ORiON, Vol. 13, No. 1/2, pp.19-36 ISSN 0259-191-X

SOME CURRENT PROBLEMS OF RELIABILITY THEORY*

FRANK BEICHELT

University of the Witwatersrand
Department of Statistics and Actuarial Science
Johannesburg, South Africa

ABSTRACT

Reliability work occupies an increasingly important place in engineering practice. Although the details differ depending on whether mechanical, electrical, chemical, or other systems are under analysis, the reliability concepts and the mathematical foundations cut across the specific fields of application. Over the past 50 years thousands of papers and dozens of books on mathematical models of reliability have been published. A comprehensive survey alone on the current developments in the mathematical theory of reliability would fill a voluminous book. Based on importance both for theory and application and taking into account the interests of the author, current investigations in four important branches of reliability theory are considered: coherent systems, stochastic networks, software reliability, and maintenance theory.

1. INTRODUCTION

No other branch of engineering science, with exception of computer- and environmental technology, has developed and advanced as substantially during the past 40 years as reliability engineering. This is mainly due to using high risk systems as nuclear power stations, to man's step into the space, and to the development of highly sensitive weapon systems. But even less spectacular modern systems of industrial production and of transportation and communication technology are usually so complex that their reliability prediction and preservation is not possible without using scientific methods. Hence it is not surprising that the rapid development of reliability engineering has given a

*) Invited plenary lecture at the Annual Conference of the German Statistical Society and the Stochastic Society within the German Mathematical Association, Wittenberg (Germany), May 1997.

decisive impetus to the phenomenal development of the mathematical theory of reliability during the past four decades.

According to the IEC document [2], reliability is defined as "the capability of a product, a system, or a service to perform its expected job under specified conditions of use over an intended period of time". For the sake of simplicity, this paper refers only to the reliability of systems or their components (subsystems). In reliability theory, the intended time period is called *mission time.* If the system is no longer capable of performing its job during its mission time, a *system failure* has occured. The concept of a system failure in this rather general sense is basic for reliability investigations. In particular, the IEC-reliability definition allows to derive special, mathematically well-defined criteria, called *reliability criteria.* The most important ones are:

1) *Survival probability*: This is the probability that there is no failure within a specified time interval [0, t].

2) *Availability*: The *point availability* is the probability A(t) that at a given time point t the system is capable of performing its job. The *stationary availability* is defined by

$$A = \lim_{t \to \infty} A(t).$$

Thus, A is the fraction of time with respect to an infinite mission time in which the system is capable of doing its job.

Reliability theory deals with the measurement, prediction, preservation and optimization of the reliability of technical systems.

Hereby *reliability* refers to one or more reliability criteria which are most appropriate and important for the mission of the system under consideration. The main problems the mathematical theory of reliability deals with are:

1) Investigation of the mutual relationship between the reliability criteria of a system and its subsystems (components).

2) Modeling the failure and aging behaviour of systems (components).

3) Statistical estimation of reliability criteria.

4) Development, investigation and optimization of measures for the preservation and restoration, respectively, of a fixed reliability level (maintenance theory).

Recent developments deal with software reliability and reliability of man-machine-systems. This contribution cannot give a full survey on all new trends and develop-

ments in the mathematical theory of reliability or cite all relevant literature. (Even the excellent volume MISRA [1] which comprises more than 700 pages is not complete.) Taking into account the interests of the author, here a selection is presented based on its importance both for theory and application.

2. BINARY COHERENT SYSTEMS

Let us consider a system **S** consisting of the components $e_1, e_2, ..., e_n$. The two possible states of the system and its components are *available* and *not available*. Let us further introduce the (0,1)-indicator variables for the states of the system and its components z_S and $z_1, z_2, ..., z_n$ in the following way:

$$z_S = \begin{cases} 1 \text{ if } \mathbf{S} \text{ is available} \\ 0 \quad \text{otherwise} \end{cases};$$

$$z_i = \begin{cases} 1 \text{ if } e_i \text{ is available} \\ 0 \quad \text{otherwise} \end{cases}.$$

The dependence of z_S on the z_i is given by the *structure function* ϕ of the system:

$$z_S = \phi(z_1, z_2, ..., z_n).$$

The system is called *coherent* if its structure function has the following properties:

1) $\phi(0,0,...,0) = 0$.

$\phi(1,1,...,1) = 1$.

2) ϕ is nondecreasing in each z_i.

