Factorization and Exact Evaluation of the Source-Terminal Diameter-Constrained Reliability

Eduardo Canale, Pablo Romero, Gerardo Rubino

Factorization and Exact Evaluation of the Source-Terminal Diameter-Constrained Reliability

Eduardo Canale[†], Pablo Romero[†] and Gerardo Rubino[‡]
† Facultad de Ingeniería, Universidad de la República.
Julio Herrera y Reissig 565. Montevideo, Uruguay.
Email: promero@fing.edu.uy
‡ Inria Rennes, Bretagne-Atlantique.
Campus de Beaulieu.
35042 RENNES Cedex
Email: rubino@inria.fr

Abstract—In classical network reliability, the system under study is a network with perfect nodes and imperfect links that fail randomly and independently. The probability that a given subset K of terminal nodes belongs to the same connected component is called *classical* or K-*Terminal reliability*. Although (and because) the classical reliability computation belongs to the class of $\mathcal{NP}$-Hard problems, the literature offers many methods for this purpose, given the importance of the models.

This article deals with diameter-constrained reliability, where terminal nodes are further required to be connected by d hops or fewer (d is a given strictly positive parameter of the metric called its *diameter*). This metric was defined in 2001, inspired by delay-sensitive applications in telecommunications.

Factorization theory is fundamental for the classical network reliability evaluation, and today it is a mature area. However, its extension to the diameter-constrained context requires at least the recognition of irrelevant links, which is an open problem. In this paper, irrelevant links are efficiently determined in the most used case, where $|K| = 2$, thus providing a first step towards a Factorization theory in diameter-constrained reliability. We also analyze the metric in series-parallel and composition graphs. The article closes with a Factoring algorithm and a discussion of trends for future work.

Keywords—Computational Complexity, Network reliability, Diameter-constrained reliability, Factorization Theory, Series-Parallel Graphs, Composition Graphs.

1. MOTIVATION

The Diameter-Constrained Reliability measure (DCR) was introduced in 2001 by L. Petingi and J. Rodríguez, inspired by delay sensitive applications [20]. In telecommunications, there are several problems where the number of hops in the communication path is a major cause of concern. In flooding-based systems, the number of hops should be controlled in order to avoid network congestion. Peer-to-peer networks originally supported file discovery protocols by means of flooding [1]. Internet Protocol version 6 (IPv6) has a "Hop limit" field, reserved for these cases [12]. Another hot topic in network design is fiber optic deployment. There, light-paths should be short in order to save bandwidth resources [15]. The performance of degraded systems

dramatically deteriorates with distance [9]. A practical example is the case of electrical networks, which suffer from the Joule effect, causing power losses. We invite the reader to find a rich discussion on diameter-constrained reliability and its applications in [7].

Since DCR subsumes the classical network reliability, its evaluation also belongs to the class of $\mathcal{NP}$-Hard computational problems. As a corollary, prior contributions in the field can be classified into three categories:

(i) exact Methods for general DCR evaluation (exponential time),
(ii) exact Methods for DCR evaluation in special families of graphs (polynomial time),
(iii) approximation methods (polynomial time).

We address the DCR evaluation in a source-terminal scenario, that is, when $|K| = 2$. Our main contributions include general exact methods for DCR evaluation, and specific methods for special graphs (cases (i) and (ii) above). More specifically, the DCR of series-parallel graphs is studied, and an algorithm for the recognition of *irrelevant* links is provided, whose deletion does not affect the correct operation of the system. This recognition supports the development of a Factorization algorithm which is another contribution of this work.

The article is organized in the following manner. Section 2 formally presents the problem. In Section 3, the computational complexity of the DCR evaluation is discussed in terms of the diameter and the number of terminals. Section 4 discusses exact methods to evaluate diameter-constrained network reliability, focused on the factorization procedure.

The main contributions of this article are concentrated in Sections 5, 6 and 7. Specifically, an efficient determination of irrelevant links is given in Section 5. New families of graphs that accept polynomial time DCR evaluation are introduced in Section 6. In Section 7, a set of elementary operations that are DCR-invariants are presented. A factorization algorithm combines them with the deletion of irrelevant links and a recursive decomposition method. The

spirit is to reduce/simplify the network as much as possible during each iteration, in order to return the source-terminal DCR efficiently. Finally, a summary of open problems and trends for future work is included in Section 8.

2. TERMINOLOGY

We follow the terminology of Michael Ball [2]. A *stochastic binary system* (SBS) is a triad (S, ϕ, p), where S a ground-set with a finite number of $|S| = m$ elements, called *components*, ϕ is a function $\phi : \mathcal{P}(S) \to \{0, 1\}$ called *structure function*, that assigns either value 1 (system is up, or operational) or value 0 (system is down, or failed) to each subset of S, and $p : S \to [0, 1]$ is a probability function, assigning to each component a probability called its *elementary reliability*. We denote by X_s the state of component $s \in S$, that is, a Bernoulli random variable with success probability $\mathbb{P}(X_s = 1) = p(s) = p_s$. Moreover, the set $\{X_s\}_{s \in S}$ is a family of independent random variables. Let $X \subseteq S$ be the random subset of S defined by $X = \{i : X_i = 1\}$. Then, the *reliability of the SBS* is the number r defined by

$$r = \mathbb{P}(\phi(X) = 1) = E(\phi(X)). \tag{1}$$

A *pathset* is a subset $x \subseteq S$ such that $\phi(x) = 1$. Here we call *cutset* any subset $x \subseteq S$ such that $\phi(x) = 0$. We warn the reader, since in other works a cutset is a subset whose complement does not include pathsets. This last definition assumes monotonicity, and we define here cutset in its most general setting. A pathset x is *perfect* if $\mathbb{P}(X_s) = 1 \, \forall s \in x$.

