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

Reporte Técnico RT 10-12

Polynomial-time topological reductions that preserve the diameter constrained reliability of a communication network

Héctor Cancela Mohamed El Khadiri Louis Petingi

2010

Polynomial-time topological reductions that preserve the diameter constrained reliability of a communication network
Cancela, Héctor; El Khadiri, Mohamed; Petingi, Louis
ISSN 0797-6410
Reporte Técnico RT 10-12
PEDECIBA
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay, 2010

Polynomial-time topological reductions that preserve the diameter constrained reliability of a communication network

HÉCTOR CANCELA
Universidad de la República
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

LOUIS PETINGI
Computer Science Department
College of Staten Island
City University of New York
2800 Victory Boulevard Staten Island, N.Y. 10314, USA
louis.petingi@csi.cuny.edu

Summary: In this paper, we propose a polynomial-time algorithm for detecting and deleting edges of a network which are irrelevant in the evaluation of the Source-to-terminal Diameter Constrained Network reliability parameter. As evaluating this parameter is known to be an NP-hard problem, the proposed procedure may lead to important computational gains when combined with an exact method to calculate the reliability. Experimental results are shown, corroborating the predicted computational reward, when this reliability preserving algorithm is integrated within an exact factorization approach based upon Moskowitz's edge decomposition theorem and applied to evaluate the Source-to-terminal Diameter Constrained reliability of specific topologies.
Resumen: En este trabajo se propone un algoritmo de tiempo polinomial para detectar y eliminar aristas de una red que son irrelevantes para el cálculo del parmetro de confiabilidad fuente-terminal con restricciones de dimetro en una red. Como la evaluacin de este par—'ametro es un problema NP-difcil, el procedimiento propuesto puede resultar en una importante ganancia computacional cuando se combina con un m—'etodo exacto para calcular la confiabilidad. Los resultados experimentales que se incluyen corroboran la ganancia computacional predicha, cuando el mtodo de reduccin es integrado dentro de un enfoque de factorizacin exacto, basado en el teorema de descomposicin en aristas de Moskowitz, y utilizado para evaluar la confiabilidad con restricciones de dimetro de algunas topologas especficas.
Key Words: Network reliability, diameter constraints, paths, factoring, topological reductions.

1 Introduction

The system under study is a communication network modeled by an undirected graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ consisting of a set of nodes (vertices) $\mathscr{V}$, a set of edges (links) $\mathscr{E}$, and a distinguished set $\mathscr{K} = \{s,t\}$ of nodes called terminal nodes (also called participating nodes). In this model nodes do not fail, but each edge l of $\mathscr{G}$ is assigned an independent probability of failure q_l (called the unreliability). The unreliability q_l is the one's complement of the reliability r_l (i.e., the probability of survival) of l.
In the classical reliability measure, the network is supposed to work if after the removal of the failed edges, the nodes s and t can be connected by at least an operational path. This is a random event that has probability $R_{st}(\mathscr{G})$. The problem of evaluating $R_{st}(\mathscr{G})$ is called the Source-to-terminal Network Reliability problem (see [1] for further information regarding this reliability model).
There are many situations where it is not enough that the terminal nodes are connected after the removal of the failed edges, but the quality of the communication depends on the existence of a path connecting s and t, whose length (measured as the number of edges) is bounded by a given integer D. The Source-to-terminal Diameter Constrained Network reliability measure, denoted by $R_{st}(\mathscr{G}, D)$, is the probability of this event and it was originally introduced by Petingi and Rodriguez in 2001 [14], for any arbitrary set of terminals $\mathscr{K} \subseteq \mathscr{V}$. As real networks are subject to failures, the Source-to-terminal Diameter Constrained reliability may be shown to be useful in different contexts. For example, this measure gives an indicator of the suitability of an existing network topology to support good quality voice over IP applications between a pair of terminal nodes. In wireless networks, $R_{st}(\mathscr{G}, D)$ could represent the probability that two participating nodes (e.g., access points in a network, terminals, internet server, etc.) would be able to communicating thru at most D hops.
As $R_{st}(\mathscr{G}, |\mathscr{V}|-1) = R_{st}(\mathscr{G})$ (since any path in a network is composed of at most $|\mathscr{V}|-1$ edges), and since the evaluation