3) For each $i = 1, 2, ..., n$ there is a vector $\mathbf{z} = (z_1, z_2, ..., z_n)$ with the properties

$$\phi(z_1, ..., z_{i-1}, 0, z_{i+1}, ..., z_n) = 0.$$

$$\phi(z_1, ..., z_{i-1}, 1, z_{i+1}, ..., z_n) = 1.$$

If $\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_w$ are the path sets of the system and

$$A_k = \prod_{i \in \mathbf{W}_k} z_i; \quad k = 1,2,...,w,$$

then the structure function can be written in the form

$$z_S = A_1 \vee A_2 \vee ... \vee A_w, \tag{1}$$

where for any (0,1)-variables x and y

$$x \vee y = x + y - xy.$$

In what follows, the z_S and z_i are assumed to be random variables. They refer to a fixed time point or to the stationary regime. Hence the availability of the system is

$$p_S = E(z_S) = E(\phi(z_1, z_2, ..., z_n)).$$

From the theoretical point of view, binary coherent systems are very well investigated. But the computational problems connected with calculating their availability belong to the class of NP-hard problems. This implies in particular that with increasing complexity of ϕ the computation time to obtain p_s increases exponentially fast. Hence it has up to now been a challenge for reliability theoreticians to find computer algorithms which allow the exact computation of the availability of complex coherent systems in a reasonable time span.

The starting point for the approach considered here is formula (1). The basic aim consists in deducing from (1) a representation of the structure function in the form

$$\phi(z_1,z_2,...,z_n) = \sum_{j=1}^{d} D_j, \quad D_i D_j = 0 \quad \text{for} \quad i \neq j; \quad i, j = 1,2,....d, \tag{2}$$

where the D_j are products of some z_i and $1-z_k$. (2) is called an *orthogonal sum representation* of the structure function. The advantage of an orthogonal sum representation is evident: the system availability can be obtained from

$$p_s = \sum_{j=1}^{d} E(D_j).$$

Assuming independence of the $z_1, z_2, ..., z_n$, the mathematical expectations $E(D_j)$ are simply the products of the corresponding expectations of the z_i and $1-z_k$. (These are the known availabilities and nonavailabilities, respectively, of the corresponding components e_i and e_k.)

Up till now there has been a competition amongst reliability theoreticians to construct orthogonal sum representations of ϕ with comparatively low complexity, in particular with a small number of terms d. The first were ABRAHAM [3], and LOCKS [4], among others. The paper of ABRAHAM especially has initiated further research: BEICHELT/SPROSS [5], HEIDTMANN [6], VEERARAGHAVAN and TRIVEDI [7]. Important work has been done by the research group of J. KOHLAS in generalizing the approaches of ABRAHAM and HEIDTMANN to noncoherent binary and multivalued systems: BERTSCHY and MONNEY [8], KOHLAS [9], ANRIG and LEHMANN [10]. It is interesting that their research was initiated by solving problems arising from the theory of artificial intelligence. Recent contributions to multivalued coherent systems gave BAXTER [11, 12], MAZARS [13], MENG [14] as well as ABOUAMMOH and AL-KADI [15].

3. STOCHASTIC NETWORKS

Network reliability analysis is required in many important engineering areas. Examples are communication networks, monitoring systems, transportation and electrical power distribution systems. In particular, computer communication networks have evolved to cope with the massive demand for transmission and processing of information. Hence it has been imperative that effective tools be developed and refined for carrying out reliability analyse of complex networks. To simplify terminology, this section refers to communication networks.

The basic topological structure of the network under consideration is assumed to be given by a connected graph without loops $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where $\mathbf{V}$ is its node set and $\mathbf{E}$ its edge set. The nodes are interpreted as end users and the edges as links between them. $\mathbf{G}$ becomes a *random graph* or a *stochastic network* $\tilde{\mathbf{G}}$ by assuming that its edges and nodes exist (are available) or do not exist (are not available) according to a given probability distribution. Hence the state variables of the nodes and the edges are binary random variables. For convenience, in what follows it is assumed that the nodes are always available. The most important reliability criteria for stochastic networks are:

1) *Connectednes probability (overall reliability)*: this is the probability that even if some edges are not available the remaining graph is connected.

2) *Two-terminal reliability*: this is the probability that - given two fixed nodes - there exists a path between these nodes consisting only of available edges.

With respect to these two reliability criteria (and other more sophisticated ones), a stochastic network is a binary coherent system as introduced in section 2. Hence the connectedness probability as well as the two-terminal reliability can in principle be computed via the *orthogonal sum representation* of the corresponding structure function. But stochastic networks are examples of coherent systems which allow the construction of more efficient methods. Methods of this kind are above all *decomposition- and reduction approaches.* All known algorithms for computing reliability criteria of planar networks whose computation time increases only polynomially with increasing network complexity, are based on such methods. Algorithms having this property are called *polynomial algorithms.* (However, reliability analysis of general networks remains principally an NP-hard problem.)

