcal{G})\right\}. \tag{8}$$

Results (7) and (8) imply that the sample mean $\widehat{Z}(\mathcal{G})$, is an unbiased estimator of $\mathcal{G}$ with variance smaller than the variance of the SMC estimator, thus leading to a variance-reduction method.

Let us now describe how the trial $Z(\mathcal{G})^{(k)}$ is generated when $\mathcal{G}$ is not trivial (that is, when $Q(\mathcal{G}) \neq 1$ and $Q(\mathcal{G}) \neq 0$). First, a series-parallel reductions procedure is called to reduce the network's size. If the network is sp-reducible, the unreliability of the sampled network is $1 - \prod_{l\in\tilde{\mathcal{G}}} r_l$. Else, after determining a $\mathcal{K}$-cutset and a $\mathcal{K}$-pathtset of $\tilde{\mathcal{G}}$, intervals $(J_i(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))_{2\leq i\leq|C_{\tilde{\mathcal{G}}}|}$ and $(J_i'(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}}))_{2\leq i\leq|P_{\tilde{\mathcal{G}}}|}$ are deduced and a trial of $\mathcal{U}$ is generated. As it belongs to only one of the $(|C_{\tilde{\mathcal{G}}}| + |P_{\tilde{\mathcal{G}}}| - 2)$ intervals, only the associated term survives in the two sums of (3). Let $\mathcal{G}_s$ be the surviving network among the $(|C_{\tilde{\mathcal{G}}}| + |P_{\tilde{\mathcal{G}}}| - 2)$ networks involved in (3). Then $Z(\mathcal{G})^{(k)}$ is obtained from the formula

$$Z(\mathcal{G})^{(k)} = q_0(C_{\tilde{\mathcal{G}}}) + \alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})Z(\mathcal{G}_s)^{(k)} \tag{9}$$

where $Z(\mathcal{G}_s)^{(k)}$ is a trial of $Z(\mathcal{G}_s)$. If $\mathcal{G}_s$ has deterministic behavior (its unreliability is equal to 1 or 0), or is sp-reducible, the process terminates by returning $Z(\mathcal{G})^{(k)} = q_0(C_{\tilde{\mathcal{G}}}) + \alpha(P_{\tilde{\mathcal{G}}}, C_{\tilde{\mathcal{G}}})Q(\mathcal{G}_s)$. Otherwise, we apply to $\mathcal{G}_s$ the same procedure that we applied to $\mathcal{G}$. The related 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 and the last node presents a network having a deterministic behavior. Because at each recursive step the number of links of the resulting network is diminished by at least 1, the size of the related structure is bounded by $|\mathcal{E}|$.

Remark 3.3 *From inequality (8), we can deduce that the ratio*

$$Q(\mathcal{G})R(\mathcal{G}) / \left(Q(\mathcal{G}) - q_0(C_{\tilde{\mathcal{G}}})\right)\left(R(\mathcal{G}) - p_0(P_{\tilde{\mathcal{G}}})\right)$$

represents a lower bound on the variance-reduction ratio of the proposed estimator with respect to the SMC one. This bound, always greater than 1, depends on the $\mathcal{K}$-pathset $P_{\tilde{\mathcal{G}}}$ and $\mathcal{K}$-cutset $C_{\tilde{\mathcal{G}}}$ chosen at the first call and it is maximal when $P_{\tilde{\mathcal{G}}}$ and $C_{\tilde{\mathcal{G}}}$ have largest $p_0(P_{\tilde{\mathcal{G}}})$ and $q_0(C_{\tilde{\mathcal{G}}})$ respectively. On the other hand, as at each new recursive call this choice is operated on a new network, the variance-reduction ratio achieved by the method will depend on the strategy adopted for selecting $\mathcal{K}$-pathset and $\mathcal{K}$-cutset. Numerical illustrations presented in Section 4 correspond to a version using the Breadth First Search procedure for obtaining a $\mathcal{K}$-pathset and using the set of adjacent links to one of the nodes in $\mathcal{K}$ as a $\mathcal{K}$-cutset. A Breadth First Search algorithm can be found in the book of Cormen, Leiserson and Rivest [4].

