Journal of the Operations Research
Society of Japan
2005, Vol. 48, No. 1, 71-88

RELIABILITY EVALUATION AND OPTIMIZATION OF DISSIMILAR-COMPONENT COLD-STANDBY REDUNDANT SYSTEMS

Amir Azaron Hideki Katagiri Kosuke Kato Masatoshi Sakawa
Hiroshima University

(Received March 25, 2004; Revised July 26, 2004)

Abstract A new methodology for reliability evaluation and optimization of non-repairable dissimilar-component cold-standby redundant systems is introduced in this paper. Each component includes some elements with series-parallel configuration. When the operating component fails, the next dissimilar component is put into operation. Therefore, the initially functioning elements are backed up by one or more cold-standby redundant elements. The lifetimes of the elements are assumed to be independent random variables with Erlang distribution. The purchase cost of each element is assumed to be an increasing function of its expected lifetime. To evaluate the system reliability, we apply shortest path technique in stochastic networks. To do that, we construct a directed stochastic network called E-network, in which each path of this network corresponds with a minimal cut of the reliability graph. Finally, we construct a multi-objective problem and apply the surrogate worth trade-off method to solve the related problem and determine the optimal allocation of reliability to the elements of the system.

Keywords: Reliability optimization, Markov processes, multiple objective programming, graph theory

1. Introduction

After initial production, improvements are often made to components of a system to upgrade system performance; for example, when designing a later version or release. Two common methods of improving system reliability are to increase the reliability of components and/or add redundancy. Many of the reliability optimization models are formulated as N-stage series systems. At each stage there is the option to add redundancy in the form of parallel components. The optimization problem is to determine the optimal number of parallel components at each stage along with the reliability of each component to maximize system reliability subject to resource constraints such as cost or weight.

This paper presents a new approach for reliability evaluation and optimization of non-repairable dissimilar-component cold-standby redundant systems. Cold-standby means that the redundant components cannot fail while they are waiting. Each component includes some elements with mixed parallel and series configurations. The lifetimes of the elements are assumed to be independent random variable, because the elements of the system work independently. This methodology also allows constant and increasing hazard functions. The purchase cost of each element is assumed to be an increasing function of its expected lifetime. In other words, it is possible to increase the expected lifetime of each element by paying higher purchase cost. From a general prospective, the problem is to determine how much should be invested in the reliability of the elements so that the total costs of the standby system (initial purchase cost) is minimized, the system MTTF (mean time to failure) is maximized and also the system VTTF (variance time to failure) is minimized.

This approach involves the use of graph theory, Markov processes, reliability analysis and multiple objective programming.

Many fielded systems use cold-standby redundancy as an effective system design strategy. Space exploration and satellite systems achieve high reliability by using cold-standby redundancy for non-repairable systems, see Sinaki [25]. Generally, the use of cold-standby redundancy provides higher system reliability compared to an analogous system architecture with active redundancy. Space inertial reference units are required to accurately monitor critical information for extended mission times without opportunities for repair. Sinaki [25] describes several design strategies to achieve high system reliability for these systems. The use of cold-standby redundancy is preferable for these systems because of the comparable system reliability advantages. Redundant computer functions and hardware are used as part of the system design. They are not activated and thus not stressed, until the functioning unit has failed. Many other systems use cold-standby redundancy as an effective strategy to achieve high reliability including textile manufacturing systems, see Pandey et al. [19], and carbon recovery systems used in fertilizer plants, see Kumar et al. [15].

There has been little research directed toward the study of cold-standby redundancy with non-repairable systems. The problem has often been solved for active redundancy with non-repairable systems, or cold-standby redundancy with repairable systems. For active redundancy, the problem has been solved using dynamic programming (Fyffe et al. [8] and Nakagawa et al. [16]), integer programming (Bulfin et al. [5] and Gen et al. [9]). Ushakov et al. [27] and Gnedenko et al. [10] presented algorithms to maximize the median time to failure. Nakashima et al. [17] solved an analogous problem to maximize the time period where system reliability remains above a preselected value. Their algorithm assumes that components have exponential lifetime, but that the distribution parameters are the decision variables to be determined in addition to the redundancy levels.

There are also a few papers, which consider the multi-objective reliability optimization for either time-independent case, see Sakawa [22], or active redundant systems (Sakawa [23] and Dhingra [7]), which optimize system reliability, cost, weight and volume, for a given mission time.

With cold-standby redundancy, it has generally been assumed that component repair is possible. Solution methodologies have been presented (Agrafiotis et al. [1], Gurov et al. [11], Harunuzzaman et al. [12], Subramanian et al. [26] and Vaurio [28]) to allocate redundancy for various system structures, maintenance strategies and repair time distributions. Painton et al. [18] have solved a reliability optimization problem that incorporates risk. In their work, component lifetime is distributed according to an exponential distribution, but the distribution parameter itself is a random variable. They used genetic algorithm to maximize a lower percentile of the mean time between failure for a fixed system structure, given defined component reliability improvement levels and repair assumptions.

When standby redundancy is used for non-repairable systems, the problem has received less attention. Albright et al. [2] have solved a reliability optimization problem for non-repairable systems with standby redundancy. They assume exponential lifetime and one component choice per subsystem. Robinson et al. [21] studied system design for non-repairable systems with cold-standby redundancy. They examined systems designed with components that have phase-type lifetime distributions. Coit [6] have determined optimal design configurations for non-repairable series-parallel systems with cold-standby redundancy. His problem formulation considers non-constant component hazard functions and imperfect switching. Psarad et al. [20] consider the problem of allocating multi-functional redundant components for deterministic and stochastic mission times. In their formulation, there is a limit on the total number of redundant components that can be used. Their algorithm is based on sufficient conditions for different classes of component lifetime and mission time distributions.

The major limitations in the reliability optimization approaches for non-repairable systems thus far are:

1. Existing system reliability optimization algorithms are most often available for active redundancy. The logarithm of system reliability for an active standby redundant system is a separable function and dynamic programming or integer programming can be easily used to determine optimal solutions to the problem.
2. Available algorithms that do address cold-standby optimization generally assume similar redundant components and exponential lifetimes.
3. The system structure is very rigidly defined. The most of articles used the common system of N-stages in series, with the option of adding redundancy at each stage. Although this is a start, there are many more complicated system configurations that should be examined. The problem lies in the difficulty of presenting more complicated structures.
4. Only one criterion for time-dependent reliability, like maximizing system reliability at a given mission time, maximizing MTTF or maximizing the time for which the system reliability exceeds a specified value, is considered in the model. In the reliability optimization problem, one often wishes to lower the risk that systems with short system lifetime are produced, but only maximizing MTTF is not always fit for the requirement, especially when the optimally designed system has a large VTTF. In the first criterion, it matters how the mission time is selected.