Decomposition Let $G^1 = (V^1, E^1)$ and $G^2 = (V^2, E^2)$ be two edge-disjoint subgraphs of **G** with the property

$$G = G^1 \cup G^2, \qquad G^1 \cap G^2 = (U, \varnothing). \tag{3}$$

A node set **U** with this property is called *separating node set* of **G**. An obvious approach is to carry out reliability analysis for $\tilde{G}^1$ and $\tilde{G}^2$ separately and to combine the results to obtain the desired reliability criterion of $\tilde{G}$. This approach was first applied in a heuristic way by ROSENTHAL [16] and ROSENTHAL and FRISQUE [17]. The first mathematically exact treatment was given by BEICHELT and TITTMANN [18]. Their approach is based on methods of higher combinatorics, in particular on properties of the set Π of all partitions of the separating node set:

$$\Pi = \{\pi_1, \pi_2, ..., \pi_b\}.$$

(A partition of **U** is a set of disjoint subsets, called *blocks*, of **U** whose union is **U**.) A partition π_j is said to be a *refinement* of π_i iff each block of π_j can be obtained by splitting a block of π_i. In Π a partial ordering "$\leq$" is introduced in the following way: $\pi_i \leq \pi_j$ iff π_j is a refinement of π_i. Thus, the partition lattice $(\Pi, \leq)$ is given. Let π_1 be its smallest element. Then π_1 consists only of a single block comprising all elements of **U**. Further, let U_j (G_j^i) be that node set (graph) arising from **U** (G^i) by fusing the nodes of each block of π_j into one node. Let the matrix $A = ((a_{jk}))$ be given by

$$a_{jk} = \begin{cases} 1, & \text{if } \pi_j \wedge \pi_k = \pi_1 \\ 0, & \text{else} \end{cases}.$$

Here $\pi_j \wedge \pi_k$ denotes the greatest partition satisfying both $\pi_j \wedge \pi_k \leq \pi_j$ and $\pi_j \wedge \pi_k \leq \pi_k$. Let R_j^i be the connectedness probability of $\tilde{G}_j^i$ and

$$R^i = (R_1^i, R_2^i, ..., R_b^i); \quad i = 1, 2.$$

Then the the following *decomposition formula* holds for the connectedness probability $R(\tilde{G})$ of $\tilde{G}$:

$$R(\tilde{G}) = (R^1)^T A^{-1} R^2.$$

Note that R^1 and R^2 depend only on the stochastic subnetworks $\tilde{G}^1$ and $\tilde{G}^2$. The "coupling" matrix **A** is independent of these subnetworks. It depends only on the cardinality of the separating node set **U**. Similar formulas have been obtained for other reliability criteria. Moreover, these decomposition formulas allow the

construction of approximate algorithms for network reliability criteria which run in polynomial time, see BLECHSCHMIDT and TITTMANN [19].

Reduction Network reduction is a powerful tool for reliability analysis of complex stochastic networks. It is characterized by simplifying the topological structure of the underlying network **G** by substituting a subgraph of **G** with a *replacement graph* to obtain a network **H** and the corresponding stochastic network $\tilde{\mathbf{H}}$ with known functional relationship between the relevant reliability criteria of $\tilde{\mathbf{G}}$ and $\tilde{\mathbf{H}}$. The most popular method is to replace a "triangle" in a network by a "star" or vice versa ("triangle-star-reduction"). The mathematically exact treatment of reduction methods became possible with the formalism of network decomposition as outlined above. The *reliability preserving network reduction* involves 3 steps:

1) Decomposition of **G** according to (3).

2) Generation of a graph **H** by replacing $\mathbf{G}^2$ in **G** with a replacement graph $\mathbf{H}^2$ satisfying

$$\mathbf{G}^1 \cap \mathbf{H}^2 = (\mathbf{U}, \varnothing).$$

3) The stochastic network $\tilde{\mathbf{H}}$ belonging to **H** has to be constructed in such a way that

$$R(\tilde{\mathbf{G}}) = h\,R(\tilde{\mathbf{H}}). \tag{4}$$

h is called *reduction constant.* Its introduction is only of computational importance. From (4) follows that the reliability criteria $R(\tilde{\mathbf{H}}_j^2)$ of the stochastic networks $\tilde{\mathbf{H}}_j^2$, which are constructed analogously to the $\tilde{\mathbf{G}}_j^2$, have to satisfy the following system of (generally nonlinear) equations

$$R_j^2 = h\,R(\tilde{\mathbf{H}}_j^2); \quad j = 1, 2, \ldots, b. \tag{5}$$

Here it is assumed that the R_j^2 are known. They have to be computed by any available method. The criteria $R(\tilde{\mathbf{H}}_j^2)$ in (5) are expressed in terms of the unknown availabilities of the edges of the stochastic replacement structure $\tilde{\mathbf{H}}_j^2$. The equation system (5) has to be solved for these availabilities. Network decomposition and network reduction are particularly efficient if they are applied successively.