of $R_{st}(\mathscr{G})$ is an NP-hard problem [1], the evaluation of $R_{st}(\mathscr{G},D)$ is also an NP-hard problem as well. This implies that it is very unlikely that a general algorithm exists to evaluate the Source-to-terminal Diameter Constrained reliability in polynomial time on the size of a network. As a consequence any tool which transforms an instance of a topology into a smaller one, in polynomial time, would be of relevant importance. Contributions in this direction were proposed for the classical Source-to-terminal Network reliability problem [16,17], and applied for the exact reliability evaluation [6,7, 13,15,18,20,21], as well as to estimate the reliability measure by application of Monte Carlo techniques [3–5,11]. This paper is devoted to a procedure which detects, in polynomial-time, irrelevant edges of a given network $\mathscr{G}$ with respect to the $R_{st}(\mathscr{G},D)$ measure (i.e., the edge e is irrelevant if $R_{st}(\mathscr{G},D) = R_{st}(\mathscr{G}-e,D)$).
The paper is organized as follows. In the following section we introduce notation and definitions pertaining to the Source-to-terminal Diameter Constrained reliability measure. In Section 3 we present combinatorial properties which are useful to identifying some irrelevant edges. Based upon these properties, an algorithm which detects and deletes these edges from a network is then introduced. In Section 4 this algorithm is integrated within a factoring method based upon the decomposition equation of Moskowitz [12] which allows to express the reliability of a network $\mathscr{G}$ in terms of the reliability of two networks obtained from $\mathscr{G}$ by fixing the state of an arbitrary edge e either operational (i.e., reliability of the edge is set to one) or failed (i.e., its reliability is set to zero). Section 5 is devoted to present experimental results which highlight the contributions and ideas suggested in this paper. Finally, in Section 6, we present conclusions and future work.

2 Model definitions and notation

This section presents definitions and notation pertaining to the Source-to-terminal Diameter Constrained reliability that will be employed in the remainder of the paper:

- $\mathscr{G} = (\mathscr{V},\mathscr{E})$ an undirected graph modeling a communication network;
- $\mathscr{V}$: the node-set of $\mathscr{G}$;
- $\mathscr{E}$: the edge-set of $\mathscr{G}$;
- $\mathscr{K} = \{s,t\} \subseteq \mathscr{V}$: the set of *terminal* nodes;
- For two nodes u and v in $\mathscr{V}$,
  – An uv-path of $\mathscr{G}$ is defined as a sequence of distinct nodes $(u = n_1, n_2, n_3, \ldots, n_L, n_{L+1} = v)$ such that $n_i n_{i+1}, 1 \leq i \leq L$, is an edge of $\mathscr{G}$. Moreover the length of this path is L.
  - The distance between u and v,

$$d(u,v) = \begin{cases} \text{length of shortest } uv\text{-path}, & \text{if } \mathscr{G} \text{ contains } uv\text{-paths.} \\ +\infty & \text{otherwise.} \end{cases} \tag{1}$$

- D : the diameter constraint;
- A D-st-path is an st-path whose length is at most D;
- For an edge $e \in \mathscr{E}$,

$$r_e = \begin{cases} 1 & \text{if } e \text{ is always operational.} \\ 0 & \text{if } e \text{ always fails.} \\ \text{the probability that edge } e \text{ is operational.} & \text{otherwise.} \end{cases} \tag{2}$$

- $q_e = (1 - r_e)$: the probability that the edge e fails (unreliability of edge e);
- $R_{st}(\mathscr{G},D) = \Pr\{\, \mathscr{G}$ contains at least one D-st-path with only operational edges$\}$: the Source-to-terminal Diameter Constrained Network reliability;
- With respect to $R_{st}(\mathscr{G},D)$, an edge $e \in \mathscr{E}$ is relevant if it belongs to at least one D-st-path; otherwise e is irrelevant;

From the definition of $R_{st}(\mathscr{G},D)$, it follows that if an edge e does not belong to any D-st-path of $\mathscr{G}$ then $R_{st}(\mathscr{G},D) = R_{st}(\mathscr{G}-e,D)$, thus e is then irrelevant.

3 Identification of irrelevant edges with respect to the $R_{st}(\mathscr{G},D)$ evaluation problem

As evaluating $R_{st}(\mathscr{G},D)$ is an NP-hard problem, any polynomial algorithm which allows to reduce the size of a topology while preserving its reliability parameter is of practical importance. This section is devoted to proposing an efficient algorithm that detects a class of irrelevant edges. As these edges are not useful within the context of the reliability's calculation of a network, they may be deleted while preserving the parameter $R_{st}(\mathscr{G},D)$. For illustration purposes suppose that we need to compute $R_{st}(G_1,5)$ (i.e., $D = 5$) for the network G_1 depicted in Figure 1. One can see that edges ab, bs, and as do not belong to any st-path, while the edges cd and ef belong to st-paths of length greater than 5; consequently these edges are irrelevant since they do not lie on any D-st-path and they can be removed without altering the value of the reliability (i.e., $R_{st}(G_1,5) = R_{st}(G_2,5)$).
The following proposition plays an important role in determining a class of irrelevant edges.

