Efficient Computation of Marginal Reliability-Importance for Reducible^{+} Networks

Steen J. Hsu, *Member, IEEE*, and Maria C. Yuang, *Member, IEEE*

***Abstract*—Marginal reliability importance (MRI) of a link with respect to terminal-pair reliability (TR) is the rate to which TR changes with the modification of the success probability of the link. It is a quantitative measure reflecting the importance of the individual link in contributing to TR of a given network. Computing MRI for general networks is an NP-complete problem. Attention has been drawn to a particular set of networks (reducible networks), which can be simplified to source-sink (2-node) networks via 6 simple reduction rules (axioms). The computational complexity of the MRI problem for such networks is polynomial bounded. This paper proposes a new reduction rule, referred to as triangle reduction. The triangle reduction rule transforms a graph containing a triangle subgraph to that excluding the base of the triangle, with constant complexity. Networks which can be fully reduced to source-sink networks by the triangle reduction rule, in addition to the 6 reduction rules, are further defined as reducible^{+} networks. For efficient computation of MRI for reducible^{+} networks, a 2-phase (2-P) algorithm is given. The 2-P algorithm performs network reduction in phase 1. In each reduction step, the 2-P algorithm generates the correlation, quantified by a reduction factor, between the original network and the reduced network. In phase 2, the 2-P algorithm backtracks the reduction steps and computes MRI, based on the reduction factors generated in phase 1 and a set of closed-form TR formulas. As a result, the 2-P algorithm yields a linearly bounded complexity for the computation of MRI for reducible^{+} networks. Experimental results from real networks and benchmarks show the superiority, by two orders of magnitude, of the 2-P algorithm over the traditional approach.**

***Index Terms*—Marginal reliability importance (MRI), network reduction technique, reducible network, terminal-pair reliability (TR).**

ACRONYMS[1]

iff	if and only if
2-P	our 2-phase algorithm in this paper
TR	terminal-pair reliability
MRI	marginal reliability importance

Notation:

G	graph/network whose links can fail s-independently of each other, with known probabilities
m	number of links in a network
$\mathrm{Rel}(G)$	terminal-pair reliability of network G
$\mathrm{MRI}(e_i)$	marginal reliability-importance of link e_i
s, t	[source, sink] vertex
p_i, q_i	[success, failure] probability of link e_i
"*", -	[compressing, deleting] operation on links
$G*e_i$	G with link e_i compressed
$G-e_i$	G with link e_i deleted
C	transformation factor
R	reduction factor
G_t	triangle subgraph of graph G
G_r	graph $G-G_t$
$e_{s,1}$, $e_{s,2}$	the 2 sides of G_t
$e_{b,1}$, $e_{b,2}$	the base of G_t.

Definitions: `Terminal-pair reliability`: Probability that 2 specified terminals (source and sink) are connected by at least 1 path.

`Marginal reliability importance`: Rate to which TR changes in association with the modification of the success probability of a link.

`Source-sink network`: Network which contains only 2 nodes (source and sink) and the link connecting them.

`Reducible network`: Network which can be fully reduced to a source-sink network by recursively applying the 6 traditional reduction rules.

`Reducible`$^{+}$ `network`: Network which can be reduced to a source-sink network by recursively applying the 6 traditional reduction rules and the triangle reduction rule.

`Triangle subgraph`: Subgraph which contains the source and 2 1-way or 2-way connected nodes to which only the source is connected.

Assumptions:

1) Each link has 2 states: success or failure.
2) The p_i are known for all links.
3) Nodes are fault free.
4) All failure events are mutually s-independent.

I. INTRODUCTION

THE ANALYSIS of TR [1]–[14] of a network has considerable attention in network management. MRI [4], [15]–[19] of a link with respect to TR has been defined as the rate to which TR changes in association with the modification of the success probability of the link. It is a quantitative measure reflecting the importance of the individual link in contributing to TR of the given network. In essence, a network achieves maximal reliability gain if the link with the highest MRI is upgraded [16]:

$$\mathrm{MRI}(e_i) = \frac{\partial \mathrm{Rel}(G)}{\partial p_i} = \mathrm{Rel}(G*e_i) - \mathrm{Rel}(G-e_i). \quad (1)$$

Manuscript received March 2, 1998; revised September 8, 1999.

S. J. Hsu is with the Department of Information Engineering, I-Shou University, 1 Section 1, Hsueh Cheng Rd, Ta-Hsu Hsiang, Kaohsiung County 84008 Taiwan, R.O.C. (e-mail: SteenHsu@isu.edu.tw).

M. C. Yuang is with the Department of Computer Science and Information Engineering, National Chiao Tung University, 1001 Ta Hsueh Rd, HsinChu 30050 Taiwan, R.O.C. (e-mail: MCYuang@csie.nctu.edu.tw).

Publisher Item Identifier S 0018-9529(01)06807-5.

[1]The singular and plural of an acronym are always spelled the same.