A comprehensive survey on reduction methods published up to 1990 was given by MISRA [20]. In 1987, KOHLAS [21] presented an excellent treatment of special reductions, the so-called *polynome-to-chain-reductions.* The theoretical foundation of decomposition-reduction methods outlined here have been extended, summarized

and illustrated by many examples in BEICHELT [22, 23]. However, important problems have still to be investigated in more detail, e.g.

1) Computer-aided construction of suitable replacement graphs.

2) Solvability of reduction equations.

3) Optimal choice of separating node sets.

4) Developing software.

For many years networks were investigated only under the flow aspect (flow networks). The obvious and important step of considering networks both under the flow and the reliability aspect has been done by RÜGER [24], LIN ET AL. [25], and JENTSCH [26].

4. SOFTWARE RELIABILITY

Nowadays computers are used in many vital areas where failures could imply catastrophic or at least costly consequences, e.g. in process control systems of space flights, nuclear power plants, air traffic systems, or ballistic missile defence systems. In view the recent progress in hardware technology, the main cause for computer system failures are due to software deficiencies. Hence in the past 20 years there has been a considerable effort in improving what has been called *software reliability.* The now generally accepted definition of software reliability was given by MUSA and OKUMOTO [27]: *Software reliability is the probability of failure-free operation of a computer program in a specified enviroment for a specified period of time.* (The analogy to the definition of hardware reliability is obvious.) The problems for engineers and statisticians resulting from the investigation of software reliability are (see BARLOW and SINGPURWALLA [28]):

1) Quantification and measurement of software reliability.

2) Assessment of the changes in software reliability over time.

3) Analysis of software failure data.

4) Decision of whether to continue or stop testing a software.

Problems of software reliability differ from hardware reliability problems mainly for four reasons: 1) The cause of software failures is human error. 2) Once all failures are removed, the software is absolutely reliable and will continue to be so. 3) There is no process which generates failures during the application (!) of a software. 4) A

software lifecycle consists of 3 main phases which are accessible to mathematical modeling: the testing and debugging phase, the validation phase, and the operational phase. (For the time being no promising approaches exist for modeling the maintenance phase.) Three early models are frequently used as benchmarks against which to compare more recent ones: these are the models of JELINSKY and MORANDA [29], LITTLEWOOD and VERALL [30], GOEL and OKUMOTO [31].

Jelinsky-Moranda Model It is assumed that 1) failures are uniformly distributed over the software and 2) the software is executed in a uniform (but random) fashion. These assumptions imply that the software failure rate at any given time point is proportional to the number of faults in the software at that time point. Thus, if N is the original number of faults in the software, the time to the first failure is exponentially distributed with parameter λN. After the first fault is detected and eliminated, the failure rate drops to $\lambda(N-1)$ and so on. Hence the time t_i between the (i-1)th and i th failure is exponentially distributed with parameter $\lambda(N-i)$. Based on the t_i, the total number of failures N can be estimated by the *likelihood function*:

$$L(t_1, t_2, \ldots, t_n) = \prod_{i=1}^{n} (N-i+1)\lambda \exp[-(N-i+1)\lambda t_i].$$

Littlewood-Verall-Model This model takes into account that attempts at removing faults from software may not be successful and may even lead to an increase in the number of faults. LITTLEWOOD and VERALL therefore describe the failure rate of a software by a stochastic process. Of course, its trend function is assumed to be decreasing.

Goel-Okumoto-Model Let M(t) be the number of faults found in [0, t]. $\{M(t), t \geq 0\}$ is assumed to be an inhomogeneous Poisson process with trend function m(t). It is assumed that the mean number of faults found in the interval $[t, t+\Delta t]$ is proportional to the mean number of faults still present in the software at time point t (this assumption is well-known from the theory of population growth):

$$m(t+\Delta t) - m(t) = a(N - m(t))\Delta t.$$

This implies the differential equation

$$\frac{d}{dt} m(t) = a(N - m(t))$$

with the boundary conditions $m(0) = 0$ and $m(\infty) = N$. Its solution is

$$m(t) = N(1 - e^{-at}).$$

Models for the test phase The models outlined so far do not explicitly take into account the life phases of a software. However, models especially created for the test phase are most important for increasing and predicting software reliability. A typical model of this kind was introduced by DOWNS [32] and extended by DOWNS and GARRONE [33]. According to the situation in the test phase, this model assumes:

1) Software testing is restricted to the execution of a sequence of logical paths through the software.

2) The input data of the software is governed by an execution schedule specifying the probabilities with which different paths are selected for execution.