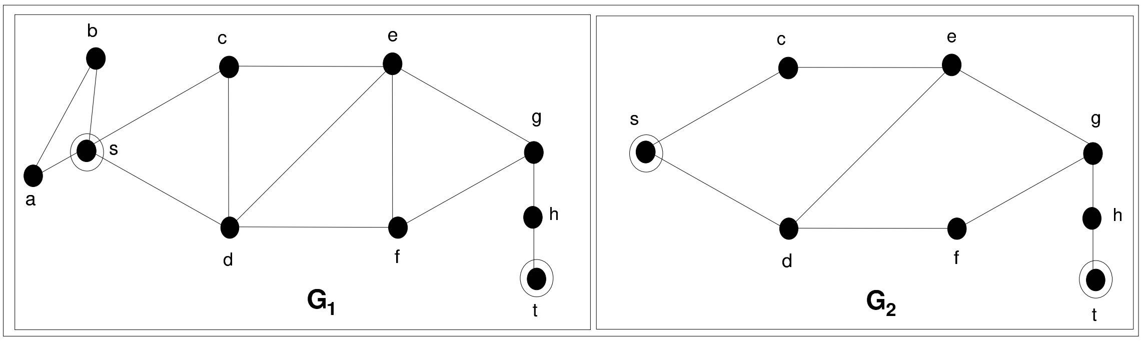

Fig. 1 Irrelevant edges. Edges ab, bs, as, cd, ef of G_1 are irrelevant when $D = 5$ and they can be removed without altering the reliability.

Proposition 1 *Let uv be an edge which belongs to some st-path of $\mathscr{G}$. If $d(s,u)+d(v,t) \geq D$ and $d(s,v)+d(u,t) \geq D$, then uv is not relevant.*

Proof Suppose that uv is relevant, then at least one of the following conditions holds:

- uv belongs to a path $(s = n_1, n_2, \ldots, n_{i-1}, u, v, n_{i+2}, \ldots, n_{L+1} = t)$, with length $L \leq D$,
- vu belongs to a path $(s = n'_1, n'_2, \ldots, n'_{i-1}, v, u, n'_{i+2}, \ldots, n'_{L'+1} = t)$, with length $L' \leq D$.

As a consequence

- $d(s,u)+d(v,t) \leq L-1 < D$, or
- $d(s,v)+d(u,t) \leq L'-1 < D$.

By contraposite if $(d(s,u)+d(v,t) \geq D)$ and $(d(s,v)+d(u,t) \geq D)$ then uv is not relevant.

A procedure to remove from $\mathscr{G}$ irrelevant edges that do not belong to any st-path as well as edges identified by Proposition 1, can be summarized as follows:

Procedure IrrelevantEdgesReduction$(\mathscr{G}, st, D)$

Input: network $\mathscr{G} = (\mathscr{V}, \mathscr{E})$, set $\mathscr{K} = \{s,t\}$, diameter D
Output: network $\tilde{\mathscr{G}}$

1. Delete edges that do not belong to any st-path.
2. For each vertex v compute $d(s,v)$ and $d(t,v)$.
3. Delete all edges uv such that $d(s,u)+d(t,v) \geq D$ and $d(s,v)+d(u,t) \geq D$.
4. If there are still edges that do not belong to any st-path, go to Step 1
   else return $\tilde{\mathscr{G}}$.

In Step 1, edges that do not belong to any st-path are identified by the application of a linear-time algorithm based on biconnectivity theory (see [9,19]), and then deleted. The other type of irrelevant edges are edges that belong to some st-path, but not to any D-st-path. In Step 2, the distances of the nodes to both s and t are calculated by application of either Dijkstra's shortest-path algorithm [8] or by the well known linear-time BFS procedure [10] of time-complexity $O(|\mathscr{E}|+|\mathscr{V}|)$. In Step 3, the distances computed in the previous step are used to identify irrelevant edges, following the conditions stated in Proposition 1, and then deleted from the graph. Finally, as deletion of irrelevant edges mentioned in the previous step may destroy some st-paths, the resulting graph may yield edges that do not lie on any st-path, thus the biconnectivity test mentioned in Step 1 must be conducted again. As all steps have in the worst case linear-time complexity and at most $|\mathscr{E}|$ edges can be deleted, then the time-complexity of the procedure IrrelevantEdgesReduction() is $O(|\mathscr{E}|(|\mathscr{E}|+|\mathscr{V}|))$.
Let us now illustrate how the procedure $IrrelevantEdgesReduction()$ works on the network shown in Figure 1.

Step 1: Edges ab, as and bs are removed as they do not belong to any st-path. Nodes a and b become isolated.
Step 2: The distances $d(s,v)$ and $d(v,t)$ are computed for every node v: $d(s,s) = 0$, $d(s,c) = 1$, $d(s,d) = 1$, $d(s,e) = 2$, $d(s,f) = 2$, $d(s,g) = 3$, $d(s,h) = 4$, $d(s,t) = 5$, $d(c,t) = 4$, $d(d,t) = 4$, $d(e,t) = 3$, $d(f,t) = 3$, $d(g,t) = 2$, $d(h,t) = 1$ and $d(t,t) = 0$. Moreover $d(s,a) = d(s,b) = d(a,t) = d(b,t) = \infty$.
Step 3: For each edge uv, we check if $d(s,u)+d(v,t) \geq D$ and $d(s,v)+d(u,t) \geq D$ (in this example D is equal to 5).

