International Journal of Reliability, Quality and Safety Engineering
Vol. 7, No. 2 (2000) 129–142

SYSTEM RELIABILITY OPTIMIZATION WITH k-out-of-n SUBSYSTEMS

DAVID W. COIT* and JIACHEN LIU
Department of Industrial Engineering
Rutgers University, Piscataway, NJ 08854, USA
*E-mail: *coit@rci.rutgers.edu*

Received 5 February 1999
Revised 4 February 2000

Optimal solutions to the redundancy allocation problem are determined for systems designed with multiple k-out-of-n subsystems in series. The objective is to select the components and redundancy levels to maximize system reliability given system-level constraints. The individual subsystems may use either active or cold-standby redundancy, or they may require no redundancy. Previously, optimization methods for this problem either pertained to k-out-of-n systems consisting of a single subsystem or to series–parallel systems ($k = 1$). Additionally, it had generally been assumed that only active redundancy was to be used. In practice design problems can vary appreciably from these restrictions and the design process may consider more complex system configurations. Unfortunately, available optimization algorithms are inadequate for many of these design problems. The methodology presented here is specifically developed to accommodate the case with k-out-of-n subsystems. Optimal solutions to the problem are found by an equivalent problem formulation and integer programming. The methodology is demonstrated on a well-known test problem with interesting results. The availability of this tool fills a void and should result in more reliable and cost-effective engineering designs.

Keywords: Redundancy Allocation Problem; Reliability Optimization; k-out-of-n Reliability.

1. Introduction

A new problem formulation and solution method is presented to determine optimal system design configurations when a system design includes multiple k-out-of-n subsystems that are designed with either active or cold-standby redundancy. The redundancy allocation problem has previously been analyzed for many different system structures, objective functions and time-to-failure distributions. Generally, the problem domain has been limited to series–parallel systems with active redundancy or k-out-of-n systems consisting of a single subsystem. The redundancy allocation problem involves the selection of components from among discrete choices and the determination of a system-level configuration to maximize system reliability given constraints on the system. Formulation of the problem to accommodate k-out-of-n

subsystems and different redundancy options offers enhanced capabilities. More engineering design problems can be analyzed, thereby providing a better tool for designers and reliability analysts.

When formulated to consider only active redundancy and restricted to series–parallel systems, efficient optimization algorithms have been developed to determine optimal designs using dynamic programming or integer programming. These existing solution methodologies are not applicable when the system design may involve active and cold-standby redundancy in different parts of the design and when there may be subsystems where more than one component is required for the system to operate ($k_i > 1$).

A methodology is presented and demonstrated here to determine optimal solutions to this more general form of the redundancy allocation problem. The problem is solved by developing an equivalent problem formulation and the application of zero–one integer programming methods. For this problem formulation, component time-to-failure is distributed according to the exponential distribution.

1.1. *Notation*

$R(t)$	system reliability at time t depending on design vectors $\mathbf{z}$ and $\mathbf{n}$
$\mathbf{n}$	$= (n_1, n_2, \ldots, n_s)$
n_i	number of components used in subsystem i
$n_{\max,i}$	upper bound for $n_i (n_i \leq n_{\max,i} \forall\, i)$
$\mathbf{z}$	$= (z_1, z_2, \ldots, z_s)$
z_i	index of component choice used for subsystem i, $z_i \in \{1, 2, \ldots, m_i\}$
m_i	number of available component choices for subsystem i
$\mathbf{k}$	$= (k_1, k_2, \ldots, k_s)$
k_i	minimum number of operating components for subsystem i
s	number of subsystems
T_{ij}	time-to-failure of the jth available component for subsystem i
λ_{ij}	component failure rate (exponential distribution parameter) for the jth available component for subsystem i, $f_{ij}(t) = \lambda_{ij} \exp(-\lambda_{ij} t)$
C, W	system-level constraint limits for cost and weight
c_{ij}, w_{ij}	cost and weight for the jth available component for subsystem i
t	mission time (fixed)

2. Redundancy Allocation Problem

The redundancy allocation problem has been studied in great detail as an efficient means to select sound design configurations. There are two general problem classifications. For the first, there are discrete component choices with known characteristics (cost, weight, etc.). The objective for this combinatorial problem is to select which components to use and the corresponding redundancy levels. For the second general problem class, component reliability (or an exponential distribution parameter) is treated as a design variable and component cost is a predefined increasing function of component reliability. This paper pertains to the first type of problem. It is a difficult combinatorial problem which has been shown to be NP-hard.[1]

The design of new products involves the specification of performance requirements, the 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 within some specified budget, numerous design alternatives must be considered, resulting in a complex combinatorial optimization problem.