3) The execution schedule is fixed in the intervals between software failures.

Under these assumptions the test process is a sequence of Bernoulli trials:

Result 0: The path tested contains a fault.

Result 1: the path tested contains no fault.

Notation

r number of paths which are tested independently of each other per unit time.

Z random number of paths which have to be tested to find the first fault.

p probability of finding no fault per unit time.

$p^{1/r}$ is the probability that a given path is faultless. Therefore, Z is geometrically distributed with parameter $p^{1/r}$. Thus,

$$E(Z) = \frac{1}{1-p^{1/r}}.$$

If T denotes the random time to detecting a fault, then

$$E(T) = \frac{1}{r(1-p^{1/r})}.$$

Providing faults are uniformly distributed over a path, the time between the detection of two faults is exponentially distributed with parameter $\lambda = -\ln p$. Within this model, DOWNS and GARRONE also consider the possibility that a fault may affect several paths. The number of affected paths is usually unknown. Consider a software with N faults and M paths. Let a fault affect k paths with probability p_k, $k = 1, 2, \ldots, M$. To adapt a model to this situation let us next assume that the software is faultless. N faults are allocated independently of each other to a random number X of paths each, where X is distributed according to $P(X = k) = p_k$; $k = 0,1,\ldots$ Then it can easily be seen that the probability of any path being faultless is given by

$$p = \left(1 - \frac{E(X)}{M}\right)^N \approx 1 - \frac{NE(X)}{M}.$$

If the faults are assumed to be uniformly distributed over a path, the time between the detection of two faults is again exponentially distributed with parameter $\lambda = -\ln p$. A similar model was considered by YAMADA, HISHATINANI and OSAKI [34]. A neural networks approach for predicting software faults during the test phase is presented by KHOSHGOFTAAR and SZABO [35]. Imperfect debugging during the test phase is taken into account by FAKHRE-ZAKERI and STUD [36]. Finally, YANG and CHAO [37] propose stopping rules for software testing.

5. MAINTENANCE THEORY

Few systems are designed to operate without maintenance. These work in environments where access is impossible or at least very difficult, for example in outer space or high-radiation fields. Usually systems are subjected to maintenance, both to preventive maintenance and corrective maintenance. In preventive maintenance, components are exchanged, lubricants applied, adjustments made and so on before failures occur. In this way the system reliability is preserved or increased by staving off aging effects. Corrective maintenance is performed on failures. The mathematical theory of maintenance provides tools for efficient organization of preventive and corrective maintenance measures.

Let $F(t)$, $\bar{F}(t)$, and $\lambda(t)$ be the failure probability, the survival probability and the failure rate of the system. If X denotes the random time to the first failure (lifetime) of the system, then $F(t) = P(X \le t)$ is the distribution function and $f(t) = F'(t)$ is the probability density of X, and the failure rate is given by

$$\lambda(t) = f(t) / \bar{F}(t).$$

To have an aging system, $\lambda(t)$ is assumed to be nondecreasing. (Otherwise preventive maintenance would make no sense.)

Renewal theory deals with the most simple model of corrective maintenance: On failure the system is replaced by an identical new one. The corresponding maintenance action is called *replacement* or *renewal* of the system. (After a replacement the system is "as good as new".) Another basic maintenance action is the *minimal repair*: A minimal repair does not affect the failure rate of the system, i.e. its failure

rate on repair has the same value as immediately before the failure. (After a minimal repair the system is "as bad as old".) More exactly, if a system failure occuring at system age x is removed by a minimal repair, then the "residual lifetime" of the system has the distribution function

$$F_x(t) = \frac{F(t+x) - F(x)}{1 - F(x)}.$$

Maintenance policies considered until the seventies scheduled replacements and minimal repairs in an arbitrary manner, i.e. without taking into account the character of the failure causing a maintenance action (BARLOW and PROSCHAN [38, 39]. A breakthrough came with the paper of BEICHELT [40] (see also BEICHELT and FISCHER [41, 42]). Here two types of system failures were introduced:

Type 1-failures: can be removed by minimal repair.
Type 2-failures: have to be removed by replacement.

Hence, failures with minor consequences will be removed by minimal repairs and failures causing severe damage to the system by replacements. Probabilities p(t) and 1 – p(t) are given that a failure occuring at time t is of type 2 or type 1, respectively. Applying only corrective maintenance, i.e. maintenance according to the failure type, and denoting by c_m the expected cost of a minimal repair and by c_r the expected cost of a replacement, the expected long-run total maintenance cost per unit time is in case of p = p(t) given by

$$K = \frac{\frac{1-p}{p} c_m + c_r}{\int_0^\infty \left[\bar{F}(t)\right]^p dt}. \qquad (6)$$

This criterion serves as a benchmark against which to compare more sophisticated maintenance policies.

