Preface

Traditionally, Reliability theory studies networks whose components subject to failure (edges and/or nodes) are binary, i.e., have two states—*up and down*. Typically, network *DOWN* state is defined as loss of terminal connectivity or a break-up into a critical number of components. There are two most often used tools for investigating network reliability—Monte Carlo simulation and the so-called D-spectra technique.

The "standard" model in network Monte Carlo [1] assumes that components fail independently, and node (or edge) i fails with probability q_i. The goal of Monte Carlo simulation is estimation of network static *DOWN* probability. In our opinion, the most efficient and accurate Monte Carlo method is the so-called evolution and merging algorithm originally suggested by M. V. Lomonosov [2, 3]. It has been shown [4] that this algorithm can be easily adapted to the case of non-reliable nodes. When all components are statistically independent and identical, i.e., have the same *down* probability q, reliability analysis can be considerably simplified by using the so-called D-spectra or signature technique. D-spectrum is a discrete distribution $\mathbf{f} = (f_1, f_2, \ldots, f_n)$, where f_i is the probability that system failure takes places at the instant of the i-th component failure. D-spectrum is system combinatorial invariant. It depends only on system structure function and has the following surprising property: the number $C(x)$ of system failure sets with exactly x components *down*, can be expressed via the D-spectrum by the following simple formula:

$$C(x) = (f_1 + \cdots + f_x) \cdot n!/(x!(n-x)!)$$

The cumulative D-spectrum $F(x) = f_1 + \cdots + f_x$ can be easily estimated by means of Monte Carlo simulation. To calculate $P(DOWN)$ we use the formula:

$$P(DOWN) = \sum_{j=1}^{n} C(j) q^j p^{n-j} \,. (*)$$

D-spectra technique works quite well for networks of small to medium size with 30–100 components. All D-spectra-based techniques can be easily extended to the binary networks having more than two states [5]. Moreover, a modification of D-spectra allows obtaining another system invariant, so-called Importance Spectrum, by means of which it becomes possible to calculate component Birnbaum Importance Measure (BIM), widely used in network design [2, 3, 6–8].

Summing up, network reliability theory successfully handles systems with two principal limitations:

(i) components fail independently;
(ii) components are binary.

We do not see an easy way to relax assumption (i), except for some *ad hoc* situations. Since violation of independence cannot be formalized into a simple "dependence" model, we cannot expect, in our opinion, rapid and decisive progress in this direction.

What about the second assumption? Almost every practical application of networks to real-life situations puts a question mark to the binary assumption regarding the component state. Here are some examples.

If the network describes the road system, link $e = (a, b)$ failure means traffic violation between nodes a and b. This violation almost never means complete disruption of transport flow. Natural disaster like flood or earthquake, may lead to only partial damage of the road segment. Therefore, between the perfect state of a link and its complete failure there should be at least one intermediate state.

Similar is the situation in flow networks. If an edge $e = (a, b)$ representing water supply pipe is in perfect state (*up*), it can deliver maximal amount of water, say of 1,000 cube/h. If it is broken (*down*) no water is delivered from a to b. But there are also situations when due to some technical reasons (e.g., leaks, maintenance works, partial damage) the water flow is reduced by 50 or 30 %. This means that an adequate description of the link asks for introducing one or several intermediate states between *up* and *down*. Communication networks consist of communication lines (channels) allowing different rates of information transmission depending on the technical state of the channel. In the absence of interferences (*up* state) the transmission speed is maximal. In case of broken channel (*down*) the speed is reduced to zero. It may happen that during peak hours, the transmission speed gets reduced by 30 %, which represents an intermediate state "between" *up* and *down*.

Social networks representing connections between individuals have also several degrees of "closeness" between two individuals who either maintain an extensive information exchange, or have no exchange at all, or have a reduced level of communication.

