IIE Transactions (2001) **33**, 471–478

Cold-standby redundancy optimization for nonrepairable systems

DAVID W. COIT

Department of Industrial Engineering, Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08854, USA
E-mail: coit@rci.rutgers.edu

Received July 1998 and accepted March 2000

A solution methodology is described and demonstrated to determine optimal design configurations for nonrepairable series–parallel systems with cold-standby redundancy. This problem formulation considers non-constant component hazard functions and imperfect switching. The objective of the redundancy allocation problem is to select from available components and to determine an optimal design configuration to maximize system reliability. For cold-standby redundancy, other formulations have generally required exponential component time-to-failure and perfect switching assumptions. For this paper, there are multiple component choices available for each subsystem and component time-to-failure is distributed according to an Erlang distribution. Optimal solutions are determined based on an equivalent problem formulation and integer programming. Compared to other available algorithms, the methodology presented here more accurately models many engineering design problems with cold-standby redundancy. Previously, it has been difficult to determine optimal solutions for this class of problems or even to efficiently calculate system reliability. The methodology is successfully demonstrated on a large problem with 14 subsystems.

1. Introduction

The redundancy allocation problem has been studied for many different system structures, objective functions and distribution assumptions. In this paper, a problem formulation and solution methodology are presented to maximize system reliability for nonrepairable system designs that use cold-standby redundancy as a strategy to increase system reliability. This methodology allows constant and increasing hazard functions and imperfect component failure detection and switching.

Cold-standby redundancy involves the use of redundant components that are shielded from the operational stresses associated with system operation. Without exposure to those stresses, the likelihood of failure is very low, and assumed to be zero, until the component is required to operate as a substitute for a failed component. When a failure does occur, it is necessary to detect the failure and to activate the redundant component. For a nonrepairable system, the failure detection and switching must be accomplished by additional system hardware that would not otherwise be required.

Many fielded systems use cold-standby redundancy as an effective system design strategy. Space exploration and satellite systems (Sinaki, 1994) achieve high reliability by using cold-standby redundancy for nonrepairable systems. Generally the use of cold-standby redundancy provides higher system reliability compared to an analogous system architecture with active redundancy. However, cold-standby is more difficult to implement because of the necessity to detect failures as they occur and activate the redundant component. This is not necessary for active redundancy.

Space inertial reference units are required to accurately monitor critical information for extended mission times without opportunities for repair. Sinaki (1994) describes several design strategies to achieve high system reliability for these systems. He contends that space-based computer systems cannot achieve the required reliability levels without the use of redundancy. The use of cold-standby redundancy is preferable for these systems because of the comparable system reliability advantages. Redundant computer functions and hardware, including gyros, 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 (Pandey *et al.*, 1996) and carbon recovery systems used in fertilizer plants (Kumar *et al.*, 1996).

Existing system reliability optimization algorithms are most often available only for active redundancy. Additionally, available algorithms that do address cold-standby optimization generally assume perfect switching and exponential failure times. The use of cold-standby requires that the system be able to detect a failure and

0740-817X

activate the redundant component. In practice, switching is often very reliable, but it can fail. The problem of imperfect switching has been studied in detail by Shankar and Gururajan (1993) and also Gurov and Utkin (1996), although there have been no previously successful research efforts to incorporate imperfect switching into system reliability optimization.

2. Redundancy allocation problem

The design of new products involves the specification of performance requirements, the evaluation and selection of components to perform clearly defined functions and the determination of a system-level architecture. Detailed engineering specifications prescribe minimum levels of reliability, maximum weight, maximum volume, etc. If the design is to be produced economically or within some specified budget, numerous design alternatives must be considered, resulting in a complex combinatorial optimization problem.

The redundancy allocation problem pertains to a system of s subsystems in series. Each subsystem is designed with one or more parallel components. If more than one component is used ($n_i > 1$), then there will be one initially operating component and $n_i - 1$ components in cold-standby waiting to be activated, if required.

For each subsystem, there are m_i functionally equivalent components that may be selected. Each available component has different levels of cost, weight, reliability and other characteristics and there is an unlimited supply of each of the m_i choices. There are system-level constraints and the problem is to select the component choices and levels of redundancy to maximize the system reliability. It is a difficult combinatorial problem which has been shown to be NP-hard by Chern (1992).