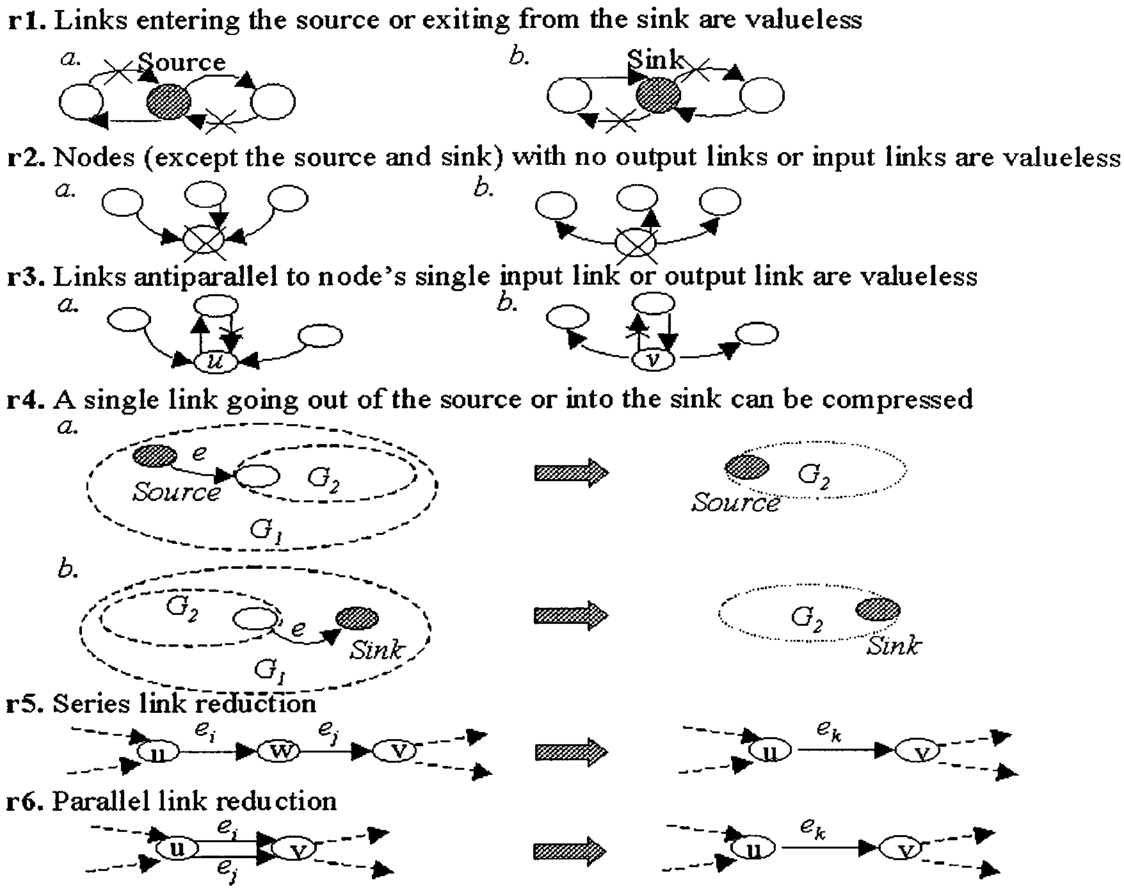


Fig. 1. 6 existing reduction rules.

Computation of MRI involves the evaluation of the TR, e.g., $\text{Rel}(G * e_i)$. The computation of TR for general networks is an NP-complete problem [5]. Nevertheless, for a particular set of networks, called reducible networks [14], which can be fully reduced to source-sink (2-node) networks by 6 simple reduction rules [6], [7], [10], [14], TR can be computed in $O(m)$ [14]. This yields a combinatorial complexity of $O(m^2)$ for computing MRI of all links for a reducible network.

This paper presents a new reduction rule: triangle reduction. The triangle reduction rule basically transforms a graph, in which the source is connected only to 2 1-way or 2-way connected nodes, forming a triangle subgraph, to a simpler graph with the link(s) connecting the 2 nodes removed. The resulting success probabilities of the corresponding links, connecting the source to the 2 nodes, are reassigned via closed-form equations. Another set of networks, called reducible$^+$ networks is introduced; they can be fully reduced to source-sink networks by the triangle reduction rule, in addition to the 6 existing reduction rules. For efficient computation of MRI for reducible$^+$ networks, a 2-phase (2-P) algorithm is presented. The 2-P algorithm reduces the network in phase 1. In each reduction step, the 2-P algorithm generates the correlation, quantified by a reduction factor, between the original network and the reduced network. In phase 2, the 2-P algorithm backtracks the reduction steps and computes MRI, based on the reduction factors generated in phase 1, and a set of closed-form TR formulas. The 2-P algorithm, as shown in this paper, yields a linearly bounded complexity, $O(m)$. Experimental results demonstrate that, compared to a traditional MRI-computation approach [7] and (1), the 2-P algorithm improves run-time by 2 orders of magnitude.

Section II first overviews reducible networks. A new notion of reducible$^+$ networks and the new reduction rule are introduced.

Section III presents the 2-P algorithm. All proofs are in the Appendix.

II. Reducible$^+$ Networks

A network is a source-sink network iff it contains only 2 nodes (the source and sink) and the link connecting them. A network is reducible iff it can be fully reduced to a source-sink network by recursively applying 6 existing reduction rules [6], [10], summarized in Fig. 1. With any 1 of the 6 reduction rules applied, a given network G can be transformed to another network G_a, such that,

$$\text{Rel}(G) = C \cdot \text{Rel}(G_a).$$

In rule r4, for instance, the transformation factor is the success probability of the essential link going out of the source (or into the sink). For the rest of the 6 rules, the transformation factor is simply 1. By repeatedly applying these 6 reduction rules, a reducible network can be reduced to a source-sink network. As a result, the TR of such network can be computed in linear time and is simply the product of the "success probability of the only link in the source-sink network" and R; R is the product of the

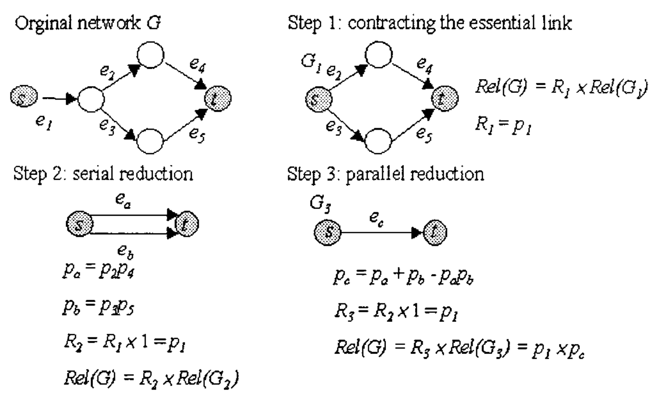


Fig. 2. Example of a reducible network G.

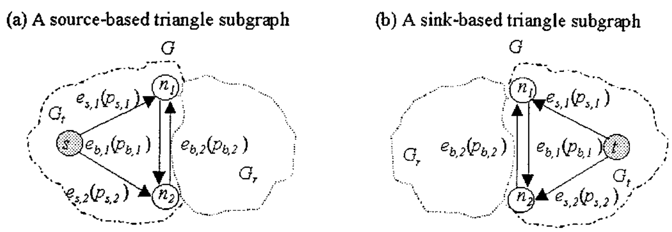


Fig. 3. Triangle subgraphs.

"transformation factor corresponding to the current reduction step," and the "reduction factor generated from the previous reduction step."

Fig. 2 is an example of a reducible network. According to rule r4, network G is reduced to G_1 by compressing the essential link, e_1, with $R_1 = p_1$. Based on rule r5, G_1 is transformed to G_2 by replacing 2 pairs of series-link, e_2, e_4, and e_3, e_5, with 2 new links, e_a, e_b, respectively. The new success probabilities are recomputed, as shown in Fig. 2. $R_2 = R_1$, because of a transformation factor of 1 in this step. According to rule r6, e_a, e_b are further reduced to e_c with success probability p_c, and R_3 is re-derived. The network TR can be directly computed and expressed as the product of p_c and the reduction factor, R_3

$$\text{Rel}(G) = R_3 \cdot p_c = p_1 \cdot (p_2 \cdot p_4 + p_3 \cdot p_5 - p_2 \cdot p_3 \cdot p_4 \cdot p_5).$$

A new reduction rule, called the triangle reduction rule [20], is introduced. The triangle reduction rule takes effect if there exists a triangle subgraph in a graph representing the network under consideration. Fig. 3(a) shows the subgraph. The notion of the triangle subgraph can be similarly applied to a subgraph including the sink instead (sink-based), as shown in Fig. 3(b). For simplicity, without further declaration, the triangle subgraph is referred throughout the rest of the paper as source-based.