This paper, not only considers the reliability optimization for complex structures (dissimilar-component cold-standby redundant systems, in which each component is composed of a number of elements with series-parallel configuration and non-constant hazard functions), but also the system is optimized with respect to the three important conflicting objectives: minimum total costs (initial purchase cost), maximum MTTF and also minimum VTTF. To evaluate the system reliability, we construct a directed stochastic network with exponentially distributed arc lengths called E-network, in which each path corresponds with a minimal cut of the reliability graph. We also prove that the system failure function is equal to the density function of the shortest path of E-network. To follow this approach, we extend the paper of Azaron et al. [3]. They developed a new approach to evaluate the reliability function of a class of redundant systems with exponentially distributed lifetimes. Then we construct the appropriate multi-objective optimal control problem, in order to determine optimal element reliabilities assuming that their respective hazard rates can be controlled. That is, the hazard rate of each element can be decreased with an increasing cost associated with this action.

This continuous-time problem is so complicated to solve by analytical methods, and therefore we try to solve it numerically. To do that, we discretize the continuous-time system and convert the optimal control problem into an equivalent nonlinear programming. This approach is similar to the work by Azaron and Fatemi Ghomi [4]. They developed a new approach for bicriteria optimal control of service rates of the service stations and also the arrival rates to the service stations in a class of Jackson networks, and solved the problem numerically by converting the optimal control problem into an equivalent nonlinear programming.

We can apply one of the multiple objective techniques to obtain the optimal element reliabilities. We use the surrogate worth trade-off method, see Hwang and Masud [13], which is an interactive approach with explicit trade-off information given, to solve this multi-objective reliability optimization problem.

The remainder of this paper is organized in the following way. In section 2, we extend the work of Azaron et al. [3] to evaluate the reliability function of dissimilar-component cold-standby systems with Erlang lifetime. In section 3, an analytical method for obtaining the distribution function of shortest path in E-networks is presented. In section 4, we present the multi-objective reliability optimization problem. In section 5, we explain about the surrogate worth trade-off method for solving our problem. In section 6, the method is illustrated through a numerical example, and finally we draw the conclusion of the paper in section 7.

2. Reliability Evaluation of Dissimilar-Component Cold-Standby Systems

A very efficient method to compute the reliability of a system is to express it as a graph. Reliability graphs consist of a set of arcs. Each arc represents an element of the system, while the nodes of the graph tie the arcs together and form the structure. Corresponding with the ith arc of the reliability graph, $i = 1, 2, \ldots, n$, there is a random variable T_i with Erlang distribution with the parameters (λ_i, n_i), which represents the lifetime of this element. When $n_i = 1$, the underlying distribution becomes exponential with the parameter λ_i. These random variables are independent, due to the fact that the elements work independently. If a system has i minimal path, denoted by $P_1, P_2, \ldots, P_i$, then it has a connection between its input and output nodes, if at least one of them is intact.

By definition, a cut of the graph is a set of arcs, which interrupts all connections between input and output when removed from the graph. A minimal cut is the one that contains no other cuts within it. Each system failure can be represented by the removal of at least one minimal cut from the graph.

As mentioned before, we consider a dissimilar-component cold-standby system, where each component is composed of a number of elements with series-parallel configuration and not all of its elements are set to function at time zero. Initially, only the elements of the first path of the reliability graph work. Upon failing one element of this path, the system is switched to the next path and the connection between the input and the output is established through this second path. This process continues until no more connection between the input and the output of the graph exists. In that case, the system fails.

In the systems, which we discuss in this paper, the minimal cuts are not coincided with the paths of the reliability graph. For example, our proposed methodology is not working in a bridge system, in that some minimal cuts and paths of the graph are coincided with each other.

Notations

- T_i: lifetime of the ith element of the system
- T: system lifetime
- C_j: jth minimal cut of the reliability graph, $j = 1, 2, \ldots, m$
- X_j: failure time of the jth minimal cut of the reliability graph

Lemma 2.1 *For $j = 1, 2, \ldots, m$, the following relation holds:*

$$X_j = \sum_{i \in C_j} T_i. \tag{2.1}$$

Proof: Taking into account the cold-standby nature of the structure, upon failure of each element of the jth minimal cut, the system is switched to the next path. Since this minimal cut is not coincided with any path of the reliability graph, then at any moment only one element of the jth minimal cut is activated. Therefore, the failure time of this cut is the sum of all its elements.

2.1. Equivalent network

To evaluate the reliability function, we construct a directed stochastic network with exponentially distributed arc lengths called E-network. There are m paths in this network, in which the jth path of this directed network corresponds with the jth minimal cut of the reliability graph of the system, $j = 1, 2, \ldots, m$. Clearly, by Lemma 2.1, the length of each path in this directed network is equal to the failure time of the corresponding cut. For constructing this network, we use the idea that if the lifetime of the ith element of the system is distributed according to an Erlang distribution with parameters (λ_i, n_i), it can be decomposed to n_i series of independent exponentially random variables with parameter λ_i. The following rule describes how to construct E-network.

Rule 1 Arc i belongs to the jth path of the E-network, if and only if $i \in C_j$. If $n_i = 1$, then the length of this arc has exponential distribution with parameter λ_i. Otherwise, if $n_i > 1$, then this arc is substituted with n_i series of arcs in the E-network, each having exponentially distributed arc length with parameter λ_i.

Let $F(t)$ represent the distribution function of shortest path (from the source to the sink node) in E-network and $R(t)$ represent the reliability function of the system. The relation between $F(t)$ and $R(t)$ can be expressed by the following theorem.

Theorem 2.1 *The system lifetime T is a random variable, as follows:*

$$T = \min_{j=1,2,\ldots,m} \{X_j\}. \tag{2.2}$$

Consequently,

$$R(t) = 1 - F(t). \tag{2.3}$$

Proof: Upon the failure of the first minimal cut of the reliability graph of the system, all connections between the input and the output are interrupted, and consequently the system fails. Therefore, the lifetime of the system would be equal to the failure time of the first minimal cut which results in (2.2). Relation (2.3) follows from the definitions of $R(t)$ and $F(t)$.

Clearly, the mean time to failure in the systems with cold-standby structure would be greater than or equal that of active standby systems. In section 3, we present an analytical method to obtain the distribution function of shortest path of E-network.

3. Shortest Path Analysis in E-Networks

In this section, we present an analytical method for obtaining the distribution function of shortest path in E-network, or in fact the distribution function of the shortest path from the source to the sink node of a directed stochastic network, in which arc lengths are exponentially distributed. To do that, we apply Kulkarni's method [14].

Let $G = (V, A)$ be a directed network, in which V and A represent the set of nodes and arcs of the network, respectively. We also assume s and t represent the source and the sink nodes of this network, respectively. The length of arc $(u, v) \in A$ is indicated by $T_{(u,v)}$, which is an exponential random variable with parameter $\lambda_{(u,v)}$.

