

RAn (Robustness Analyser)

Fabrice Saffre
BTexact Technologies
Aadastral Park, Antares 2 pp 5
Martlesham IP5 3RE
United Kingdom
fabrice.saffre@bt.com

1. Introduction

In this paper, we present RAn, a simple and user friendly software tool for analysing topological robustness of random graphs. The underlying model and the methodology are briefly described, then RAn's potential as a research and design tool is illustrated through a basic practical example.

Robustness of complex networks has been extensively discussed in the scientific literature for the last few years. Several authors have pointed out that different topologies would react differently to node failure and/or broken links (see e.g. Albert et al., 2000; Cohen et al., 2000) and that mathematical techniques used in statistical physics could effectively be used to describe their behaviour (see e.g. Callaway et al., 2000). It has also been demonstrated that most artificial networks, including the Internet and the World Wide Web, can be described as complex systems, often featuring "scale-free" properties (see e.g. Albert et al., 1999; Faloutsos et al., 1999; Tadic, 2001).

In this context, it is becoming increasingly obvious that the robustness of a wide variety of real distributed architectures (telecommunication and transportation networks, power grids etc.) is essentially a function of their topology, and could therefore be evaluated on the basis of their blueprint. Similarly, several alternative designs could be compared before their actual implementation, in order, for example, to balance redundancy costs against increased resilience.

2. The model

Efficient quantification and comparison requires selecting a consistent set of global variables that are considered a suitable summary of network behaviour under stress. In a previous work (Saffre and Ghanea-Hercock, 2001), we found that the decay of the average relative size of the largest component $\langle S \rangle$ could effectively be modelled using a basic non-linear equation of the form:

$$\langle S \rangle = \frac{X}{X + e^{\beta x}} \quad [1a]$$

where X and β are constants, while x is the fraction of nodes which have been removed from the original network. Depending on the topology, a better fitting can sometimes be obtained for a slightly different expression:

$$\langle S \rangle = \frac{X}{X + x^\beta} \quad [1b]$$

Equations [1a] and [1b] obey a very similar logic though, and their relative efficiency in describing the system's behaviour can actually be used as a first indication to discriminate between 2 "qualitatively" different categories of architecture.

In any case, if expression [1a] or [1b] give a satisfactory approximation of the decay of a specific network's largest component, then the corresponding X and β global variables are all that is required to quantify its resilience to cumulative node failure. For increased clarity, it might be preferable to use an adjusted value of X :

$$X_c = \frac{\ln(X)}{\beta} \quad [2a]$$

or

$$X_c = \sqrt[\beta]{X} \quad [2b]$$

for [1a] and [1b] respectively. X_c is then the value of x for which the average relative size of the largest component is equal to 0.5, that is: the critical fraction of "missing" nodes above which, on average, less than 50% of the surviving elements are still interconnected. Finally, the value of β itself roughly indicates the slope of the curve around this critical value.

3. Operation

The 3 global variables mentioned in the previous section (β , X and X_c) are automatically computed by RAn, after performing a statistical analysis on data produced using Monte Carlo simulation techniques. Network structure and simulation parameters have to be specified by the user. A Graphical User Interface (GUI) allows these be entered/modified very easily (see Fig. 1). After a properly formatted topology file has been generated for the network to analyse, the user may launch RAn to perform robustness tests.

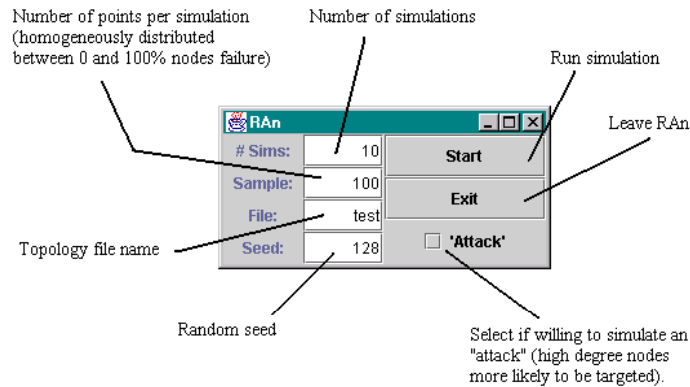


Figure 1: RAn Graphical User Interface (GUI).