The redundancy allocation problem studied here pertains to a system of s independent k-out-of-n subsystems in series. Each subsystem may be designed with one component ($k_i = n_i = 1$), parallel redundant components ($k_i = 1 < n_i$), or components in a k-out-of-n configuration. Figure 1 presents a typical example of the system configuration being considered. Within a particular subsystem, the redundant components are either active or in a standby mode. For a k-out-of-n subsystem with cold-standby redundancy, k of the original n components are fully active and susceptible to failure. As a component fails, it is detected and one of the redundant components is activated. It is assumed that the system has perfect failure detection and switching.

For each subsystem, there are m_i equivalent components that may be selected. There is an unlimited supply of each of the m_i choices. Each available component has different levels of cost, weight and other characteristics. Component time-to-failure is independent and is distributed according to the exponential distribution, $T_{ij} \sim \text{EXP}(\lambda_{ij})$. There are system-level constraints on cost and weight (and others) and the objective is to select the component choices and levels of redundancy to maximize the system reliability.

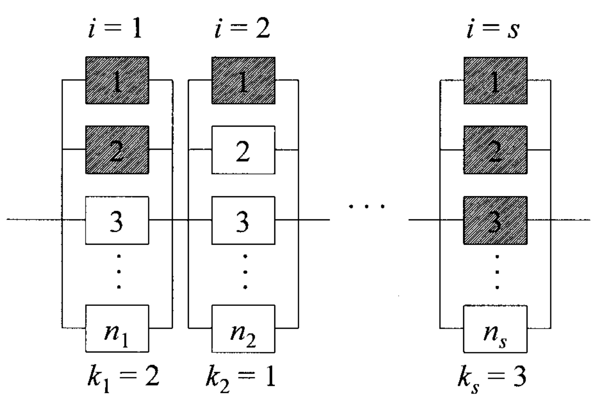

Fig. 1. System with k-out-of-n subsystems.

A general problem formulation to maximize system reliability is defined as Problem P1 and presented as follows:

Problem P1:

$$
\begin{aligned}
\max \quad & R(t)\,, \\
\text{s.t.} \quad & \sum_{i=1}^{s} c_{iz_i} n_i \le C\,, \\
& \sum_{i=1}^{s} w_{iz_i} n_i \le W\,, \\
& n_i \in \{k_i, k_i+1, k_i+2, \ldots, n_{\max,i}\}\,, \\
& z_i \in \{1, 2, \ldots, m_i\}\,.
\end{aligned}
$$

The problem has most often been formulated considering only series–parallel systems ($k_i = 1$) or k-out-of-n systems consisting of a single subsystem ($s = 1$). There has only been very limited research directed towards optimization of systems with cold-standby redundancy and k-out-of-n subsystems. Previous optimization methods also require the assumption that only one type of redundancy will be used throughout the system. That is, the system design will use only active redundancy or only cold-standby redundancy throughout the subsystem.

In practice, system design may involve some subsystems that use cold-standby and other subsystems that use active redundancy. The choice of redundancy type is often dictated by the particular failure mechanisms and technology of the part types being used. If it is not efficient or possible to detect a failed component and activate a redundant component (without significantly interrupting service), then active redundancy will be used. Furthermore, there may be a time lag associated with the switching that is unacceptable for some safety critical systems. Alternatively, if power, cost or space restrictions preclude the simultaneous operation of redundant components, then cold-standby will be used. If either redundancy strategy is possible for a particular subsystem, then cold-standby redundancy will generally be preferable when the failure detection and switching capability is highly reliable. In practice, it is unrealistic to limit a system design to exclusively one redundancy strategy.

