Decomposition Techniques for Evaluating Network Reliability

Luigi Fratta
Politecnico di Milano, Milano, Italy
Ugo Montanari
University of Pisa, 56100 Pisa, Italy

ABSTRACT

In this paper an efficient technique to evaluate the terminal reliability of a network consisting of unreliable independent undirected arcs is presented. This technique is an extension of the quite useful series-parallel reductions to the case where it is possible to isolate subnetworks of the given network connected to the rest of the network through three or more nodes. It is shown that this technique, based on recursive decompositions, leads to a linear computational complexity for any class of "m=m structured" networks.

1. INTRODUCTION

When dealing with analysis and synthesis of communication networks a very important and hard task is the evaluation of the network reliability.

The parameter which is usually taken as a measure of network reliability is the terminal reliability defined as the probability that there exists at least one path between the two extrema of the network [1]. Many efforts have been spent recently to find efficient analysis procedures to compute such a parameter in networks whose elements have statistically independent failure probabilities.

The existing methods are mainly based on case analysis [2-5] or on paths or cut-sets enumeration [6-8] and their computational complexity grows exponentially with the number of stochastic variables. Their practical application is therefore limited to networks with, say, up to 20 nodes and 40 arcs. Among these methods the most efficient ones take advantage of the series-parallel reductions which allow to eliminate one variable at each application, thus inexpensively reducing the problem complexity. Thus a network which can be solved just by applying series-parallel reductions would have a computational complexity linear with the number of variables. Unfortunately this is not generally the case.

In order to extend the applicability of series-parallel reduction a first generalization has been presented in [9] where a subnetwork connected to the rest of the network through two nodes is separately analyzed and replaced by an arc of suitable failure probability. In the present paper this generalisation is further extended to any kind of subnetwork in order to reduce as much as possible the computational complexity when solving very large communication networks.

2. NETWORKS OF POLYGONS

An hypergraph G = G(N,P) consists of a set of nodes N and a set of polygons P. To every polygon AaP is associated a sequence of nodes (a1,a2,...,am) which constitute the attaching nodes of the polygon. The value m=1,2,... is called the rank of the polygon; if m=2 the polygon is called arc (the standard component of usual networks), if m=3 it is called triangle and so on.

For example in the hypergraph reported in Fig. 1 we have two triangles A,B and one arc C with attaching points (b,d,a), (e,b,c) and (d,e) respectively. The initial node of every polygon and the sense of percurrence are marked by an arrow.

A network H is obtained from an hypergraph by associating to every polygon A an independent stochastic variable xA defined by its probability distribution FA(xA). The possible values of xA are all the "connection states" of the polygon A. Each connection state is characterized by a partition of the attaching nodes of A. For instance the probability Pa(σ), where σ is the connection state \{[a][b,d]\} of the triangle A, represents the probability that nodes b and d are connected through A and node a is disconnected from both b and d (at least as far as A is concerned).

The number \(T_m\) of connection states of a polygon is function of its rank m: more precisely we have [10, p. 73]

\[
T_m = \sum_{k=1}^{m} \binom{m}{k}
\]

where \(\binom{m}{k}\) are the Stirling numbers of the second kind and can be computed by the following recurrent formula

\[
\binom{m}{k} = \begin{cases} 
1 & \text{if } k = 1 \\
\binom{m-1}{k-1} + k \binom{m-1}{k} & \text{if } 1 < k < m
\end{cases}
\]

A few values for \(T_m\) are given in Table I.

<table>
<thead>
<tr>
<th>m</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>(T_m)</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>15</td>
<td>52</td>
<td>203</td>
<td>877</td>
<td>4140</td>
</tr>
</tbody>
</table>

In Table II are reported possible probability distributions of the stochastic variables xA, xB and xC of our example. Note that \(a_1\) states for the i-th attaching point of any polygon. Furthermore in the following we denote as \(p^A_{j}\) the probability of j-th state of polygon A. For instance \(p^A_1 = .15\). Given a network H, as previously defined, and specified a sequence of nodes called extrema (for instance nodes a, b and c in Fig. 1) we are interested in evaluating the probabilities of the connection states of the whole network seen as a polygon attached to its extrema.