The total duration of the simulation process is obviously highly dependent on parameter values and network size. As an order of magnitude, analysing the resilience to random failure of a network up to ten thousand nodes large (with a similar number of connections) is typically done in less than 2 minutes on a standard PII Desktop PC.

After the simulation phase is over, RAn analyses the data and the results are summarised as a series of automatically generated files. In addition to these, RAn also provides a graphical summary, as illustrated in Fig. 2.

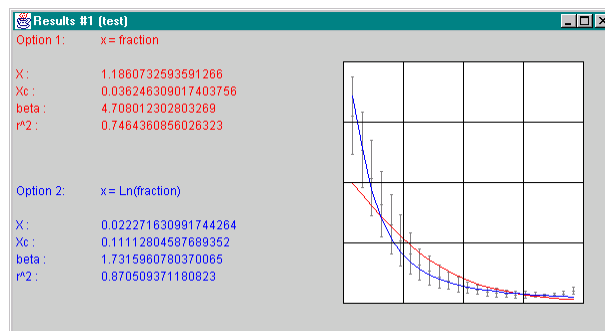


Figure 2: Results window displayed by RAn after the robustness analysis is completed. It includes values for all global variables and a graph showing simulation data (average +/- standard deviation) as well as both fitting curves. In this example (1000 nodes, 999 links, scale-free) expression [1b] (option 2) provides better fitting (brittle network).

If the "Attack" option is selected in the GUI (see Fig. 1), RAn makes the assumption that the attacker possesses partial information about network topology,

and follows a "best guess" strategy in order to choose which node to target next. This is modelled by attributing to each surviving node a probability of being selected that is linearly proportional to its degree k :

$$P_i = \frac{(k_i + 1)}{\sum_{j=1}^n (k_j + 1)} \quad [3]$$

Equation [3] is obviously a variant of the preferential attachment rule presented by Barabasi et al. (1999). Unfortunately, P_i has to be recalculated after each attack in order to take into account the changing probability distribution caused by the elimination of one of the nodes. This increased complexity is the reason why testing a network's resilience for directed attack is considerably more intensive (and time consuming) than for random failure. However, this could probably be improved by optimising the algorithm.

Finally, it might be worth mentioning that the "Attack" scenario, because of its stochastic nature, could also be used to model special forms of accidental damage where connectivity level is involved. For example, it is conceivable that in a network where congestion is a cause for node failure, key relays (high degree nodes) would also be more likely to suffer breakdown, which could easily be modelled using expression [3].

4. Example scenario

This section consists in a scenario emphasising how RAN could be used as a design tool when planning network architecture. The planned network is a relatively large 3000 nodes system. The cheapest way to have them all interconnected (from a strictly topological point of view!) would involve 2999 links. They could all be arranged in a single "star" or in a closed "loop", but more realistic architectures would probably involve inter-connected sub-domains of different size and/or topology. Because it has been shown that most networks belong to this category, we did "grow" a scale-free network of the appropriate size (3000 nodes, one link per node except the 1st one) to use as the basic blueprint. Obviously, the process of generating such blueprint would be different if a real system was being designed, because it would have to take into account many other parameters (node type and capability, geographical location, connection type...). However, this makes no difference for RAN, as long as the corresponding topology is translated into the appropriate file format.

Assuming that this topology is actually the blueprint for a real architecture, the network designer could use RAN to compute statistics about its resilience to node failure, in terms of the cohesion of its largest component (initially including all nodes). After the complete process is over (simulation + analysis took just under 1 min for this example), RAN displays the following results window:

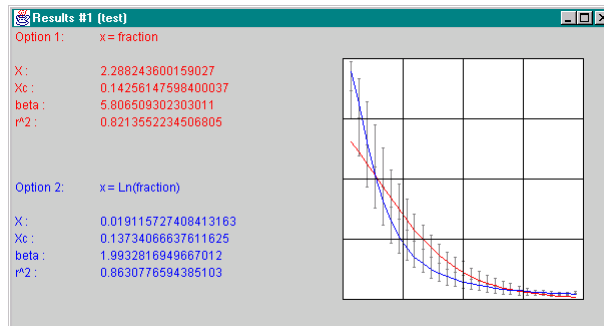


Figure 3: Analysis results for a model scale-free network (3000 vertices, 2999 edges).

