# Static network performability analysis

G. Rubino, ENSTB & IRISA
IRISA, Campus de Beaulieu, 35042 Rennes Cédex, France
rubino@{irisa,rennes.enst-bretagne}.fr

## Abstract

This work deals with the quantitative analysis of the topology of a communication network whose components are subject to failures and repairs. After reviewing the standard *network reliability* theory, we consider performability measures. The purpose of the paper is to propose new performability metrics to evaluate this aspect of a communication network, and a Monte Carlo scheme to estimate them.

## 1 Overview

The most widely used model to assess the reliability of the topology of a communication network consists of a stochastic graph, that is, a graph whose components are weighted by probabilities. Theses probabilities are the *elementary reliabilities* or shortly, the reliabilities of the components. Let us call this object a *stochastic network*. Formally, a stochastic network is a pair $\mathcal{N} = (\mathcal{G}, r)$ where $\mathcal{G}$ is a graph and $r$ denotes the family of elementary reliabilities. The context is a *static* one, that is, the system is considered at a fixed point in time $\tau$ (possibly, $\tau = +\infty$), and the reliability $r_i$ of component $i$ is simply the probability that component $i$ is working at time $\tau$. We are in a binary world, components and systems are either *up* or *down*, and the objective is to analyze the probability $R$ that the whole system is up. To define this, some criteria based on connectivity properties is used. The basic ones are the fact that two fixed nodes can communicate (source-to-terminal reliability) or the fact that every pair of nodes can communicate (all-terminal reliability). For a survey in the area, see [1] and [2]. We summarize the area in Section 2.

Nowadays, the components of a network are highly reliably; moreover, in many cases the topologies are quite dense. A consequence of this is the fact that the output measure $R$ can be very close to one. This makes difficult the comparison between alternative topologies; the differences between system reliabilities can even be less than the precision used to measure (or to estimate) data. A second point to make here is that, in any case, the standard reliability metrics are not very informative, they only measure the fact that a quite global property (for instance, the fact that the network is connected) is satisfied. This suggests to propose the use of a performability index which could take into account not only the fact that the network works, but also some information about the way it works. Some effort has already been done in this direction. Basically, the idea is to leave the binary world and to change it into a multi-state one, taking into account supplementary information such as costs, capacities, etc. We review these models in Section 3.

In this text, we propose to keep the basic connectivity-based approach, and to consider metrics of the form $E(Y \mid I)$, where $Y$ measures, in some way, how much work can be done by the system (for instance, in term of possible connections) and $I$ is some undesirable event (for instance, the fact that the graph is no more connected). Classical reliability analysis will lead to compute simply the probability of $I$. Here, we propose to compute, for instance, the expected number of communicating pairs given that the network is no more connected. Since this type of quantity takes into account both dependability aspects and performance ones, it is a performability measure. The evaluation of these numbers is a very hard task, harder (from the complexity theory point of view) than the evaluation of standard reliability measures. For this reason, we show how to estimate it by means of an efficient Monte Carlo procedure. In spite of the fact that the probability of event $I$ is usually very low, we will show how a recently proposed algorithmic scheme [3] can be transported into this context and leads to a satisfactory procedure. This material is developed in Section 4.

## 2 Network reliability: a connectivity-based approach

We are given a stochastic network $\mathcal{N} = (\mathcal{G}, r)$ modeling some communication network. The objective is to evaluate reliability measures associated with the topology (so, with graph $\mathcal{G}$). For simplicity, we assume that the components of the network that are subject to failures are its lines, modeled by the edges of $\mathcal{G}$. Moreover, the graph is undirected (lines are bi-directional), connected and without loops. At the point in time $\tau$ of interest, the set of operating lines defines a (random) partial graph of $\mathcal{G}$, which we denote by $G$. We are then given a subset $\mathcal{K}$ of nodes, that is, $\mathcal{K} \subseteq \mathcal{V}$, and $R$ is the probability that the nodes in $\mathcal{K}$ are connected, that is, that they belong to the same connected component of $G$. In this case, we say that $G$ is $\mathcal{K}$-connected. Two important particular cases are $\mathcal{K} = \{s, t\}$, the so-called *source-to-terminal* case, and $\mathcal{K} = \mathcal{V}$, the *all terminal* case.