Minimal pathsets (cutsets) are called *minpaths* (*mincuts*). A component $i \in S$ is *irrelevant* if $\phi(x) = \phi(x - \{i\})$ for all possible subsets $x \subseteq S$. In words, a component is irrelevant when its state never affects the global system's state. We will work here with the probability space $(S, \mathbb{P}, \mathcal{P}(S))$ where $\mathbb{P}$ is given by $\mathbb{P}(x) = \prod_{s \in x} p_s \prod_{s \notin x} (1 - p_s)$.

Given a simple graph $G = (V, E)$ and subsets $A, B \subseteq V$, an $A - B$ path is a path with one end in A and the other one in B. If a is a node, to alleviate the notation we will say an $a - B$ path instead of an $\{a\} - B$ path. The classical network reliability problem considers a simple (no self-loops) undirected graph $G = (V, E)$ and a terminal set $K \subseteq V$. The corresponding SBS is defined by the following triad:

- the ground set is E;
- the structure function ϕ is 1 on a subset $E' \subseteq E$ if and only if in the subgraph (V, E'), for any two terminals s and t there is a path connecting them, i.e., an $s - t$-path with all its links in E';
- the elementary reliabilities are given by $p : E \to [0, 1]$.

The classical network reliability is historically termed connectedness probability as well [21]. In the diameter-constrained scenario, the structure function is modified, and $\phi(E') = 1$ if for any two terminals s and t there is an $s - t$-path with d links or fewer, having all its links in E', where d is a given positive integer called diameter. The reader must be warned about the fact that the diameter is used in general graph theory as a graph metric equal to the greatest distance between any pair of vertices. Here, it is a given parameter of DCR. For instance, the

graph of Figure 1 has diameter 3. However, we can look at its DCR corresponding to the source s, terminal t and diameter parameter $d = 6$. If all its links have the same elementary reliability p, we will see at the end of Section 7 that its DCR is $p^2 + p^5 - p^6$.
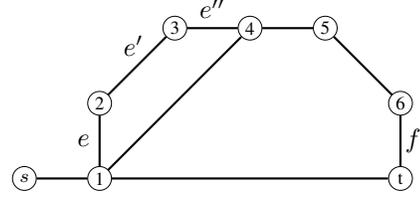

Fig. 1. Sample graph G with irrelevant links $\{1, 2\}, \{2, 3\}, \{3, 4\}$ when $d = 6$.

We denote by $R_{K,G}^d$ the diameter-constrained reliability metric of graph G with terminal set K and diameter parameter d.

3. COMPUTATIONAL COMPLEXITY

Recognizing and counting minimum cardinality mincuts/minpaths are at least as hard as computing the reliability of an SBS [2]. Arnie Rosenthal observed that minimum cardinality recognition in the classical network reliability problem is precisely the Steiner Tree Problem in graphs [23]. Since this problem is included in Karp's list [14], the classical network reliability computation belongs to the class of $\mathcal{NP}$-Hard problems. Since the exact diameter-constrained reliability computation is an extension of classical reliability evaluation, it also belongs to the class of $\mathcal{NP}$-Hard problems.

Later effort has focused on particular cases of the DCR, in terms of the number of terminals $k = |K|$ and the diameter d. When $d = 1$ all pairs of terminals must have a direct link between them, and $R_{K,G}^1 = \prod_{\{u,v\} \subseteq K} p_{u,v}$, where $p_{u,v}$ is the probability of operation of link $\{u, v\} \in E$ or 0 if $\{u, v\} \notin E$. Furthermore, the reliability of critical graphs (in the sense that their diameter seen here as the general graph metric is increased under any link deletion) is again the product of all their elementary reliabilities. An example of a critical graph is the complete graph with diameter $d = 1$, or a complete bipartite graph with diameter $d = 2$. The problem is still simple when $k = d = 2$. In fact,

$$R_{\{u,v\},G}^2 = 1 - (1 - p_{u,v}) \prod_{w \in V - \{u,v\}} (1 - p_{u,w} p_{w,v}).$$

Héctor Cancela and Louis Petingi showed in [8] that when $d \geq 3$, the problem is $\mathcal{NP}$-Hard even in the source-terminal case. The authors show that the diameter-constrained reliability evaluation of a special topology is at least as hard as counting the number of vertex covers of a bipartite graph; see [8] for a complete proof. A full complexity analysis of different subproblems as a function of $k = |K|$ and d is available in prior works [4]. The problem is still hard if more terminal nodes are added. The whole picture about the complete complexity analysis is shown in Table 1 as a function of the different pairs (k, d), where n denotes the number of nodes in the given graph.

	k (fixed)		$k = n$ or free
	2	3 ...	
2	$O(n)$ [8]	$O(n)$ [5]	$\mathcal{NP}$-Hard [3]
3			
d ⋮		$\mathcal{NP}$-Hard [8]	
			$\mathcal{NP}$-Hard [3]
$n-2$			
$n-1$			
⋮		$\mathcal{NP}$-Hard [23]	$\mathcal{NP}$-Hard [21]

TABLE 1

DCR COMPLEXITY IN TERMS OF THE DIAMETER d AND THE NUMBER OF TERMINALS $k = |K|$; OBSERVE THAT IF $d \geq n-1$, THEN THE PROBLEM BECOMES THE CLASSICAL RELIABILITY ONE.