Fig. 3(a) denotes the 2 nodes to which the source is connected as n_1, n_2. The 2 links emanating from s to n_1, n_2, referred to as the sides of the triangle, are labeled, $e_{s,1}$, $e_{s,2}$, with success probabilities $p_{s,1}$, $p_{s,2}$, respectively. The link connecting $n_1(n_2)$ to $n_2(n_1)$, referred to as the base of the triangle, is labeled $e_{b,1}(e_{b,2})$ with success probability $p_{b,1}(p_{b,2})$. If n_1 and n_2 are 2-way connected, the base of the triangle is comprised of 2 links. As a result, the 3 nodes (s, n_1, n_2), the 2 sides $(e_{s,1}, e_{s,2})$, and the base ($e_{b,1}$ and/or $e_{b,2}$) constitute the triangle subgraph, G_t. The rule for the 2-link base is formally

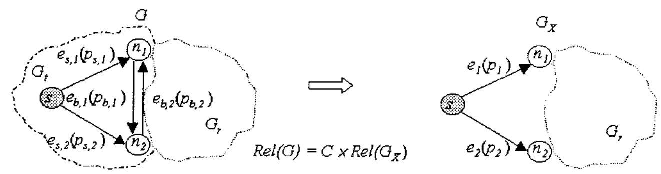


Fig. 4. Triangle reduction rule.

stated and proved in this paper. For the 1-link base, similar results can be obtained by replacing either $p_{b,1}$ or $p_{b,2}$ with 0.

A. Triangle Reduction Rule

In a given graph G, see Fig. 4, if there exists a triangle subgraph with 3 nodes $(s, n_1 \quad n_2)$, 2 sides $(e_{s,1}, e_{s,2})$, and the base ($e_{b,1}$ and/or $e_{b,2}$), G can be transformed to G_X with the base removed. The new p_1 of link e_1 connecting s to n_1, and p_2 of link e_2 connecting s to n_2, are reassigned as

$$p_1 = \frac{\psi_N}{\psi_{D1} + \psi_N}, \tag{2}$$

$$p_2 = \frac{\psi_N}{\psi_{D2} + \psi_N}, \tag{3}$$

$$\psi_N \equiv q_{s,1} \cdot p_{s,2} \cdot p_{b,2} + p_{s,1} \cdot q_{s,2} \cdot p_{b,1} + p_{s,1} \cdot p_{s,2}$$

$$\psi_{D1} \equiv q_{s,1} \cdot p_{s,2} \cdot p_{b,2}, \quad \psi_{D2} \equiv p_{s,1} \cdot q_{s,2} \cdot q_{b,1}.$$

Rel(G) becomes the product of the terminal-pair reliability of the transformed graph G_X and the C:

$$\text{Rel}(G) = \text{Rel}(G_X) \cdot C, \tag{4}$$

$$C = \frac{(\psi_{D1} + \psi_N) \cdot (\psi_{D2} + \psi_N)}{\psi_N}. \tag{5}$$

Proof: See the Appendix, Section 1.

The computational complexity of triangle-reduction rests on "examining the existence of triangle subgraphs" and "computing the transformation." Apparently, both tasks require computation complexity of constant time: $O(1)$.

B. Reducible$^+$ Networks

Incorporating the triangle reduction rule, another set of networks (reducible$^+$ networks) is introduced. A network is reducible$^+$ if it can be reduced to a source-sink network by recursively applying the 6 reduction rules and the triangle reduction rule. Fig. 5 is an example of a reducible$^+$ network. According to the triangle reduction rule, network G is reduced to G_1 by replacing the triangle subgraph with 2 new links, e_a, e_b. The new success probabilities and the reduction factor R_1 are recomputed, as shown in Fig. 5. By applying serial and parallel reductions, G_1 reduces to G_3, a source-sink network. The TR of network G can be directly computed.

III. The 2-P Algorithm

To compute efficiently the MRI for reducible$^+$ networks, the 2-P algorithm is presented. It has a reduction phase (reduction is performed), and a backtracking phase (the MRI are derived). In the reduction phase, the algorithm generates the reduction factor between the original network and the reduced network in each

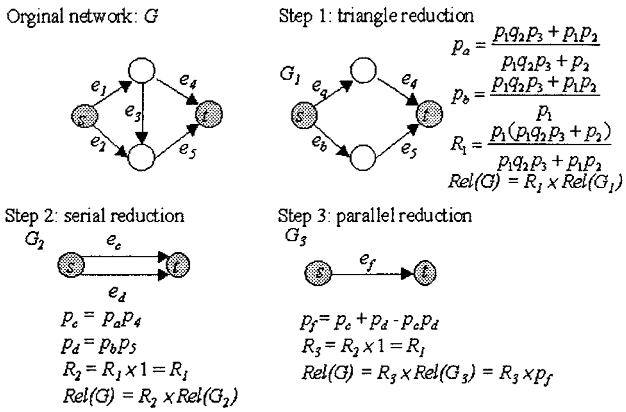


Fig. 5. Example of a reducible$^+$ network, G.

reduction step. In the backtracking phase, when the algorithm is backtracking reduction-step j, it computes $\text{Rel}(G_{j-1} * e_i)$ and $\text{Rel}(G_{j-1} - e_i)$ of link e_i; e_i is replaced (or removed) in reduction step j, based on a set of closed-form (backward-TR) formulas. The MRI are then computed based on the reduction factor and the backward-TR formulas. Section III-A introduces 6 backward-TR formulas as lemmas. Section III-B provides the MRI computation.

A. Backward TR Formulas

Lemma 1: If link e_i is valueless in network G, then

$$\text{Rel}(G - e_i) = \text{Rel}(G * e_i) = \text{Rel}(G).$$

Proof: See the Appendix, Section 2.

Lemma 2: If e_i is the essential link going out of the source (or into the sink) in network G, then

$$\text{Rel}(G - e_i) = 0, \quad \text{Rel}(G * e_i) = \frac{\text{Rel}(G)}{p_i}.$$

Proof: See Appendix, Section 3.

