I. Gertsbakh

Reliability Theory

With Applications to Preventive Maintenance

Springer

Ilya Gertsbakh

Reliability Theory

With Applications to Preventive Maintenance

With 51 Figures

Springer

Professor Ilya Gertsbakh
Ben Gurion University
Department of Mathematics
P.O. Box 653
84105 Beersheva
Israel

CIP-Data applied for
Die Deutsche Bibliothek - CIP-Einheitsaufnahme
Gercbach,Il'ja B.:
Reliability theory: with applications to preventive maintenance / Ilya Gertsbakh
Berlin; Heidelberg; New York; Barcelona; Hong Kong; London; Milan; Paris; Singapore; Tokyo: Springer, 2000
(Engineering online library)
ISBN 3-540-67275-3

ISBN 3540-67275-3 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

Printed in Germany

Typesetting: Camera-ready copy from author
Cover-Design: MEDIO Innovative Medien Service GmbH, Berlin
Printed on acid-free paper SPIN 10763074 62/3020 5 4 3 2 1 0

Ilya Gertsbakh

Reliability Theory

To my teacher Khaim Kordonsky

Preface

The material in this book was first presented as a one-semester course in Reliability Theory and Preventive Maintenance for M.Sc. students of the Industrial Engineering Department of Ben Gurion University in the 199 7/98 and 1998/99 academic years.

Engineering students are mainly interested in the applied part of this theory. The value of preventive maintenance theory lies in the possibility of its implementation, which crucially depends on how we handle statistical reliability data. The very nature of the object of reliability theory – system lifetime – makes it extremely difficult to collect large amounts of data. The data available are usually incomplete, e.g. heavily censored. Thus, the desire to make the course material more applicable led me to include in the course topics such as modeling system lifetime distributions (Chaps. 1, 2) and the maximum likelihood techniques for lifetime data processing (Chap. 3).

A course in the theory of statistics is a prerequisite for these lectures. Standard courses usually pay very little attention to the techniques needed for our purpose. A short summary of them is given in Chap. 3, including widely used probability plotting.

Chapter 4 describes the most useful and popular models of preventive maintenance and replacement. Some practical aspects of applying these models are addressed, such as treating uncertainty in the data, the role of data contamination and the opportunistic scheduling of maintenance activities.

Chapter 5 presents the maintenance models which are based on monitoring a "prognostic" parameter. Formal treatment of these models requires using some basic facts from Markov-type processes with rewards (costs). In recent years, there has been a growing interest in maintenance models based on monitoring the process of damage accumulation. A good example is the literature dealing with the preventive maintenance of such "nontypical" objects as bridges, concrete structures, pipelines, dams, etc. The chapter concludes by considering a general methodology for planning preventive maintenance when a system has a multidimensional state parameter. The main idea is to make the maintenance decisions depending on the value of a one-dimensional system "health index."

The material of Chap. 6 is new for a traditional course. It is based on the recent works of Kh. Kordonsky and considers the choice of the best *time scale* for age replacement. It would not be an exaggeration to say that the correct choice of the time scale is a central issue in any sphere of reliability applications.

Chapter 7 shows an example of *learning* in the process of servicing a system. Several strong assumptions were made to make the mathematics as simple as possible. It is important to demonstrate to students that the combination of prior knowledge with new data received in the process of decision making is, in fact, a universal phenomenon, which may have various useful applications.

It takes me, on the average, two weeks in the classroom (3 hours weekly) to deliver the material of one chapter. In addition, I spend some time explaining the most useful procedures of *Mathematica* needed for the numerical analysis of the theoretical models and for solving the exercises. Getting to the "real" numbers and graphs always gives students a good feeling and develops better intuition and understanding, especially if the material is saturated with statistical notions. The course concludes with detailed solutions of the exercises, including a numerical investigation by means of *Mathematica.*

Ilya Gertsbakh
Beersheva, January 2000

Contents

Chapter 1

System Reliability as a Function of Component Reliability

The whole is simpler than the sum of its parts
Willard Gibbs

1.1 The System and Its Components

In reliability theory, as in any theory, we think and operate in terms of *models.* In this chapter we investigate a model of a *system,* which consists of *elements* or *components.* Our purpose is to develop a formal instrument to enable us to receive information about a system's reliability from information about the reliability of its components. The exposition in this section does not involve probabilistic notions.