4 Numerical comparisons to efficient Monte-Carlo methods

To illustrate the interest of the RVR-PC method, based on the estimator $\widehat{Z}(\mathcal{G})$ (4), we present some numerical comparisons to the following three methods, which, up to our knowledge, are among the most efficient ones in the published literature: the RVR-C method based on the estimator $\widehat{F}(\mathcal{G})$ proposed by Cancela and El Khadiri [1, 2], the Merge Process method (MP) proposed by Elperin et al. [5] and the method which uses the Cross-Entropy technique to further improve the Merge Process performances (CE-MP) proposed by Hui et al. [6]. For the examples, we consider the following networks, which have been chosen because previous publications have included numerical results that we will use in the comparisons:

- grid networks G_3 and G_6 (see Figure 1), where links are assigned equal unreliability $q = 10^{-3}$ or $q = 10^{-6}$ and $\mathcal{K}$ is the set of the four corner nodes as in the work of Hui et al. [6];
- complete networks C_{10}, C_{15}, C_{20}, C_{25} and C_{30} where links are assigned equal unreliability $q = 0.55$ and $\mathcal{K} = \mathcal{V}$ as in in the work of Elperin et al. [5] (C_n has n nodes and $n(n-1)/2$

links).

For those networks exact values of $Q(\mathcal{G})$ are tabulated at column 3 of Table 1. Exact unreliabilities of grid networks are given by Hui et al. in [6] and for complete topolgies, we used a Maple program to compute them based on the following recursive formula given by Colbourn [3]: if $\mathcal{Q}(C_n) = u_n$ and if the common unreliability of the lines is q, then $u_1 = 0$, $u_2 = q$ and for $n \geq 3$,

$$u_n = \sum_{j=1}^{n-1} \binom{j-1}{n-1} q^{j(n-j)}(1-u_j),$$

Each exact unreliability $Q(\mathcal{G})$ serves

- in the computation of the relative error parameter

  $$RE_{\widehat{W}}(\%) = 100 \times |\widehat{W} - Q(\mathcal{G})|/Q(\mathcal{G}) \tag{10}$$

  which helps to analyze the quality of the estimates produced by a sample mean estimator $\widehat{W}$;

- for checking if the TLC-based 95%–level confidence interval

  $$CI_{\widehat{W}} = \left]\widehat{W} - 1.96\sqrt{\mathrm{Var}\left\{\widehat{W}\right\}}, \widehat{W} + 1.96\sqrt{\mathrm{Var}\left\{\widehat{W}\right\}}\right[\tag{11}$$

  associated with the estimate of $Q(\mathcal{G})$ by the sample mean estimator $\widehat{W}$ contains or not the exact reliability $Q(\mathcal{G})$ (let us recall that if the sample size N is large enough, the central limit theorem implies that the probability that $Q(\mathcal{G})$ belongs to $CI_{\widehat{W}}$ is greater than 95%);

- in the computation of the exact variances of the SMC estimator; these variances are needed to compute the variance-reduction ratios

  $$VRR_{\widehat{W}} = \mathrm{Var}\left\{\widehat{Y}(\mathcal{G})\right\}/\mathrm{Var}\left\{\widehat{W}\right\} = Q(\mathcal{G})(1-Q(\mathcal{G}))/(N \times \mathrm{Var}\left\{\widehat{W}\right\}). \tag{12}$$

  For a fixed sample size N, if $VRR_{\widehat{W}} = v$, then the length of the confidence interval related to the estimate obtained by the estimator $\widehat{W}$ is $\sqrt{v}$ times smaller than the length of SMC's confidence interval.

For a fixed sample size N, the best method in terms of accuracy is the one leading to a 95%–level confidence interval covering the exact value and such that its variance-reduction ratio with respect to the SMC method is the largest one (the length of its confidence interval is the smallest one). As $\mathrm{Var}\left\{\widehat{Z}(\mathcal{G})\right\}$ and $\mathrm{Var}\left\{\widehat{F}(\mathcal{G})\right\}$ are unknown, their unbiased estimators

$$\widehat{V}_Z = \sum_{i=1}^{N} (Z^{(i)}(\mathcal{G}) - \widehat{Z}(\mathcal{G}))^2/(N-1)N \tag{13}$$

and

$$\widehat{V}_F = \sum_{i=1}^{N} (F^{(i)}(\mathcal{G}) - \widehat{F}(\mathcal{G}))^2/(N-1)N \tag{14}$$

are used in combination with the exact variances of the SMC estimator given in column 4 of Table 1, in order to calculate the corresponding variance-reduction ratios. For MP and CE-MP methods, results given by their authors are used.