An elementary event of a network H is obtained by assigning to each polygon of H a connection state. Since the stochastic variables associated to the polygons are independent, the probability of the elementary event is of course given by the product of the probabilities of the assigned connection states.
TABLE II

<table>
<thead>
<tr>
<th>J</th>
<th>Conn. State</th>
<th>$x_A$</th>
<th>$x_B$</th>
<th>$x_C$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>${a_1,a_2,a_3}$</td>
<td>.7</td>
<td>.6</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>${a_1},{a_2,a_3}$</td>
<td>.1</td>
<td>.09</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>${a_2},{a_1,a_3}$</td>
<td>.15</td>
<td>.11</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>${a_2},{a_1,a_2}$</td>
<td>.04</td>
<td>.15</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>${a_1},{a_2},{a_3}$</td>
<td>.01</td>
<td>.05</td>
<td>-</td>
</tr>
</tbody>
</table>

Every elementary event accomplishes a connection state of the whole network. Therefore the probability of a network connection state is given by the sum of the probabilities of all the elementary events accomplishing that state. In our example assigning connection states 4, 1 and 1 to polygons A, B and C respectively we obtain an elementary event for the network in Fig. 1 of probability $P_A \cdot P_B \cdot P_C = .0216$.

This elementary event may be represented by the graph in Fig. 2 and thus it accomplishes the connection state $\{\{a\},\{b,c\}\}$ of the whole network.

To systematically compute the connection state accomplished by an elementary event it is sufficient to make the union of all the equivalence relations defining the given connection states of the components and to form the transitive closure of the resulting relation. This corresponds to a partition of all the nodes in the network. Projecting this partition on the set of extrema we finally get the accomplished connection state for the whole network.

The analysis of all elementary events for our network is shown in Fig. 3 where the table entries are elementary events labelled by the connecting state they accomplish.

The probabilities of the connection states of $H$ obtained summing up all the elementary events are

$$P_H = P(\{(a,b,c)\}) = \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{3} = \frac{1}{27}$$

$$P_H = P(\{(a)\}) = \frac{1}{3} \cdot \frac{1}{3} \cdot 1 = \frac{1}{9}$$

$$P_H = P(\{(b)\}) = \frac{1}{3} \cdot 1 \cdot 1 = \frac{1}{3}$$

$$P_H = P(\{(c)\}) = 1 \cdot 1 \cdot 1 = 1$$

3. ANALYSIS METHODS

In the previous section the probability distribution of the connection states of a network has been defined. From the definition itself a completely enumerative method for its computation stems. Its complexity however is proportional to the product of the numbers of the connection states of all polygons in the network.

In the following two complementary methods are presented which allow a more efficient analysis of a network $H$.

3.1 DECOMPOSITION METHOD

This method consists essentially of the following operations:

(i) Select a subnetwork $H'$ (containing $n$ polygons) of $H$ which is connected to the rest of $H$ through a sequence of $m$ connecting nodes.

(ii) Analyze the subnetwork $H'$ separately considering the $m$ connecting nodes as its extrema.

(iii) Substitute in $H$ the subnetwork $H'$ with a polygon of rank $m$ and having as probability distribution of its connection states the values computed in (ii) for the subnetwork $H'$.

(iv) Analyze the reduced network obtained in (iii).

When this method is applicable keeping $m$ small it is very convenient since it tends to linearize the computing time which would be otherwise exponential. This property has been already appreciated in the special case of $m=2$ for solving networks consisting only of arcs. In this case, in fact, the decomposition method is nothing else than the generalized series-parallel reduction [9].

For a more detailed complexity analysis let us consider the class of $n\cdot m$ structured networks. Every such a network can be obtained by the iterated application of the decomposition technique described above with $n \geq 2$ and $n$ and $m$ bounded by $n$ and $m$. Furthermore no polygon with more than $m$ attaching points must be present in the original network. For example a series-parallel arc network is a 2-2 structured network.

The computational complexity of performing operation (ii) by complete enumeration is bounded by $\Theta(2^n)$ where $\Theta$ is a constant.

As the number of decompositions is bounded by the number $N$ of polygons in the original network the overall complexity, $\Theta$, is bounded by

$$\Theta \leq N \cdot \Theta(2^n)$$

314-2
We have therefore a linear growth of the computational complexity for any class of \(n\times m\)-structured networks even if the multiplicative coefficient \(B \cdot (\tau_{m})\) may increase very fast with both \(m\) and \(n\).

As an example of 3-2 structured networks we give in Fig. 4 the complete symbolic solution of the class of ladder network.

\[
\begin{align*}
\text{Figure 4}
\end{align*}
\]

3.2 CASE ANALYSIS METHOD

The case analysis method used for analysing networks with arcs [5] can be straightforwardly extended to polygon networks as follow:

(i) Select a polygon \(A\) with rank \(m\) of \(H\).