A system is a set of components (elements). Only *binary* components will be considered, i.e. components having only two states: operational (up) and failed (down). The state of component i, $i = 1, \ldots, n$, will be described by a binary variable x_i: $x_i = 1$ if the component is up; $x_i = 0$ if the component is down.

It will be assumed that the whole system can only be in one of two states: up or down. The dependence of a system's state on the state of its components will be determined by means of the so-called *structure* function $\phi(\mathbf{x})$, where $\mathbf{x} = (x_1, x_2, \ldots, x_n)$: $\phi(\mathbf{x}) = 1$ if the system is up; $\phi(\mathbf{x}) = 0$ if the system is down.

We also use the notation $\mathbf{x} < \mathbf{y}$. This means that the components of $\mathbf{x}$ are less then or equal to the components of $\mathbf{y}$, i.e. $x_i \leq y_i$, but for at least one component j, $x_j < y_j$.

Example 1.1.1: Series system (Fig. 1.1a)
This system is up if and only if all its components are up. Formally,

$$\phi(\mathbf{x}) = \prod_{i=1}^{n} x_i = \min_{1\leq i\leq n} x_i. \quad (1.1.1)$$

Example 1.1.2: Parallel system (Fig. 1.1b)
The system is up if and only if at least one of its components is up. Formally,

$$\phi(\mathbf{x}) = 1 - \prod_{i=1}^{n}(1 - x_i) = \max_{1\leq i\leq n} x_i. \# \quad (1.1.2)$$

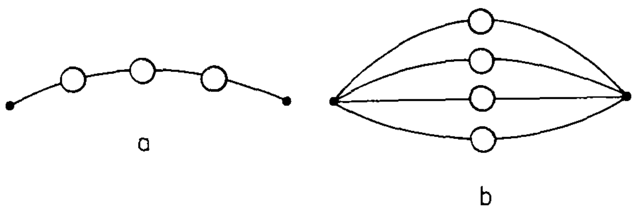

Figure 1.1. Representation of series (a) and parallel system (b)

Example 1.1.3: k-out-of-n system
This system is up if and only if at least k out of its n components are operating. Formally,

$$\phi(\mathbf{x}) = 1, \text{ if } \sum_{i=1}^{k} x_i \geq k, \quad (1.1.3)$$

and $\phi(\mathbf{x}) = 0$ otherwise
Example 1.1.4: Cable TV transmitter (Fig. 1.2)
The system is designed to transmit from the central station S to three local stations S_1, S_2, S_3. The stations are connected by cables numbered $1, 2, 3, 4, 5$, which are the system components. The system is operational (up) if all substations are connected directly or through another substation to the central station.

One can check that

$$\begin{aligned}\phi(\mathbf{x}) &= 1 - (1 - x_2x_3x_5)(1 - x_2x_4x_5) \\ &\times (1 - x_2x_3x_4)(1 - x_1x_3x_4)(1 - x_1x_3x_5)(1 - x_1x_2x_5)(1 - x_1x_2x_4).\end{aligned}$$

We explain later how to derive this formula.

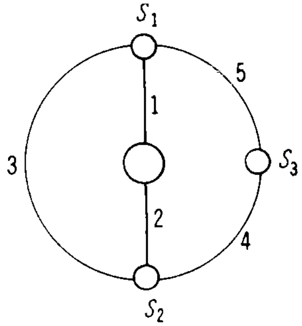

Figure 1.2. Cable TV transmission system

Example 1.5: Series connection of parallel systems (Fig. 1.3)
For this system, $\phi(\mathbf{x}) = [1 - (1 - x_1)(1 - x_2][1 - (1 - x_3)(1 - x_4)]$.

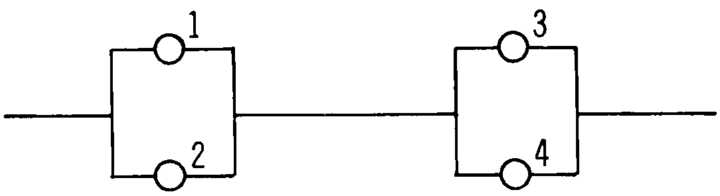

Fig. 1.3. Series connection of parallel systems

Example 1.1.6: Parallel connection of series systems (Fig. 1.4)
Check that for this system $\phi(\mathbf{x}) = 1 - (1 - x_1x_2)(1 - x_3x_4x_5)$.