There has been significant research activity devoted to different forms of the redundancy optimization problem. Kuo *et al.*[2] provides a comprehensive overview of system reliability optimization problem formulations and solution methodologies. For a single subsystem, there have been research findings describing optimal configurations for the k-out-of-n subsystem. For multiple subsystems in series, reliability optimization research has largely been limited to series–parallel systems. This is equivalent to k-out-of-n subsystems with k_i always set equal to one.

Pham[3] describes an optimal design method for k-out-of-n redundant systems. The optimization problems are formulated and solved to minimize expected total cost of k-out-of-n systems. The optimal number of units can be obtained for a system consisting of a single k-out-of-n subsystem. Similarly, Pham[4] also demonstrates methods to optimally determine the most economical number of components in k-out-of-n subsystems. In this case, methods are presented to determine the optimal values of k (for fixed n) and n (for fixed k) to minimize the mean total cost of k-out-of-n subsystems. Suich and Patterson[5] have also derived several cost models for a single k-out-of-n subsystem and presented a numerical solution for k and n that minimizes the total cost of a k-out-of-n system.

Pham and Malon[6] consider the problem of achieving optimal system size n, and a threshold k for k-out-of-n subsystem with competing failure modes. Methods to determine optimal k (given n); optimal n (given k); and optimal k, n are described. In each case, only one subsystem is considered.

Bai, Yun and Chung[7] describe the problem of determining the optimal number of redundant units in k-out-of-n subsystems with common-cause failures (CCFs). The mean cost rate is obtained, and the number of redundant units minimizing the mean cost rate is shown to be finite and unique. Again, only one subsystem is considered.

Chiang and Chiang[8] considered a relayed mobile communication system. Such a system can be considered as a consecutive k-out-of-n subsystem. They present equations for computing the mean number of stations needed for a successful relay and studied the optimal choice of k to minimize the mean number. Hwang and Shi[9] demonstrate that it is always better to replace a consecutive k-out-of-n subsystem by a consecutive l-out-of-n line but with k redundancy. The problem of choosing an optimal k still has no closed-form solution but is more tractable than the original problem studied by Chiang and Chiang. There has been other research pertaining to optimization problems for k-out-of-n subsystems.[10–14] For each of these, only one subsystem is considered.

Kuo *et al.*[2] provide the most comprehensive description of system reliability optimization. There are generally two classes of problems. The first is where component reliability is a continuous design variable and cost is expressed as an explicit function of component reliability. The second is where components are selected from discrete choices. Both problems are difficult and many solution strategies have been proposed for both. Generally, the first problem classification has been analyzed using problem-specific variants of nonlinear programming, often relying on the problems special structure. The second class has been addressed with integer programming, dynamic programming and heuristic approaches.

For multiple subsystems in series, the problem has been formulated to maximize system reliability and solved using dynamic programming[15,16] integer programming[17,18] and genetic algorithms.[19–21] Systems with k-out-of-n subsystems have not received significant attention, although Misra and Sharma[22] and Coit and Smith[21] do consider these systems.

Misra and Sharma[22] consider the most general case of system designs with mixed redundancies. They use a combination of direct search methods and random search.

Coit and Smith[21] develop a problem-specific genetic algorithm (GA) to analyze series–parallel systems and to determine the design configuration to maximize reliability when there are multiple component choices available for each k-out-of-n subsystem. Only active redundancy is considered. While this GA has repeatedly been demonstrated to yield good solutions, it cannot guarantee optimality, and furthermore, the final solution may be dependent on the initial population that is randomly selected.

A similar problem has been addressed by Nakashima and Yamato.[23] They consider improving the reliability of multivalue output systems by the use of n redundant subsystems. The mean loss of multivalue output systems with multichannels can be minimized by adopting k-out-of-n redundancy for each channel. The optimal k depends on the probability and loss matrices, but it can be specified in some special cases. Only active redundancy is considered.