System reliability, $R(t; \mathbf{z}, \mathbf{n})$, depends on the component selections, $\mathbf{z}$, and the levels of redundancy, $\mathbf{n}$. z_i and n_i are the component choice and redundancy level respectively for subsystem i. The formulation assumes that the redundant component(s) for each subsystem are of the same type as the originally operating component, which is selected from any of the m_i choices.

A general problem formulation to maximize $R(t; \mathbf{z}, \mathbf{n})$ is presented as Problem P1. This formulation has only two constraints. The solution methodology presented in this paper can be readily extended to accommodate any number of linear constraints.

Problem P1:

$$\max g(\mathbf{z}, \mathbf{n}) = R(t; \mathbf{z}, \mathbf{n})$$

subject to

$$\sum_{i=1}^{s} c_i(z_i) n_i + \sum_{i:n_i>1} c_{s,i} \le C$$

$$\sum_{i=1}^{s} w_i(z_i) n_i + \sum_{i:n_i>1} w_{s,i} \le W$$

$$n_i \in \{1, 2, \ldots, n_{\max}\}$$

$$z_i \in \{1, 2, \ldots, m_i\}$$

$c_i(j)$ and $w_i(j)$ are the cost and weight of the jth available component for subsystem i while $c_{s,i}$ and $w_{s,i}$ are the cost and weight of the failure detection and switching required if $n_i > 1$. C and W are the maximum system cost and weight. $n_{\max}$ is a user-selected parameter. It is the maximum number of components that can be used in any subsystem. Often, this can be selected based on practical constraints. However, if an analyst has no rationale to select $n_{\max}$ or does not wish to limit the number of redundant components, than an upper-bound value can be selected to include the entire feasible region as defined by the system-level constraints.

There has been little research directed towards the study of cold-standby redundancy with nonrepairable systems. The problem has often been solved for active redundancy with nonrepairable systems, or cold-standby redundancy with repairable systems. For active redundancy, the problem has been solved using dynamic programming (Fyffe *et al.*, 1968; Nakagawa and Miyazaki, 1981), integer programming (Bulfin and Liu, 1985; Gen *et al.*, 1990) and genetic algorithms (Coit and Smith, 1996; Gen and Cheng, 1997; Painton and Campbell, 1995).

With cold-standby redundancy, it has generally been assumed that component repair is possible. The problem has been modeled as a Markov chain and formulated to minimize steady-state operating costs. Solution methodologies have been presented (Agrafiotis and Tsoukalas, 1994; Gurov *et al.*, 1995; Harunuzzaman and Aldemir, 1996; Subramanian and Anantharaman, 1995; Vaurio, 1997) to allocate redundancy for various system structures, maintenance strategies and repair time distributions.

When standby redundancy is used for nonrepairable systems, the problem has received less attention. Yearout *et al.* (1986) reviewed and categorized 156 references specifically describing research in standby redundancy. They identified only one reference, by Albright and Soni (1984) which pertained to optimization of nonrepairable systems with standby redundancy. Albright and Soni assume exponential time-to-failure and one component choice per subsystem. Tillman *et al.* (1977) reviewed 144 references describing reliability optimization research. Of those 144, only 14 pertained to standby redundancy.

Messinger and Shooman (1970) describe a dynamic programming algorithm and several heuristics to allocate spare components, which could equivalently be used to allocate cold-standby redundant components. Hwang *et al.* (1971) use an integer programming problem formulation to determine optimal cold-standby redundancy levels with one component choice for each subsystem, exponential time-to-failures and perfect switching. Robinson and Neuts (1989) studied system design for nonrepairable systems with cold-standby redundancy. They examined systems designed with components that have phase-type time-to-failure distributions.

Prasad *et al.* (1999) 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. The total number of spare components can be allocated to any of the subsystems, although the component degradation processes are application specific depending on the subsystem. Their algorithm is based on sufficiency conditions for different classes of component lifetime and mission time distributions.

3. System reliability with cold-standby redundancy

The reliability of a series–parallel system with cold-standby redundancy and perfect switching is given by Equation (1). Equation (1) is a general equation that is appropriate for any component time-to-failure distribution. System reliability is the product of subsystem reliability values. Determination of subsystem reliability is presented in Appendix A.

$$R(t)=\prod_{i=1}^{s}\left(r_i(t)+\sum_{x=1}^{n_i-1}\int_0^t r_i(t-u)f_i^{(x)}(u)\mathrm{d}u\right) \quad (1)$$

where $r_i(t)$ is the reliability at time t for component used for subsystem i; and $f_i^{(x)}(t)$ is the pdf for the xth failure arrival for subsystem i, i.e., sum of x iid component failure times.