To consider both accuracy and execution time, we consider the relative efficiency parameter $W_{M/SMC}$ (sometimes called speed-up). The relative efficiency is the product of the variance ratio defined in (12) and the time ratio T_{SMC}/T_M where T_M denote the mean execution time of method M. This parameter is a natural efficiency indicator, as it has two very intuitive interpretations: if we consider that we run SMC for the same total average execution time as M, then the variance of the estimator by SMC will be $W_{M/SMC}$ larger than the one of the estimator computed by M. Alternatively, if we run SMC for a number of iterations enough to obtain the same variance as M, then SMC computing effort will be, on the average, $W_{M/SMC}$ times higher than the M's one. Execution times of the SMC, RVR-PC and RVR-C methods correspond to our implementations using the Microsoft Visual C++ 6.0 language and a computer based on the Intel Celeron processor with 2.66 GHZ frequency and 256 Mo of RAM memory. We tabulate in column 5 of Table 1 the mean time par trial for the SMC method. Then, we only give time ratios of the RVR-PC and RVR-C methods with respect to the SMC one, in the corresponding tables. For the MP method, we report speed-ups given by Elperin et al. [5]. For the CE-MP method, we report only variance-reduction ratios (execution times are not considered by Hui et al. in [6]).

4.1 Comparison of RVR-C, RVR-PC and CE-MP methods on grid topologies

As in the work of Hui et al. [6], where the estimator based on Merge Process and Cross-Entropy techniques is proposed, we use the sample size $N = 10^6$, networks G_3 and G_6, the terminals are the corner nodes and the common link unreliability is $q = 10^{-3}$ and $q = 10^{-6}$.

Table 2 shows the results obtained by RVR-C. From column 5 we can see that for both grids, G_3 and G_6, RVR-C is able to give confidence intervals covering the exact $Q(\mathcal{G})$ value when the link unreliabilities are 10^{-3}; but when the link unreliabilities are fixed to 10^{-6}, dramatically increasing the rareness of the network failures, RVR-C confidence intervals do not include $Q(\mathcal{G})$. The variance reduction rates with respect to SMC are modest for $q = 10^{-3}$; the large values shown for 10^{-6} are not significant, as the estimation is not correct in that case. Additional experiments with larger sample sizes (up to 10^9 iterations) show that, while the estimated relative errors are always in the 50% range, the confidence interval still do not cover the exact value.

The RVR-PC method gives much better results, as shown in Table 3. The exact values are always within the confidence intervals, the relative errors are good for the four cases and there are large variance-reduction ratios and speed-ups obtained over SMC.

The CE-MP method's results are presented in Table 4; here also the estimations are consistent, the confidence intervals cover the exact values, and good variance-reduction ratios are achieved. The comparison between RVR-PC and CE-MP show that the former still enjoys an small advantage, in terms of variance-reduction ratios. As execution times are not given by Hui et al. in [6], we can't compare CE-MP and RVR-PC methods with respect to speed-ups values.

4.2 Comparison of RVR-C, RVR-PC and MP methods on complete topologies

As in the work of Elperin et al. [5] where the estimator based on Merge Process is proposed, we consider complete topologies C_n for $n = 10, 15, 20, 25, 30$, with common link unreliability $q = 0.55$. All nodes are considered as terminals (the measure evaluated is the all-terminal reliability) and the sample size is $N = 10^4$.

We show in Table 5 the results obtained by employing RVR-C. We observe that relative errors grow when the networks get increasingly more reliable (then, when failures are increasingly rare events). Moreover, for the C_{25} and C_{30} topologies, relative errors exceed 90% and the confidence intervals do not contain the exact unreliabilities. For these topologies, we can conclude that a sample size 10^4 is not enough to obtain estimates with reasonable relative errors when $Q(\mathcal{G})$ is of order 10^{-6} or smaller. The fourth column of the table shows the variance reduction ratios; it is clear that for the three first topologies, the method only improves marginally upon SMC; the large values for the last two topologies are meaningless, as the method did not give a good estimation of $Q(\mathcal{G})$.

Table 6 shows the results obtained by the proposed new method, RVR-PC. We can see now that RVR-PC gets increasingly better results when the networks are more reliable; the relative error obtained using the same sample size $N = 10^4$ not only are always smaller than the corresponding ones by the RVR-C method, but they also are reasonably good for the C_{25} and C_{30} topologies. This is confirmed by looking at column 5, where we can see that all confidence intervals cover the exact value of $Q(\mathcal{G})$. This table analyzes two additional topologies, C_{40} and C_{50}, which were included to confirm that the behavior of RVR-PC was robust when considering still more reliable topologies. Columns 4 and 6 show the variance-reduction ratio and the relative efficiency with respect to the SMC method. These indicators increase when $Q(\mathcal{G})$ decreases and illustrate that

RVR-PC offers substantial gains.