Lemma 3: If network G_1 with two series links, e_i, e_j, is reduced (rule r5) to a new network G_2, with the replaced link e_k, then

$$\begin{aligned} \text{Rel}(G_1 - e_i) &= \text{Rel}(G_1 - e_j) = \text{Rel}(G_2 - e_k), \\ \text{Rel}(G_1 * e_i) &= q_j \cdot \text{Rel}(G_2 - e_k) + p_j \cdot \text{Rel}(G_2 * e_k), \\ \text{Rel}(G_1 * e_j) &= q_i \cdot \text{Rel}(G_2 - e_k) + p_i \cdot \text{Rel}(G_2 * e_k). \end{aligned}$$

Proof: See the Appendix, Section 4.

Lemma 4: If network G_1 with 2 parallel links, e_i, e_j, is reduced (rule r6) to a new network G_2, with the replaced link e_k, then

$$\begin{aligned} \text{Rel}(G_1 * e_i) &= \text{Rel}(G_1 * e_j) = \text{Rel}(G_2 * e_k), \\ \text{Rel}(G_1 - e_i) &= q_j \cdot \text{Rel}(G_2 - e_k) + p_j \cdot \text{Rel}(G_2 * e_k), \\ \text{Rel}(G_1 - e_j) &= q_i \cdot \text{Rel}(G_2 - e_k) + p_i \cdot \text{Rel}(G_2 * e_k). \end{aligned}$$

Proof: See the Appendix, Section 5.

Lemma 5: If network G_X has only 2 links, e_1, e_2, emanating from the source (or into the sink), then

$$\begin{aligned} \text{Rel}(G_X - e_1 * e_2) &= \frac{\text{Rel}(G_X - e_1)}{p_2}, \\ \text{Rel}(G_X * e_1 - e_2) &= \frac{\text{Rel}(G_X - e_2)}{p_1}; \end{aligned} \tag{6}$$

$$\begin{aligned} \text{Rel}(G_X * e_1 * e_2) &= \frac{p_1 \cdot \text{Rel}(G_X * e_1) - q_2 \cdot \text{Rel}(G_X - e_2)}{p_1 \cdot p_2} \\ &= \frac{p_2 \cdot \text{Rel}(G_X * e_2) - q_1 \cdot \text{Rel}(G_X - e_1)}{p_1 \cdot p_2}. \end{aligned} \tag{7}$$

Proof: See the Appendix, Section 6.

Lemma 6: If network G containing a triangle subgraph with 2 sides, $e_{s,1}$, $e_{s,2}$, and the base, $e_{b,1}$ and $e_{b,2}$, is reduced (triangle reduction) to a new network G_X, with the replaced links e_1, e_2, then

$$\begin{aligned} \text{Rel}(G - e_{s,1}) &= \alpha \cdot p_{s,2} \cdot q_{b,2} + \gamma \cdot p_{s,2} \cdot p_{b,2}, \\ \text{Rel}(G - e_{s,2}) &= \beta \cdot p_{s,1} \cdot q_{b,1} + \gamma \cdot p_{s,1} \cdot p_{b,1}, \\ \text{Rel}(G - e_{b,1}) &= \alpha \cdot q_{s,1} \cdot p_{s,2} \cdot q_{b,2} + \beta \cdot p_{s,1} \cdot q_{s,2} + \gamma \\ &\quad \cdot (q_{s,1} \cdot p_{s,2} \cdot p_{b,2} + p_{s,1} \cdot p_{s,2}), \\ \text{Rel}(G - e_{b,2}) &= \alpha \cdot q_{s,1} \cdot p_{s,2} + \beta \cdot p_{s,1} \cdot q_{s,2} \cdot q_{b,1} + \gamma \\ &\quad \cdot (p_{s,1} \cdot q_{s,2} \cdot p_{b,1} + p_{s,1} \cdot p_{s,2}), \\ \text{Rel}(G * e_{s,1}) &= \beta \cdot q_{s,2} \cdot q_{b,1} + \gamma \cdot (p_{s,2} + q_{s,2} \cdot p_{b,1}), \\ \text{Rel}(G * e_{s,2}) &= \alpha \cdot q_{s,1} \cdot q_{b,2} + \gamma \cdot (p_{s,1} + q_{s,1} \cdot p_{b,2}), \\ \text{Rel}(G * e_{b,1}) &= \alpha \cdot q_{s,1} \cdot p_{s,2} \cdot q_{b,2} + \gamma \\ &\quad \cdot (p_{s,1} + q_{s,1} \cdot p_{s,2} \cdot p_{b,2}), \\ \text{Rel}(G * e_{b,2}) &= \beta \cdot p_{s,1} \cdot q_{s,2} \cdot q_{b,1} + \gamma \\ &\quad \cdot (p_{s,2} + p_{s,1} \cdot q_{s,2} \cdot p_{b,1}). \\ \alpha &\equiv \text{Rel}(G_X - e_1 * e_2), \\ \beta &\equiv \text{Rel}(G_X * e_1 - e_2), \\ \gamma &\equiv \text{Rel}(G_X * e_1 * e_2). \end{aligned} \tag{8}$$

Proof: See the Appendix, Section 7.

The goal is to derive $\text{MRI}(e_i)$ of link e_i in a given network. Theorem 1 shows the computation of the MRI based on: a) the reduction factors and b) backward TR formulas detailed in lemma 6.

Theorem 1: In G_j, the MRI of a link e_i belonging to both the original network G and G_j, is

$$\begin{aligned} \text{MRI}(e_i) &= R_j \cdot [\text{Rel}(G_j * e_i) - \text{Rel}(G_j - e_i)]; \\ R_j &\equiv \text{reduction factor in reduction step } j. \end{aligned} \tag{9}$$

Proof: See the Appendix, Section 8.