For constructing the proper stochastic process, it is convenient to visualize the stochastic network as a communication network with the nodes as stations capable of receiving and transmitting messages and arcs as one-way communication links connecting pairs of nodes. The messages are assumed to travel at a unit speed so that $T_{(u,v)}$ denotes the travel time from node u to v. As soon as a node receives a message over one of the incoming arcs, it transmits it along all the outgoing arcs and then disables itself (*i.e.* loses the ability to receive and transmit the future messages). Now, let $X(t)$ be the set of all disabled nodes at time t. $X(t)$ is called the state of the network at time t.

Definition 3.1 *To describe the evolution of the stochastic process* $\{X(t), t \geq 0\}$ *for each* $X \subset V$*, where* $s \in X$ *and* $t \in \overline{X} = V - X$*, we define the following sets:*

$\overline{X}_1 = \{v \in \overline{X}$*: each path which connects any node of this set to the sink node* t*, includes at least one member of* $X\}$,

$S(X) = X \cup \overline{X}_1$.

Example 3.1 *In the network depicted in Figure 1, if we consider* $X = \{1, 2\}$*, then* $\overline{X}_1 = \emptyset$*, because for connecting node* $\{3\}$ *or* $\{4\}$ *to the sink node* $\{5\}$*, there is at least one path which does not include any member of* X*. However, if we consider* $X = \{1, 4\}$*, then the only path that connects node* $\{2\}$ *to node* $\{5\}$ *passes through node* $\{4\}$*, but node* $\{3\}$ *does not belong to* $\overline{X}_1$*, because it can be connected to* $\{5\}$ *directly and the path 3-5 does not include any node of* X*. Therefore,* $\overline{X}_1 = \{2\}$*, and* $S(X) = \{1, 2, 4\}$.

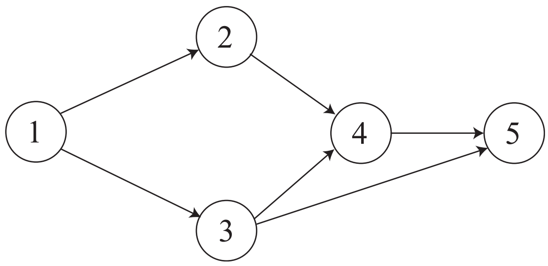

Figure 1: Graph of Example 3.1

Definition 3.2

$$\Omega = \{X \subset V / \overline{X}_1 = \emptyset\}, \tag{3.1}$$

$$\Omega^* = \Omega \cup V. \tag{3.2}$$

In the above example, $\Omega^* = \{\{1\}, \{1,2\}, \{1,3\}, \{1,2,3\}, \{1,2,4\}, \{1,2,3,4\}, \{1,2,3,4,5\}\}$. *The first six elements of* Ω^* *are the members of* Ω *while the last element of this set is* V.

Definition 3.3 *If* $X \subset V$ *such that* $s \in X$ *and* $t \in \overline{X}$*, then a cut is defined as:*

$$C(X, \overline{X}) = \{(u, v) \in A / u \in X, v \in \overline{X}\}. \tag{3.3}$$

There is a unique minimal cut contained in $C(X, \overline{X})$, denoted by $C(X)$. If $X \in \Omega$, then $C(X, \overline{X}) = C(X)$.

It is shown that $\{X(t), t \geq 0\}$ is a continuous-time Markov process with state space Ω^* and the infinitesimal generator matrix $Q = [q(X, Y)](X, Y \in \Omega^*)$ given by (see Kulkarni [14] for details)

$$q(X, Y) = \begin{cases} \sum_{(u,v)\in(X)} \lambda_{(u,v)} & \text{if } Y = S(X \cup \{v\}), \\ -\sum_{(u,v)\in C(X)} \lambda_{(u,v)} & \text{if } Y = X, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

We assume that the states in Ω^* are numbered $1, 2, \ldots, N = |\Omega^*|$ so that the Q matrix is upper triangular.

Let T represent the length of the shortest path in this E-network. Then, it is clear that,

$$T = \min\{t > 0 : X(t) = N/X(0) = 1\}. \tag{3.5}$$

Therefore, the length of the shortest path in the network is equal to the time until $\{X(t), t \geq 0\}$ gets absorbed in the final state N, starting from state 1. The objective is to compute $F(t) = P\{T \leq t\}$ or the distribution function of shortest path in the E-network.

Chapman-Kolmogorov backward or forward equations can be applied to compute $F(t)$. By using the backward algorithm, if we define

$$P_i(t) = P\{X(t) = N/X(0) = i\} \quad i = 1, 2, \ldots, N. \tag{3.6}$$

Then, $F(t) = P_1(t)$.

The system of differential equations for the vector $P(t) = [P_1(t), P_2(t), \ldots, P_N(t)]^T$ is given by

$$\begin{aligned} \dot{P}(t) &= Q.P(t), \\ P(0) &= [0, 0, \ldots, 1]^T. \end{aligned} \tag{3.7}$$

By taking advantage of the upper triangular nature of Q, the differential equations (3.7) can be easily solved. After computing $F(t)$, the system reliability is computed from equation (2.3).

4. Multi-Objective Reliability Optimization Problem

In this section, we develop a model to optimally allocate the reliability to the elements of the cold-standby system, when the expected lifetime of each element depends on its purchase cost and one can select an element with the desired expected lifetime. In fact, we may decrease the hazard rate of an element or replace it with a better element by buying a more expensive element. In that case, the mean time to failure of the system will be increased. However, clearly it causes the initial purchase cost of the system to be increased, accordingly. Consequently, an appropriate trade-off between cost and reliability is required.

To achieve the above-mentioned goals, we develop a multi-objective problem, in which three objectives are sought simultaneously, minimizing initial purchase cost, maximizing MTTF and also minimizing VTTF.

The purchase cost of each element is assumed to be an increasing function of its expected lifetime. The expected lifetime of an element with Erlang distribution and the parameters (λ_i, n_i) is equal to n_i/λ_i ($n_i = 1$ means that this element is distributed according to an exponential distribution with parameter λ_i). Therefore, the initial purchase cost C can be computed as follows:

$$C = \sum_{i=1}^{n} g_i\left(\frac{n_i}{\lambda_i}\right), \tag{4.1}$$

where $g_i(n_i/\lambda_i)$ is an increasing function respect to n_i/λ_i (n_i is assumed to be a constant value). MTTF and VTTF are given by

$$MTTF = \int_0^\infty t\dot{P}_1(t)dt, \tag{4.2}$$

$$VTTF = \int_0^\infty t^2\dot{P}_1(t)dt - \left[\int_0^\infty t\dot{P}_1(t)dt\right]^2. \tag{4.3}$$

Taking into account the above assumptions, the infinitesimal generator matrix, Q, is not constant, but it would be a function of the control vector $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_n]^T$. Therefore, the system of differential equations for the vector $P(t) = [P_1(t), P_2(t), \ldots, P_N(t)]^T$ is given by

$$\begin{aligned} \dot{P}(t) &= Q(\lambda).P(t), \\ P_i(0) &= 0 \quad i = 1, 2, \ldots, N-1, \\ P_N(t) &= 1. \end{aligned} \tag{4.4}$$