This book is devoted to the reliability analysis of ternary (or trinary) networks, i.e., to networks whose components subject to failure have *three* states: *up, down* and an intermediate state which we call *mid*. It turns out that the D-spectra technique can be extended to the case of components with three states. The price for this extension is introducing a more complicated version of D-spectrum, the *ternary D-spectrum.*

Ternary D-spectrum is a collection of so-called cumulative r-spectra $F_r(x)$, $r = 0, 1, \ldots, n-1; x = 1, 2, \ldots, n-r$. Here $F_r(x)$ is the probability that the network is *DOWN* if r of its components are *up*, x components are *down* and the remaining $(n - r - x)$ are in state *mid*. Formally, instead of a vector $\mathbf{f}$ in binary case, we have a set of vectors for ternary case. The approximate computation of the ternary spectrum is carried out by a quite straightforward Monte Carlo procedure.

Ternary spectrum is a network combinatorial invariant, and if it is known in addition to the fact that all components are statistically independent and identical, system *DOWN* probability can be computed by means of a simple formula similar to (*). The material of this book is organized as follows.

Section 1.1–1.4 of Chap. 1 are devoted to the definition and properties of the ternary D-spectrum. The knowledge of the ternary D-spectrum $\{F_r(x)\}$ allows counting the network failure sets with given structure. Let $C(r;x)$ be the number of failure sets having r components *up*, x components *down* and the remaining $n-r-x$ components in *mid* state. Then we prove that

$$C(r;x) = F_r(x) \cdot \frac{n!}{r!x!(n-r-x)!} \cdot \ (**)$$

If all network components are independent and identical, (**) allows to find network *DOWN* probability by means of the following formula:

$$P(DOWN) = \sum_{\{r,x:0 \le r+x \le n\}} C(r;x) p_2^r p_1^{(n-r-x)} p_0^x,$$

where p_2, p_1, p_0 are components *up, mid*, and *down* probability, respectively.

Section 1.5 of Chap. 1 is devoted to a modification of ternary D-spectrum called ternary importance spectrum which allows to evaluate network component importance measures. These measures are a modification of Birnbaum Importance Measures [3, 5–8] adjusted to ternary components.

Section 1.6 describes how to obtain an approximation to the ternary D-spectrum and to the component importance measures using Monte Carlo simulation techniques.

Chapter 2 consists of two parts. The first part (Sects. 2.1–2.4) is a numerical illustration of the theory developed in Chap. 1. In Sect. 2.1 we consider reliability calculations for an H_4 network. The network is a hypercube of order four, it has 16 nodes and 32 edges. Next, we define in it two sets of terminals, T_1 and T_2. An edge $e = (a, b)$ in state *up* provides high communication speed between a and b. If this edge is in state *mid*, the $a \leftrightarrow b$ communication goes with reduced speed; *down* state for an edge means that this edge does not exist. Edge state is chosen randomly and independently, according to probabilities p_2, p_1, p_0 for *up, mid*, and *down* state, respectively. System *UP* state is defined as the existence of high-speed communication between nodes of T_1 *and* the existence of a path of operational edges between any pair of nodes of T_2. We present data on network reliability and on the ternary D-spectrum.

Section 2.2 considers a stochastic source—terminal problem for a dodecahedron network. In this network, an edge $e = (a, b)$, except for edges going out of s and into t, is in fact a *pair* of directed links for $a \to b$ and $b \to a$ directed flows. Each link has capacity 6, 3, or 0 for *up*, *mid* and *down* state, respectively. The network has two *DOWN* states, *DOWN*2 and *DOWN*1, for the flow less than

L_1 or L_2, respectively ($L_2 < L_1$). We present data on network reliability for various values of edge probability vectors $p = (p_2, p_1, p_0)$.

Section 2.3 is an example of a rectangular grid network with 100 nodes and 180 edges. Components subject to failure are the nodes. If a node is *down* all edges adjacent to it are erased and the node gets isolated. If a node is in *mid* state, it has only horizontal or vertical edges, depending on the position of the node. For this network we calculate the probability that the largest connected node set (an analogue to a "giant" component) has less than L nodes.

The first part of Chap. 2 is concluded by Sect. 2.4 which presents edge importance data and their analysis for H_4 network. The second part (Sect. 2.5) deals with networks which have statistically independent and *nonidentical* components. Component i has state distribution $\mathrm{p}^{(i)} = (p_2^{(i)}, p_1^{(i)}, p_0^{(i)})$ meaning that the component is in state *up, mid*, and *down* with probability $p_2^{(i)}, p_1^{(i)}$ *and* $p_0^{(i)}$, respectively. In this situation, different failure sets with the same number of components in *up, mid, down* have different probabilistic weights, and this makes it not possible to use the ternary spectrum technique for finding system *DOWN* probability. What remains in this more complex case is to resort to a fast and accurate Monte Carlo method. Such a method is based on a modification of M. V. Lomonosov's evolution algorithm [2, 3]. The algorithm is described in Sects. 2.5.1 and 2.5.2. Its action is illustrated by numerical examples of flow and grid networks.