B. The Detailed Algorithm

```
Algorithm The_2-P(G, R)
Input: A network G with source s, sink
  t, and the failure probabilities of
  the links; the corresponding reduction
  factor, R; Initially, R = 1;
```

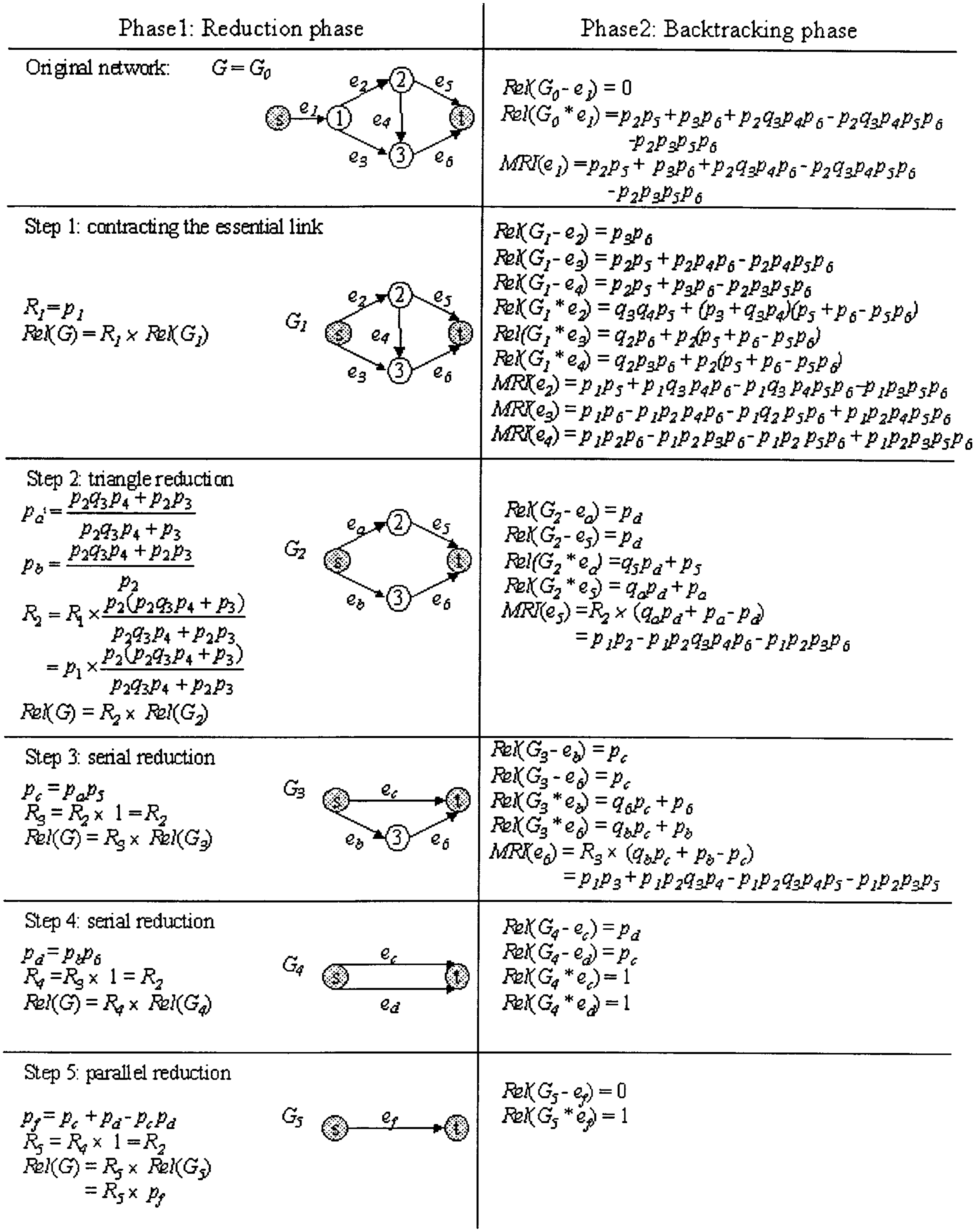


Fig. 6. Example of the MRI evaluation, based on the 2-P algorithm.

```
Output: The MRI of all links;
Begin
IF G is a source-sink network containing
  a single link e_i,
  THEN return: Rel(G − e_i) = 0, Rel(G * e_i) = 1;
  ELSE
IF G is not a reducible+ network
  THEN exit;
  ELSE
  Reduce G to G' using any 1 of the 7
reduction axioms;
  CR = R* Transformation factor;
```

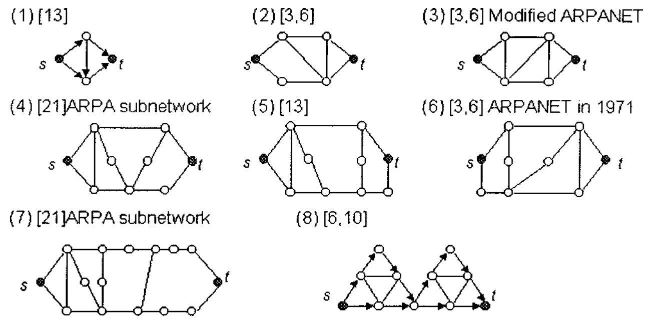


Fig. 7. Benchmarks and real networks.

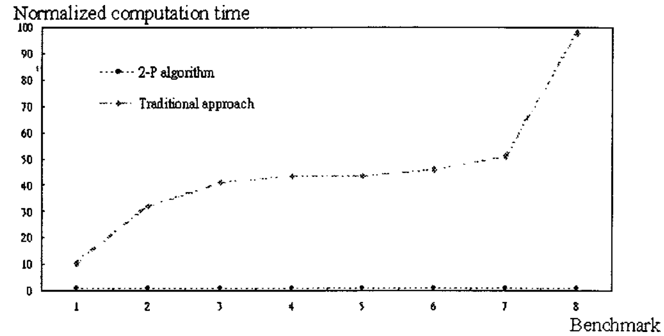


Fig. 8. Comparisons of computation time.

```
    /* CR is the current reduction factor
*/
  The\_2-P(G, CR);
  Compute Rel(G−e) and Rel(G∗e), based on
the
    backward TR formulas;
  IF link e_i is in G and not in G′
    THEN Compute MRI(e) based on theorem
1;
    END_IF;
  END_IF;
END_IF;
End Algorithm
```

Fig. 6 illustrates the computation of MRI via an example. The original network G is transformed to a source-sink network G_5 through 5 reduction steps in the reduction phase. Then, $\text{Rel}(G_5 - e_f) = 0$ and $\text{Rel}(G_5 * e_f) = 1$. In the backtracking phase, for example, because a) e_d replaces e_b and e_6 in reduction step 4, resulting in the reduced network G_4, and b) e_6 is the only link contained in G, then the MRI of e_6 is computed, as shown in the figure. The MRI of the remaining links can be similarly derived.

C. Computational Complexity Analysis

The reduction phase involves at most $m-1$ reduction steps to transform a reducible$^+$ network to a source-sink network. The backtracking phase requires constant time to evaluate closed-form expressions at each of MRI backtracking steps. This yields a complexity of $O(m)$ for computing MRI based on the 2-P Algorithm.

D. Experimental Results

An experiment compared the 2-P Algorithm and the traditional MRI-computation approach using (1) [7], in Sun ServexStation 5 using a collection of real networks and benchmarks [3], [6], [10], [13], [21] as shown in Fig. 7. Fig. 8 displays the computation time of the traditional approach with respect to the normalized computation time of the 2-P MRI algorithm. Fig. 8 shows that the 2-P algorithm outperforms the traditional approach by 2 orders-of-magnitude.