4. EXACT METHODS

As stated before, computing network reliability metrics is both important and hard. This explains the high number of papers published for computing, bounding or estimating them, plus many other ones dedicated to related issues (topological network design, model transformation, among others). For instance, in [22] some basic approaches are discussed, and a powerful method (called Factorization) is also described. We focus on the latter, but to understand the complexity of the problem, let us briefly consider some basic ways of representing and, in theory, computing these metrics.

The first one, also called Poincare's formula, is based on a full enumeration by minpaths (or mincuts). Assume that $M_1, \ldots, M_\ell$ is the whole list of minpaths of a certain SBS (S, ϕ, p). Let us denote by $E_1, \ldots, E_l$ the corresponding events $E_i = \{M_i \subseteq X\}$ (i.e., the minpath M_i occurs). The Inclusion-Exclusion method returns the reliability using the following expression:

$$r = \mathbb{P}(\bigcup_{i=1}^{\ell} E_i) = \sum_{i=1}^{\ell-1} (-1)^{i-1} \sum_{\emptyset \neq I \subseteq \{1,2,\ldots,\ell\}; \, |I|=i} \mathbb{P}(\bigcap_{j \in I} E_j). \quad (2)$$

This is basically worse than the brute force approach consisting in exploring all subsets x and checking which ones lead to $\phi(x) = 1$. In the general case, and considering classical reliability metrics, the number of minpaths is exponential with the number of components m. Moreover, the number of terms from Expression (2) is even larger. This is of theoretical interest only, unless we are in a situation where most of the terms cancel out.

Observe that the events $\{M_i\}_{i=1,\ldots,\ell}$ are not disjoint. An alternative is to re-write Expression (2), finding a mutually-exhaustive union of disjoint events. This is the key idea behind the Sum of Disjoint Products method. The reader is invited to consult [22] for further details.

Let us now have a closer look at the Factorization approach. The basic idea is to condition on the state of some component s:

$$r = p_s E(\phi(X) \mid X_s = 1) + (1 - p_s) E(\phi(X) \mid X_s = 0). \quad (3)$$

A shortcoming of this recursive method in its basic form is that it is strongly exponential. However, if the system has irrelevant components they can be discarded at the beginning of the evaluation procedure, and the whole computational process becomes largely accelerated. The determination of irrelevant components depends on the specific structure under study and the metric we are considering.

The first work about Factorization in the field of network reliability is authored by Fred Moskowitz, inspired by electrical networks [18]. In graph terms, we can write the factoring formula

$$R_{K,G} = p_e R_{K',G/e} + (1 - p_e) R_{K,G-e}, \quad (4)$$

where $e \in E$ is any link of graph $G = (V, E)$ with elementary reliability p_e, $G - e = (V, E - e)$ is the result of removing link e from G, G/e is the result of contracting link e in G (that is, removing link e and merging its endpoints), and K' is the new terminal-set after link contraction. Informally, we sometimes say that $G - e$ is the "deletion" graph and G/e is the "contraction" graph.

Since the contraction operation is not diameter invariant, Expression (4) (sometimes called deletion-contraction formula) does not hold for the diameter-constrained reliability metric. However, a similar expression holds:

$$R_{K,G}^d = p_e R_{K,G|_e}^d + (1 - p_e) R_{K,G-e}^d, \quad (5)$$

where $G|_e$ is the graph G with $p_e = 1$. Expression (5) suggests a recursive solution, where links are either perfect (i.e., with elementary reliability 1) or deleted in turn, until a halting condition is met (either the network has a pathset of probability 1, or there is no "feasible" pathset, that is, a pathset where the number of links is $\leq d$).

Recall that a recursive application of Expression (5) is strongly exponential, and reductions/simplifications to successive graphs should be performed. When we use the term *Factorization methods* we include these aspects as well. As previously stated, in classical network reliability, Factorization theory is a mature area. The central element in the theory is how to find an optimal pivotal link e in order to reduce the number of operations by means of a recursive application of Equation (5). Chang and Satyanarayana found a remarkable characterization of the optimal pivot [24], [25]. They first present a graph invariant called the *domination* $D_K(G)$ of a graph G for a given terminal set K. They consider the successive graphs obtained from link deletions and contractions. In this process, the resulting binary tree of successive graphs presents exactly $D_K(G)$ leaves, or more. The equality thus results in the optimal factorization process, and the authors find the best pivot for all successive graphs. The process is significantly improved by a previous recognition and deletion of irrelevant links and the simplification of series and parallel links. Unfortunately, domination theory can't be easily extended (for instance, it doesn't work if the graph is directed). Its possible extensions to the diameter-constrained case deserves further research.

5. DETERMINATION OF IRRELEVANT LINKS IN DCR

The central work of Factorization theory in the classical reliability model is paper [24] by Chang and Satyanarayana. The authors consider series-parallel reductions (i.e., equivalent graphs with fewer nodes and/or links), and characterize the properties of any graph obtained during the recursion process in an optimal situation (optimality with respect to the number of graphs built during the evaluation procedure, after a link deletion or a contraction). The way they proceed ensures that we don't actually need to look for irrelevant links, because they proved that there always exists at least one link leading to an optimal procedure (of course, starting with a graph where all links are relevant), and in all optimal procedures we never have irrelevant components. However, during the development, they discuss irrelevance of links, and they characterize it. The characterization is simple and elegant. We present it here for completeness. Consider a simple graph $G = (V, E)$ and a terminal set $K \subseteq V$.

Proposition 1: The graph G presents irrelevant links if and only if there exists v such that $G - \{v\}$ has a connected component without terminals.