- For cd we have: $d(s,c)+d(d,t) = 1+4 = 5 \geq D$ and $d(s,d)+d(c,t) = 1+4 = 5 \geq D$. Then cd is irrelevant and it will be deleted from G_1.
- Similarly the edge ef is identified as irrelevant and it is deleted from G_1.
- For the remaining edges, the conditions stated in Proposition 1 are not satisfied and therefore these edges are not deleted from G_1.

Step 4: All edges from the resulting graph after execution of Step 3 lie on some st-path, consequently the procedure terminates. The resulting graph G_2 is returned with $R_{st}(G_2, 5) = R_{st}(G_1, 5)$.

The distances inequalities stated in Proposition 1 and employed in the algorithm described above are sufficient conditions for identifying irrelevant edges, but not necessary ones; as exemplified by the network shown in Figure 2, with respect to the measure $R_{st}(\mathscr{G}, D)$, when $D = 6$. By examination of this topology, it follows that edge ab belongs to some st-path, $d(s,a) = 1$, and $d(b,t) = 2$, yielding the inequality $d(s,a) + d(b,t) = 3 < D$; but this edge does not belong to any D-st-path, as the only two D-st-paths are (s,a,t) and (s,a,d,e,f,t). Therefore the edge ab is irrelevant and it can be eliminated from the network, even if it does not satisfy the conditions stated in Proposition 1. The determination of the existence of necessary conditions to identify irrelevant edges efficiently from a computational viewpoint remains an open problem.

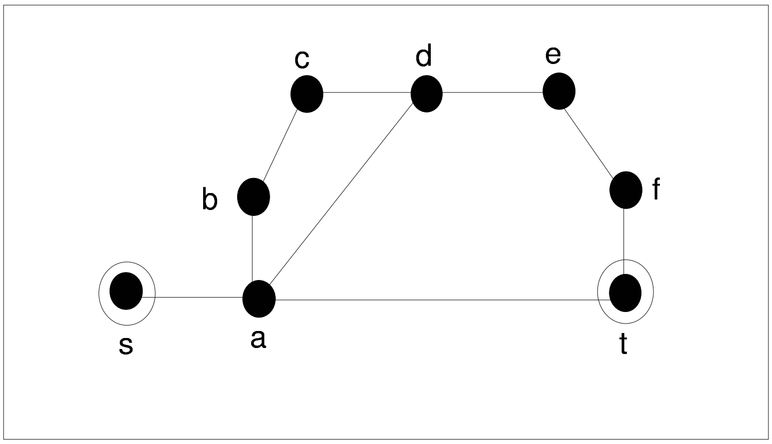

Fig. 2 Topology with irrelevant edges not identified by Proposition 1.

4 Integration of the reduction procedure within an exact evaluation context

In this section we present a Factoring algorithm to calculate the Source-to-terminal Diameter Constrained reliability of a network $\mathscr{G}$, which includes the procedure $\mathsf{IrrelevantEdgesReduction}()$, presented in Section 3, in order to identify and delete irrelevant edges of a network. Given that the edges of $\mathscr{G}$ have been assigned independent operational probabilities, we say that the random state of an edge e is *undetermined* if $0 < r_e < 1$.
The following proposition is based on the decomposition equation introduced by Moskowitz [12], which offers a method to express the reliability of a network as a function of the reliabilities of the two networks obtained from $\mathscr{G}$ by fixing the state of a selected edge e either up (i.e., e is operational) or down (i.e., e failed). Moskowitz decomposition was extensively used within the context of the classical reliability $R_{st}(\mathscr{G})$ (see [6, 7, 13, 15, 18, 20, 21]).

Proposition 2 *For any network $\mathscr{G}$ that has at least one edge e whose random state is undetermined then*

$$R_{st}(\mathscr{G}, D) = r_e R_{st}(\mathscr{G} * e, D) + (1 - r_e) R_{st}(\mathscr{G} - e, D) \tag{3}$$

where

- *e is an edge with undetermined random state in $\mathscr{G}$ (i.e., $0 < r_e < 1$).*
- *$\mathscr{G} * e$ is the network obtained from $\mathscr{G}$ by fixing the edge e as operational (i.e., let $r_e = 1$).*
- *$\mathscr{G} - e$ is the network obtained from $\mathscr{G}$ by fixing edge e as failed (i.e., let $r_e = 0$, or equivalently e is deleted).*

A procedure $\mathsf{Factoring}()$ can be implemented by application of the recursive function established by Equation (3) which describes a binary tree in which each node j of this tree represents a subgraph of $\mathscr{G}$, $\mathscr{G}_j$ (the root node represents the original network $\mathscr{G}$), in which its edges are either operational, have failed, or whose random states are undetermined. For each of the possible subgraphs $\mathscr{G}_j$'s, its Source-to-terminal Diameter Constrained reliability is then calculated as:

$$R_{st}(\mathscr{G}_j, D) = \begin{cases} 0 & \text{if there is no } D\text{-}st\text{-path in } \mathscr{G}_j. \\ 1 & \text{if } \mathscr{G}_j \text{ contains a } D\text{-}st\text{-path with only oper. edges.} \\ r_e R_{st}(\mathscr{G}_j * e, D) + (1 - r_e) R_{st}(\mathscr{G}_j - e, D) & \text{otherwise, where } e \text{ has undetermined random state.} \end{cases} \tag{4}$$