APPENDIX

1) Proof of the Triangle Reduction Rule: According to the factoring theorem [10], $\text{Rel}(G)$ can be partitioned into the 16 subproblems in Fig. 9, corresponding to 4 graphs, G_a, G_b, G_c, G_d. For example, G_b is related to G by the presence of link $e_{s,1}$ and the absence of links $e_{s,2}$, $e_{b,1}$: $G_b = G * e_{s,1} - e_{s,2} - e_{b,1}$. According to r4a and r1, the s is compressed with n_1, and valueless link $e_{b,2}$ is removed, resulting in 2 equal-valued subproblems,

$$\begin{aligned}\text{Rel}(G_b) &= \text{Rel}(G * e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2}) \\ &= \text{Rel}(G * e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2}).\end{aligned}$$

As a result, G_X can be associated with G_b by the presence of link e_1 and the absence of link e_2; thus

$$\text{Rel}(G_b) = \text{Rel}(G_X * e_1 - e_2).$$

Apply the same logic of relating G_a, G_c, G_d to G_X; the result is

$$\begin{aligned}\text{Rel}(G_X) &= p_1 \cdot q_2 \cdot \text{Rel}(G_X * e_1 - e_2) + q_1 \cdot p_2 \\ &\quad \cdot \text{Rel}(G_X - e_1 * e_2) + p_1 \cdot p_2 \\ &\quad \cdot \text{Rel}(G_X * e_1 * e_2) \\ &= p_1 \cdot q_2 \cdot \text{Rel}(G_b) + q_1 \cdot p_2 \cdot \text{Rel}(G_c) + p_1 \\ &\quad \cdot p_2 \cdot \text{Rel}(G_d). \end{aligned} \tag{10}$$

Also

$$\begin{aligned}\text{Rel}(G) &= p_{s,1} \cdot q_{s,2} \cdot q_{b,1} \cdot \text{Rel}(G_b) + q_{s,1} \cdot p_{s,2} \\ &\quad \cdot q_{b,2} \cdot \text{Rel}(G_c) + \psi_{1,1} \cdot \text{Rel}(G_d) \\ \psi_{1,1} &\equiv (p_{s,1} \cdot q_{s,2} \cdot p_{b,1} + q_{s,1} \cdot p_{s,2} \cdot p_{b,2} + p_{s,1} \cdot p_{s,2}).\end{aligned} \tag{11}$$

Multiply (10) by transformation factor C,

$$\begin{aligned}C \cdot \text{Rel}(G_X) &= C \cdot p_1 \cdot q_2 \cdot \text{Rel}(G_b) + C \cdot q_1 \cdot p_2 \cdot \text{Rel}(G_c) \\ &\quad + C \cdot p_1 \cdot p_2 \cdot \text{Rel}(G_d).\end{aligned} \tag{12}$$

Equate (11) and (12),

$$\text{Rel}(G) = \text{Rel}(G_X) \cdot C, \tag{13}$$

$$\begin{aligned}&p_1 \cdot q_2 = \frac{p_{s,1} \cdot q_{s,2} \cdot q_{b,1}}{C}, \quad q_1 \cdot p_2 = \frac{q_{s,1} \cdot p_{s,2} \cdot q_{b,2}}{C}, \\ &p_1 \cdot p_2 = \frac{\psi_{1,1}}{C}.\end{aligned} \tag{14}$$

Rearrange (14); then directly derive (2), (3), (5), and thus prove the theorem.

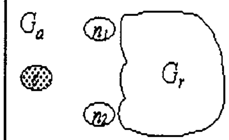
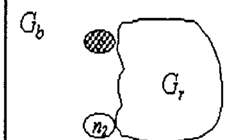
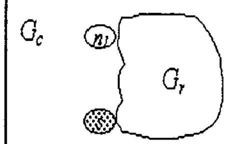
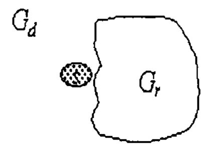
Decomposition	Subproblems for G	Corresponding graph after factoring	Subproblems for Gx
$G_a = G - e_{s,1} - e_{s,2}$	4 subproblems: $Rel(G_a) = Rel(G - e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} * e_{b,1} * e_{b,2})$		$Rel(G_a)$ $= Rel(G_x - e_1 - e_2) = 0$
$G_b = G * e_{s,1} - e_{s,2} - e_{b,1}$	2 subproblems: $Rel(G_b) = Rel(G * e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2})$		$Rel(G_b)$ $= Rel(G_x * e_1 - e_2)$
$G_c = G - e_{s,1} * e_{s,2} - e_{b,2}$	2 subproblems: $Rel(G_c) = Rel(G - e_{s,1} * e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} * e_{b,1} - e_{b,2})$		$Rel(G_c)$ $= Rel(G_x - e_1 * e_2)$
$G_d = G * e_{s,1} * e_{s,2}$	8 subproblems: $Rel(G_d) = Rel(G * e_{s,1} - e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} - e_{s,2} * e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} * e_{b,1} * e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} * e_{b,1} * e_{b,2})$		$Rel(G_d)$ $= Rel(G_x * e_1 * e_2)$
$Rel(G) = p_{s,1} q_{s,2} q_{b,1} \times Rel(G_b) + q_{s,1} p_{s,2} q_{b,2} \times Rel(G_c)$ $(p_{s,1} q_{s,2} p_{b,1} + q_{s,1} p_{s,2} p_{b,2} + p_{s,1} p_{s,2}) \times Rel(G_d)$			$Rel(G_x) =$ $p_1 q_2 \times Rel(G_b)$ $+ q_1 p_2 \times Rel(G_c)$ $+ p_1 p_2 \times Rel(G_d)$

Fig. 9. Association of $\text{Rel}(G)$ and $\text{Rel}(G_X)$.

2) Proof of Lemma 1: According to the factoring theorem [10], $\text{Rel}(G)$ can be expressed as

$$\text{Rel}(G) = q_i \cdot \text{Rel}(G - e_i) + p_i \cdot \text{Rel}(G * e_i). \tag{15}$$

Because e_i is a valueless link, according to rules r1–r3,

$$\text{Rel}(G) = \text{Rel}(G - e_i). \tag{16}$$

From (15) and (16), the lemma is proved.

3) Proof of Lemma 2: According to rule r4,

$$\text{Rel}(G) = p_i \cdot \text{Rel}(G * e_i). \tag{17}$$

Based on the factoring theorem [10],

$$\text{Rel}(G) = q_i \cdot \text{Rel}(G - e_i) + p_i \cdot \text{Rel}(G * e_i). \tag{18}$$

The lemma is directly proved from (17) and (18).