The evaluation of $R$ is a computational hard problem. Formally, it belongs to the $\#P$–complete class, a family of $NP$–hard problems not known to be in $NP$. This implies that a $\#P$–complete problem is at least as hard as a $NP$–complete one. Even in very restricted classes of graphs, the computation of $R$ remains in this complexity class. For instance, it is shown in [4] that the computation of the source-to-terminal reliability is in the $\#P$–complete class even if the graph is planar (in fact, $s,t$-planar) and has vertex degree at most equal to three. From the practical point of view, this means that a graph with, say, more than one hundred elements (nodes, lines) can not be exactly evaluated (except for special topologies). An efficient alternative is then to use a Monte Carlo method. The standard version consists of generating a $N$-sample $G^{(1)}, \ldots, G^{(N)}$ of $G$, and to estimate $R$ by

$$\widehat{R} = \frac{1}{N} \sum_{n=1}^{N} 1_{\left\{G^{(n)} \text{ is } \mathcal{K}\text{-connected}\right\}}. \qquad (1)$$

In many cases, this approach can handle medium and large size models, but, of course, it has its own drawbacks; its efficiency is sensitive to the numerical values of the data. In particular, the standard estimator becomes of no use in the *rare event* case, that is, when $R$ is close to one. Unfortunately, this a very interesting case. However, the sensitivity of Monte Carlo techniques to the effective values of the set of data can, in fact, be used to design efficient schemes dealing to new estimators having a reduced variance (reduced with respect to the variance of $\widehat{R}$, which value is $R(1-R)/N$). See references in [2]. A second important feature of the Monte Carlo approach is its flexibility. It can be adapted to the analysis of many different measures, at least in its standard form. This is the main fact exploited in this work (Section 4).

## 3 Network performability: using more data

We mainly follow here Section 6 of [1], where the reader can find details and many references. Let us modify the previously described model in the following way. Instead of associating with each edge $i$ its reliability $r_i$, we associate with it a discrete random variable (r.v.) $X_i$. For instance, $X_i$ may represent some measure of the capacity of the line to transport information. With some probability, the capacity can be 0, due to a fatal failure, or can be at some "intermediate" level, if the line operates in a "degradated" mode, or is at a maximal level, for instance after the renewal of a failed component. The system state is also a discrete r.v. $\Phi$. In the previous example, an usual measure for the whole system is $\Phi$ = the maximum $s,t$-flow between two fixed nodes $s$ and $t$. This is clearly a performability measure, since it takes into account both the performance of the system and its behavior face to failures and repairs, that is, dependability aspects. The problem is to obtain informations about $\Phi$ (moments, distribution). For instance, in the recent work [5] the authors develop a (Monte Carlo) method to estimate $\Pr(\Phi > d)$ where $d$ is a fixed demand threshold. Another possible measure could be $\Phi$ = fraction of the offered traffic which is lost due to the capacity values, etc.

The main observation to be made is that the evaluation of these metrics is much more difficult than the evaluation of classical reliability ones. It can be seen that for many performability frameworks such as the previously described one, the network reliability context is a particular case. This fact means two things. First, the class of tractable models is much more reduced than in the standard reliability setting and, second, since the related problems are more difficult, available methods are less numerous. A third consequence of this is the fact that the Monte Carlo approach becomes even more important in this case. In [1] one can find pointers to several works performed on the exact evaluation of these metrics as well as on deriving statistical estimators.

## 4 Proposing connectivity-based network performability measures

In [2], we proposed a different approach. We keep the connectivity-based context, that is, we assume that the only available or relevant data is the one described in the model of Section 2. So, our input is again a stochastic network $\mathcal{N} = (\mathcal{G}, r)$. The underlying assumption of the connectivity-based point of view is that as long as a path exists between two nodes in a communication network, then these nodes can communicate. Assume that this assumption is valid. We will show here how to do a performability analysis taking into account the ability of a network to support communications between its different nodes, without using supplementary data. Then, we will exhibit a Monte Carlo scheme to perform the evaluations.