We can integrate the procedure IrrelevantEdgesReduction(), described in Section 3, within Factoring(), to possibly delete irrelevant edges in each of the states $\mathscr{G}_j$ of the binary tree generated by the application of the recursive function stated by Equation 3. The inclusion of this procedure serves a two-folded purpose:

i) By possibly reducing the size of an state, the procedure will shorten the computational effort needed for its evaluation.
ii) Suppose that an irrelevant edge e is to be considered for a particular state $\mathscr{G}_j$. Then, as the random state of e does not influence the network reliability, we have that $R_{st}(\mathscr{G}_j, D) = R_{st}(\mathscr{G}_j * e, D) = R_{st}(\mathscr{G}_j - e, D)$. But $R_{st}(\mathscr{G}_j, D) = r_e R_{st}(\mathscr{G}_j * e, D) + (1 - r_e) R_{st}(\mathscr{G}_j - e, D)$; thus eliminating e allows to divide this computational effort in half.

Formally, the following procedure, called FactoringWithReduction(), calculates the Diameter Constrained reliability of a network while enforcing the deletion of the irrelevant edges identified by the procedure IrrelevantEdgesReduction():

Procedure FactoringWithReduction($\mathscr{G}, s, t, D, RedIsPoss$)

Input: network $\mathscr{G} = (\mathscr{V}, \mathscr{E})$, s, t, D, and $RedIsPoss$
Output: the parameter $R_{st}(\mathscr{G}, D)$
1. Check end recursion condition:
   1.1. If $\mathscr{G}$ contains a D-st-path having only operational edges return(1).
   1.2. If there is no D-st-path in $\mathscr{G}$ return(0).
2. Apply reduction procedure if irrelevant edges may exist: If (RedIsPoss = 1) IrrelevantEdgesReduction($\mathscr{G}, st, D$).
3. Select randomly an edge e in $\mathscr{G}$ with undetermined state.
4. Solve recursively for $\mathscr{G} - e$: $R_{st}(\mathscr{G} - e, D) :=$ (FactoringWithReduction($\mathscr{G} - e, s, t, D, 1$)).
5. Solve recursively for $\mathscr{G} * e$: $R_{st}(\mathscr{G} * e, D) :=$ (FactoringWithReduction($\mathscr{G} * e, s, t, D, 0$)).
6. Compute $R_{st}(\mathscr{G}, D)$: return($R_{st}(\mathscr{G}, D) = (1 - r_e) R_{st}(\mathscr{G} - e, D) + r_e R_{st}(\mathscr{G} * e, D)$).

It is interesting to note that the fact of fixing an edge e down (i.e., $r_e = 0$) transforms the topology of the problem, and some previously relevant edges could become irrelevant, opening the door for further simplifications of the network; this is expressed by setting the $RedIsPoss$ flag equal to one. On the other hand, when fixing an edge up, the topology is unchangeable, then no new simplifications will occur ($RedIsPoss = 0$). This flag will determine when the procedure IrrelevantEdgesReduction() should be executed, saving relevant computational effort.

5 Experimental results

In this section we illustrate numerical results to reflect the computational gain obtained in computing the Source-to-terminal Diameter Constrained reliability of different topologies by comparing the execution time taken by application of the FactoringWithReduction() method, and the execution time taken by the Factoring() procedure described in the previous section. For these tests, comparisons were performed on three classes of topologies: 1) the Dodecahadron, 2) the 5 by 5 Grid topology, and 3) the Circulant topology with n nodes and jumps 1 and $n/2$, denoted as $C^n_{1,n/2}$ (see Fig. 3); moreover comparisons were performed taking into account different values for the diameter constraint D. In addition highly-dense topologies such as complete graphs on n nodes (denoted as K_n) were also tested.