The performances of the MP method are shown in Table 7. The estimations for the first four topologies are good, but in the case of the most reliable topology considered, C_{30}, the confidence interval does not contain the exact value and the estimation itself is an order of magnitude away from $Q(\mathcal{G})$ (this might also be due to a mistype in the source of these values. The MP code was not available to repeat these experiments and confirm). For the cases where the estimation is good, the relative errors, variance-reduction ratios and speed-ups with respect to the SMC method are slightly worse than the ones abtained by RVR-PC.

Comparing the results shown by the three tables, we can see that RVR-PC is the most robust technique, giving estimations close to the true $Q(\mathcal{G})$ values for all the considered topologies, and obtaining the best precisions for a given sample size, as well as the best relative efficiency values. The MP method, albeit also giving good results, has consistently lower variance reduction ratios and relative efficiency figures. The RVR-C method does not obtain significant improvements over SMC, and for the most reliable networks is not able to give dependable estimates when run with a 10^4 sample size.

5 Conclusions

The exact evaluation of the $\mathcal{K}$-*network unreliability* parameter belongs to the family of NP–hard problems. An alternative is to perform estimations by Monte Carlo simulation methods. In the rare event case (highly reliable networks), the standard Monte Carlo method is also expensive when accurate estimates are required. For such cases, variance-reduction methods must be used. We proposed here a new recursive variance-reduction method based on series-parallel reductions and a decomposition procedure which simultaneously exploits both a $\mathcal{K}$-pathset and a $\mathcal{K}$-cutset in order to transform the original problem into similar ones on smaller networks. This method is a modification of a previous RVR method where only $\mathcal{K}$-cutsets were involved at each recursive call.

As illustrated on many configurations used in the published literature, the new RVR method not only improves significantly upon the old one but it is also competitive when compared to the Merge process Monte Carlo method and to a method which uses the Cross-Entropy technique to improve the Merge Process performances.

Numerical illustrations presented here correspond to a version using the Breadth First Search procedure for obtaining a $\mathcal{K}$-pathset and using the set of adjacent links to one of the nodes in $\mathcal{K}$ as a $\mathcal{K}$-cutset. As the lower bound on the variance-reduction ratio with respect to the SMC one depends on the (un)reliabilities of the links in the $\mathcal{K}$-pathset and the $\mathcal{K}$-cutset selected from the original network and at each new recursive call this choice is operated on a new network, the variance-reduction ratio achieved by the method will depend on the strategy adopted for selecting them. A future extension of this work is the search of an optimal strategy.

References

[1] H. Cancela and M. El Khadiri. Series-Parallel Reductions in Monte-Carlo Network Reliability Evaluation. *IEEE Transactions on Reliability*, 47(2):159–164, Juin 1998.

[2] H. Cancela and M. El Khadiri. The Recursive Variance-Reduction Simulation Algorithm for Network Reliability Evaluation. *IEEE Transactions on Reliability*, 52(2):207–212, Juin 2003.

[3] C. J. Colbourn. *The Combinatorics of Network Reliability.* Oxford University Press, 1987.

[4] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to Algorithms.* The MIT Press, Cambridge, Massachusetts, 1990.

[5] T. Elperin, I. Gertsbakh, and M. Lomonosov. Estimation of network reliability using graph evolution models. *IEEE Transactions on Reliability*, 40(5):572–581, December 1991.

[6] K.P. Hui and N.Bean and M. Kraetzl and D.P. Kroese. The Cross-Entropy Method for Network Reliability Estimation", *Annals of Operations Research*,134: 101–118, 2005.

[7] S.M. Ross. A new simulation estimator of system reliability. *Journal of Applied Mathematics and Stochastic Analysis*, 7(4):331–336, 1994.

[8] A. Satyanarayana and R.K. Wood. A linear-time algorithm for computing k-terminal reliability in series-parallel networks. *SIAM Journal on Computing*, 14(4):818–832, November 1985.