### 4.1 New performability measures

Let us denote by $CC(G)$ the number of connected components of $G$. We have $CC(\mathcal{G}) = 1$ (by hypothesis), and, for instance, $\Pr(CC(G) = 1) = R_{\mathcal{V}}$, where $R_{\mathcal{V}}$ denotes the all-terminal reliability. A way to compare two alternative topologies could be the analysis of the distribution of $CC(G)$, or of some of its moments. Observe that the probabilities $\Pr(CC(G) > k)$ are usually very small for $k \geq 2$. Another possibility is the following. Let us denote by $NCP(G)$ the number of connected pairs of nodes in $G$. For instance, $NCP(\mathcal{G}) = n(n-1)/2$ if $n$ is the number of nodes in the network, since the given graph $\mathcal{G}$ is assumed to be connected. The expectation of the r.v. $NCP(G)$ is related to the source-to-terminal reliability: denoting by $R_{s,t}$ the source-to-terminal reliability between nodes $s$ and $t$, we have

$$NCP(G) = \sum_{\text{all } s \neq t} 1_{\{s \text{ and } t \text{ connected in G}\}},$$

and thus, taking expectations in both sides,

$$\mathrm{E}(NCP(G)) = \sum_{\text{all } s \neq t} R_{s,t}.$$

Since the system reliabilities are usually very high, our proposal is to evaluate the topologies by conditional metrics such as $\mathrm{E}(NCP(G) \mid CC(G) > 1)$. Even if the system reliabilities for two different topologies are very close to one, the proposed conditional expectation can allow to distinguish between them, when the system is working in a degradated mode, allowing only communications between parts of the network. In the last section, we will discuss about another performability measure of this type.

Needless to say, the exact analysis of this class of measure is a very complicated task. However, this is not true if one uses the Monte Carlo approach, at least in a standard way. In [2], we illustrate these proposals using the scholar example depicted in Figure 1 (widely used for illustrative purposes in the research community) and a naive (see next subsection) implementation of the standard or crude estimator.

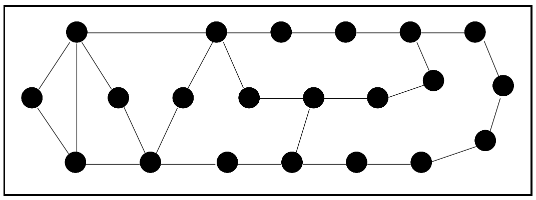

Figure 1: A version of the Arpanet communication network

The elementary reliabilities were first set to 0.9 to avoid long computing times. We obtain the following values: $\mathrm{E}(CC(G)) \approx 1.31$ and $\mathrm{E}(NCP(G)) \approx 197$. Observe that $NCP(\mathcal{G}) = 210$. If we estimate the proposed conditional expectation, we obtain $\mathrm{E}(NCP(G) \mid CC(G) > 1) \approx 158$. In this case, the all-terminal reliability was $R_{\mathcal{V}} \approx 0.752$. Increasing the elementary reliabilities to 0.95, we obtain $R_{\mathcal{V}} \approx 0.933$, saying us that, as expected, the network behaves much better than in the previous case (in the same way, $\mathrm{E}(CC(G))$ decreases to $\approx 1.07$). However, the expected number of communicating pairs does not change very much: we get $\mathrm{E}(NCP(G)) \approx 207$ and $\mathrm{E}(NCP(G) \mid CC(G) > 1) \approx 168$. There is an increase of 24% in the system reliabiliy, but only of 6.3% in the proposed conditional metrics.

### 4.2 Monte Carlo estimation

We briefly present here the main idea of [3]. The point in that paper is that the standard estimator (of $R$ in [3]), can be very efficient even in the rare event case! First, we must remark that the efficiency we are addressing here is an algorithmic concept. We say that what we mostly want and need is an *efficient algorithm*, rather than an efficient estimator (this also has a precise meaning in statistics).