In Equation (1), subsystem reliability is the sum of n_i probabilities associated with n_i mutually exclusive events that result in successful subsystem operation for mission time t. $r_i(t)$ is the probability that no redundant components are required. The subsequent n_i-1 additive terms (in the summation) represent the mutually exclusive probabilities that there are between one and n_i-1 failures *and* a redundant component is still operating at time t. For example, the first term $(x=1)$ is the probability that the first component fails, but the second component is still operating at time t. Since, the first failure time is a random variable, it is necessary to integrate over all possible failure times from zero to t. The convolution integrals and $f_i^{(x)}(t)$ are intractable for most applicable time-to-failure distributions.

For cold-standby redundancy, a detection and switching mechanism is required to sense the presence of a failed component and to activate a redundant component, if one is available. The switch itself may fail. $\rho_i(t)$ is defined to be the reliability of the detection/switching mechanism for subsystem i. For subsystems with a single component $(n_i=1)$, no switch is required.

There are two distinct imperfect switching cases: (i) continual monitoring and detection, and (ii) detection and switching only at time of failure. For case (i), the switching/detection capability is monitoring system functionality at all times to detect a failure. In this case, the switch/detection capability can fail at any time and it is represented by a continuous function, $\rho_i(t)$. Prior to switch failure, all required switches are effective, and after the switch failure, no switches will be effective. The system does not necessarily fail when the detection/switching mechanism fails because it may not be required to detect and switch during the remainder of the mission. For case (ii), a detection/switching failure can only occur in response to an observed failure with probability ρ_i. The probability that the system fails due to detection and switching in response to the xth component failure can be modeled as a geometric random variable with probability mass function of $\rho_i^{x-1}(1-\rho_i)$.

If there is imperfect failure detection and switching, system reliability is given by the following equations for the two imperfect detection/switching cases.

Case (i): Continual monitoring and detection, $\rho_i(t)$

$$R(t)=\prod_{i=1}^{s}\left(r_i(t)+\sum_{x=1}^{n_i-1}\int_0^t \rho_i(u)r_i(t-u)f_i^{(x)}(u)\mathrm{d}u\right) \quad (2a)$$

Case (ii): Detection and switching only at time of failure, ρ_i

$$R(t)=\prod_{i=1}^{s}\left(r_i(t)+\sum_{x=1}^{n_i-1}\rho_i^x\int_0^t r_i(t-u)f_i^{(x)}(u)\mathrm{d}u\right) \quad (2b)$$

The problem formulation is further developed based on case (i), although an analogous formulation could be developed for case (ii). It is difficult to determine a closed-form version of Equation (2a). A convenient lower-bound on system reliability, $\tilde{R}(t)$, can be determined because $\rho_i(u)\geq\rho_i(t)$ for all $u\leq t$.

$$R(t)\geq\tilde{R}(t)=\prod_{i=1}^{s}\left(r_i(t)+\rho_i(t)\sum_{x=1}^{n_i-1}\int_0^t r_i(t-u)f_i^{(x)}(u)\mathrm{d}u\right) \quad (3)$$

The limit of $R(t)-\tilde{R}(t)$ is zero as $\rho_i(t)$ approaches one ($\rho_i(t)$ is generally close to 1.0 even with imperfect switching). When there is perfect switching $(\rho_i(t)=1\ \ \forall i)$, then the approximation in Equation (3) yields the exact system reliability. This approximation is used as the objective function for the system reliability maximization problem. A system reliability approximation can be expressed as a function of $\mathbf{z}$ and $\mathbf{n}$ as follows.

$$\tilde{R}(t;\mathbf{z},\mathbf{n}) = \prod_{i=1}^{s}\left(r_i(t;z_i) + \rho_i(t)\sum_{x=1}^{n_i-1}\int_0^t r_i(t-u;z_i)f_{iz_i}^{(x)}(u)\mathrm{d}u\right) \tag{4}$$

If component time-to-failure is exponential, then a closed-form version of Equation (4) can be expressed by considering the occurrences of subsystem failure as a homogeneous Poisson process prior to the n_ith failure. In this case, subsystem reliability is the probability that there are strictly less than n_i failures, which is Poisson distributed. However, an exponential assumption is too restrictive for most actual engineering design problems, which are likely to use some mechanical or electro-mechanical components with increasing hazard functions.