As a by-product, the 2-type-failure model allows the mathematically exact treatment of the so-called *repair-limit-replacement-policy.* Under this policy a system will be replaced if the repair cost exceeds a given limit L. Otherwise a minimal repair is carried out. Hence, the two failure types specified above are now generated by the random repair cost C. Let R(x) be the distribution function of C. Then p = 1 - R(L).

Substituting this p in (6) yields the expected total maintenance cost rate per unit time:

$$K(L) = \frac{\frac{1}{\bar{R}(L)} \int_0^L \bar{R}(x)dx + c_r - L}{\int_0^\infty \left[\bar{F}(t)\right]^{\bar{R}(L)} dt}.$$

The significance of the paper BEICHELT [40] only became then obvious when the same results were published some years later by BROWN and PROSCHAN [43] for constant p and by BLOCK, BORGES and SAVITS [44] for time-dependent p, see also BEICHELT [45], BEICHELT and FRANKEN [46].

Repairs of general degree Replacements and minimal repairs are extreme cases in quantifying the effect of maintenance actions on the system. (However, it also makes sense to allow maintenance actions making a system "better than new".) A natural generalization of the two-type-failure model is to consider repairs for which the effect on the system reliability is "between the effects of minimal repairs and replacements". To characterize such *repairs of general degree* let us consider the sequence

$$t_1, t_2, \ldots, t_i, \ldots$$

of failure times of a system. A repair occuring at time t_i has the degree δ_i given by

$$\delta_i = \frac{\lambda(t_i - 0) - \lambda(t_i + 0)}{\lambda(t_i - 0) - \lambda(0)}. \qquad (7)$$

In general, $0 \le \delta_i \le 1$. Minimal repairs and replacements are characterized by $\delta_i = 0$ and $\delta_i = 1$, respectively. Hence the failure rate of the system after the i th repair is

$$\lambda(t_i + 0) = \lambda(t_i - 0) - \delta_i[\lambda(t_i - 0) - \lambda(0)]\ ; \quad i = 1, 2, \ldots$$

To quantify the effect of repairs on system reliability, KIJIMA [47] introduced the concept of the *virtual age* of a system, which attracted more attention of researchers than criterion (7). The idea of the virtual age approach is that after a repair following a failure the system is "younger" than immediately before the repair. More exactly, the residual lifetime of a system after a repair is stochastically greater than its residual lifetime immediately before the failure inducing this repair. (Thus, the virtual age of a technical system corresponds to the biological age of a human being.) It is not the calender age of a system that is the decisive characteristic of its reliability,

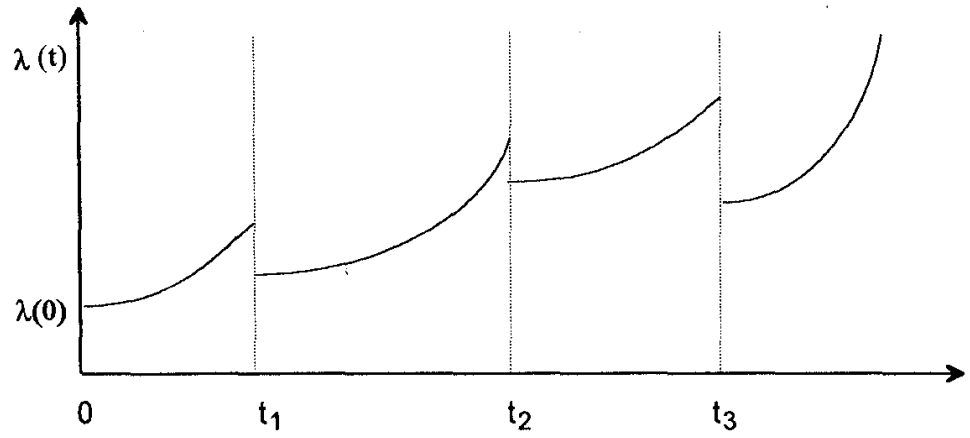

Fig. 1 Qualitative behaviour of the failure rate in case of repairs with general degree of repair

but its virtual age. This corresponds to the obvious fact that a well-maintained system is more reliable than a poorly maintained one.

Let Y(t) be the virtual age of a system having the calender time age t. Immediately before the i th failure takes place at calender time t_i, the system has the virtual age

$$Y(t_i - 0) = Y(t_{i-1} + 0) + \Delta t_i ,$$

where

$$\Delta t_i = t_i - t_{i-1} ; \quad i = 1, 2, \ldots ; \; t_0 = 0 .$$

Assuming $0 \le g_i \le 1$, KIJIMA proposed two models for the virtual age process $\{Y(t), t \ge 0\}$:

Model 1: The virtual age of the system after the i th repair is

$$Y(t_i + 0) = Y(t_{i-1} + 0) + (1 - g_i)\,\Delta t_i .$$

Therefore, the i th repair can only remove those damages arising in $[t_{i-1}, t_i]$.