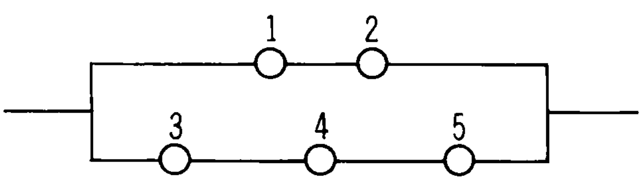

Figure 1.4. Parallel connection of series systems

It is important to have a systematic way of constructing a formula for the structure function $\phi(\cdot)$. This will be done by using the notions of ***minimal paths*** and *minimal cuts*. Before doing this let us impose some natural demands on $\phi(\mathbf{x})$.

Definition 1.1.1: *Monotone system*
A system with structure function $\phi(\cdot)$ is called *monotone* if it has the following properties:
(i) $\phi(0,0,\ldots,0) = 0, \quad \phi(1,1,\ldots,1) = 1$;
(ii) $\mathbf{x} < \mathbf{y} \Rightarrow \phi(\mathbf{x}) \leq \phi(\mathbf{y})$.
In words: the system is down if all its elements are down; it is up if all its elements are up; and the state of the system cannot become worse if any of its elements changes its state from down to up.

Definition 1.1.2: *Cut vector, cut set, path vector, path set*
A state vector $\mathbf{x}$ is called a *cut vector* if $\phi(\mathbf{x}) = 0$. The set $C(\mathbf{x}) = \{i : x_i = 0\}$ is then called a *cut set*. If, in addition, for any $\mathbf{y} > \mathbf{x}$, $\phi(\mathbf{y}) = \mathbf{1}$, then the corresponding cut set is called *minimal cut set* or simply minimal cut.

A state vector $\mathbf{x}$ is then called a *path vector* if $\phi(\mathbf{x}) = 1$. The set $A(\mathbf{x}) = \{i : x_i = 1\}$ is then called a *path set*. If, in addition, for any $\mathbf{y}$, $\mathbf{y} < \mathbf{x}$, $\phi(\mathbf{y}) = 0$, then the corresponding path set is called *minimal path set*, or minimal path.

A minimal cut set is a minimal set of components whose failure causes the failure of the whole system.

If all elements of the path set are "up" then the system is up. A minimal path is a minimal set of elements whose functioning (i.e. being up) ensures that the system is up. The minimal path set cannot be reduced, as it has no redundant elements.

Examples 1.1.5, 1.1.6 continued
For Example 1.1.5, $\mathbf{x}_1 = (1,1,1,0)$ is a path vector. The corresponding path set is $\{1,2,3\}$. It is *not*, however, a minimal path set because if element 2 is turned down the system will still be up. $\{1,3\}$ is the minimal path set. There are three other minimal path sets. Find them!

For Example 1.1.6, there are two minimal path sets, $\{1,2\}$ and $\{3,4,5\}$.

The system in Fig. 1.3 has two minimal cuts: $\{1,2\}$ and $\{3,4\}$.

The set $\{1,2,3\}$ is also a cut set but not a minimal one. The system in Fig. 1.4 has six minimal cuts of the form $\{i,j\}$, where $i = 1,2$ and $j = 3,4,5$.

Theorem 1.1.1: *Structure function representation*
Let $P_1, P_2, \ldots, P_s$ be the minimal path sets of the system. Then

$$\phi(\mathbf{x}) = 1 - \prod_{j=1}^{s} \left(1 - \prod_{i \in P_j} x_i\right) . \tag{1.1.4}$$

Let $C_1, C_2, \ldots, C_k$ be the minimal cut sets of the system. Then

$$\phi(\mathbf{x}) = \prod_{j=1}^{k} \left(1 - \prod_{i \in C_j} (1 - x_i)\right) . \tag{1.1.5}$$

Proof
Assume that there is at least one minimal path set, all elements of which are up, say P_1. Then $\prod_{i \in P_1} x_i = 1$ and this leads to $\phi(\mathbf{x}) = 1$. Suppose now that the system is up. Then there must be one minimal path set having all of its elements in the up state. Thus the right-hand side of (1.1.4) is 1. Therefore, $\phi(\mathbf{x}) = 1$ if and only if there is one minimal path set having all its elements in the up state. This proves (1.1.4).