Observe that if $G - \{v\}$ has a connected component without terminals, this component is irrelevant. If there is no such cut-vertex v and the graph G is two-node connected, there exists a path between every pair of terminals containing any specified link [13].

Even though the recognition of irrelevant links is well understood in the classical reliability model, it is an open problem under the extended diameter-constrained reliability. The characterization from Proposition 1 doesn't work for DCR, and a possible counterexample is the source-terminal DCR in an elementary cycle with diameter $d = 1$.

In [7], Héctor Cancela et al. proposed a sufficient condition for a link to be irrelevant, and stated that its efficient determination in a source-terminal context was open. Later effort has been carried-out by Louis Petingi, with a stronger sufficient condition [19]. Consider an arbitrary graph $G = (V, E)$, a two-terminal set $K = \{s, t\}$, a diameter d and a specific link $e = \{x, y\}$ under study. We will denote by $d_G(u, v)$ the distance between u and v in graph G. Two sufficient conditions for $e = \{x, y\}$ to be irrelevant for $R^d_{\{s,t\}, G}$ are, in chronological order of publication:

(I) $d_G(s, x) + d_G(y, t) \geq d$ and $d_G(s, y) + d_G(x, t) \geq d$;

(II) $d_{G-e}(s, x) + d_{G-e}(y, t) \geq d$
and $d_{G-e}(s, y) + d_{G-e}(x, t) \geq d$.

Let us introduce here an immediate to check and obviously stronger sufficient condition:

(III) $d_{G-y-t}(s, x) + d_{G-s-x}(y, t) \geq d$
and $d_{G-x-t}(s, y) + d_{G-s-y}(x, t) \geq d$.

First, we show that the three sufficient conditions above fail to detect all irrelevant links. Then, we describe an efficient method to detect all irrelevant links in an arbitrary graph.

Let us consider the graph G sketched in Figure 1. Observe that $e = \{1, 2\}$ is irrelevant for $d = 5$, and even for $d = 6$. However, $d_G(s, 1) + d_G(2, t) = 1 + 2 = 3 < 5$, so

Condition (I) does not detect that e is irrelevant for $d = 5$ nor for $d = 6$. Observe that $d_{G-e}(s, 1) + d_{G-e}(2, t) = 5$ and $d_{G-e}(s, 2) + d_{G-e}(1, t) = 4 + 1 = 5$, so Condition (II) detects that e is irrelevant when $d = 5$, but not for $d = 6$. Finally, $d_{G-2-t}(s, 1) + d_{G-1-s}(2, t) = 1 + 5 = 6$ and $d_{G-1-t}(2, s) + d_{G-s-2}(1, t) = \infty$, so Condition (III) detects that e is irrelevant in both cases. However, the reader can check that link $e' = \{2, 3\}$ is irrelevant when $d = 6$, and that none of the three sufficient conditions detects this.

The conditions (i), (ii) and (iii) helped to introduce the problem of detecting irrelevant components in a model (together with referencing previous work on this topic). Now, we will determine all irrelevant links in an efficient way, always in the fundamental 2-terminal case.

We start by observing the trivial fact (that follows by definition) stating that if a link e is irrelevant for a diameter d, then it will also be irrelevant for any diameter $d' \leq d$. Let $G = (V, E)$ be a simple graph, $K = \{s, t\}$ the terminal set, d a positive integer (diameter) and $e = \{x, y\} \in E$ a given link. Under these conditions, e *is relevant if and only if there is some $s - t$ minpath P composed of at most d links, such that $e \in P$.* We will say we have two node-disjoint $\{x, y\} - \{s, t\}$ paths if there exists either a path from s to x and a path from y to t with no common node, or a path from s to y and a path from x to t again not sharing any node. In Theorem 1 we will use the following characterization:

Lemma 1: Given $G = (V, E)$ and $\{s, t\} \subseteq V$, a link $e = \{x, y\}$ is relevant for the computation of the $s - t$-DCR with respect to diameter d if and only if there are two node-disjoint $\{x, y\} - \{s, t\}$ minpaths with total length at most $d - 1$.

Proof: If $e = \{x, y\}$ is relevant, then there exists an $s - t$ minpath P composed of at most d links such that $e \in P$. In that case, $P - e$ is the union of two node-disjoint $\{x, y\} - \{s, t\}$ paths. Conversely, if we have two such minpaths P_1 and P_2 then $P_1 \cup e \cup P_2$ is an $s - t$-minpath as well, with d or fewer links. ∎

Theorem 1: There is a polynomial time algorithm to determine if a link is irrelevant or not, for the source-terminal DCR.

Proof: Let $e = \{x, y\}$ be the link under study, s and t the terminals of the system, and d the diameter. We build two paths as in Lemma 1, in the following way. Add first two artificial nodes u and v, with u adjacent to x and y, and v adjacent to s and t. Suurballe's algorithm [26] finds two disjoint paths P_1 and P_2 with minimum sum-length, joining u and v, in polynomial time. Then, removing the artificial nodes and making the paths (necessarily minpaths) connect, say, s with x and t with y, we have two minpaths with minimum sum-length for testing. ∎

Remark 1: The complexity of the resulting method is $O(m^2 \log_{(1+m/n)} n)$, since the Suurballe algorithm is called for each link, and it runs in $O(m \log_{(1+m/n)} n)$ time. We can wonder if the detection of irrelevant links provides hints to detect other irrelevant links. In fact, this is the case in

[27] and later in [28] to speed up the algorithm. The Suurballe-Tarjan algorithm finds two arc-disjoint directed paths with min lengths-sum for all possible pairs of source-terminal vertices [28]. It runs in $O(m \log_{(1+m/n)} n)$ time and $O(m)$ space. However, it solves the directed scenario. A node-splitting is then required first to call the Suurballe-Tarjan algorithm in our case. This is introduced in [10] and described again in [26]. Nevertheless, just for the sake of completeness, we will describe the node-splitting needed here. The transformation consists in two steps. The first one transforms the graph into a digraph, while the second one moves the problem of link-disjoint paths into node-disjoint paths. In the first step, we transform the graph G into a digraph $\vec{G}$, by replacing each link with two opposite arcs. In the second step, we transform the digraph $\vec{G}$ into a digraph $\vec{G}'$ by splitting each node v into two nodes v', v'', one of them, say v', adjacent to the other (v''). The arcs adjacent to (from) v are transformed into arcs adjacent to v' (from v'').