$\mathscr{G}$	$\{s,t\}$	r_e	D	$1 - R_{\{s,t\}}(\mathscr{G}, D)$	Tree's size Fact	CPU time (s) Fact	Tree's size FactWRed	CPU time (s) FactWRed
$C^{20}_{1,10}$	$\{1,20\}$	0.9	5	$1.529019999999999e-02$	$8.7031807000e+07$	535	$9.100000000e+01$	<1
$C^{20}_{1,10}$	$\{1,20\}$	0.9	9	$1.370034110439999e-02$	$1.0167910300e+08$	654	$1.329700000e+04$	1
$C^{20}_{1,10}$	$\{1,20\}$	0.9	13	$3.328128111167163e-03$	$1.3448759300e+08$	920	$1.539617700e+07$	111
$C^{20}_{1,10}$	$\{1,20\}$	0.9	17	$3.283248606737214e-03$	$1.3489784700e+08$	960	$1.570485500e+07$	114
Dod	$\{1,20\}$	0.9	5	$1.230716705876657e-02$	$4.1937978300e+08$	5656	$4.055900000e+04$	<1
Dod	$\{1,20\}$	0.9	9	$2.923769913020631e-03$	$5.8604278300e+08$	7524	$3.732774300e+07$	373
Dod	$\{1,20\}$	0.9	13	$2.879975697913531e-03$	$5.8563926100e+08$	10257	$4.301811500e+07$	422
Dod	$\{1,20\}$	0.9	17	$2.879601513657377e-03$	$5.8552952500e+08$	11548	$4.311722700e+07$	426
K_9	$\{1,9\}$	0.9	2	$8.938717389999e-07$	$1.1979456511e+10$	29595	$1.167000000e+03$	1
K_9	$\{1,9\}$	0.9	4	$2.000012525263e-08$	$2.2968666710e+09$	5713	$2.095771757e+09$	11971
K_9	$\{1,9\}$	0.9	6	$2.000012504139e-08$	$2.1920252150e+09$	5126	$2.089202149e+09$	10051
K_9	$\{1,9\}$	0.9	8	$2.000012504139e-08$	$2.1912126710e+09$	4736	$2.088560265e+09$	9926

Table 1 Experimental results for measuring the effect of the IrrelevantEdgesReduction() procedure when combined to the factoring equation of Moskowitz.

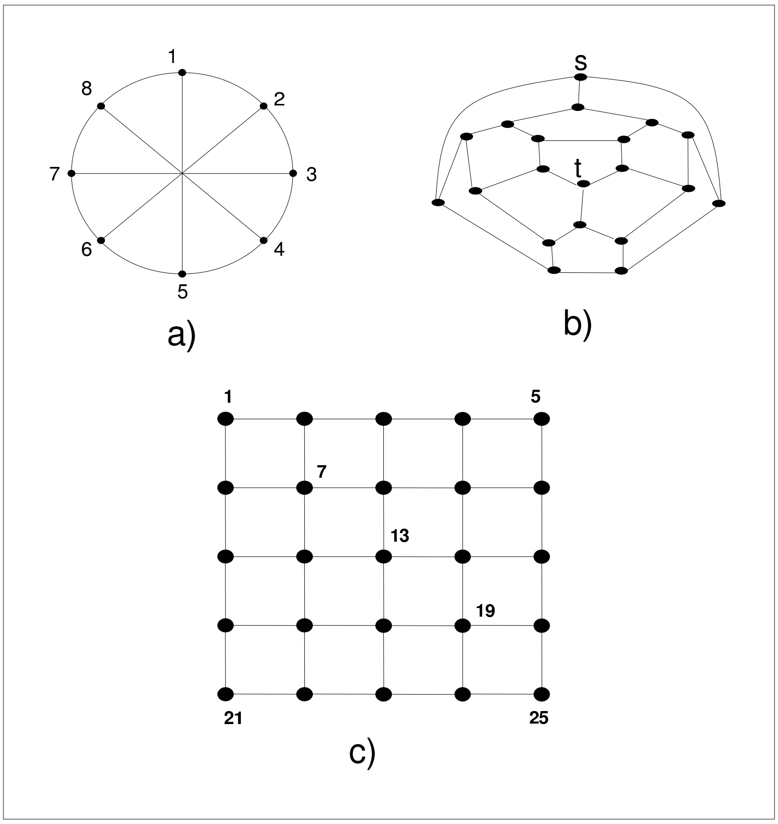

Fig. 3 Illustration of three topologies: a) Circulant $C^8_{1,4}$, b) Dodecahedron, and c) 5 by 5 Grid.

$\mathscr{G}$	$\{s,t\}$	r_e	D	$1-R_{\{s,t\}}(\mathscr{G},D)$	Tree's size Fact	CPU time (s) Fact	Tree's size FactWRed	CPU time (s) FactWRed
5x5GRID	$\{1,21\}$	0.999	8	$2.008010993794891e-006$	-	>2 days	$2.662850000e+05$	2
5x5GRID	$\{1,5\}$	0.999	8	$2.008010993794890e-06$	-	>2 days	$1.958150000e+05$	2
5x5GRID	$\{1,7\}$	0.999	8	$1.002002033838198e-06$	-	>2 days	$3.222610000e+05$	4
5x5GRID	$\{1,13\}$	0.999	8	$1.002003018090846e-06$	-	>2 days	$3.696286700e+07$	538
5x5GRID	$\{1,19\}$	0.999	8	$1.002006016230354e-06$	-	>2 days	$9.793337850e+08$	13506
5x5GRID	$\{1,25\}$	0.999	8	$2.004007123796960e-06$	-	>2 days	$2.311453991e+09$	28929

Table 2 Experimental results for measuring the effect of the IrrelevantEdgesReduction() procedure when combined to the factoring equation of Moskowitz on the the 5 X 5 Grid topology. Execution times more than 2 days were aborted.