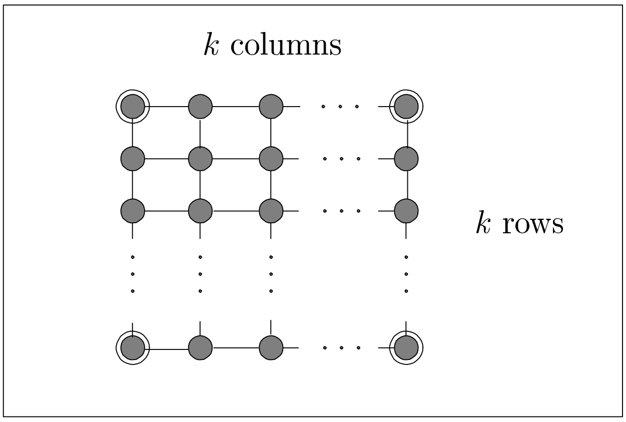

Figure 1: G_k: the Grid Network Topology

Network	q	$Q(\mathcal{G})$[6]	$N \times \text{Var}\left\{\widehat{Y}(\mathcal{G})\right\} = \text{Var}\{Y(\mathcal{G})\} = Q(\mathcal{G})R(\mathcal{G})$
G_3	10^{-3}	4.01199×10^{-06}	4.01197×10^{-6}
G_3	10^{-6}	4.00001×10^{-12}	4.00001×10^{-12}
G_6	10^{-3}	4.00800×10^{-06}	4.00798×10^{-6}
G_6	10^{-6}	4.00001×10^{-12}	4.00001×10^{-12}
C_{10}	0.55	4.58481×10^{-02}	4.37460×10^{-02}
C_{15}	0.55	3.47489×10^{-03}	3.46281×10^{-03}
C_{20}	0.55	2.33295×10^{-04}	2.33240×10^{-04}
C_{25}	0.55	1.46772×10^{-05}	1.46770×10^{-05}
C_{30}	0.55	8.86419×10^{-07}	8.86418×10^{-07}
C_{40}	0.55	2.99368×10^{-09}	2.99368×10^{-08}
C_{50}	0.55	9.47855×10^{-12}	9.47855×10^{-12}

Table 1: Exact unreliabilities of networks used for numerical illustrations, exact variances of network random state $Y(\mathcal{G})$ and mean execution time of one trial of $Y(\mathcal{G})$.

Network	q	Estimate of $Q(\mathcal{G})$	RE(%)	VRR	$Q(\mathcal{G}) \in$ 95%-CI ?	$W_{RVR-C/SMC}$
G_3	10^{-3}	3.01102×10^{-06}	$2.50 \times 10^{+01}$	$3.01 \times 10^{+00}$	Yes	$5.18 \times 10^{+01}$
G_3	10^{-6}	1.99996×10^{-12}	$5.00 \times 10^{+01}$	$2.00 \times 10^{+18}$	No	
G_6	10^{-3}	3.00402×10^{-06}	$2.50 \times 10^{+01}$	$3.00 \times 10^{+00}$	Yes	$1.22 \times 10^{+02}$
G_6	10^{-6}	1.99996×10^{-12}	$5.00 \times 10^{+01}$	$2.00 \times 10^{+18}$	No	

Table 2: Performances of the RVR-C method for the evaluation of G_3 and G_6. Terminals are the four corner nodes and $N = 10^6$.

Network	q	Estimate of $Q(\mathcal{G})$	RE (%)	VRR	$Q(\mathcal{G}) \in$ 95%-CI ?	$W_{RVR-PC/SMC}$
G_3	10^{-03}	4.01321×10^{-06}	3.04×10^{-02}	$4.49 \times 10^{+05}$	Yes	$1.21 \times 10^{+07}$
G_3	10^{-6}	4.00130×10^{-12}	3.22×10^{-02}	$1.44 \times 10^{+11}$	Yes	$3.72 \times 10^{+12}$
G_6	10^{-03}	4.01758×10^{-06}	2.39×10^{-01}	$1.12 \times 10^{+05}$	Yes	$5.17 \times 10^{+04}$
G_6	10^{-06}	4.00894×10^{-12}	2.23×10^{-01}	$1.10 \times 10^{+11}$	Yes	$5.31 \times 10^{+10}$

Table 3: Performances of the RVR-PC method for the evaluation of G_3 and G_6. Terminals are the four corner nodes and $N = 10^6$.