We illustrate both steps in Figure 2. If the arcs of the form (v', v'') have weight 0, then, two $u'-v'$-arc-disjoint paths of minimum lengths-sum joining two vertices in $\vec{G}'$ correspond to two $u-v$-node-disjoint paths of minimum lengths-sum in G joining the corresponding vertices. Thus, if we had m links and n vertices in G, then we can add $m+1$ new vertices and $2m+2$ new arcs. Next, we can split the original nodes and links, which leads to $n+2m$ arcs and $2n$ nodes. Summing up, will have $2n+m+1$ vertices and $4m+n+2$ links. So, the complexity is $O\big((4m+n+2)\log_{1+(4m+n+2)/(2n+m+1)}(2n+m+1)\big)$, which is smaller than $O(m^2 \log_{(1+m/n)} n)$ because $m \geq n-1$.

Remark 2: An efficient recognition of irrelevant links is still an open problem when the terminal set is arbitrary. Theorem 1 provides an answer when $|K| = 2$. Given a graph $G = (V, E)$, a terminal set $K \subseteq V$ and a fixed diameter d, we would like to answer whether some link $e \in E$ is irrelevant or not. A first approach is to study whether e is irrelevant to connect every pair of nodes $v, w \in K$. However, there are scenarios where e is irrelevant for each pair $v, w \in K$ but it is relevant when we consider the whole terminal set. Figure 3 shows an example when $d = 2$.
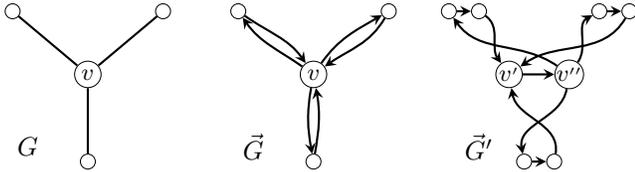

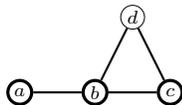Fig. 2. Transformation of a graph in order to apply the Suurballe algorithm.

Fig. 3. Links $\{b, d\}$ and $\{c, d\}$ are irrelevant when $K = \{a, b, c\}$ and $d = 2$. However, they are in the minpath $b - d - c$ when $K = \{b, c\}$.

6. EXACT DCR EVALUATION IN SPECIAL GRAPHS

Definition 1: A *two-terminal graph (TTG)*, G_s^t, is a graph G equipped with two terminals s and t. Given two TTGs, $X_{s_1}^{t_1}$ and $Y_{s_2}^{t_2}$, their *parallel composition* $G_s^t = P(X_{s_1}^{t_1}, Y_{s_2}^{t_2})$ is the TTG created from the disjoint union of graphs X and Y by identifying $s = s_1 = s_2$ and $t = t_1 = t_2$. On the other hand, the *series composition* $G_s^t = S(X_{s_1}^{t_1}, Y_{s_2}^{t_2})$ is the TTG with terminals s_1 and t_2 created from the disjoint union of graphs X and Y by identifying $t_1 = s_2$.

Notice that both operations are a particular case of *amalgamation* [11], but we found it easier to give a direct definition.

Lemma 2: [16, Corollary p.1417] Let G be a two-terminal graph that is the series composition of TTGs X_s^t and Y_s^t. Then

$$R_{\{s,t\},G}^d = \sum_{i=d_X(s,t)}^{d-d_Y(s,t)} (R_{\{s,t\},X}^i - R_{\{s,t\},X}^{i-1}) R_{\{s,t\},Y}^{d-i}, \quad (6)$$

where $d_X(s,t)$ and $d_Y(s,t)$ are the distances between the terminals s and t in the respective graphs X and Y.

Lemma 3: Let G be a two-terminal (simple) graph that is the parallel composition of TTGs X_s^t and Y_s^t. Then

$$R_{\{s,t\},G}^d = R_{\{s,t\},X}^d + R_{\{s,t\},Y}^d - R_{\{s,t\},X}^d R_{\{s,t\},Y}^d. \quad (7)$$

Proof: It is clear that a minpath in P belongs either to X or to Y but not to both, because if a path belongs to both X and Y, it necessarily has s or t as an internal vertex which makes the path not minimal. So, the probability of existence of a minpath in X is independent of the existence of a minpath in Y, thus the product in the formula. ∎

1. Series-Parallel G Graphs

Definition 2: A *two-terminal series-parallel graph (TTSPG)* is a graph that can be constructed by a sequence of series and parallel compositions starting from a single-edge graph K_2 with assigned terminals.

Definition 3: We say that a graph is a *two-terminal series-parallel G graph (TTSPGG)* if it can be constructed by a sequence of series and parallel compositions starting from a single TTG G.