Due to the physical or technological limitations, the lifetime of available elements for substituting the ith element has the lower and upper bounds, which we denote them by L_i and U_i. Therefore, the set of constraints (4.5) should be satisfied.

$$L_i \le T_i \le U_i \quad i = 1, 2, \ldots, n. \tag{4.5}$$

Each T_i is a random variable, and therefore the above constraints cannot be always satisfied. According to the framework of the chance-constrained programming, we do not require that the constraints (4.5) should always be satisfied, but rather that they be satisfied with certain given probabilities. Thus the constraints (4.5) appear as

$$P(T_i \le L_i) \le \alpha_i \quad i = 1, 2, \ldots, n, \tag{4.6}$$

$$P(T_i \ge U_i) \le \beta_i \quad i = 1, 2, \ldots, n, \tag{4.7}$$

where α_i and β_i are the given probabilities. The distribution of T_i is Erlang with parameters (λ_i, n_i) and consequently the above set of constraints is substituted by

$$e^{-\lambda_i L_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i L_i)^k}{k!} \ge 1 - \alpha_i \quad i = 1, 2, \ldots, n, \tag{4.8}$$

$$e^{-\lambda_i U_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i U_i)^k}{k!} \le \beta_i \quad i = 1, 2, \ldots, n. \tag{4.9}$$

Taking into account the above notations and assumptions, the appropriate multi-objective optimal control problem would be

$$\begin{aligned} \text{Min} \quad & f_1(\lambda) = \sum_{i=1}^{n} g_i\left(\frac{n_i}{\lambda_i}\right) \\ \text{Max} \quad & f_2(\lambda) = \int_0^\infty t\dot{P}_1(t)dt \\ \text{Min} \quad & f_3(\lambda) = \int_0^\infty t^2\dot{P}_1(t)dt - \left[\int_0^\infty t\dot{P}_1(t)dt\right]^2 \\ \text{s. t.} \quad & \dot{P}(t) = Q(\lambda).P(t) \\ & P_i(0) = 0 \quad i = 1, 2, \ldots, N-1 \\ & P_N(t) = 1 \\ & e^{-\lambda_i L_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i L_i)^k}{k!} \ge 1 - \alpha_i \quad i = 1, 2, \ldots, n \\ & e^{-\lambda_i U_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i U_i)^k}{k!} \le \beta_i \quad i = 1, 2, \ldots, n \\ & P(t), \lambda \ge 0 \end{aligned} \tag{4.10}$$

This continuous-time problem is so complicated to solve by analytical methods, and therefore we try to solve it numerically. To do that, we discretize the continuous-time system and convert the optimal control problem into an equivalent nonlinear programming problem. In other words, we transform the differential equations into equivalent difference equations as well as transform the integral terms into equivalent summation terms. To follow this approach, the time interval is divided into K equal portions with length Δt. If Δt is sufficiently small, it can be assumed that $P(t)$ varies only in times $0, \Delta t, \ldots, (K-1)\Delta t$. Therefore, if we consider $P(k\Delta t)$ or the kth value of P as $P(k)$, then the continuous-time system $\dot{P}(t) = Q(\lambda).P(t)$ is transformed into the following discrete-time system:

$$P(k+1) = P(k) + Q(\lambda).P(k)\Delta t \quad k = 0, 1, \ldots, K-1. \tag{4.11}$$

Similarly, the integral terms are equivalent to the following summation terms:

$$\sum_{k=0}^{K-1} k\Delta t \left(P_1(k+1) - P_1(k)\right), \tag{4.12}$$

$$\sum_{k=0}^{K-1} (k\Delta t)^2 \left(P_1(k+1) - P_1(k)\right) - \left[\sum_{k=0}^{K-1} k\Delta t \left(P_1(k+1) - P_1(k)\right)\right]^2. \tag{4.13}$$

Since each $P_i(k)$, for $i = 1, 2, \ldots, N-1$, $k = 1, 2, \ldots, K$, is a distribution function, then,

$$P_i(k) \leq 1 \quad i = 1, 2, \ldots, N-1, \; k = 1, 2, \ldots, K. \tag{4.14}$$

Theoretically, when K approaches to infinity and Δt approaches to zero, the optimal results of the original problem will be obtained, but in this case the computational time also approaches to infinity, which is not practical in reality. Practically, we should select a finite value for K. The accuracy of the discrete-time approximation model is guaranteed by using a small value for Δt and a great value for K. A feasible solution should posses the following property: $P_1(K) \geq 1 - \varepsilon$, in which ε should approach zero. If a solution does not have this property, the value of Δt is increased in order to satisfy this necessary condition.

Finally, the following multi-objective nonlinear programming problem is approximately equivalent to the original continuous-time model and from which $\lambda^* = [\lambda_1^*, \lambda_2^*, \ldots, \lambda_n^*]^T$ or the optimal control vector is obtained.

$$\begin{aligned}
\text{Min} \quad & f_1(\lambda) = \sum_{i=1}^{n} g_i\left(\frac{n_i}{\lambda_i}\right) \\
\text{Max} \quad & f_2(\lambda) = \sum_{k=0}^{K-1} k\Delta t \left(P_1(k+1) - P_1(k)\right) \\
\text{Min} \quad & f_3(\lambda) = \sum_{k=0}^{K-1} (k\Delta t)^2 \left(P_1(k+1) - P_1(k)\right) - \left[\sum_{k=0}^{K-1} k\Delta t \left(P_1(k+1) - P_1(k)\right)\right]^2 \\
\text{s. t.} \quad & P(k+1) = P(k) + Q(\lambda).P(k)\Delta t \quad k = 0, 1, \ldots, K-1 \\
& P_i(0) = 0 \quad i = 1, 2, \ldots, N-1 \\
& P_N(k) = 1 \quad k = 0, 1, \ldots, K \\
& e^{-\lambda_i L_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i L_i)^k}{k!} \geq 1 - \alpha_i \quad i = 1, 2, \ldots, n \\
& e^{-\lambda_i U_i} \sum_{k=0}^{n_i-1} \frac{(\lambda_i U_i)^k}{k!} \leq \beta_i \quad i = 1, 2, \ldots, n \\
& P_i(k) \leq 1 \quad i = 1, 2, \ldots, N-1, \; k = 1, 2, \ldots, K \\
& P(k), \; \lambda \geq 0
\end{aligned} \tag{4.15}$$

If we consider the initial and terminal state conditions for $P(k)$ implicitly, and substitute each $P_i(0)$ with zero and $P_N(k)$ with one, nonlinear programming problem (4.15) would have $2K(N-1)+2n$ constraints and $2K(N-1)+3n$ variables, including the slack variables.