The methods FactoringWithReduction() and Factoring() were implemented in C++ and experiments were conducted on a 2.4 Ghz Celeron computer, running on XP operating system.

Table 1 and Table 2 reflect the computational results obtained, and the data shown in columns 1 thru 5 represent the type of topology, the label of the source and terminal nodes of the topology, the unique operational probability assigned to each edge, the diameter bound D, and the value obtained by both procedures for the Source-to-terminal Diameter Constrained unreliability (the complement to one of the reliability), respectively. Columns 6 and 7, and 8 and 9, represent the tree's size and CPU time taken by Factoring() and by FactoringWithReduction() procedures in order to compute the unreliability of a topology, respectively. The tree's size is the number of nodes in the binary tree produced by the recursive calls executed by each procedure, as explained in Section 4.

The results yielded by these tables show a consistent computational gain obtained when elimination of irrelevant edges were integrated within the factoring procedure, especially when low-dense topologies were tested (e.g., Circulants, Dedocahedron, and the 5 by 5 Grid topologies). From Table 1, it is also possible to observe that for topologies composed on n nodes, the computational gain obtained when detecting irrelevant edges is particularly important for low values of the diameter bound D, and it becomes less significant when D approaches its maximum value $n-1$ ($R_{st}(\mathscr{G},D)$ approaches the classical reliability value $R_{st}(\mathscr{G})$ as explained in the Introduction). In the case of highly-dense topologies such as K_9, although no computational gain was observed, the number of recursive calls (i.e., tree's size) performed by FactoringWithReduction() was substantially less that one yielded by Factoring(); however the trade-off between tree's size and computational time shows that the Factoring() procedure without deletion of irrelevant edges produces better results. Table 2 is also interesting because it shows a case where the reliability of a quite modest sized network (with just

25 nodes) could not be computed within 48 hs using standard factorization, while employing the reductions allows to compute the reliability in all cases. It can be also observed from all the experiments performed, that independently of the type of topologies tested, FactoringWithReduction() performs better than Factoring(), with regards to the ratio between their tree's sizes, particularly when low values of the diameter bound D had been considered.

Further experiments were conducted to observe the behavior of the FactoringWithReduction() procedure applied on networks of higher order and for different values of the diameter bound D. Table 3 and Table 4 present results when Circulant network topologies were tested. In Table 3, fixed topologies were tested while values of the diameter bound D varied. These results show that for some classes of graphs such as Circulants, deletion of irrelevant edges can play a very important computational role when incorporated within exact methods to evaluate the Source-to-Diameter Constrained reliability of a network.

Method	Network	r_e	D	$1-R_{\{1,22\}}(\mathscr{G},D)$	Tree's size	CPU time (s)
FactWRed	$C^{22}_{1,11}$	0.99	5	$1.068119900200002e-04$	$9.1000000e+01$	<1
FactWRed	$C^{22}_{1,11}$	0.99	9	$1.039792532863799e-04$	$1.3297000e+04$	<1
FactWRed	$C^{22}_{1,11}$	0.99	13	$2.123401100995179e-06$	$7.0805383e+07$	788
FactWRed	$C^{22}_{1,11}$	0.99	17	$2.123210555152134e-06$	$7.3731159e+07$	619
FactWRed	$C^{22}_{1,11}$	0.99	19	$2.123210555151751e-06$	$7.3736879e+07$	614

Table 3 Experimental results for measuring the effect of the IrrelevantEdgesReduction() procedure when combined with the factoring equation of Moskowitz. Tests were conducted on Circulants topologies, and for arbitrary values of the diameter bound D

Method	Network	r_e	D	$1-R_{\{1,n\}}(\mathscr{G},D)$	Tree's size	CPU's time (s)
FactWRed	$C^{30}_{1,15}$	0.99	14	$1.039788003521266e-004$	$2.347793e+06$	24
FactWRed	$C^{40}_{1,20}$	0.99	14	$1.039788003521266e-004$	$2.347793e+06$	34
FactWRed	$C^{50}_{1,25}$	0.99	14	$1.039788003521266e-004$	$2.347793e+06$	42
FactWRed	$C^{100}_{1,50}$	0.99	14	$1.039788003521266e-004$	$2.347793e+06$	107

Table 4 Effect of the IrrelevantEdgesReduction() procedure when combined with the factoring equation of Moskowitz as a function of the number of nodes n of the Circulant topology, and for fixed value of the diameter bound D.

6 Conclusions and future work

In this paper we presented combinatorial properties to identify irrelevant edges within the context of the Source-to-terminal Diameter Constrained network reliability, that yield an efficient procedure for detecting and eliminating a class of irrelevant edges of a network while preserving the reliability. Moreover we showed how this procedure can be used in combination with a factoring method based on the decomposition equation of Moskowitz, for computing the Source-to-terminal Diameter Constrained network reliability measure. While computing this measure is an NP-hard problem, and as such intractable for large networks, we have shown experimentally that applying topological reductions based on the notion of irrelevancy can help to considerably improve the efficiency of the factoring method. Experiments conducted on different topologies confirmed a substantial computational gain, except when highly-dense graphs (e.g., complete graphs) were tested.