Another tractable form of the system approximation (Equation (4)) can be determined if component time-to-failure is distributed according to an Erlang distribution, i.e., gamma distribution with shape parameter restricted to positive integers. If times-to-failure, T_{ij}, for available components are distributed according to the Erlang distribution with parameters λ_{ij} and k_{ij}, then component reliability and probability density function are given by,

$$r_i(t;j) = \mathrm{e}^{-\lambda_{ij}t}\sum_{l=0}^{k_{ij}-1}\frac{(\lambda_{ij}t)^l}{l!}, \quad f_{ij}(t) = \frac{\lambda_{ij}{}^{k_{ij}}t^{k_{ij}-1}\mathrm{e}^{-\lambda_{ij}t}}{\Gamma(k_{ij})}$$

Consider $N_i(t)$ as the number of component failures for subsystem i from time zero to t. For $N_i(t) < n_i$, $N_i(t)$ can be described by a renewal process with Erlang interarrival times. An Erlang random variable, with parameters λ_{ij} and k_{ij}, is equivalent to the sum of k_{ij} iid exponential random variables with parameter λ_{ij}. Now consider $N_i'(t)$ as a Poisson process with parameter λ_{ij}. Then, $\Pr\{N_i(t) < x\} = \Pr\{N_i'(t) < k_{ij}x\}$, and,

$$\begin{aligned}
&\int_0^t r_i(t-u;j)f_{ij}^{(x)}(u)\mathrm{d}u \\
&= \Pr\{T_1 + T_2 + \cdots + T_x + T_{x+1} > t \,|\, T_1 + T_2 + \cdots + T_x < t\} \\
&= \Pr\{N_i(t) = x\} \\
&= \Pr\{x \le N_i(t) < x+1\} \\
&= \Pr\{k_{ij}x \le N_i'(t) < k_{ij}(x+1)\} \\
&= \sum_{l=k_{ij}x}^{k_{ij}(x+1)-1}\mathrm{e}^{-\lambda_{ij}t}\frac{(\lambda_{ij}t)^l}{l!}.
\end{aligned} \tag{5}$$

$$\sum_{x=1}^{n_i-1}\int_0^t r_i(t-u;j)f_{ij}^{(x)}(u)\mathrm{d}u = \sum_{l=k_{ij}}^{k_{ij}n_i-1}\mathrm{e}^{-\lambda_{ij}t}\frac{(\lambda_{ij}t)^l}{l!}. \tag{6}$$

The results presented as Equations (5) and (6) can be used with the approximation from Equation (4) to approximate system reliability as,

$$\tilde{R}(t;\mathbf{z},\mathbf{n}) = \prod_{i=1}^{s}\left[\mathrm{e}^{-\lambda_{iz_i}t}\left(\sum_{l=0}^{k_{iz_i}-1}\frac{(\lambda_{iz_i}t)^l}{l!} + \rho_i(t)\sum_{l=k_{iz_i}}^{k_{iz_i}n_i-1}\frac{(\lambda_{iz_i}t)^l}{l!}\right)\right] \tag{7}$$

The Erlang distribution is appropriate for hardware reliability and provides greater flexibility and realism for engineering design problems. Example Erlang hazard functions are presented in Fig. 1 for an arbitrary fixed scale parameter and various shape parameters. A wide variety of different increasing hazard functions can be modeled with this distribution. Additionally, when k_{ij} equals one, the Erlang distribution is equivalent to the exponential distribution. The Erlang distribution offers superior characteristics than the exponential distribution for modeling component time-to-failure.

4. Solution methodology

A solution approach was developed to maximize system reliability by transforming the problem and defining new decision variables to yield an equivalent problem. The problem was transformed by taking the logarithm of Equation (7) and by defining new 0–1 decision variables, y_{ijp}. This linearizes the problem and allows for the use of integer programming algorithms.