5. Surrogate Worth Trade-Off Method

For solving the multi-objective nonlinear programming (4.15), we use the surrogate worth trade-off method, which is an interactive approach with explicit trade-off information given. It is a virtue that all the alternatives during the solution process are non-dominated. Thus the decision maker in not bothered with any other kind of solutions. The major steps in the SWT method to solve the multi-objective problem (4.15) are

Step 1 Determine the ideal solution for each of the objectives in problem (4.15). Then set up the multi-objective problem in the form of (5.1).

$$\begin{aligned} \text{Min} \quad & f_1(\lambda) \\ \text{s. t.} \quad & f_2(\lambda) \geq \varepsilon_2 \\ & f_3(\lambda) \leq \varepsilon_3 \\ & \lambda \in S \text{ (Feasible region of problem (4.15))} \end{aligned} \tag{5.1}$$

Step 2 Identify and generate a set of non-dominated solutions by varying εs parametrically in problem (5.1). Assuming $\mu_j,\ j=2,3$ as the Lagrange multipliers corresponding with the first set of constraints of problem (5.1), the non-dominated solutions are the ones, which have non- zero values for μ_j.

Step 3 Interact with the DM (Decision-Maker) to assess the surrogate worth function w_j, or the DM's assessment of how much (from -10 to 10) he prefers trading μ_j marginal units of the first objective for one marginal unit of the jth objective $f_j(\lambda)$, given the other objectives remaining at their current values.

Step 4 Isolate the indifference solutions. The solutions, which have $w_j=0$ for all j, are said to be indifference solutions. If there exists no indifference solution, develop approximate relations for all worth functions $w_j=\hat{w}_j\ (f_j, j=2,3)$, by multiple regressions. Solve the simultaneous equations $\hat{w}_j(f)=0$ for all j to obtain f^* (f^* does not include f_1^*). Then, solve problem (5.2). Present this solution to the DM, and ask if this is an indifference solution. If yes, it is a preferred solution; proceed to Step 5. Otherwise, repeat the process of generating more non-dominated solutions around $\hat{w}_j(f)=0$ and refining the estimated f^* until it results in an indifference solution.

$$\begin{aligned} \text{Min} \quad & f_1(\lambda) \\ \text{s. t.} \quad & f_2(\lambda) \geq f_2^*(\lambda) \\ & f_3(\lambda) \leq f_3^*(\lambda) \\ & \lambda \in S \end{aligned} \tag{5.2}$$

Step 5 The optimal solution $f_1^*(\lambda)$ along with f^* and λ^* would be the optimal solution to the multi-objective problem (4.15).

In real-world problems, it is very difficult to design an element so as to satisfy the condition that the control vector λ is exactly equal to λ^*, but in most real cases, it is possible to produce an element in different prototypes with one month difference between the expected lifetimes of the consecutive prototypes. Therefore, considering year as the time unit in (4.15) and $[x]$ as the integer part of x, we can design an element with the expected lifetime (months) equal to $([n_i/0.0833\lambda_i^*] + 1)$ for substituting the ith element, if its optimal parameter is equal to λ_i^*, because the expected lifetime of an element with Erlang distribution is equal to n_i/λ_i and one month is equal to 0.0833 year.

6. Numerical Example

For controlling a spacecraft, there are three non-repairable dissimilar components in a cold-standby redundancy scheme, which are shown in Figure 2.

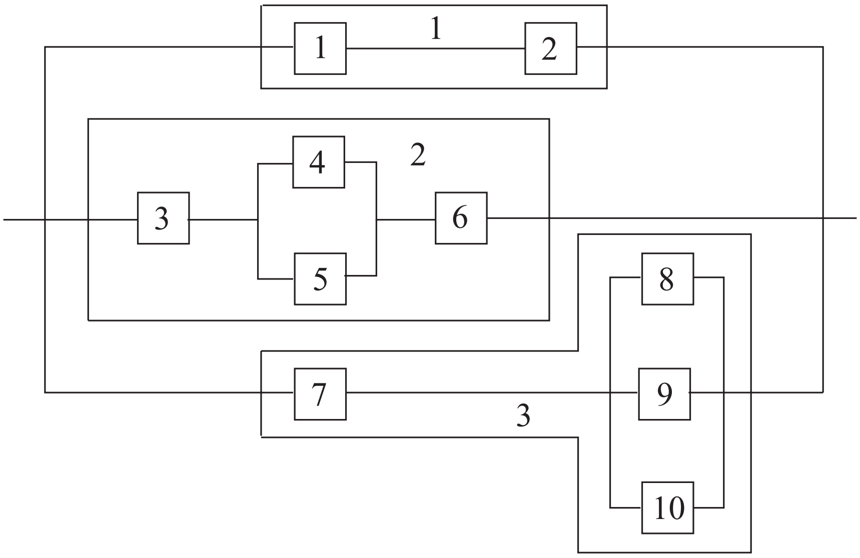

Figure 2: Cold-standby redundant system

At the beginning, the operating component is component 1, which includes a laptop computer (element 1) and a power supply (element 2) in a series configuration. When this component fails, the redundant component 2, which includes PC I (element 3), CD Drive I (element 4) and CD Drive II (element 5) as the cold-standby redundant elements and also a monitor (element 6), is put into operation. If component 2 fails, then the redundant component 3, which includes PC II (element 7) and Hard Drive I (element 8), Hard Drive II (element 9) and Hard Drive III (element 10) as the cold standby redundant elements, goes into operation. Table 1 shows the characteristics of the elements. The cost unit is in hundred dollars and the time unit is in year. We are interested in the optimal allocation of reliability to the elements of the system. First, we construct E-network following Rule 1, as depicted in Figure 3. In this network, the arcs 2 and $2'$ have both exponential distribution with parameter λ_2, according to Rule 1. The stochastic process $\{X(t), t \geq 0\}$ related to the

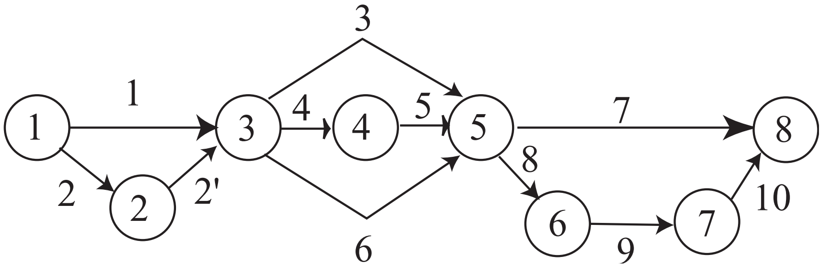

Figure 3: E-network

Table 1: Characteristics of the elements

i	Distribution	Parameters	$g_i\left(\frac{n_i}{\lambda_i}\right)$	L_i	U_i
1	Exponential	λ_1	$7\left(\frac{1}{\lambda_1}\right)^2+5$	0.01	8
2	Erlang	$(\lambda_2, n_2=2)$	$\left(\frac{2}{\lambda_2}\right)+2$	0.01	9
3	Exponential	λ_3	$8\left(\frac{1}{\lambda_3}\right)+6$	0.01	5
4	Exponential	λ_4	$4\left(\frac{1}{\lambda_4}\right)+3$	0.01	3
5	Exponential	λ_5	$4\left(\frac{1}{\lambda_5}\right)+3$	0.01	3
6	Exponential	λ_6	$10\left(\frac{1}{\lambda_6}\right)+4$	0.01	7
7	Exponential	λ_7	$3\left(\frac{1}{\lambda_7}\right)^2+7$	0.01	6
8	Exponential	λ_8	$5\left(\frac{1}{\lambda_8}\right)+2$	0.01	2
9	Exponential	λ_9	$5\left(\frac{1}{\lambda_9}\right)+2$	0.01	2
10	Exponential	λ_{10}	$5\left(\frac{1}{\lambda_{10}}\right)+2$	0.01	2