3. System Reliability with *k*-out-of-*n* Subsystems

A system design may employ active redundancy for some subsystems and cold-standby redundancy for other subsystems. The choice of redundancy strategy may be based either on reliability benefits or practical design constraints. The choice of redundancy strategy may be dictated by the practicality of sensing a failure and activating a component in cold-standby. For other subsystems, there may already been a decision to use a single component in series. System reliability is a function of the design variables represented by $\mathbf{z}$ and $\mathbf{n}$, given a mission time t, and minimum component numbers $\mathbf{k}$. For notational brevity, system reliability will simply be expressed as $R(t)$, i.e., $R(t) = R(\mathbf{z}, \mathbf{n}|t, \mathbf{k})$.

The reliability of a system, with both active and cold-standby redundancy is given in Eq. (1). $R_i(t, z_i, n_i, k_i)$ is the reliability of the ith subsystem.

$$R(t) = \prod_{i=1}^{s} R_i(t, z_i, n_i, k_i)\,. \tag{1}$$

A k-out-of-n subsystem with cold-standby redundancy requires k_i components to be operating, and thus, they are prone to failure. The remaining $n_i - k_i$ redundant components are not exposed to operating stress and not prone to failure. As components do fail, the failures are sensed and a redundant component is activated. Prior to subsystem failure, there are always a collective k_i operating components. The subsystem failure process can be considered as a Poisson process with rate $\lambda_{ij}k_i$. The expected time to failure of the ith subsystem can be determined by summing the expected interarrival times of the first $n_i - k_i + 1$ failure events of the Poisson process. If X is defined as the Poisson process interarrival time, then

$$MTTF_i = (n_i - k_i + 1)E[X] = (n_i - k_i + 1)\frac{1}{\lambda_{ij}k_i} = \frac{n_i - k_i + 1}{k_i}\,E[T_{ij}]\,, \tag{2}$$

where $MTTF_i$ = mean time to failure of the ith subsystem with cold standby redundancy.

Subsystem reliability for cold-standby subsystems is the probability that there are less than or equal to $n_i - k_i$ failures observed until time t. This probability can now be computed from the Poisson distribution. Subsystem reliability for active redundancy is computed using standard techniques. System reliability is now given by the following equation:

$$R(t) = \prod_{i \in A} \sum_{l=k_i}^{n_i} \binom{n_i}{l} (\exp(-\lambda_{i,z_i} t))^l (1 - \exp(-\lambda_{i,z_i} t))^{n_i - l}$$

$$\times \prod_{i \in S} \exp(-\lambda_{i,z_i} k_i t) \sum_{l=0}^{n_i - k_i} \frac{(\lambda_{i,z_i} k_i t)^l}{l!} \tag{3}$$

A = set of all subsystems using active redundancy
S = set of all subsystems using cold-standby redundancy

4. Solution Methodology

A methodology for maximization of system reliability was developed by transforming the problem and defining new decision variables to yield an equivalent zero–one integer programming problem. This approach is based on the research and algorithms from Misra and Sharma.[24] The problem was transformed by taking the logarithm of Eq. (3) and by defining new 0–1 decision variables y_{ijp}. This linearizes the problem and allows for the use of integer programming algorithms. y_{ijp} is defined as follows:

$$y_{ijp} = \begin{cases} 1\,, & p \text{ of the } j\text{th component is used for subsystem } i\,, \\ 0\,, & \text{otherwise}\,. \end{cases}$$

The following methodology is used to determine an optimal solution to Problem P1. The resulting solution maximizes system reliability.

Step 1. Partition each subsystem i into sets A, S, N.
A = set of all subsystems limited to active redundancy
S = set of all subsystems limited to cold-standby redundancy
N = set of all subsystems which will not use additional redundancy ($n_i = k_i$)