Because this sort of network is basically a tree-like hierarchical structure with no built-in redundancy (1 link per node), it is not very robust to node failure. Indeed, RAN finds that, on average, removing only about 14% of all vertices (equivalent to severing all their links) is enough to reduce the size of the largest component to 50% of the surviving population ($X_c \sim 0.14$). So in effect, RAN tells the designer that if 500 nodes out of 3000 are malfunctioning, chances are the largest sub-set of relays that are still interconnected contains less than a half of the 2500 surviving nodes. In other words, it is likely that in this situation, around 1250 otherwise perfectly operational nodes are in fact cut from (and, obviously, unable to exchange any information with) the core of the network.

As can be expected, testing the same architecture for "attack" gives even more concerning results. In this scenario, killing only about 2% of the population (but this time selecting preferentially highly connected nodes) is enough to reach the same situation. So when applied to a typical scale-free architecture, RAN correctly and automatically predicts the type of network behaviour described by Albert et al. (2000), with the added advantage of summarising it by a set of global variables.

In the hypothesis that the designer wants to increase the robustness of the planned network, alternative blueprints could be produced, then analysed using RAN in order to compare their performance against that of the original, "cheapest", structure. For example, a straightforward way of increasing robustness is to add at least some backup links, so that alternative routes are available between nodes in case the primary (presumably most efficient) path becomes unavailable due to node failure(s). In our example, the designer could want to test the influence of doubling the total number of connections (raising it to 5999 links). As shown on Fig. 4a, this has a rather spectacular effect on robustness:

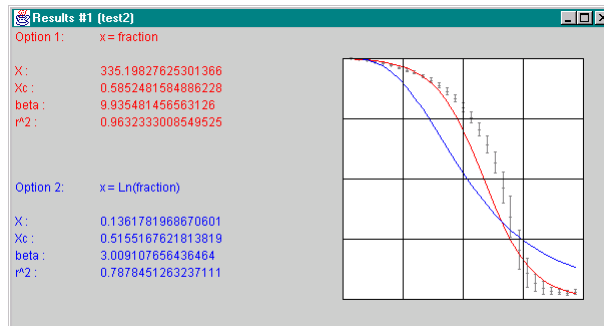


Figure 4a: Analysis results for a higher connection density (3000 vertices, 5999 edges).

With 3000 new connections added to the original blueprint, the network becomes much more resilient to node failure: it now takes about 60% nodes missing before more than a half of the surviving population is cut from the largest component. It is also clear that option 1 (expression [1a]) now gives a much better fitting than option 2 (expression [1b]), suggesting a "qualitative" change in network behaviour. Moreover, RAn provides additional information in the form of the evolution of the standard deviation around the average value. Indeed, until up to 50 percent nodes have failed, the relative size of the largest component appears extremely stable from one simulation to the other, unlike in the original architecture (meaning the reaction of the network to cumulative stress has become more predictable). Finally, the ability of the network to withstand directed attack is even more dramatically increased, as shown on Fig. 4b. Indeed, instead of requiring the removal of only 2% of the nodes, it is now necessary to kill up to 40% to break the largest component, even though the most highly connected vertices are still specifically targeted.

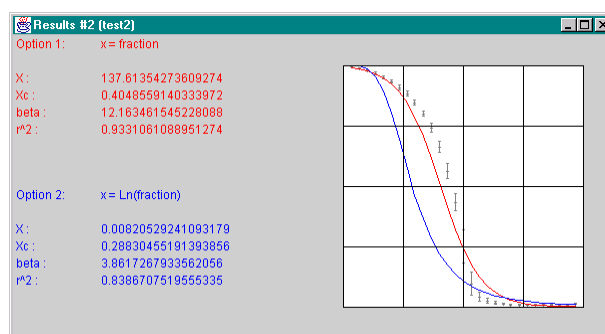


Figure 4b: same network as in Fig. 4a, this time submitted to directed attack.