As future work, we are planning on extending the study of these reliability preserving reductions to further improve their use in combination with other exact methods of evaluation, as well as to introduce new combinatorial properties in which edge irrelevancy could be spotted while improving the computational time of calculating the Diameter Constrained reliability, both for the Source-to-terminal case, as well as for networks having an arbitrary set $\mathscr{K}$ of terminal nodes, $|\mathscr{K}| > 2$.

Application of these topological reductions could play an important role in network design problems as well. For example in [2], different optimization techniques were presented in order to find an alternative network to a given topology (theoretically composed of a large number of nodes and edges) with higher reliability while obeying certain cost constraints. Edge irrelevancy could be also combined with these techniques to reduce their computational time and consequently yield alternatives topologies with higher reliability.

References

1. M.O. Ball. Computational complexity of network reliability analysis: An overview. *IEEE Trans. Reliab.*, R-35(3):230–239, August 1986.

2. P. Burgos, H. Cancela, A. Godoy, and L. Petingi. Source-terminal reliability maximization with path length constraints. In *International Conference on Industrial Logistics (ICIL'03)*, pages 76–83, Vaasa, Finland, 2003.
3. H. Cancela and M. El Khadiri. Series-parallel reductions in Monte Carlo network-reliability evaluation. *IEEE Trans. on Reliab.*, 47(2):159–165, June 1998.
4. H. Cancela and M. El Khadiri. On the RVR simulation algorithm for network reliability evaluation. *IEEE Transactions on Reliability*, 52(2):207–212, 2003.
5. H. Cancela, M. El Khadiri, and G. Rubino. In G. Rubino and B. Tuffin, editors, *Rare event simulation using Monte Carlo methods*, chapter 7 in Rare events analysis by Monte Carlo techniques in static models, G. Rubino and B.Tuffin (eds.), John Wiley & Sons, pp. 145–170, 2009.
6. J. Carlier and O. Theologou. An efficient program for computing undirected networks reliability. In M. Rudianski R. Avenhavs, H. KarKar, editor, *Defense Decision Making*, Springer-Verlag, 1991.
7. J. Carlier and O. Theologou. Factoring & reductions for networks with imperfect vertices. *IEEE Trans. on Reliab.*, 40(2):210–217, June 1991.
8. E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Math.*, 1, 1959.
9. M. El Khadiri. *Direct evaluation and by simulation of communication network reliability parameters: Sequential and memory distributed parallel algorithms (in French)*. Computer science phd. thesis, Rennes I, Campus de Beaulieu, 35042 Rennes, France, December 1992.
10. D.E. Knuth. *The Art Of Computer Programming*. Vol 1. 3rd ed. Addison-Wesley, ISBN 0-201-89683-4, 1997.
11. A. Konak. Combining network reductions and simulation to estimate network reliability. In S.G. Henderson, B. Biller, M. H. Hsieh, J. D. Tew, and R. R. Barton, editors, *Proceedings of the 2007 Winter Simulation Conference*, pages 2301–2305, 2007.
12. F. Moskowitz. The analysis of redundancy networks. *IEEE Trans. on Communication and Electronic. Reliab.*, 39:627–632, 1958.
13. L.B. Page and J.E. Perry. A practical implementation of the factoring theorem for network reliability. *IEEE Trans. Reliab.*, 37(3):259–267, August 1988.
14. L. Petingi and J. Rodriguez. Reliability of networks with delay constraints. In *Congressus Numerantium 152*, pages 117–123, 2001.
15. L.I.P. Resende. Implementation of a factoring algorithm for reliability evaluation of undirected networks. *IEEE Trans. Reliab.*, 37(5):462–468, December 1988.
16. A. Satyanarayana and R.K. Wood. Polygon-to-chain reductions and network reliability. Technical Report ORC 82-4, Operations Research Center, University of California, Berkeley, USA, 1982.
17. A. Satyanarayana and R.K. Wood. A linear-time algorithm for computing k-terminal in series-parallel networks. *SIAM J. Comput.*, 14(4):818–832, November 1985.
18. A. Satyarayana and M.K. Chang. Network reliability and the factoring theorem. *Networks*, 13:107–120, 1983.
19. R.E. Tarjan. Depth-first search and linear graph algorithms. *SIAM J. Comput.*, 1(2):146–160, June 1972.
20. K. Wood. A factoring algorithm using polygon-to-chain reductions for computing k-terminal network reliability. *Networks*, 15:173–190, 1985.
21. K. Wood. Factoring algorithms for computing k-terminal network reliability. *IEEE Trans. Reliab.*, R-35(3):269–278, August 1986.