We omit the proof of (1.1.5), which is similar.

It follows from Theorem 1.1.1 that any monotone system can be represented in two equivalent ways: as a series connection of parallel subsystems each being a minimal cut set, or as a parallel connection of series subsystems each being a minimal path set. Therefore, there are two ways to represent structure functions. After corresponding simplifications, these become identical, as the following example shows.

Example 1.1.5 continued
The structure function given above for the system in Fig. 1.3 is based on minimal cuts $\{1,2\}$ and $\{3,4\}$. The system also has four minimal paths: $\{1,3\}, \{1,4\}$, $\{2,3\}, \{2,4\}$. Thus, $\phi(\mathbf{x}) = 1 - (1 - x_1x_3)(1 - x_1x_4)(1 - x_2x_3)(1 - x_2x_4)$.

The structure function based on minimal cuts was presented in Example 1.1.5. Both formulas produce identical results. To verify this, it is necessary to simplify both expressions. Note that for binary variables, $x_i^k = x_i$ for any integer k.

More information on monotone systems and their structure function can be found in the literature, e.g. in Barlow and Proschan (1975), Chap. 1, and Gertsbakh (1989), Chap. 1.

1.2 Independent Components: System Reliability and Stationary Availability

Contrary to Sect. 1.1, let us now assume that the state of component i is described by a binary *random* variable X_i, defined by

$$P(X_i = 1) = p_i, \quad P(X_i = 0) = 1 - p_i, \tag{1.2.1}$$

where 1 and 0 correspond to the operational (up) and failure (down) state, respectively.

It will be assumed that all components are mutually independent. This implies a considerable formal simplification: for independent components, the joint distribution of $X_1, X_2, \ldots, X_n$ is completely determined by component reliabilities $p_1, p_2, \ldots, p_n$.

Denote by $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ the system state vector. This is now a random vector. Correspondingly, the system structure function $\phi(\mathbf{X}) = \phi(X_1, \ldots, X_n)$ becomes a binary random variable: $\phi(\mathbf{X}) = 1$ corresponds to the system up state and $\phi(\mathbf{X}) = 0$ corresponds to the system down state.

Definition 1.2.1: *System reliability*
System reliability r_0 is the probability that the system structure function equals 1:

$$r_0 = P(\phi(\mathbf{X}) = 1)\,. \tag{1.2.2}$$

Since $\phi(\cdot)$ is a binary random variable, the last formula can be written as

$$r_0 = E[\phi(\mathbf{X})]\,. \tag{1.2.3}$$

Expression (1.2.3) is very useful since the operation of taking expectation $E[\cdot]$ is a very powerful tool for reliability calculations. The following examples show how to compute system reliability via its structure function.

Example 1.2.1: Reliability of a series system
$\phi(\mathbf{X}) = \prod_{i=1}^{n} X_i$, and therefore

$$r_0 = E[\phi(\mathbf{X})] = \prod_{i=1}^{n} p_i. \tag{1.2.4}$$

Example 1.2.2: Parallel system
Here $\phi(\mathbf{X}) = 1 - \prod_{i=1}^{n}(1 - X_i)$. Thus

$$r_0 = E[\phi(\mathbf{X})] = 1 - \prod_{i=1}^{n}(1 - p_i)\,. \tag{1.2.5}$$

Example 1.2.3: Series connection of parallel systems (Example 1.1.5)
From the expression for $\phi(\mathbf{X})$ it follows immediately that

$$r_0 = E[\phi(\mathbf{X})] = [1 - (1 - p_1)(1 - p_2)][1 - (1 - p_3)(1 - p_4)]\,. \tag{1.2.6}$$

Example 1.2.4: 2-out-of-4 system with identical elements
For this system,

$$r_0 = E[\phi(\mathbf{X})] = P\left(\sum_{i=1}^{4} X_i \geq 2\right) = \sum_{m=2}^{4} \binom{n}{m} p^m (1 - p)^{4-m}\,. \tag{1.2.7}$$