Let us come back to the context of Section 2 and the estimation of the source-to-terminal reliability. Formally, we know that the *precision* of an *estimator* is proportional to its standard deviation. This follows from the central limit theorem, the standard way to derive confidence intervals. We propose to define the *efficiency index* of an implementation of an estimator by its mean execution time multiplied by the precision of the implemented estimator. In symbols, if we denote by $\sigma_N$ the standard deviation of $\widehat{R}_N$ (putting explicitly the size of the sample) and if $\mathcal{A}$ is some implementation of $\widehat{R}$, we define the efficiency index of $\mathcal{A}$ by

$$\mathrm{ei}(\mathcal{A}, N) = \mathrm{E}(T(\mathcal{A}(G), N))\sigma_N, \qquad (2)$$

where $T(\mathcal{A}(G), N)$ is the execution time of $\mathcal{A}$ applied to the input $G$ and using a $N$-sample. What we want is small values of ei. If two different implementations $\mathcal{A}_1$ and $\mathcal{A}_2$ of two estimators verify $\mathrm{ei}(\mathcal{A}_1, N) = \mathrm{ei}(\mathcal{A}_2, N)/2$, then $\mathcal{A}_1$ gives us the same precision than $\mathcal{A}_2$ in (expected) half time, or its speed is twice the speed of $\mathcal{A}_2$ for the same precision.

Consider now the "naive" implementation of $\widehat{R}$ which consists of a loop from 1 to $N$, sampling at each step from $G$ and testing whether it is $s, t$-connected or not. Let us denote it by $\mathcal{A}_0$. We define the *relative efficiency index* of any implementation $\mathcal{A}$ of the same estimator by the ratio

$$\mathrm{relEi}(\mathcal{A}, N) = \frac{\mathrm{ei}(\mathcal{A}_0, N)}{\mathrm{ei}(\mathcal{A}, N)}. \qquad (3)$$

This means that we want values of relEi as high as possible. In [3], we propose the following alternative implementation $\mathcal{A}$ of $\widehat{R}$ which satisfies $\mathrm{relEi}(\mathcal{A}, N) > 1$. We denote by $\gamma$ some elementary path between $s$ and $t$ and by , the event "every link in $\gamma$ is up", whose probability is

$$p_\gamma = \prod_{i \in \gamma} r_i.$$

Let us denote by $F$ the r.v. "first element in a $\infty$-sample $G^{(1)}, G^{(2)}, \ldots$ of $G$ where , does not occurs". Clearly, $F \geq 1$ is a geometric r.v.; we have

$$\Pr(F = n) = p_\gamma^{n-1}(1 - p_\gamma) \quad \text{and} \quad \mathrm{E}(F) = \frac{1}{1 - p_\gamma}.$$

Now, let $G$ denote a new random graph defined on the set of partial graphs of $\mathcal{G}$ where at least one link in $\gamma$ is down, and with probability proportional to the probability of $G$. The new algorithm implementing $\widehat{R}$ is then the following (as in the `C` language, '=' means affectation, and '+=' means "incremented by"):

```
build some path γ;
#success = 0, #samples = 0;
while #samples < N do
   f = sample from F;
   if #samples + f > N then
      #success += N - #samples;
      #samples = N;
   else
      #success += f - 1;
      g' = sample from G';
      if s and t are connected in g' then
         #success += 1;
      #samples += f;
   endif
enddo
estimator = #success/N;
```

The idea is to sample first from $F$. If we obtain the value $f$, it means that in the first $f-1$ realizations, the graph is $s,t$-connected, so, we increment the counter `#success` by the value $f-1$. For the $f$th one, we must check if $s$ and $t$ are connected. Since we know that event , does not occur, we have now to sample from $G'$ and not from $G$. We have no enough room here to explain how to perform this (see [3]). We have discussed the `else` part of the `if` instruction. The `then` part simply handles the case when we get out of the fixed $N$-sample.

Now, observe that the mean execution time of $\mathcal{A}$ for a $N$-sample is less than the mean execution time of $\mathcal{A}_0$ since for each call of $F$ we do not need to explore $F-1$ graphs. Specifically, for not very small values of $N$, we have

$$\frac{\mathrm{E}(T(\mathcal{A}_0, N))}{\mathrm{E}(T(\mathcal{A}, N))} = \mathrm{E}(F) = \frac{1}{1-p_\gamma},$$

leading to

$$\mathrm{relEi}(\mathcal{A}, N) = \frac{1}{1-p_\gamma}.$$

When the elementary reliabilities are high, the relative efficiency index can be high as well. However, given the form of $p_\gamma$, it goes down to zero when the size of the graph grows up. For this reason, in [3] a second version is proposed, which works with a set of disjoint paths between $s$ and $t$, improving by an important factor the index. We refer to [3] for details.