Definition 4: A graph is called a *series-parallel (sp-graph)*, if it is a TTSPG for some pair of vertices. Similarly, a graph is called *series-parallel G (sp-G-graph)*, if it is a TTSPGG for some pair of vertices.

Proposition 2: The source-terminal diameter reliability of an sp-G-graph can be computed in polynomial time when the terminal vertices coincide with those of the graph.

Proof: Let H_s^t be a sp-G-graph. We want to prove that $R_{\{s,t\},H}^d$ can be computed in polynomial time for any fixed d. We proceed by induction: if $H_s^t = P(X_s^t, Y_s^t)$, we can compute both $R_{\{s,t\},X}^d$ and $R_{\{s,t\},Y}^d$ in polynomial time, so by (7), we can do the same for $R_{\{s,t\},G_s^t}^d$.

In the same way, but considering (6), if $H_s^t = S(X_s^t, Y_s^t)$, we can compute $R_{\{s,t\},H}^d$ in polynomial time.

Finally, since G is fixed, we can compute its reliability in constant, so polynomial, time. ∎

5

Example 1: Let $G_s^t = K_4 - \{e\}$, and let s, t be the extremes of the edge e. Figure 4 shows the computation of its $R_{\{s,t\},G}^2$ using the recursions from Proposition 2.

$$= p_{13}p_{12} + p_{23} - p_{13}p_{12}p_{23}.$$

Fig. 4. Recursive Computation of $R_{\{s,t\}}^2$ of an sp-graph, where $[G]^d$ means $R_{\{s,t\},G}^d$ and the terminals are drawn in bold.

Example 2: Let P_n be the s–t path $s = v_0, v_1, \ldots, v_n = t$. If $p_{i:j} = p_{v_i v_{i+1}} \ldots p_{v_{j-1} v_j}$, then

$$R_{\{s,t\},P_n}^d = \begin{cases} p_{0:n} & d \geq n, \\ 0 & d < n. \end{cases}$$

Example 3: Consider the cycle C_n with nodes $v_1, \ldots, v_n, v_1$. If $a_i = p_{1:i+1}$, $b_i = p_{i+1:n}p_{v_1 v_n}$, then

$$R_{\{1,i+1\},C_n}^d = \begin{cases} a_i & i \leq d < n - i, \\ b_i & n - i \leq d < i, \\ a_i + b_i - a_i b_i & \max\{i, n-i\} \leq d, \\ 0 & d < \min\{i, n-i\}. \end{cases}$$

7. FACTORIZATION ALGORITHM

1. Factor

The central component of our reduction/simplification process is the recognition of DCR invariants. Clearly, the deletion of irrelevant links does not affect the reliability, and it represents a DCR invariant. By Theorem 1, we can efficiently determine and delete irrelevant links in a first stage. Lemmas 2 and 3 show that the transformation of series-parallel graphs is another invariant, governed by Equations (6) and (7). We can recognize parallel graphs by a simple deletion of the terminal nodes s and t. In fact, if $G - \{s,t\}$ has more than one component, these components are connected in parallel with respect to s and t. We will use the parallel invariant and replace the series by a stronger condition authored by Migov [16]. Recall that Proposition 1 shows further that the components in $G - \{v\}$ without terminals are irrelevant. This is a sufficient condition for a component to be irrelevant from the DCR point of view as well. Further, Migov built a stronger condition to find irrelevant blocks for the source-terminal scenario, that is explained in the following paragraph.

Consider a connected graph $G = (V, E)$ with two terminals s, t, and let us denote $C = \{v_1, \ldots, v_r\}$ the r cut-nodes in G. These cut-nodes fully determine nonseparable subgraphs of G, whose intersections are precisely the cut-nodes, called *blocks* (they are also called biconnected components of the graph). Consider the block-graph G^C, which is the bipartite graph whose nodes

are the cut-nodes on the one hand and the blocks on the other, where a cut-node and a block are adjacent if the latter includes the former. By the definition of block, G^C cannot have cycles, so G^C is a tree, called the *tree-block of G*. Since s and t belong to some blocks, there exists a unique path of blocks P^C in G^C that connects the blocks with the terminals. Migov formally proves that the other blocks can be discarded, that is, they are irrelevant. Furthermore, the source-terminal DCR evaluation can be finally found using equivalent subproblems, by means of a series composition [16].

We sum up the DCR invariants in a source-terminal configuration:

- *Tree-Blocks*: Receives a TTG G_s^t and returns the main path P^C from the tree-block G^C using Migov's method [16].
- *Series*: Receives d, TTGs G_s^t, H_u^v and finds the DCR of the series composition $S(G_s^t, H_u^v)$ using Equation (6).
- *Parallel*: Receives d, TTGs G_s^t, H_s^t and finds the DCR of the parallel composition $P(G_s^t, H_s^t)$ using Equation (7).
- *Delete*: Finds and deletes all irrelevant links using the construction from Theorem 1 and the Suurballe-Tarjan algorithm.

These invariants are building blocks of a Factorization method, combined with the deletion-contraction formula in Equation (5).

Algorithm 1 $R = Main(G, s, t, d)$

1: $P^C \leftarrow \texttt{MainPath}(G, s, t)$
2: **for** $i = 1$ **to** $|P^C|$ **do**
3: $\quad (B_i, s_i, t_i) \leftarrow \texttt{Blocks}(i, P^C)$
4: $\quad$ **for** $D = 1$ **to** $d - |P^C| + 1$ **do**
5: $\qquad R_i^D \leftarrow Factor(B_i, s_i, t_i, D)$
6: $\quad$ **end for**
7: **end for**
8: $R \leftarrow \texttt{Series}((R_i^D), d)$
9: **return** R

Main algorithm finds the main path P^C in the Tree-Block (Line 1) and immediately identifies the important blocks $B_1, \ldots, B_{|P_C|}$ that jointly compose the main path (Lines 2-3). The DCR metric is found for each block and feasible diameter $D \leq d - |P^C| + 1$ using the *Factor* algorithm (Lines 4-7). These reliabilities are stored in a matrix (R_i^D). It is used to find the source-terminal DCR using Equation (6) iteratively (function Series, Line 8). The DCR is finally returned in Line 9. All the invariants not presented here are exploited inside the *Factor* algorithm.