Step 2. Compute,

$$\alpha_{ijp} = c_{ij} p \quad \text{for} \quad 1 \le i \le s, 1 \le j \le m_i, k_i \le p \le n_{\max,i}$$

$$\beta_{ijp} = w_{ij} p \quad \text{for} \quad 1 \le i \le s, 1 \le j \le m_i, k_i \le p \le n_{\max,i}$$

for $i \in A$,

$$\gamma_{ijp} = \begin{cases} -\lambda_{ij} k_i t\,, & p = k_i \\ \ln\left(\sum_{l=k_i}^{n_i} \binom{n_i}{l} (\exp(-\lambda_{ij} t))^l (1 - \exp(\lambda_{ij} t))^{n_i - l}\right), & \\ & k_i < p \le n_{\max,i} \end{cases}$$

for $i \in S$,

$$\gamma_{ijp} = \begin{cases} -\lambda_{ij} k_i t\,, & p = k_i \\ -\lambda_{ij} k_i t + \ln\left(\sum_{l=0}^{n_i - k_i} \frac{(\lambda_{ij} k_i t)^l}{l!}\right), & k_i < p \le n_{\max,i} \end{cases}$$

for $i \in N$,

$$\gamma_{ijp} = -\lambda_{ij} k_i t, p = k_i$$

$$n_{\max,i} = k_i$$

Step 3. Solve the following 0–1 integer program (Problem P2) using any convenient branch-and-bound or cutting plane algorithm.

Problem P2:

$$\max \sum_{i=1}^{s} \sum_{j=1}^{m_i} \sum_{p=k}^{n_{\max,i}} \gamma_{ijp} y_{ijp}$$

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

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

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

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

Step 4. Interpret the results. There will be exactly $s\, y_{ijp}$ values equal to one in the optimal solution and the remainder will be equal to zero.

(1) For $i \in A$ and $y_{ijp} = 1$, the optimal system design uses p of the jth available component choice in active redundancy for subsystem i.
(2) For $i \in S$ and $y_{ijp} = 1$, the optimal system design uses p of the jth available component choice in cold-standby redundancy for subsystem i (i.e., k_i active component and $p - k_i$ in cold-standby).
(3) For $i \in N$ and $y_{ijp} = 1$, the optimal system design uses one of the jth available component choice.

To solve the problem, the user is required to pre-specify t, $\mathbf{k}$ and $n_{\max,i}$ and determine sets A, S and N. These decisions should be based on mission requirements and preliminary design decisions. If it is technologically possible to assign a particular subsystem to set A, S, or N but the decision is unclear, then the subsystem should be assigned to set S unless the detection and switching capabilities are suspect. In that case, the subsystem should be assigned to A. $n_{\max,i}$ is an upper-bound for n_i. Usually, $n_{\max,i}$ can be selected based on pragmatic restrictions. Otherwise, $n_{\max,i}$ can be computed as

$$n_{\max,i} = \max_j \min \left[\frac{C}{c_{ij}}, \frac{W}{w_{ij}} \right].$$

α_{ijp}, β_{ijp} and γ_{ijp} are expressed entirely as a function of specified component and problem parameters. Any problem with exponential component times-to-failure and known cost and weight measures will yield a unique set of α_{ijp}, β_{ijp} and γ_{ijp} values.

Problem P2 is linear and in the form of a standard 0–1 integer program with $\sum_{i=1}^{s}(n_{\max,i} - k_i + 1)m_i$ decision variables. Optimal solutions can be found using standard algorithms developed specifically for 0–1 integer programs.[25–27] There are many readily available automated packages (e.g., CPLEX, LINDO) to solve integer programs. The optimal solution is found using branch-and-bound or cutting plane methods that are based on a linear programming relaxation and successive application of the simplex algorithm.

For very large problems, the number of variables and/or constraints may become prohibitive and the capacities of available automated algorithms may be exceeded. In this case, it may be necessary to use other approaches for very large problems. Linear program complexity is largely dictated by the number of constraints. In the problem formulations presented here, there were only two constraints, but there may be many more in practice. The use of surrogate constraints methods[16,17] is one viable option. This involves the combination of all constraints into a single surrogate constraint creating a surrogate problem that can be readily solved using special algorithms for the knapsack problem. Then, a series of surrogate problems is solved with different constraint combinations to determine the optimal solution to the original problem.

If there are many constraints but only one component choice per subsystem ($m_i = 1$), then the specialized integer programming algorithm proposed by Misra and Sharma[28] can be used to solve Problem P2. This algorithm is designed specifically for reliability optimization problems and exploits the special structure of the problem for coherent systems.