shortest path analysis of this E-network has 8 states in the order of

$$\Omega^* = \{\{1\},\{1,2\},\{1,2,3\},\{1,2,3,4\},\{1,2,3,4,5\},\{1,2,3,4,5,6\},\{1,2,3,4,5,6,7\},\{1,2,3,4,5,6,7,8\}\}.$$

The rate diagram for the continuous-time Markov process $\{X(t),\ t \geq 0\}$ is shown in Figure 4. Table 2 shows matrix $Q(\lambda)$.

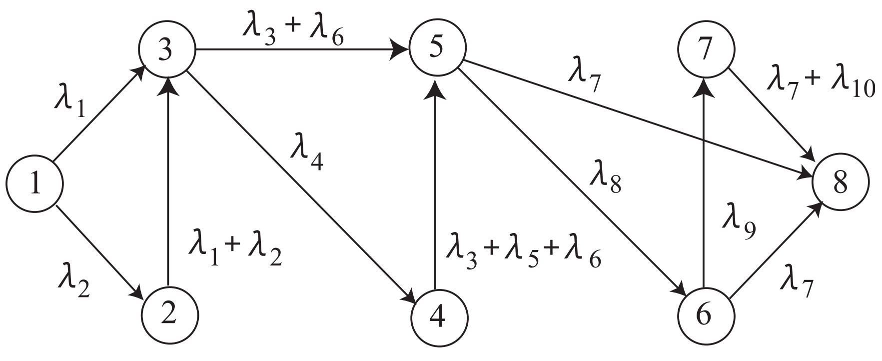

Figure 4: Rate diagram

The single objective optimization for generating a set of non-dominated solutions is formulated as:

$$\text{Min} \quad f_1(\lambda) = \sum_{i=1}^{10} g_i\left(\frac{n_i}{\lambda_i}\right)$$

$$\text{s. t.} \quad f_2(\lambda) = \sum_{k=0}^{K-1} k\Delta t(P_1(k+1) - P_1(k)) \geq \varepsilon_2$$

Table 2: Matrix $Q(\lambda)$

State	1	2	3	4	5	6	7	8
1	$-(\lambda_1+\lambda_2)$	λ_2	λ_1	0	0	0	0	0
2	0	$-(\lambda_1+\lambda_2)$	$\lambda_1+\lambda_2$	0	0	0	0	0
3	0	0	$-(\lambda_3+\lambda_4+\lambda_6)$	λ_4	$\lambda_3+\lambda_6$	0	0	0
4	0	0	0	$-(\lambda_3+\lambda_5+\lambda_6)$	$\lambda_3+\lambda_5+\lambda_6$	0	0	0
5	0	0	0	0	$-(\lambda_7+\lambda_8)$	λ_8	0	λ_7
6	0	0	0	0	0	$-(\lambda_7+\lambda_9)$	λ_9	λ_7
7	0	0	0	0	0	0	$-(\lambda_7+\lambda_{10})$	$\lambda_7+\lambda_{10}$
8	0	0	0	0	0	0	0	0

$$
\begin{aligned}
& f_3(\lambda)=\sum_{k=0}^{K-1}(k\Delta t)^2(P_1(k+1)-P_1(k))-\left[\sum_{k=0}^{K-1}k\Delta t(P_1(k+1)-P_1(k))\right]^2 \leq \varepsilon_3 \\
& P_1(k+1)=P_1(k)-(\lambda_1+\lambda_2)P_1(k)\Delta t+\lambda_2 P_2(k)\Delta t+\lambda_1 P_3(k)\Delta t \\
& \qquad k=0,1,\ldots,K-1 \\
& P_2(k+1)=P_2(k)-(\lambda_1+\lambda_2)P_2(k)\Delta t+(\lambda_1+\lambda_2)P_3(k)\Delta t \quad k=0,1,\ldots,K-1 \\
& P_3(k+1)=P_3(k)-(\lambda_3+\lambda_4+\lambda_6)P_3(k)\Delta t+\lambda_4 P_4(k)\Delta t+(\lambda_3+\lambda_6)P_5(k)\Delta t \\
& \qquad k=0,1,\ldots,K-1 \\
& P_4(k+1)=P_4(k)-(\lambda_3+\lambda_5+\lambda_6)P_4(k)\Delta t+(\lambda_3+\lambda_5+\lambda_6)P_5(k)\Delta t \\
& \qquad k=0,1,\ldots,K-1 \\
& P_5(k+1)=P_5(k)-(\lambda_7+\lambda_8)P_5(k)\Delta t+\lambda_8 P_6(k)\Delta t+\lambda_7\Delta t \quad k=0,1,\ldots,K-1 \\
& P_6(k+1)=P_6(k)-(\lambda_7+\lambda_9)P_6(k)\Delta t+\lambda_9 P_7(k)+\lambda_7\Delta t \quad k=0,1,\ldots,K-1 \\
& P_7(k+1)=P_7(k)-(\lambda_7+\lambda_{10})P_7(k)\Delta t+(\lambda_7+\lambda_{10})\Delta t \quad k=0,1,\ldots,K-1 \\
& P_i(0)=0 \quad i=1,2,\ldots,7 \\
& \mathrm{e}^{-\lambda_i L_i}\sum_{k=0}^{n_i-1}\frac{(\lambda_i L_i)^k}{k!} \geq 1-\alpha_i \quad i=1,2,\ldots,10 \\
& \mathrm{e}^{-\lambda_i U_i}\sum_{k=0}^{n_i-1}\frac{(\lambda_i U_i)^k}{k!} \leq \beta_i \quad i=1,2,\ldots,10 \qquad (6.1) \\
& P_i(k) \leq 1 \quad i=1,2,\ldots,7,\ k=1,2,\ldots,K \\
& P_i(k) \geq 0 \quad i=1,2,\ldots,7,\ k=1,2,\ldots,K \\
& P(k),\ \lambda \geq 0
\end{aligned}
$$

In nonlinear model (6.1), $P_N(k)=P_8(k)=1$ for $k=0,1,\ldots,K$, and we have substituted