However, simply doubling the number of links could be regarded as an unacceptable solution because of financial considerations. The network designer in

charge of the project could then want to look for alternative, cheaper ways of improving robustness, perhaps by testing the benefit of partial route redundancy. Again, RAN would allow him/her to make projections on the basis of yet another blueprint. For example, this 3rd option could involve only 1000 extra-connections compared to the original topology, bringing it to 3999 (see Fig. 5 for results).

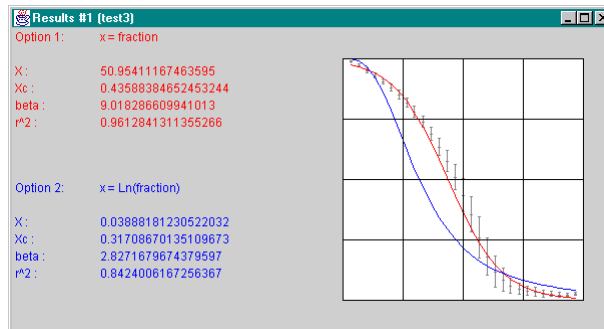


Figure 5: Analysis results for intermediate connection density (3000 vertices, 3999 edges).

Obviously, there is no free lunch: robustness is not increased in the same proportion as before. However, considering that only 33% extra links were created instead of 100%, this solution might in fact be a better one. Indeed, the critical size X_c is shifted to ~ 0.44 , meaning a factor 3 was gained compared to the original blueprint, against a factor 4 when *doubling* the number of connections, so the last choice is certainly more cost-effective!

Of course, those results by themselves are not very meaningful, since they relate to a constructed model, but they demonstrate that RAN is potentially a powerful tool for the network designer. Indeed, provided that the appropriate topology files are available, obtaining all this valuable and detailed information (including the value of β , which was not discussed in the example but gives a useful indication of how fast the network is likely to collapse when approaching critical size) is only a matter of minutes.

5. Conclusion

RAN is a combined simulation/analysis tool designed to study topological robustness. It is obviously not intended as a stand-alone application for network analysis, as it doesn't take into account other critical aspects of network operation like traffic or routing management for example. Its purpose is to provide a suitable way of estimating the speed and profile of the largest component's decay under cumulative node failure, a necessary first step in assessing a system's ability to withstand damage.

However, developing strategies to increase cohesion of the largest component (a task that can effectively be conducted using RAN to test alternative designs) is not sufficient to guarantee quality of service. This would also require being able to adapt

network operations to a changing topology (which is an entirely different problem). It cannot be denied though that maintaining all nodes within the largest component is a first and necessary condition in order to increase network robustness. In that respect, RAN could be a most valuable tool for the network designer, even though it would require being combined with others to generate an accurate and practical simulation of a realistic architecture.

References

- Albert R., H. Jeong, and A.-L. Barabasi (1999), "*Diameter of the World-Wide Web*", Nature **401**, pages 130-131.
- Albert R., H. Jeong, and A.-L. Barabasi (2000), "*Error and attack tolerance of complex networks*", Nature **406**, pages 376-382.
- Barabasi A.-L., R. Albert and H. Jeong (1999), "*Mean-field theory for scale-free random networks*". Physica A **272**, pages 173-187.
- Callaway D.S., M.E.J. Newman, S. H. Strogatz, and D.J. Watts, "*Network Robustness and Fragility: Percolation on Random Graphs*" (2000), Phys. Rev. Letters **85**, pages 5468-5471.
- Cohen R., K. Erez, D. ben-Avraham and S. Havlin (2000), "*Resilience of the Internet to random breakdowns*", Phys. Rev. Letters **85**, pages 4626-4628.
- Faloutsos M., P. Faloutsos, and C. Faloutsos (1999), "*On Power-Law Relationships of the Internet Topology*", ACM SIGCOMM '99, Comput. Commun. Rev. **29**, pages 251-263.
- Tadic B. (2001), "*Dynamics of directed graphs: the world-wide Web*", Physica A **293**, pages 273-284.
- Saffre F. and R. Ghanea-Hercock (2001), "*Robustness in Complex Network: a simplified model*", International Conference on Dynamical Networks in Complex Systems.