Model 2: The virtual age of the system after the i th repair is

$$Y(t_i + 0) = (1 - g_i)\,Y(t_i - 0) .$$

Thus, repairs can principally remove all preceding damages to the system. If

$$g_i \equiv 0 \; (g_i \equiv 1)$$

only minimal repairs (replacements) are carried out.

Repairs of general degree have up to now been the subject of international research and will surely also continue being so for some years; see e.g. GUO and LOVE [48], MAKIS and JARDINE [49], LIU, MAKIS and JARDINE [50].

LITERATURE

[1] K. B. MISRA, *New Trends in System Reliability Evaluation,* Elsevier Science Publishers, Amsterdam (1993).

[2] I. E. C DOCUMENT, *List of basic terms, definitions and related mathematics for reliability,*Publication No. 271, 2nd ed., International Electrotechnical Commission (1974).

[3] J. ABRAHAM, *An improved algorithm for network reliability,* IEEE Trans. Reliab. R-28, 58-61 (1979).

[4] M. O. LOCKS, *Recursive disjoint products: A review of 3 algorithm,* IEEE Trans. Reliab. R-33, 31-35 (1982).

[5] F. BEICHELT and L. SPROSS, *An improved Abraham method for generating disjoint sums,* IEEE Trans. Reliab. R-36, 70-74 (1987).

[6] K.D. HEIDTMANN, *Sums of disjoint products by subproduct inversion,* IEEE Trans. Reliab. R-38, 305-311 (1989).

[7] M. VEERARAGHAVAN and K. S. TRIVEDI, *Multiple Variable Inversion Techniques,* Chapt. 2 in: K. B. MISRA (ed.), *New Trends in System Reliability Evaluation,* Elsevier, pp 39-74 (1993).

[8] R. BERTSCHY and P. A. MONNEY, *A generalization of the algorithm of Heidtmann to non-monotone formulas,* Journ. of Computational and Applied Mathematics 76, 55-96 (1996).

[9] J. KOHLAS, *Allocation of arguments and evidence theory,* Theoretical Compter Science 171, 221-246 (1997).

[10] B. ANRIG and N. LEHMANN, *Calculating disjoint terms for formulas containing finite set constraints,*J. of Comp. and Appl. Mathem, to appear in 1997.

[11] L. A. BAXTER, *Continuum structures* I, Journ. Appl. Prob. 21, 802-815 (1984).

[12] L. A. BAXTER, *Continuum structures* II, Math. Proc. Cambr. Phil. Soc. 99, 331-338 (1986).

[13] N. A. MAZARS, *A unified theory for coherent systems in reliability* I, II, Appl. Stochastic Models and Data Analysis 5, 179-201; 203-231 (1989).

[14] F. C. MENG, *Characterizing the Barlow-Wu-structure functions,* Naval Research Logistics 41, 661-668 (1994).

[15] A. M. ABOUAMMOH and M. AL-KADI, *Multistate coherent systems of order k,* Microelectr. & Reliab. 35, 1415-1421 (1995).

[16] A. ROSENTHAL, *Computing the reliability of complex networks,* SIAM Journ. of Appl. Mathem. 32, 384-393 (1977).

[17] A. ROSENTHAL and D. FRISQUE, *Transformations for simplifying network reliability calculations,* Networks 7, 97-111 (1977).

[18] F. BEICHELT and P. TITTMANN, *A splitting formula for the K-terminal reliability of stochastic networks,* Stochastic Models. Communications in Statistics 8, 307-327 (1992).

[19] A. BLECHSCHMIDT and P. TITTMANN, *Reliability bounds based on network splitting,* Journ. Inform. Proc. Cybernetics 27, 317-326 (1991).

[20] K. B. MISRA, *Reliability Analysis and Prediction-A Methodology Oriented Treatment,* Elsevier Science Publishers, Amsterdam (1992).

[21] J. KOHLAS, *Availability and Maintenance*Teubner, Stuttgart (1987).

[22] F. BEICHELT, *Reliability- and Maintenance Theory* Teubner, Stuttgart (1993).

[23] F. BEICHELT, *Decomposition and Reduction Techniques,* in: *New Trends in System Reliability Evaluation* (ed.: K. B. MISRA), Elsevier Science Publishers, Amsterdam, 75-116 (1993).