$P_8(k)$ with 1 in this model. It is also clear that $n_i = 1$ for $i = 1, 3, 4, \ldots, 10$ and $n_2 = 2$. The values of other parameters are assumed to be $K = 10$, $\Delta t = 0.3$ and $\alpha_i = \beta_i = 0.05$ for $i = 1, 2, \ldots, 10$. We use LINGO to obtain the ideal solutions and to derive the trade-off ratios in the above problem. The ideal solutions are

$$\overline{f}_1 = 70.1641,\ \overline{f}_2 = 1.8398,\ \overline{f}_3 = 0.0839.$$

Then $\varepsilon_j,\ j = 2, 3$ are varied parametrically to obtain several solutions, and all dominated solutions are to be discarded. Table 3 summarizes a set of non-dominated solutions for the present problem. There exists no indifference solution. Therefore, we develop approximate relations for all worth functions $w_j = \hat{w}_j(f_j,\ j = 2, 3)$, by multiple regressions as follows:

$$\hat{w}_2(f_2, f_3) = 131.3955 - 79.2392 f_2 - 1.2148 f_3,$$
$$\hat{w}_3(f_2, f_3) = 32.8724 + 4.4444 f_2 - 163.5017 f_3.$$

These two functions are set equal to zero; the corresponding equations are solved simultaneously to give an estimate of the preferred values $f_2^* = 1.6544$ and $f_3^* = 0.246$. Problem (6.1) is solved with $\varepsilon_2 = f_2^* = 1.6544$ and $\varepsilon_3 = f_3^* = 0.246$, which yields the following solution:

$$f^* = (f_1^*, f_2^*, f_3^*) = (85.1059, 1.6544, 0.246),$$
$$\mu_2^* = -85.5898,\ \mu_3^* = 124.9094.$$

When these trade-off ratios and the solution f^* are presented for validation, the DM indicates this is an indifference solution and the algorithms is terminated.

Table 4 shows the optimal allocation of reliability to the ith element of the system λ_i^* and the expected lifetime (months) of the optimal designed element for substituting this element considering λ_i^*, for $i = 1, 2, \ldots, 10$. According to the data in Table 4, we conclude that for example, the desired element for substituting the laptop computer (element 1) would be a laptop computer with exponentially distributed lifetime and the expected lifetime equal to $([1/0.0833\lambda_1^*] + 1) = 14$ months, which can be easily designed in application, and we have to pay 1325.06 dollars for purchasing this kind of laptop computer.

Table 5 shows the reliability of spacecraft ($R(k) = 1 - P_1(k)$) at the time ($k\Delta t$), for $k = 1, 2, \ldots, 10$. According to the data in Table 5, it seems that the spacecraft exists no longer than $9 * 0.3 = 2.7$ years and the reliability of the spacecraft at this time is equal to 0.0553.

7. Conclusion

In this paper, we introduced a new methodology for reliability evaluation and optimization of non-repairable dissimilar-component cold-standby redundant systems, in which each component is composed of a number of elements with series-parallel configuration and non-constant hazard functions. We assumed for each element, the lifetime varies within a range and its expected lifetime is related with its purchase cost. More precisely, the purchase cost of each element is an increasing function of its expected lifetime. The system elements work independently and the lifetime of each element is a random variable with Erlang distribution.

Our approach is a powerful one for the reliability evaluation of non-repairable dissimilar-component cold-standby redundant systems. Computing the reliability of these systems, if is not impossible, is at least so complicated for most real case problems, because either the convolution integrals are intractable or the size of the state space would be enormous. For example, this problem could be solved by using clever complete enumeration of network

Table 3: A set of non-dominated solutions

f_1	f_2	f_3	μ_2	w_2	μ_3	w_3
88.3908	1.62	0.2	−107.9295	+5	164.623	+7
80.8953	1.58	0.23	−73.8353	+6	99.0391	+3
79.7004	1.55	0.22	−69.3345	+8	91.6105	+4
76.2838	1.53	0.25	−49.9411	+9	59.3037	0
78.4831	1.57	0.25	−60.2788	+7	76.2734	0
73.7376	1.52	0.3	−31.6802	+10	27.6091	−10
89.9790	1.65	0.21	−111.1706	0	174.8078	+4
89.8562	1.68	0.23	−104.3369	−2	167.9948	+2
85.9026	1.68	0.26	−81.5721	−2	122.9142	−2
82.8087	1.66	0.27	−75.0578	−1	104.9042	−4
87.2306	1.7	0.26	−87.2624	−3	135.1928	−1
85.8463	1.61	0.21	−96.5132	+4	141.1671	+4
80.6608	1.63	0.27	−67.3632	+3	89.5039	−5
88.2403	1.68	0.24	−95.7491	−3	148.9311	+1
83.1142	1.65	0.26	−77.7326	+1	109.0118	−2
83.6134	1.6	0.22	−82.3608	+6	122.5921	+3
86.3073	1.6	0.2	−100.5636	+5	148.1791	+6
90.2506	1.75	0.27	−91.1219	−8	154.0819	−5
87.9452	1.74	0.28	−82.9593	−7	133.9218	−7
86.3184	1.72	0.28	−79.7458	−5	123.1266	−6
84.0528	1.72	0.3	−70.9107	−4	104.293	−9
82.6558	1.7	0.3	−68.8018	−2	97.0604	−9
73.1377	1.5	0.3	−28.3461	+10	22.1547	−8
109.7418	1.75	0.2	−273.9583	−10	547.2812	+10
90.5768	1.72	0.25	−99.3223	−6	164.2597	+3
99.1315	1.74	0.23	−147.9979	−8	275.2248	+5

states, see Shooman [24]. According to our methodology, for a complete constructed network with l nodes and $l(l-1)$ arcs representing the elements of the system (the worst case example), the size of the state space would be $2^{l-2}+1$, but the size of the state space in Shooman's method would be equal to $3^{l(l-1)}$, because each element can be in one of these three states: work, fail and standby. In the numerical example, the state space has only 8 states, but according to the Shooman's method, the size of the state space would be 3^{10}. Therefore, this numerical example shows the efficiency of our new methodology.

To select the desired system elements, we developed a multi-objective model with three objectives, minimization of the total purchase costs, maximization of the mean time to failure of the system, and also minimization of the variance time to failure of the system.

The resulting continuous-time problem was so complicated to solve analytically. Therefore, we solved it numerically, by discretizing the relevant continuous-time system and converting the optimal control problem into an equivalent nonlinear programming problem.

To solve the relevant multiple objective programming, we used the surrogate worth trade-off method. The major advantages of the SWT method are

1. The method requires the DM to consider only two objectives at a time while assessing the worth function, even if there are many objectives in the problem.