Problem P2:

$$\max\ \ln[\tilde{R}(t;\mathbf{z},\mathbf{n})] = \sum_{i=1}^{s}\sum_{j=1}^{m_i}\sum_{p=1}^{n_{\max}}\gamma_{ijp}y_{ijp}$$

subject to

$$\sum_{i=1}^{s}\sum_{j=1}^{m_i}\sum_{p=1}^{n_{\max}}\alpha_{ijp}y_{ijp} \le C$$

$$\sum_{i=1}^{s}\sum_{j=1}^{m_i}\sum_{p=1}^{n_{\max}}\beta_{ijp}y_{ijp} \le W$$

$$\sum_{j=1}^{m_i}\sum_{p=1}^{n_{\max}}y_{ijp} = 1 \quad \forall\, i$$

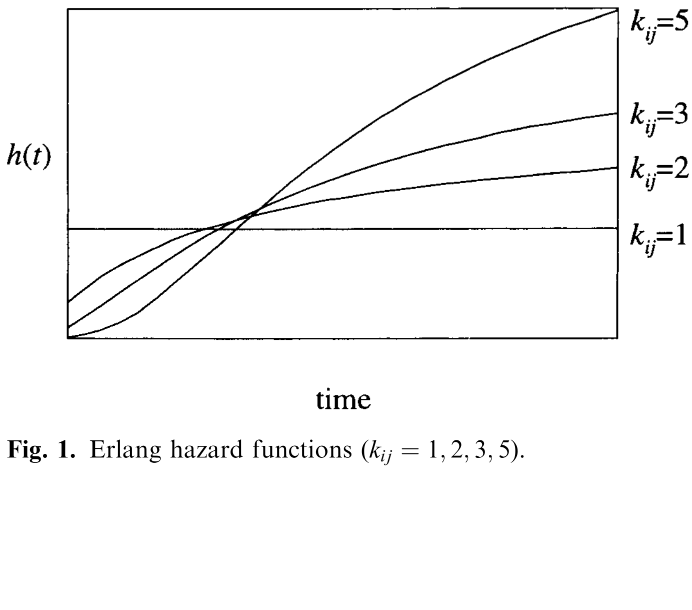

Fig. 1. Erlang hazard functions ($k_{ij} = 1, 2, 3, 5$).

$$y_{ijp} \in \{0, 1\},$$

$$y_{ijp} = \begin{cases} 1, & \text{if } n_i = p \text{ and } z_i = j, \\ 0, & \text{otherwise.} \end{cases}$$

γ_{ijp}, α_{ijp} and γ_{ijp} are constants defined as follows.

For $p = 1$,

$$\gamma_{ijp} = -\lambda_{ij}t + \ln\left(\sum_{l=0}^{k_{ij}-1} \frac{(\lambda_{ij}t)^l}{l!}\right). \tag{8}$$

$$\alpha_{ijp} = c_i(j), \quad \beta_{ijp} = w_i(j). \tag{9}$$

For $1 < p \leq n_{\max}$,

$$\gamma_{ijp} = -\lambda_{ij}t + \ln\left(\sum_{l=0}^{k_{ij}-1} \frac{(\lambda_{ij}t)^l}{l!} + \rho_i(t) \sum_{l=k_{ij}}^{k_{ij}p-1} \frac{(\lambda_{ij}t)^l}{l!}\right). \tag{10}$$

$$\alpha_{ijp} = pc_i(j) + c_{s,i}, \quad \beta_{ijp} = pw_i(j) + w_{s,i}. \tag{11}$$

Solution of Problem P2 provides solutions to the original problem formulation, Problem P1. The constraints in Problem P2 are functionally the same as in Problem P1 except that new constraints were needed so that only one y_{ijp} is equal to one for each subsystem. Maximization of the logarithm of system reliability in Problem P2 also maximizes system reliability, as in Problem P1. Functionally, the only difference is that the approximation for system reliability is used in Problem P2. Of course, the approximation provides the exact reliability when perfect switching is assumed, as has been done by most other researchers.

α_{ijp}, β_{ijp} and γ_{ijp} are expressed entirely as a function of specified component and problem parameters. Any problem where available components have known Erlang time-to-failure parameters and cost and weight measures will yield a unique set of α_{ijp}, β_{ijp} and γ_{ijp} values. The switch reliability, cost and weight must also be known, and system mission time, t, must be specified. The switch reliabilities, $\rho_i(t)$, can be from any parametric distribution or estimated nonparametrically. This problem has been formulated with two constraints but it can be readily expanded to accommodate additional linear constraints.

Problem P2 is linear and in the form of a standard 0–1 integer program. Optimal solutions can be found using standard algorithms. There are $n_{\max} \times \sum_{i=1}^{s} m_i$ decision variables. In the optimal solution, there will be exactly s y_{ijp} variables equal to one and the remainder will be equal to zero.