[24] W. J. RÜGER, *Reliability analysis of networks with capacity-constraints and failures at branches and nodes,*IEEE Trans. Reliab. R-35, 523-528 (1986).

[25] J.-S. LIN et al., *On the reliability evaluation of a capacitated-flow network in terms of minimal path sets,* Networks 25, 131-138 (1995).

[26] R. JENTSCH, *Reliability Analysis in Flow Networks,* in: Advances in Stochastic Models for Reliability, Quality and Safety (ed.: W. Kahle), Birkhäuser, Boston, Basel, Berlin, 235-236 (1997).

[27] J. D. MUSA and K. OKUMOTO, *Software reliability models: Concepts, classification, comparisions, and practice,* in: J. K. SKWIRZYNSKI (ed.), *Electronic Systems Effectiveness and Life Cycle Costing,*Springer, NewYork, 395-423 (1982).

[28] R. E. BARLOW and N. D. SINGPURWALLA, *Assessing the reliability of computer software and computer networks: An opportunity for partnership with com puter scientists,*The American Statistician 39, 88-94 (1985).

[29] Z. JELINSKY and P. B. MORANDA, *Software reliability research,* in: W. FREIBERGER (ed.), *Statistical Computer Performance Evaluation,* Academic Press, 465-484 (1972).

[30] B. LITTLEWOOD and J. L. VERALL, *A Bayesian reliability model with stochastistically monotone failure rate,* IEEE Trans. on Reliab. R-23, 108-114 (1974).

[31] A. L. GOEL and K. OKUMOTU, *Time-dependent error-detection rate model for software reliability and other performance measures,* IEEE Trans. on Reliab. R-28, 206-211 (1979).

[32] T. DOWNS, *An approach to the modeling of software testing with some applications,*IEEE Trans. on Software Engineering, SE-11, 375-386 (1985).

[33] T. DOWNS, P. GARRONE, *Some new models of software testing with performance comparisions,* IEEE Trans. on Reliab. 40, 322-328 (1991).

[34] S. YAMADA, J. HISHATINANI and S. OSAKI, *Test effort dependent software reliability measurement,* Int. J. System Science 22, 73-83 (1991).

[35] T. M. KHOSHGOFTAAR and R. M. SZABO, *Using neural networks to predict software faults during testing,* IEEE Trans. Reliab. 45, 456-462 (1996).

[36] I. FAKHRE-ZAKERI and E. STUD, *Mixture models for reliability of software with imperfect debugging,* IEEE Trans. on Reliab. R-44, 104-113 (1995).

[37] M. YANG and A. CHAO, *Reliability-estimation & stopping rules for software testing,*IEEE Trans. on Reliab. R-44, 315-321 (1995).

[38] R. E. BARLOW and F. PROSCHAN, *The mathematical theory of reliability,* Wiley, NewYork (1965).

[39] R. E. BARLOW and F. PROSCHAN, *Statistical theory of reliability and life testing,* Holt, Rinehart and Winston, NewYork (1975).

[40] F. BEICHELT, *A general preventive maintenance policy,* Math. Operationsforschung und Statistik 7, 927-932 (1976).

[41] F. BEICHELT and K. FISCHER, *General failure model applied to preventive maintenance policies,* IEEE Trans. on Reliab. R-29, 39-41 (1980).

[42] F. BEICHELT and K. FISCHER, *On a basic equation of reliability theory,* Microelectr. & Reliab. 19, 367-369 (1980).

[43] M. BROWN and F. PROSCHAN, *Imperfect repair,* J. Appl. Prob. 20, 851-859 (1983).

[44] H. W. BLOCK, W. S. BORGES and T. H. SAVITS, *Age dependent minimal repair,* Journ. Appl. Prob. 22, 370-385 (1985).

[45] F. BEICHELT, *A unifying treatment of replacement policies with minimal repair,* Naval Research Logistics 40, 58-75 (1993).

[46] F. BEICHELT and P. FRANKEN, *Reliability and Maintenance-Mathematical Methods* (in German), Hanser, München-Wien (1984).

[47] M. KIJIMA, *Some results for repairable systems with general repair,* Journ. of Appl. Prob. 26, 89-102 (1989).

[48] R. GUO and C. E. LOVE, *Statistical analysis for age model of imperfect repairable systems,* Quality and Reliab. Engin. Internat. 8, 133-146 (1992).

[49] V. MAKIS and K. S. JARDINE, *A note on optimal replacement policy under general repair,* European Journ. of Oper. Research 69, 75-82 (1993).

[50] X.-G. LIU, V. MAKIS and K. S. JARDINE, *A replacement model with overhauls and repairs,* Naval Research Logistics 42, 1063-1079 (1995).

Acknowledgement

The author thanks the editor for many hints which substantially improved the quality of the paper.