If the problem is sufficiently large that none of these approaches are viable, then the genetic algorithm proposed by Coit and Smith[21] can be extended for this problem formulation. This approach acts directly on the Problem P1 formulation. It does not guarantee optimal solutions, although it has been demonstrated repeatedly to yield high-quality solutions that are optimal or near-optimal.

This problem has been formulated with two constraints but it can be readily expanded to accommodate additional linear constraints. The redundancy allocation problem can also be formulated to minimize system cost given a constraint for system reliability R (a reliability requirement or minimal acceptable reliability). System cost becomes the objective function and the following constraint is added:

$$\sum_{i=1}^{s}\sum_{j=1}^{m_i}\sum_{p=k_i}^{n_{\max,i}} \gamma_{ijp}\, y_{ijp} \geq \ln[R]\,.$$

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, Hines

Table 1. Component data for example.

Subsystem			Component Choice 1			Component Choice 2			Component Choice 3			Component Choice 4		
i	k_i	type	λ_{ij}	c_{ij}	w_{ij}	λ_{ij}	c_{ij}	w_{ij}	λ_{ij}	c_{ij}	w_{ij}	λ_{ij}	c_{ij}	w_{ij}
1	1	A	.001054	1	3	.000726	1	4	.000943	2	2	.000513	2	5
2	2	A	.000513	2	8	.000619	1	10	.000726	1	9	—		
3	1	A	.001625	2	7	.001054	3	5	.001393	1	6	.000834	4	4
4	2	A	.001863	3	5	.001393	4	6	.001625	5	4	—		
5	1	A	.000619	2	4	.000726	2	3	.000513	3	5	—		
6	2	A	.000101	3	5	.000202	3	4	.000305	2	5	.000408	2	4
7	1	A	.000943	4	7	.000834	4	8	.000619	5	9	—		
8	2	S	.002107	3	4	.001054	5	7	.000943	6	6	—		
9	3	S	.000305	2	8	.000101	3	9	.000408	4	7	.000943	3	8
10	3	S	.001863	4	6	.001625	4	5	.001054	5	6	—		
11	3	S	.000619	3	5	.000513	4	6	.000408	5	6	—		
12	1	S	.002357	2	4	.001985	3	5	.001625	4	6	.001054	5	7
13	2	S	.000202	2	5	.000101	3	5	.000305	2	6	—		
14	3	S	.001054	4	6	.000834	4	7	.000513	5	6	.000101	6	9

NOTES: A = active redundancy, S = cold-standby redundancy, units for λ_{ij} are failures/hour and Lee.[15] The system is designed with 14 subsystems. For each subsystem, there are three or four component choices. Component cost, weight and exponential distribution parameter (λ_{ij}) are provided in Table 1. The objective is to maximize system reliability at a time of 100 h given constraints for system cost ($C = 130$) and system weight ($W = 170$).

The problem was revised by randomly selecting $k_i \in \{1, 2, 3\}$ for each subsystem and by assigning each subsystem to set A or S. The original problem was for series–parallel systems (k_i=1) with exclusively active redundancy. The component exponential distribution parameters in Table 1 yield the same component reliability data (for $t = 100$) as originally presented by Fyffe, Hines and Lee.[15] For each subsystem, the selection of k_i and redundancy type has been indicated in Table 1. The maximum number of components within a subsystem has been defined to be six ($n_{\max,i} = 6$).

For this problem, there are 244 0–1 decision variables in the reformulated problem. The number of prospective unique solutions to the problem is larger than 1.6×10^{17}. 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 .4466, a system cost of 118 and a system weight of 170.

Table 2 also presents the optimal solution to the original problem with active redundancy and $k_i = 1$. The comparison in Table 2 is interesting, but it must be emphasized that by changing k_i and redundancy type, the problem is fundamentally different. For 13 of the 14 subsystems, the optimal solutions to the two problems

Table 2. Example results.