(ii) Consider all connection states \(\tau_i\) \((i = 1, \ldots, T_m)\) of \(A\) with probability \(p_{A}^{\tau}\) and for any \(\tau_i\) construct the corresponding degraded network(*). \(H_i\).

(iii) Analyze separately all networks \(H_i\) computing its probabilities \(p_{H_i}^{\tau}\).

(iv) The probabilities of \(H\) are given by

\[
\hat{p}_H = \frac{1}{T_m} \sum_{i=1}^{T_m} p_{A}^{\tau} p_{H_i}^{\tau}
\]

The case analysis is carried on until we reach a degraded network whose probabilities are all zero but one. Such a network can be recognized by the fact that all pair of extremal classes are either coalesced or disconnected. This method is still essentially enumerative, but the termination condition above allows, generally, to analyze the network \(H\) without considering all elementary events.

3.3 MIXED METHOD

The two previous methods can be conveniently combined. For instance, when no decomposition is possible with a small value of \(m\) the application of the case analysis method is advisable. Then on some of the degraded networks obtained a decomposition with small \(m\) may be possible.

In the following we report the full analysis of the network shown in Fig. 1, by using the mixed method. However for explicative purposes the choices made are not necessarily the best in every specific situation.

(*) The degraded network \(H_i\) corresponding to the connection state \(\tau_i\) is obtained from \(H\) by coalescing all nodes of \(H\) which belong to the same class in the partition \(\tau\) and by erasing polygon \(A\).
position is more convenient in comparison with case analysis method. We can therefore conclude that one must try the decomposition method with a smaller number of connecting points when the whole problem is large and thus no further branching is required.

The application of the case analysis technique to subnetworks 2, 3 and 5 generates only 16 terminal networks while the method based on the elementary events would have produced 30 cases.

The nodes marked with a circle in the case tree correspond to applications of the decomposition method. For instance in the node reached by path $p_2$ we apply the decomposition shown in Fig. 7(a). The solution of the corresponding network is represented in Fig. 6(b).

$$
\begin{array}{c}
\text{(a)} \\
\text{(b)} \\
\text{(c)}
\end{array}
$$

Figure 7

Note that this solution requires a further decomposition shown in Fig. 7(b) whose solution is in Fig. 6(c). The relations associated to circled nodes in the case tree indicate the correspondences among the states of the decomposed network and those of the original network. In our case we have for instance that connection state 2 of D contributes to connection state 4 of H. Similarly, when decomposing node reached by path $p_2$ we apply the decomposition in Fig. 7(c) whose solution is represented in Fig. 6(d).

4. CONCLUSIONS

Among the methods presented in the previous section the mixed one is in general clearly better. However it requires at any application the choice of using either the decomposition or the case analysis method. Furthermore, once the technique has been decided, to perform step (1) it is necessary to take a decision which is undefined.

No matter which method is chosen and no matter which decision is taken the algorithm is correct and terminates.

However the computing times can be vastly different according to the subnetwork H or the polygon A selected in step (1) of decomposition or case analysis method respectively. Thus any implementation of this method must include a heuristic procedure for taking such a decision.

Here we give a few hints for possible heuristics. A first remark is that the computation effort grows very fast with the number of extremes of the network and the ranks of the polygons as suggested by Table 1 and formula 1. On the other hand from 1 we observe that the repetitive application of the decomposition method tends to maintain the complexity linear with the size of the network. While the case analysis method is intrinsically exponential.

We can therefore conclude that one must try the decomposition method with a "small" number of connecting points first. This number can be larger at the initial recursion step when the whole problem is large and thus the decomposition is more convenient in comparison with case analysis.

A second hint is that a decomposition is always convenient when the number of connecting points is smaller or equal to the maximum rank of the polygons in the subnetwork.

When the alternative is between a decomposition with a large number of connecting points and a case analysis, the choice is critical. In this case an interaction with human intuition may be quite valuable especially at the first iterations of the method. As an example of a crucial step in a practical case we give in Fig. 8 a possible decomposition allowing to solve the U-K trunk networks [11, p. 344]. This step gives a decomposition in five subnetworks with at most four extrema.

$$
\begin{array}{c}
\text{Figure 8}
\end{array}
$$

REFERENCES