Let us come back now to the estimation of the proposed performability measures. We can adapt the previous scheme, only changing the path $\gamma$ by a spanning tree. If all the links of a spanning tree work, then the network is connected. To derive a more efficient scheme here, we must build a set of disjoint spanning trees having as many trees as possible. In the case of the example given before, we have only one possible spanning tree. Observe that when all the reliabilities are identical, say $r_i = 1-\varepsilon$ for all link $i$, since every spanning tree has exactly $n-1$ links where $n = |\mathcal{V}|$, we have

$$\frac{1}{1-p_\gamma} = \frac{1}{1-(1-\varepsilon^{n-1})} \approx \frac{1}{n-1}\frac{1}{\varepsilon}.$$

For instance, with $r_i = 0.9$ as in one of the examples in [2], since $n = 21$ in the Arpanet graph, we only get a relative index $\mathrm{relEi}(\mathcal{A}, N) \approx 1.13$, almost no gain with respect to the naive implementation (as a matter of fact, this is of no importance since this is not a rare event situation). If we consider the more realistic case of $r_i \approx 1-10^{-4}$, then we get a relative efficiency index of about 500. For $r_i = 1-10^{-5}$, the index value is $\approx$ 5000, etc. Of course, in the case of different elementary reliabilities, we must look for a maximum-weighted spanning tree, using as weights the reliabilities and as operator the product. In a larger graph, the situation is similar as the case of reliability analysis. Using a set of disjoint spanning trees will improve the efficiency of the scheme, with the meaning explained before.

## 5 Current and further research

We are currently analyzing the behavior of the proposed Monte Carlo scheme in graphs having many dozens of links. Also, we are exploring other possible performability measures. Specifically, consider a backbone network and assume that the designer constraint was to build a topology having the property that each pair of nodes can communicate by at least two disjoint paths. That is, we want $G$ bi-connected (and we assume that, by construction, $\mathcal{G}$ has that property). If due to some failure(s), the network is no more bi-connected while still connected, communications between every pair of nodes are supported, while in a less reliable mode. This suggests to perform the analysis conditioned to the situation "$G$ is connected but it is not bi-connected". For instance, we can look at the r.v. $NCP2 =$ "number of pair of nodes connected by at least 2 disjoint paths". The main example is

$$\mathrm{E}(NCP2 \mid G \text{ connected}, G \text{ not bi-connected}).$$

Observe that $NCP2$ can be written

$$NCP2 = \sum_{B,\ \text{block of } G} \frac{|B|(|B|-1)}{2}.$$

The problem, from the evaluation point of view, is to find the equivalent of the $\gamma$ paths, which can be found efficiently, and such that the auxiliary $G'$ graphs can be sampled also in an efficient way. Our research effort is directed in this direction.

## References


[1] M.O. Ball, C Colbourn, and J.S Provan. Network reliability. In M. Ball, T. Magnanti, C. Monma, and G. Nemhauser, editors, *Handbook of Operations Research and Management Science: Network Models*, pages 673–762. Elsevier, Amsterdam, 1995.

[2] G Rubino. Network reliability evaluation. In K. Bagchi and J. Walrand, editors, *State-of-the art in performance modeling and simulation*, pages 275–302. Gordon and Breach Books, 1996. For a copy, send a mail to `rubino@irisa.fr`.

[3] M. El Khadiri and G. Rubino. Accelerating the standard monte carlo evaluation of highly reliable binary systems. In *Second International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. University of Salzburg, Austria, July 1996. For a copy, send a mail to `rubino@irisa.fr`.

[4] J. S. Provan. The complexity of reliability computations in planar and acyclic graphs. *SIAM Journal of Computing*, 15(3), Aug. 1986.

[5] S. Bulteau and M. El Khadiri. A variance reduction monte carlo algorithm for flow network reliability evaluation. In *Second International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*. University of Salzburg, Austria, July 1996. For a copy, send a mail to `sbulteau@irisa.fr`.