Network	q	Estimate of $Q(\mathcal{G})$ [6]	RE (%)	VRR[6]	$Q(\mathcal{G}) \in$ 95%-CI ?
G_3	10^{-3}	4.01172×10^{-06}	6.73×10^{-03}	$2.17 \times 10^{+05}$	Yes
G_3	10^{-6}	3.99876×10^{-12}	3.12×10^{-02}	$2.16 \times 10^{+11}$	Yes
G_6	10^{-3}	4.00239×10^{-06}	1.40×10^{-01}	$1.07 \times 10^{+05}$	Yes
G_6	10^{-6}	3.99869×10^{-12}	3.30×10^{-01}	$1.06 \times 10^{+11}$	Yes

Table 4: Performances of the MP-CE method for the evaluation of G_3 and G_6. Terminals are the four corner nodes and $N = 10^6$.

Network	Estimate of $Q(\mathcal{G})$	RE (%)	VRR	$Q(\mathcal{G}) \in$ 95%-CI ?	$W_{RVR-C/SMC}$
C_{10}	4.50953×10^{-02}	$1.64 \times 10^{+00}$	$2.43 \times 10^{+00}$	Yes	$2.07 \times 10^{+01}$
C_{15}	3.70929×10^{-03}	$6.74 \times 10^{+00}$	$1.95 \times 10^{+00}$	Yes	$3.04 \times 10^{+00}$
C_{20}	2.43334×10^{-04}	$4.30 \times 10^{+00}$	$1.93 \times 10^{+00}$	Yes	$1.36 \times 10^{+00}$
C_{25}	1.17419×10^{-06}	$9.20 \times 10^{+01}$	$4.47 \times 10^{+14}$	No	
C_{30}	5.90947×10^{-08}	$9.33 \times 10^{+01}$	$4.21 \times 10^{+18}$	No	

Table 5: Performances of the RVR-C method for the evaluation of complete networks. $\mathcal{K} = \mathcal{V}$, the common link unreliability q is equal to 0.55 and the sample size N is equal to 10^4.

Network	Estimate of $Q(\mathcal{G})$	RE (%)	VRR	$Q(\mathcal{G}) \in$ 95%-CI ?	$W_{RVR-PC/SMC}$
C_{10}	4.58982×10^{-02}	1.09×10^{-01}	$2.72 \times 10^{+02}$	Yes	$8.16 \times 10^{+02}$
C_{15}	3.47094×10^{-03}	1.14×10^{-01}	$5.54 \times 10^{+03}$	Yes	$2.16 \times 10^{+03}$
C_{20}	2.33403×10^{-04}	4.64×10^{-02}	$1.10 \times 10^{+05}$	Yes	$1.03 \times 10^{+04}$
C_{25}	1.46809×10^{-05}	2.51×10^{-02}	$2.11 \times 10^{+06}$	Yes	$7.44 \times 10^{+04}$
C_{30}	8.85976×10^{-07}	5.00×10^{-02}	$4.27 \times 10^{+07}$	Yes	$7.67 \times 10^{+05}$
C_{40}	2.99422×10^{-09}	1.82×10^{-02}	$1.69 \times 10^{+09}$	Yes	$1.54 \times 10^{+07}$
C_{50}	9.48517×10^{-12}	6.98×10^{-02}	$6.47 \times 10^{+12}$	Yes	$3.43 \times 10^{+10}$

Table 6: Performances of the RVR-PC method for the evaluation of complete networks. $\mathcal{K} = \mathcal{V}$, the common link unreliability q is equal to 0.55 and the sample size N is equal to 10^4.

Network	Estimate of $Q(\mathcal{G})$ [5]	RE(%)	VRR[5]	$Q(\mathcal{G}) \in$ 95%-CI ?	$W_{MP/SMC}$[5]
C_{10}	4.56×10^{-02}	5.41×10^{-01}	$7.00 \times 10^{+01}$	Yes	$2.10 \times 10^{+01}$
C_{15}	3.46×10^{-03}	4.28×10^{-01}	$8.87 \times 10^{+02}$	Yes	$1.69 \times 10^{+02}$
C_{20}	2.32×10^{-04}	5.55×10^{-01}	$1.52 \times 10^{+04}$	Yes	$3.28 \times 10^{+03}$
C_{25}	1.47×10^{-05}	1.55×10^{-01}	$2.70 \times 10^{+05}$	Yes	$4.72 \times 10^{+04}$
C_{30}	8.89×10^{-06}	$9.03 \times 10^{+02}$	$5.08 \times 10^{+07}$	No	

Table 7: Performances of the MP method [5] for the evaluation of complete networks. $\mathcal{K} = \mathcal{V}$, the common link unreliability q is equal to 0.55 and the sample size N is equal to 10^4.