The optimal design can be interpreted directly from the final solution. For example consider a system with four subsystems ($s = 4$) and a final solution given by $y_{134} = y_{213} = y_{322} = y_{412} = 1$ and all other y_{ijp} equal to zero. Then, the optimal solution (given in Fig. 2) uses four of the third available component for subsystem one, three of the first available component for subsystem 2, and so on.

For very large problems, the number of variables and/or constraints may become prohibitive to solve the problem using integer programming. In this case, it may be necessary to use other approaches to more readily solve very large problems. The use of surrogate constraints methods (Bulfin and Liu, 1985; Nakagawa and Miyazaki, 1981) is one viable option to iteratively solve a series of simpler problems instead of one large problem. The genetic algorithm proposed by Coit and Smith (1996) can also be used to produce solutions to the problem, although it cannot guarantee optimality.

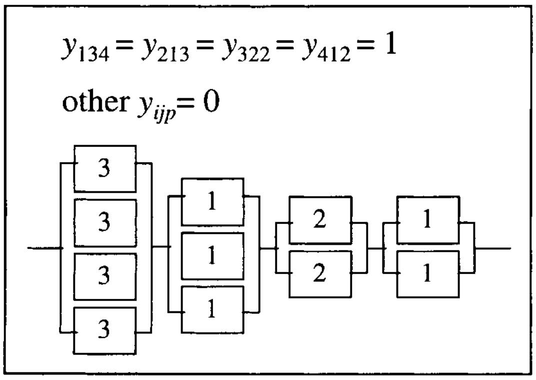

Fig. 2. Example solution.

The redundancy allocation problem is sometimes formulated to minimize system cost given a constraint for system reliability (a reliability requirement or minimal acceptable reliability). Problem P2 can be readily reformulated to accommodate this case. System cost becomes the objective function and the following constraint is added where R is the system reliability requirement.

$$\ln[\tilde{R}(t; \mathbf{z}, \mathbf{n})] = \sum_{i=1}^{s} \sum_{j=1}^{m_i} \sum_{p=1}^{n_{\max}} \gamma_{ijp} y_{ijp} \geq \ln[R].$$

5. Illustrative example

A system reliability optimization example is provided to demonstrate the methodology. The example has been adapted from the example provided by Fyffe *et al.* (1968) where system reliability was maximized using active redundancy. The system is designed with 14 subsystems. For each subsystem, there are three or four component choices. Component cost, weight and Erlang distribution parameters are provided in Table 1. The objective is to maximize system reliability at a time of 100 hours given constraints for system cost ($C = 130$) and system weight ($W = 170$). The system uses cold-standby redundancy and the reliability of a switch (at 100 hours) is 0.99 for each subsystem. The cost and weight of the switches are negligible. The maximum number of components within a subsystem has been defined to be six ($n_{\max} = 6$).

All example parameters are the same as those used by Fyffe *et al.* (1968) except the Erlang parameters, switch reliability and mission time. Fyffe *et al.* (1968) only

Table 1. Component data for example

i	*Choice 1 (j = 1)*				*Choice 2 (j = 2)*				*Choice 3 (j = 3)*				*Choice 4 (j = 4)*			
	λ_{ij}	k_{ij}	$c_i(j)$	$w_i(j)$	λ_{ij}	k_{ij}	$c_i(j)$	$w_i(j)$	λ_{ij}	k_{ij}	$c_i(j)$	$w_i(j)$	λ_{ij}	k_{ij}	$c_i(j)$	$w_i(j)$
1	0.005 32	2	1	3	0.000 726	1	1	4	0.004 99	2	2	2	0.008 18	3	2	5
2	0.008 18	3	2	8	0.000 619	1	1	10	0.004 31	2	1	9	*			
3	0.0133	3	2	7	0.0110	3	3	5	0.0124	3	1	6	0.004 66	2	4	4
4	0.007 41	2	3	5	0.0124	3	4	6	0.006 83	2	5	4	*			
5	0.000 619	1	2	4	0.004 31	2	2	3	0.008 18	3	3	5	*			
6	0.004 36	3	3	5	0.005 67	3	3	4	0.002 68	2	2	5	0.000 408	1	2	4
7	0.0105	3	4	7	0.004 66	2	4	8	0.003 94	2	5	9	*			
8	0.0150	3	3	4	0.001 05	1	5	7	0.0105	3	6	6	*			
9	0.002 68	2	2	8	0.000 101	1	3	9	0.000 408	1	4	7	0.000 943	1	3	8
10	0.0141	3	4	6	0.006 83	2	4	5	0.001 05	1	5	6	*			
11	0.003 94	2	3	5	0.003 55	2	4	6	0.003 14	2	5	6	*			
12	0.002 36	1	2	4	0.007 69	2	3	5	0.0133	3	4	6	0.0110	3	5	7
13	0.002 15	2	2	5	0.004 36	3	3	5	0.006 65	3	2	6	*			
14	0.0110	3	4	6	0.000 834	1	4	7	0.003 55	2	5	6	0.004 36	3	6	9