4) Proof of Lemma 3: After removing $e_i(e_j)$ from network G_1, then $e_j(e_i)$ becomes a valueless link. Thus,

$$\text{Rel}(G_1 - e_i) = \text{Rel}(G_1 - e_j) = \text{Rel}(G_2 - e_k). \tag{19}$$

Based on the factoring theorem [10],

$$\text{Rel}(G_1 * e_i) = \frac{\text{Rel}(G_1) - q_i \cdot \text{Rel}(G_1 - e_i)}{p_i}$$
$$\text{Rel}(G_1 * e_j) = \frac{\text{Rel}(G_1) - q_j \cdot \text{Rel}(G_1 - e_j)}{p_j}. \tag{20}$$

From rule r5,

$$\text{Rel}(G_1) = \text{Rel}(G_2) = q_k \cdot \text{Rel}(G_2 - e_k) + p_k \cdot \text{Rel}(G_2 * e_k). \tag{21}$$

The lemma is directly proved from (19)–(21).

5) Proof of Lemma 4: After compressing $e_i(e_j)$ in G_1, then $e_j(e_i)$ becomes redundant. Therefore,

$$\text{Rel}(G_1 * e_i) = \text{Rel}(G_1 * e_j) = \text{Rel}(G_2 * e_k). \tag{22}$$

Based on the factoring theorem [10],

$$\text{Rel}(G_1 - e_i) = \frac{\text{Rel}(G_1) - p_i \cdot \text{Rel}(G_1 * e_i)}{q_i}$$
$$\text{Rel}(G_1 - e_j) = \frac{\text{Rel}(G_1) - p_j \cdot \text{Rel}(G_1 * e_j)}{q_j}. \tag{23}$$

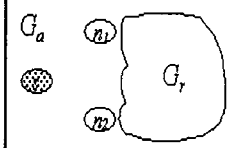
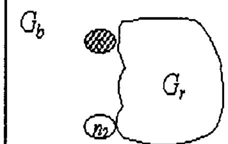
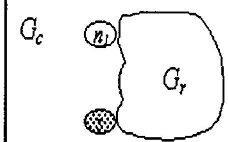
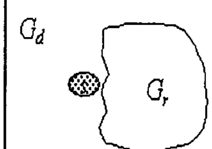
Decomposition	Subproblems for G	Corresponding graph after factoring	Subproblems for Gx
$G_a = G - e_{s,1} - e_{s,2}$	4 subproblems: $Rel(G_a) = Rel(G - e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} - e_{s,2} * e_{b,1} * e_{b,2})$		$Rel(G_a)$ $= Rel(G_x - e_1 - e_2) = 0$
$G_b = G * e_{s,1} - e_{s,2} - e_{b,1}$	2 subproblems: $Rel(G_b) = Rel(G * e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2})$		$Rel(G_b)$ $= Rel(G_x * e_1 - e_2)$
$G_c = G - e_{s,1} * e_{s,2} - e_{b,2}$	2 subproblems: $Rel(G_c) = Rel(G - e_{s,1} * e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} * e_{b,1} - e_{b,2})$		$Rel(G_c)$ $= Rel(G_x - e_1 * e_2)$
$G_d = G * e_{s,1} * e_{s,2}$	8 subproblems: $Rel(G_d) = Rel(G * e_{s,1} - e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} - e_{s,2} * e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G - e_{s,1} * e_{s,2} * e_{b,1} * e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} - e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} - e_{b,1} * e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} * e_{b,1} - e_{b,2})$ $= Rel(G * e_{s,1} * e_{s,2} * e_{b,1} * e_{b,2})$		$Rel(G_d)$ $= Rel(G_x * e_1 * e_2)$

Fig. 10. Relationships among the subproblems of Rel(G) and Rel(G_X).

Then, from rule r6,

$$\begin{aligned}\mathrm{Rel}(G_1) &= \mathrm{Rel}(G_2)\\ &= q_k \cdot \mathrm{Rel}(G_2 - e_k) + p_k \cdot \mathrm{Rel}(G_2 * e_k). \quad (24)\end{aligned}$$

The lemma is directly proved from (22)–(24).

6) Proof of Lemma 5: After removing e_1 (e_2) from G_X, then e_2 (e_1) becomes an essential link of G_X. Equation (6) is derived from lemma 2. Based on the factoring theorem [10],

$$\begin{aligned}\mathrm{Rel}(G_X * e_1 * e_2) &= \frac{\mathrm{Rel}(G_X * e_1) - q_2 \cdot \mathrm{Rel}(G_X * e_1 - e_2)}{p_2}\\ &= \frac{\mathrm{Rel}(G_X * e_2) - q_1 \cdot \mathrm{Rel}(G_X - e_1 * e_2)}{p_1}. \quad (25)\end{aligned}$$

Substitute Rel$(G_X * e_1 - e_2)$ and Rel$(G_X - e_1 * e_2)$ in (25), based on (6). Eq. (7) of the lemma is proved.

7) Proof of Lemma 6: Based on the factoring theorem [10], Rel(G) can be partitioned into 16 subproblems corresponding to 4 graphs, G_a, G_b, G_c, G_d [20], as shown in Fig. 10. In this figure, for example, graph G_b is related to graph G by the presence of link $e_{s,1}$, and by the absence of links $e_{s,2}$, $e_{b,1}$: $G_b = G * e_{s,1} - e_{s,2} - e_{b,1}$. The reduction, based on rules r4a and r1, results in 2 equal-valued subproblems,

$$\begin{aligned}\mathrm{Rel}(G_b) &= \mathrm{Rel}(G * e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})\\ &= \mathrm{Rel}(G * e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2}).\end{aligned}$$

As a result, G_X can be associated with G_b by the presence of link e_1 and by the absence of link e_2: Rel$(G_b) =$ Rel$(G_X * e_1 - e_2)$.

Apply the same logic of relating other graphs to G_X; the resulting equations are given in Fig. 10 under "Subproblems for G" and "Subproblems for G_X."