Algorithm 2 $R = Factor(G, s, t, d)$

1: **if** `HasPerfectPath`(G, s, t, d) **then**
2:     **return** $R = 1$.
3: **else if** `Distance`$(s, t) > d$ **then**
4:     **return** $R = 0$.
5: **end if**
6: $G' \leftarrow$ `Delete`(G, s, t, d)
7: $(G, \ldots, G_r) \leftarrow$ `Parallel`(G', s, t)
8: **if** $G \neq G'$ **or** $r > 1$ **then**
9:     **for** $i = 1$ **to** r **do**
10:       $R_i \leftarrow Factor(G_i, s, t, d)$
11:     **end for**
12:     $R \leftarrow$ `Union`$(R_1, \ldots, R_r)$
13: **end if**
14: **return** $(1 - p_e) \times Factor(G - e, s, t, d)$
$$+ p_e \times Factor(G/e, s, t, d)$$

$Factor$ receives the graph G, two terminals s, t and a diameter d, and returns $R = R^d_{\{s,t\},G}$. The block of Lines 1-5 tests the termination (i.e., if either G has a perfect pathset joining the terminals in at most d steps or if there is no feasible pathset in G). Irrelevant links are then deleted (Line 6), and the graph is immediately divided in equivalent parallel subproblems (Line 7). If the previous lines produced a new graph, a recursive DCR evaluation takes place for each subproblem (Lines 9-11), and the whole DCR is found using Equation (7) (function Union from Line 12). Finally, the ultimate reduction is performed by the deletion-contraction formula, which returns the recursive evaluation (Line 13). Observe that the operation G/e is not a contraction; the graph G is identical but e is a perfect link with $p_e = 1$. This explains the first halting test using a path with perfect links. The pivotal link e is here uniformly chosen at random. The reader can find a comparison among different link selecion heuristics in [17]. The optimal choice from a computational point of view is still an open problem in the diameter-constrained context.

The reader can appreciate that previous implementations of factorization do not exploit series-parallel graphs nor block structures [6]. Furthermore, the recognition of elementary paths is a linear-time procedure, and parallel graphs can be recognized in polynomial time as well (cubic in the order of the graph).

2. Example

For a better understanding of $Main$ Algorithm, we will use network G from Figure 1 with diameter $d = 6$ and identical probabilities $p_e = p \in (0, 1)$ as a pictorial example. Node 1 is identified as the only cut-vertex. Therefore, the DCR can be obtained by a series composition of two blocks (Lines 1-3). $Factor$ algorithm is called for both blocks using different diameters (Lines 4-6). The second block has irrelevant links $\{e, e', e''\}$, which are deleted (see Line 6, $Factor$). The second block is identified as a parallel one between $1 - t$ and $1 - 4 - 5 - 6 - t$. The DCR of the parallel substructure is found using Equation (7) (Function $Union$ in Line 12). The resulting DCR is found using Equation (6) (Function $Series$, Line 8).
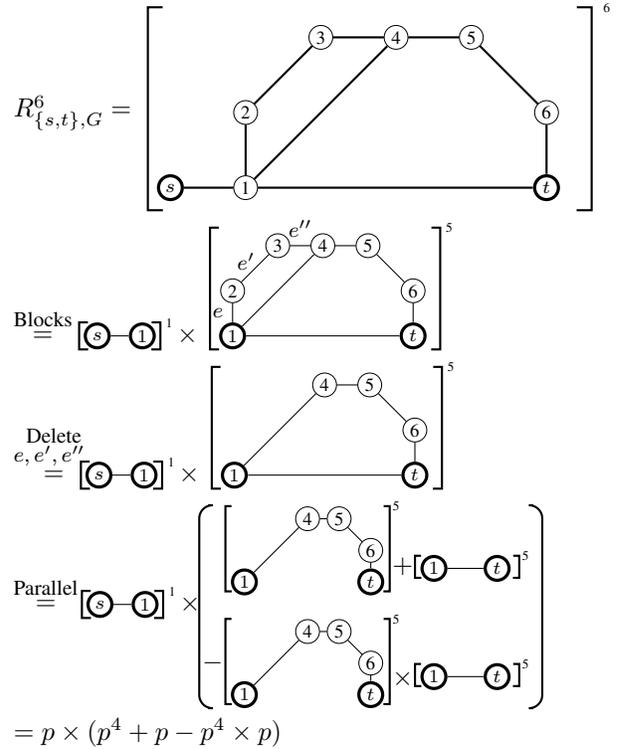

Fig. 5. DCR Evaluation in steps.

In this case, the result $R^6_{\{s,t\},G} = p \times (p + p^4 - p^4 \times p) = p^2 + p^5 - p^6$ is returned. The reader can find a step-by-step analysis in Figure 5.

8. Conclusions and Trends for Future Work

In communication systems, a hop-limit is a major cause of concern. We can find examples in flooding-based systems, peer-to-peer networks, communication protocols of the Internet ecosystem such as IPv6, etc. As a consequence, a metric called diameter-constrained reliability (DCR) was introduced in 2001, inspired in delay sensitive applications.