presented reliability values for some implicit, unstated mission time. A mission time of 100 hours was chosen. Then, Erlang shape parameters (k_{ij}) were randomly selected from 1, 2, 3 or 4, ranging from constant hazard function to increasing hazard functions. λ_{ij} was then computed so that the component reliability values would be the same as those presented by Fyffe *et al.* (1968).

α_{ijp}, β_{ijp} and γ_{ijp} values were computed using Equations (8)–(11) and the information in Table 1. This resulted in a linear formulation in the form of Problem P2. For this problem, there are 288 0–1 decision variables in the reformulated problem. The number of prospective solutions to the problem is larger than 2.1×10^{18}. The problem was then solved on a Personal Computer using readily available linear programming software (Hyper-LINDO).

The optimal solution is given in Table 2. It corresponds to a system with system reliability of 0.9863, a system cost of 123 and a system weight of 170. For comparative purposes, Table 2 also presents the optimal solution to the problem with active redundancy.

Interestingly, for four of the 14 subsystems (8, 9, 11 and 12), the optimal solution does not even involve the same component selection. For two of the subsystems (8 and 12), there is a different number of redundant components. These results are further evidence that optimization algorithms specifically for standby redundancy were needed.

Table 2. Example results

i	*Cold-standby*		*Active (Fyffe et al., 1968)*	
	z_i	n_i	z_i	n_i
1	3	3	3	3
2	1	2	1	2
3	4	3	4	3
4	3	3	3	3
5	2	3	2	3
6	2	2	2	2
7	1	2	1	2
8	3	2	1	4
9	2	2	3	2
10	2	3	2	3
11	3	2	1	2
12	4	2	1	4
13	2	2	2	2
14	3	2	3	2

6. Conclusions

Maximization of system reliability was performed when cold-standby is used as a redundancy strategy. This formulation is more general than other available formulations of the problem and explicitly considers nonconstant component hazard functions, multiple component choices for each subsystem and imperfect switching. Solutions were obtained by integer programming by the development of an equivalent problem formulation for Erlang distributed component time-to-failure. The example clearly indicates that the optimal design for cold-standby differs from the active standby case.

Acknowledgement

David Coit's work was supported by NSF CAREER grant DMII-9874716.

References

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

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

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

Chern, M.S. (1992) On the computational complexity of reliability redundancy allocation in a series system. *Operations Research Letters*, **11**, 309–315.

Coit, D.W. and Smith, A.E. (1996) Reliability optimization of series-parallel systems using a genetic algorithm. *IEEE Transactions on Reliability*, **45**(2), 254–260.

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

Gen, M. and Cheng, R. (1997) *Genetic Algorithms and Engineering Design*, John Wiley and Sons, New York, NY.

Gen, M., Ida, K. and Lee, J.U. (1990) 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**, 88–96.

Gurov, S.V. and Utkin, L.V. (1996) Cold standby systems with imperfect and non-instantaneous switch-over mechanism. *Microelectronics and Reliability*, **36**(10), 1425–1438.

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

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

Hwang, C.-L., Fan, L.T., Tillman, F.A. and Kumar, S. (1971) Optimization of life support system reliability by an integer programming method. *AIIE Transactions*, **3**, 229–238.

Kumar, S., Kumar, D. and Mehta, N.P. (1996) Behavioral analysis of shell gasification and carbon recovery process in a urea fertilizer plant. *Microelectronics and Reliability*, **36**(4), 671–673.

Messinger, M. and Shooman, M. (1970) Technique for optimum spares allocation: a tutorial review. *IEEE Transactions on Reliability*, **R-19**(4), 156–166.