In reality, networks usually interact with each other and failure in one network causes failure in another one. For example, functioning of a city road network strongly depends on the traffic light power supply system: several non-functioning traffic lights ("nodes") may cause traffic jams in large areas. Another example is power supply network and communication network which strongly interact with each other.

The simplest form of two interacting networks is sharing the same set of nodes by two independent networks. For example, the power supply and water supply networks in the same geographic area share the same set of nodes (houses or residencies). Section 3.1 presents several simple results concerning the size of the set of nodes which receive "full" supply, i.e., are adjacent to edges of both types. Here we use some basic facts from the theory of large random Poisson networks [9].

Section 3.2 considers a system of two or more finite interacting networks. Here the interaction means that a node v_a of network A delivers "infection" to a randomly chosen node v_b in B which in turn, bounces back and infects another randomly chosen node w_a in network A, and so on. As a result, a random number Y of nodes in B gets "infected" and fails. We compute, using D-spectra technique, the *DOWN* probability for network B. This model is generalized to the case of several peripheral networks attacking one "central" binary network B. In this "attack," some nodes in B will receive more than one hit. The use of DeMoivre combinatorial formula combined with the D-spectra technique allows us to obtain in a closed form an expression for network B *DOWN* probability.

Finally, Sect. 3.3 extends the results of Sect. 3.2 to the case when the "central" network is ternary. In that case, we must take into account that nodes which were hit once or more will be in different states. It is assumed that a node hit only once changes its state from *up* to *mid*. When this node receives another hit, it turns into *down* and remains in it forever. Network *DOWN* probability for this case can be estimated by a Monte Carlo algorithm.

George Box used to cite the aphorism: "all models are wrong; some models are useful." We hope that some models presented in this book might be useful to reliability researchers involved in network study and design, and to reliability engineers interested in applications of the theory to practical calculations of network reliability parameters.

Acknowledgments

My work was partially supported by the Australian Research Council under grant number CE140100049.

March 2014

Ilya Gertsbakh
Yoseph Shpungin
Radislav Vaisman

References

1. Kroese D, Taimre T, Botev ZI (2011) Handbook of Monte Carlo methods, Chap. 16. Wiley, New York, pp. 549–576
2. Elperin T, Gertsbakh IB, Lomonosov M (1991) Estimation of network reliability using graph evolution models. IEEE Trans Reliab 40(5):572–581
3. Gertsbakh I, Shpungin Y (2009) Models of network reliability: analysis, combinatorics and Monte Carlo. CRC Press, Boca Raton
4. Gertsbakh I, Shpungin Y, Vaisman R (2014) Network reliability Monte Carlo with nodes subject to failure. Int J Perform Eng 10(2) 161–170
5. Gertsbakh I, Shpungin Y (2011) Network reliability and resilience, Springer Briefs in Electrical and Computer Engineering, Springer. Berlin, Heidelberg
6. Barlow RE, Frank Proschan (1975) Statistical theory of reliability and life testing. Holt, Rinehart and Winston, Inc., New York
7. Birnbaum ZW (1969) On the importance of different components in multicomponent system. In: Krishnaiah PR (ed) Multivarite analysis-II. Academic Press, New York, pp 581–592
8. Gertsbakh I, Shpungin Y (2012) Combinatorial approach to computing importance indices of coherent systems. Probab Eng Inf Sci 26:117–128
9. Newman MEJ (2010) Networks. An introduction. Oxford University Press, Oxford

Springer

http://www.springer.com/978-3-319-06439-0

Ternary Networks
Reliability and Monte Carlo
Gertsbakh, I.; Shpungin, Y.; Vaisman, R.
2014, XIV, 62 p. 18 illus., 7 illus. in color., Softcover
ISBN: 978-3-319-06439-0