Based on the factoring theorem [10], Rel$(G - e_{s,1})$ can be partitioned into 8 subproblems and expressed as:

$$\begin{aligned}&\mathrm{Rel}(G - e_{s,1})\\ &\quad = q_{s,2} \cdot q_{b,1} \cdot q_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} - e_{s,2} - e_{b,1} - e_{b,2})\\ &\qquad + q_{s,2} \cdot q_{b,1} \cdot p_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} - e_{s,2} - e_{b,1} * e_{b,2})\\ &\qquad + q_{s,2} \cdot p_{b,1} \cdot q_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} - e_{s,2} * e_{b,1} - e_{b,2})\\ &\qquad + q_{s,2} \cdot p_{b,1} \cdot p_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} - e_{s,2} * e_{b,1} * e_{b,2})\\ &\qquad + p_{s,2} \cdot q_{b,1} \cdot q_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} * e_{s,2} - e_{b,1} - e_{b,2})\\ &\qquad + p_{s,2} \cdot q_{b,1} \cdot p_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} * e_{s,2} - e_{b,1} * e_{b,2})\\ &\qquad + p_{s,2} \cdot p_{b,1} \cdot q_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} * e_{s,2} * e_{b,1} - e_{b,2})\\ &\qquad + p_{s,2} \cdot p_{b,1} \cdot p_{b,2} \cdot \mathrm{Rel}(G - e_{s,1} * e_{s,2} * e_{b,1} * e_{b,2}). \quad (26)\end{aligned}$$

From the equations in Fig. 10, and (26), the first equation in (8) is directly derived. The rest of the equations in (8) are similarly derived.

8) Proof of Theorem 1: Based on the factoring theorem [10]

$$\text{Rel}(G_j) = p_i \cdot \text{Rel}(G_j * e_i) + (1 - p_i) \cdot \text{Rel}(G_j - e_i). \quad (27)$$

Since G_j is reduced from G at reduction-step j,

$$\begin{aligned}\text{Rel}(G) &= R_j \cdot \text{Rel}(G_j) \\ &= R_j \cdot [p_i \cdot \text{Rel}(G_j * e_i) + (1 - p_i) \cdot \text{Rel}(G_j - e_i)].\end{aligned} \quad (28)$$

Differentiate (28) with respect to p_i; the result is (9); the theorem is proved.

References

[1] S. Rai, A. Kumar, and E. V. Prasad, "Computer terminal reliability of computer network," *Reliability Engineering*, vol. 16, pp. 109–119, 1986.

[2] S. Rai and A. Kumar, "Recursive technique for computing system reliability," *IEEE Trans. Reliability*, vol. R-36, pp. 38–44, Apr. 1987.

[3] S. Soh and S. Rai, "CAREL: Computer aided reliability evaluator for distributed computing networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 2, pp. 199–213, Apr. 1991.

[4] J. A. Abraham, "An improved algorithm for network reliability," *IEEE Trans. Reliability*, vol. R-28, pp. 58–61, Apr. 1979.

[5] M. O. Ball, "Computational complexity of network reliability analysis: An overview," *IEEE Trans. Reliability*, vol. R-35, pp. 230–239, Aug. 1986.

[6] Y. G. Chen and M. C. Yuang, "A cut-based method for terminal-pair reliability," *IEEE Trans. Reliability*, vol. 45, pp. 413–416, Sep. 1996.

[7] N. Deo and M. Medidi, "Parallel algorithm for terminal-pair reliability," *IEEE Trans. Reliability*, vol. 41, pp. 201–209, Jun 1992.

[8] S. Hariri and C. S. Raghavendra, "SYREL: A symbolic reliability algorithm based on path and cutset methods," *IEEE Trans. Computers*, vol. C-36, pp. 1224–1232, Oct. 1987.

[9] M. Macgregor, W. D. Grover, and U. M. Maydell, "Connectability: A performance metric for reconfigurable transport networks," *IEEE J. Selected Areas in Communication*, vol. 11, pp. 1461–1469, Dec. 1993.

[10] L. B. Page and J. E. Perry, "Reliability of directed networks using the factoring theorem," *IEEE Trans. Reliability*, vol. 38, pp. 556–562, Dec. 1989.

[11] D. Torrieri, "An efficient algorithm for the calculation of node-pair reliability," in *Proc. IEEE MILCOM'91*, Nov. 1991, pp. 187–192.

[12] ——, "Calculation of node-pair reliability in large networks with unreliable nodes," *IEEE Trans. Reliability*, vol. 43, pp. 375–377, Sep. 1994.

[13] Y. B. Yoo and N. Deo, "A comparison of algorithms for terminal-pair reliability," *IEEE Trans. Reliability*, vol. 37, pp. 210–215, Jun. 1988.

[14] J. Sharma, "Algorithm for reliability evaluation of a reducible network," *IEEE Trans. Reliability*, vol. R-25, pp. 337–339, Dec. 1976.

[15] K. Nakashima and K. Yamato, "Variance-importance of system components," *IEEE Trans. Reliability*, vol. R-31, pp. 99–100, Apr. 1982.

[16] R. E. Barlow and F. Proschan, *Statistical Theory of Reliability and Lifetesting: Probability Models*: Holt, Rinehart, Winston, 1975.

[17] J. Hong and C. Lie, "Joint reliability-important of two edges in an undirected network," *IEEE Trans. Reliability*, vol. 42, pp. 17–23, March 1993.

[18] L. Page and J. Perry, "Reliability polynomials and link importance in networks," *IEEE Trans. Reliability*, vol. 43, pp. 51–58, March 1994.

[19] M. Armstrong, "Joint reliability-important of components," *IEEE Trans. Reliability*, vol. 44, pp. 408–412, Sep. 1995.

[20] S. J. Hsu and M. C. Yuang, "Efficient computation of terminal-pair reliability using triangle reduction in network management," in *Proc. IEEE ICC*, 1998, pp. 281–285.

[21] E. Hansler, G. K. McAuliffe, and R. S. Wilkov, "Exact calculation of computer network reliability," *Networks*, vol. 4, pp. 95–112, 1974.

Steen J. Hsu (born 1963, Taiwan) received the M.S. (1992) in computer science, and Ph.D. (1999) in information engineering from the National Chiao Tung University, Taiwan. He is an Assistant Professor in the Department of Information Engineering at the I-Shou University, Taiwan, since 1999. His research interests include network reliability analysis, QoS routing, and wireless LAN.

Maria C. Yuang received the BS (1978) in applied mathematics from the National Chiao Tung University, Taiwan; the M.S. (1981) in computer science from the University of Maryland, College Park; and Ph.D. (1989) in electrical engineering and computer science from the Polytechnic University, Brooklyn, New York. From 1981 to 1990, she was with AT&T Bell Laboratories and Bell Communications Research (Bellcore), where she was a member of technical staff working on high-speed networking and protocol engineering. She was an Adjunct Professor at the Department of Electrical Engineering, Polytechnic University, during 1989–1990. In 1990, she joined National Chiao Tung University, Taiwan, where she is a Professor of the Department of Computer Science and Information Engineering. Her research interests include high speed networking, multimedia communications, performance analysis, and network management.