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

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

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

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

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

Shankar, B.K. and Gururajan, M. (1993) Two-unit cold standby system with imperfect repair and excessive availability period. *Microelectronics and Reliability*, **33**(4), 509–512.

Sinaki, G. (1994) Ultra-reliable fault tolerant inertial reference unit for spacecraft, in *Proceedings of the Annual Rocky Mountain Guidance and Control Conference*, Univelt Inc., San Diego, CA. pp. 239–248.

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

Tillman, F.A., Hwang, C.-L. and Kuo, W. (1977) Optimization techniques for system reliability with redundancy – a review. *IEEE Transactions on Reliability*, **R-26**(3), 148–155.

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

Yearout, R.D., Reddy, P. and Grosh, D.L. (1986) An algorithmic approach to increased reliability through standby redundancy. *IEEE Transactions on Reliability*, **R-35**, 285–292.

Appendix A

Determination of subsystem reliability

A subsystem has one principal operating component and $n-1$ cold-standby redundant components. When one component fails, the redundant component is activated. For cold-standby redundancy, the redundant component has not been exposed to stress, and therefore, a redundant component (if one is available), has no probability of failure prior to being activated. T_i is defined as the ith iid component failure time and S_i is the sum of the first i component failure times.

Reliability of a single subsystem with perfect switching is equal to the probability that a component is still operating at time t.

$$R(t) = \Pr\{S_n > t\} = \Pr\{T_1 + T_2 + \cdots + T_n > t\},$$

$$\begin{aligned} R(t) &= \Pr\{T_1 > t\} + \Pr\{T_1 + T_2 > t \cap T_1 < t\} \\ &\quad + \Pr\{T_1 + T_2 + T_3 > t \cap T_1 + T_2 < t\} \\ &\quad + \cdots + \Pr\{T_1 + T_2 + \cdots + T_n \\ &\quad > t \cap T_1 + T_2 + \cdots + T_{n-1} < t\}. \end{aligned}$$

$f_{S_j}(t)$ is the probability density function of the sum of j iid component failure times; $r(t)$ is the reliability for a component at time, t, and $f_{S_1}(t) = f_{T_1}(t)$.

$$\begin{aligned} R(t) = r(t) &+ \int_0^\infty \Pr\{T_2 > t - T_1 \cap T_1 < t | T_1 = u\} f_{T_1}(u)\mathrm{d}u \\ &+ \int_0^\infty \Pr\{T_3 > t - S_2 \cap S_2 < t | S_2 = u\} f_{S_2}(u)\mathrm{d}u + \cdots \\ &+ \int_0^\infty \Pr\{T_n > t - S_{n-1} \cap S_{n-1} < t | S_{n-1} = u\} f_{S_{n-1}}(u)\mathrm{d}u. \end{aligned}$$

$$\begin{aligned} R(t) = r(t) &+ \int_0^t \Pr\{T_2 > t - u\} f_{T_1}(u)\mathrm{d}u \\ &+ \int_0^t \Pr\{T_3 > t - u\} f_{S_2}(u)\mathrm{d}u \\ &+ \cdots + \int_0^t \Pr\{T_n > t - u\} f_{S_{n-1}}(u)\mathrm{d}u. \end{aligned}$$

$$R(t) = r(t) + \int_0^t r(t-u) f_{T_1}(u)\mathrm{d}u + \int_0^t r(t-u) f_{S_2}(u)\mathrm{d}u$$
$$+ \cdots + \int_0^t r(t-u) f_{S_{n-1}}(u)\mathrm{d}u$$
$$= r(t) + \sum_{j=1}^{n-1} \int_0^t r(t-u) f^{(j)}(u)\mathrm{d}u, \quad f^{(j)}(u) = f_{S_j}(u).$$

Biography

David W. Coit is an Assistant Professor in the Industrial Engineering Department at Rutgers University. He received a B.S. degree in Mechanical Engineering from Cornell University, an MBA from Rensselaer Polytechnic Institute and M.S. and Ph.D. degrees in Industrial Engineering from the University of Pittsburgh. From 1980 to 1992, he was employed at IIT Research Institute (IITRI), Rome NY. From 1980 to 1988, he was a reliability analyst and project manager at IITRI. From 1988 to 1992, he was the Manager of Engineering at IITRIs Assurance Technology Center. His current research involves reliability prediction and optimization, and stochastic optimization techniques. He is a member of IIE, IEEE and INFORMS.

Contributed by the Reliability Modeling and Optimization Department.