The exact DCR computation belongs to the class of $\mathcal{NP}$-Hard problems, since it subsumes the classical network reliability computation. The theory of diameter-constrained reliability inherits several ideas from classical reliability, but one of the most powerful ones, Factorization Theory, is hard to fully extend to the diameter-constrained context. One of the main difficulties is the determination of irrelevant links. In this paper we provide the first results in this direction for the two-terminal case, focusing on the detection of links irrelevant for DCR, in arbitrary graphs and series-parallel compositions. An efficient method for their determination is presented. Then, DCR invariants (another component of a Factorization Theory) are discussed, and a Factoring algorithm putting these elements together is introduced. An example shows the behavior of this algorithm on small graphs.

There are several aspects that deserve future work. The general recognition of irrelevant links for an arbitrary terminal set is still an open problem. An analysis of DCR

7

invariants would help in the reduction process. The optimal choice of the pivotal link is already available in the classical network reliability model. However, the same question is open in the diameter-constrained reliability model. An efficient construction of sums of disjoint products should also be explored.

Acknowledgements

References

[1] E. Adar and B.A. Huberman, Free riding on gnutella, First Monday 5 (2000).

[2] M.O. Ball, Computational complexity of network reliability analysis: An overview, IEEE Trans Reliab 35 (1986), 230 –239.

[3] E. Canale, H. Cancela, F. Robledo, P. Romero, and P. Sartor, Diameter constrained reliability: Complexity, distinguished topologies and asymptotic behavior, Networks 66 (2015), 296–305.

[4] E. Canale, H. Cancela, F. Robledo, P. Romero, and P. Sartor, Full complexity analysis of the diameter-constrained reliability, Int Trans Oper Res 22 (2015), 811–821.

[5] E. Canale, H. Cancela, F. Robledo, G. Rubino, and P. Sartor, On computing the 2-diameter-constrained K-reliability of networks, Int Trans Oper Res 20 (2013), 49–58.

[6] E. Canale, P. Romero, and G. Rubino, Factorization theory in diameter constrained reliability, 8th Int Workshop Resilient Networks Design Modeling (RNDM), Sept 2016, pp. 66–71.

[7] H. Cancela, M.E. Khadiri, and L. Petingi, Polynomial-time topological reductions that preserve the diameter constrained reliability of a communication network, IEEE Trans Reliab 60 (2011), 845–851.

[8] H. Cancela and L. Petingi, Reliability of communication networks with delay constraints: Computational complexity and complete topologies, Int J Math Math Sci 2004 (2004), 1551–1562.

[9] T.M. Cover and J.A. Thomas, Elements of information theory (Wiley series in telecommunications and signal processing), Wiley-Interscience, 2006.

[10] L.R. Ford and D.R. Fulkerson, Flows in networks, Princeton University Press, 1962.

[11] J.L. Gross and T.W. Tucker, Topological graph theory, Wiley-Interscience, 1987.

[12] I.N.W. Group, Internet Protocol, Version 6 (IPv6) Specification (RFC 2460).

[13] F. Harary, Graph theory, Addison-Wesley Series in Mathematics, Addison Wesley, 1969.

[14] R.M. Karp, "Reducibility among combinatorial problems," Complexity of computer computations, R.E. Miller and J.W. Thatcher (Editors), Plenum Press, 1972, pp. 85–103.

[15] Z. Liu, W. Guo, Q. Shi, W. Hu, and M. Xia, Sliding scheduled lightpath provisioning by mixed partition coloring in WDM optical networks, Optical Switching Networking 10 (2013), 44 – 53.

[16] D.A. Migov, Computing diameter constrained reliability of a network with junction points, Automation Remote Control 72 (2011), 1415–1419.

[17] D.A. Migov and S.N. Nesterov, Methods of speeding up of diameter constrained network reliability calculation, Proc 15th Int Conference Comput Sci Its Appl – ICCSA 2015: Banff, AB, Canada, June 22-25, Part II, Cham, Springer International Publishing, 2015, pp. 121–133.

[18] F. Moskowitz, The analysis of redundancy networks, Am I Electrical Engineers, Part I: Commun Electronics, Trans 77 (1958), 627–632.

[19] L. Petingi, Diameter-related properties of graphs and applications to network reliability theory, WSEAS Trans Math. 12 (2013), 884–894.

[20] L. Petingi and J. Rodriguez, Reliability of networks with delay constraints, Congress Numer 152 (2001), 117–123.

[21] S.J. Provan and M.O. Ball, The complexity of counting cuts and of computing the probability that a graph is connected, SIAM J Comput 12 (1983), 777–788.

[22] S. Rai, M. Veeraraghavan, and K.S. Trivedi, A survey of efficient reliability computation using disjoint products approach, Networks 25 (1995), 147–163.

[23] A. Rosenthal, Computing the reliability of complex networks, SIAM J Appl Math 32 (1977), 384–393.

[24] A. Satyanarayana and M.K. Chang, Network reliability and the factoring theorem, Networks 13 (1983), 107–120.

[25] A. Satyanarayana and R. Wood, A linear-time algorithm for computing k-terminal reliability in series-parallel networks, SIAM J Comput 14 (1985), 818–832.

[26] J.W. Suurballe, Disjoint paths in a network, Networks 4 (1974), 125–145.

[27] J.W. Suurballe, The single-source, all-terminal problem for disjoint paths, Technical report, Bell Laboratories, 1982.

[28] J.W. Suurballe and R.E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (1984), 325–336.