Table 4: Optimal reliabilities and the expected lifetimes (months) of the designed elements

i	1	2	3	4	5
λ_i^*	0.9211	1.3137	0.9486	1.8112	1.8112
Expected lifetime	14	10	13	7	7

i	6	7	8	9	10
λ_i^*	1.0606	0.5329	2.3114	2.3114	2.3114
Expected lifetime	12	23	6	6	6

Table 5: $R(k)$ for $k = 1, 2, \ldots, 10$

k	1	2	3	4	5	6	7	8	9	10
$R(k)$	1	1	0.9734	0.8931	0.7276	0.4803	0.26	0.125	0.0553	0.0231

2. Both linear and nonlinear models can be exercised.
3. The method is applicable to both static and dynamic problems. Preference information from multiple decision makers with varied opinions can be processed. Taking into account the above advantages, the SWT method is an appropriate method to solve the multi-objective reliability optimization problem.

As mentioned, it is very difficult to design an element so as to satisfy the condition that the control vector λ is exactly equal to λ^* in most real-world problems. We indicated how to deal with this problem in Section 5. Another way of dealing with this problem is to consider λ as a discrete vector in the multi-objective problem (4.15), but the limitation of this way is that computing the Lagrange multipliers in the related nonlinear discrete problem (5.1) is almost impossible. Therefore, in this case, SWT method is not working anymore and we need to apply another multi-objective method like goal programming, goal attainment or STEM to solve this multi-objective discrete reliability optimization problem.

Our methodology can also be extended in the following directions:

1. The model can include other objective functions like the maximization of system reliability at the mission time or the maximization of time period where the system reliability remains above a preselected value.
2. The model can be extended to dissimilar-component systems, in which the configuration of the elements of any component is general.
3. The model can be extended to the general types of non-constant hazard functions and imperfect switching.

References

[1] G. K. Agrafiotis and M. Z. Tsoukalas: Reliability analysis and optimization applications of a two-unit standby redundant system with spare units. *Microelectronic and Reliability*, **34**(9) (1994), 1469–1475.

[2] S. C. Albright and A. Soni: Evaluation of costs of ordering policies in large machine repair problem. *Naval Research Logistics Quarterly*, **31**(3) (1984), 387–398.

[3] A. Azaron, H. Katagiri, M. Sakawa and M. Modarres: Reliability function of a class of time-dependent systems with standby redundancy. *European Journal of Operational*

Research, **164**(2) (2005), 378–386.

[4] A. Azaron and S. M. T. Fatemi Ghomi: Optimal control of service rates and arrivals in Jackson networks. *European Journal of Operational Research*, **147**(1) (2003), 17–31.

[5] R. L. Bulfin and C. Y. Liu: Optimal allocation of redundant components for large systems. *IEEE Transactions on Reliability*, **R-34** (1985), 241–247.

[6] D. W. Coit: Cold-standby redundancy optimization for nonrepairable systems. *IIE Transactions*, **33**(6) (2001), 471–478.

[7] A. Dhingra: Optimal apportionment of reliability & redundancy in series systems under multiple objectives. *IEEE Transactions on Reliability*, **41**(4) (1992), 576–582.

[8] D. E. Fyffe, W. W. Hines and N. K. Lee: System reliability allocation and a computational algorithm. *IEEE Transactions on Reliability*, **R-17** (1968), 64–69.

[9] M. Gen, K. Ida and J. U. Lee: A computational algorithm for solving 0-1 goal programming with GUB structures and its application for optimization problems in system reliability. *Electronics and Communications in Japan, Part 3*, **73** (1990), 88–96.

[10] B. Gnedenko and I. Ushakov: *Probabilistic reliability engineering* (John Wiley, New York, 1995).

[11] S. V. Gurov, L. V. Utkin and I. B. Shubinsky: Optimal reliability allocation of redundant units and repair facilities by arbitrary failure and repair distributions. *Microelectronics and Reliability*, **35**(12) (1995), 1451–1460.

[12] M. Harunuzzaman and T. Aldemir: Optimization of standby system maintenance schedules in nuclear power plants. *Nuclear Technology*, **113**(3) (1996), 354–367.

[13] C. Hwang and A. Masud: *Multiple objective decision making, methods and applications* (Springer-Verlag, 1979).

[14] V. G. Kulkarni: Shortest paths in networks with exponentially distributed arc lengths. *Networks*, **16**(3) (1986), 255–274.

[15] S. Kumar, D. Kumar and N. P. Mehta: Behavioral analysis of shell gasification and carbon recovery process in a urea fertilizer plant. *Microelectronics and Reliability*, **36**(5) (1996), 671–673.

[16] Y. Nakagawa and S. Miyazaki: Surrogate constraints algorithm for reliability optimization problems with two constraints. *IEEE Transactions on Reliability*, **R-30** (1981), 175–180.

[17] K. Nakashima and K. Yamato: Optimal design of a series-parallel system with time-dependent reliability. *IEEE Transactions on Reliability*, **26**(2) (1977), 119–120.

[18] L. Painton and J. Campbell: Genetic algorithms in optimization of system reliability. *IEEE Transactions on Reliability*, **44**(2) (1995), 172–178.

[19] D. Pandey, M. Jacob and J. Yadav: Reliability analysis of a powerloom plant with cold-standby for its strategic unit. *Microelectronics and Reliability*, **36**(1) (1996), 115–119.

[20] V. R. Psarad, W. Kuo and K. M. Oh-Kim: Optimal allocation of s-identical, multi-functional spares in a series system. *IEEE Transactions on Reliability*, **48**(2) (1999), 118–126.

[21] D. G. Robinson and M. F. Neuts: Standby redundancy in reliability: a review. *IEEE Transactions on Reliability*, **R-38**(4) (1989), 430–435.

[22] M. Sakawa: Multiobjective optimization by the surrogate worth trade-off method. *IEEE Transactions on Reliability*, **R-27**(5) (1978), 311–314.

[23] M. Sakawa: Reliability design of a standby system by a large-scale multiobjective optimization method. *Microelectronics and Reliability*, **20**(3) (1980), 191–204.

[24] M. Shooman: *Probabilistic Reliability: An Engineering Approach* (2nd ed., Krieger Publishing, 1991).

[25] G. Sinaki: Ultra-reliable fault tolerant inertial reference unit for spacecraft. *In Proceeding of the Annual Rocky Mountain Guidance and Control Conference* (Univelt Inc. Publishers, San Diego, California, 1994), 239–248.

[26] R. Subramanian and V. Anantharaman: Probabilistic analysis of a three unit cold-standby redundant system with repair. *Microelectronics and Reliability*, **35**(6) (1995), 1001–1008.

[27] I. Ushakov and R. Harrison: *Handbook of reliability engineering* (John Wiley, New York, 1994).

[28] J. K. Vaurio: On time-dependent availability and maintenance optimization of standby units under various maintenance policies. *Reliability Engineering and System Safety*, **56**(1) (1997), 79–89.

Hideki Katagiri
Graduate School of Engineering
Hiroshima University
1-4-1 Kagamiyama, Higashi-Hiroshima,
Hiroshima, 739-8527 Japan
E-mail: `katagiri@msl.sys.hiroshima-u.ac.jp`