	Optimal Solution		Solution-Fyffe[15]	
i	z_i	n_i	z_i	n_i
1	3	2	3	3
2	1	2	1	2
3	4	1	4	3
4	3	3	3	3
5	2	1	2	3
6	2	2	2	2
7	2	1	1	2
8	1	3	1	4
9	3	3	3	2
10	2	4	2	3
11	1	4	1	2
12	1	2	1	4
13	2	2	2	2
14	3	4	3	2

involved the same component selection (z_i). Alternatively, the number of components used (n_i) is the same for only four of the 14 subsystems. The differences for n_i in the two optimal solutions is not unexpected. For systems with k-out-of-n subsystems, n_i must be at least as large as k_i.

6. Conclusions

A solution methodology is presented to determine optimal solutions for the redundancy allocation problem for systems consisting of multiple k-out-of-n subsystems in series. This problem had not previously been satisfactorily solved and the proposed method offers the capability to solve a greater range of engineering design problems.

Acknowledgment

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

References

1. M. S. Chern, "On the computational complexity of reliability redundancy allocation in a series system," *Operations Research Letters* **11** (1992), pp. 309–315.
2. W. Kuo, V. R. Prasad, F. A. Tillman, and C.-L. Hwang, *Fundamental and Applications of Reliability Optimization* (Cambridge University Press, Cambridge, 2000).
3. Hoang Pham, "Optimal Design of k-out-of-n Redundant Systems," *Microelectronics and Reliability* **32**(1/2) (1992), pp. 119–126.
4. Hoang Pham, "On the Optimal Design of k-out-of-n: G Subsystems," *IEEE Trans. Reliab.* **45**(2) (1996), pp. 254–260.
5. R. C. Suich and R. L. Patterson, "k-out-of-n: G systems; Some Cost Considerations," *IEEE Trans. Reliab.* **40** (1991), pp. 259–264.
6. Hoang Pham and D. M. Malon, "Optimal design of systems with competing failure modes," *IEEE Trans. Reliab.* **43**(2) (1994), pp. 251–254.
7. D. S. Bai, W. Y. Yun, and S. W. Chung, "Redundancy optimization of k-out-of-n systems with common-cause failures," *IEEE Trans. Reliab.* **40**(1) (1991), pp. 56–59.
8. D. T. Chiang and R. F. Chiang, "Relayed communication via consecutive-k-out-of-n," *IEEE Trans. Reliab.* **R-30** (1986), pp. 65–67.
9. F. K. Hwang and D. Shi, "Optimal relayed mobile communication systems," *IEEE Trans. Reliab.* **38**(4) (1989), pp. 457–459.
10. A. Lesanovsky, "Minimal size of the system with two dual modes of failure having prescribed reliability," *Microelectronics and Reliability* **33**(4) (1993), pp. 543–563.
11. A. Rushdi and K. A. AI-Hindi, "Table of the Lower Boundary of the Region of Useful Redundancy for k-out-of-n System," *Microelectronics and Reliability* **33**(7) (1993), pp. 979–992.
12. Gurov, Sergey, Utkin, Lev, Shubinsky, and B. Igor, "Optimal reliability allocation of redundant units and repair facilities by arbitrary failure and repair distributions," *Microelectronics and Reliability* **35**(12) (1995), pp. 1451–1460.
13. A. A. Charl, "Optimal redundancy of k-out-of-n: G system with two kinds of CCFs," *Microelectronics and Reliability* **34**(6) (1994), pp. 1137–1139.
14. Kalyan, Radha, Kumar, and Santosh, "Study of protean systems-redundancy optimization in consecutive k-out-of-n: F systems," *Microelectronics and Reliability* **30**(4) (1990), pp. 635–638.

15. D. E. Fyffe, W. W. Hines, and N. K. Lee, "System reliability allocation and a computational algorithm," *IEEE Trans. Reliab.* **R-17** (1968), pp. 64–69.
16. Y. Nakagawa and S. Miyazaki, "Surrogate constraints algorithm for reliability optimization problems with two constraints," *IEEE Trans. Reliab.* **R-30** (1981), pp. 175–180.
17. R. L. Bulfin and C. Y. Liu, "Optimal allocation of redundant components for large systems," *IEEE Trans. Reliab.* **R-34** (1985), pp. 241–247.
18. 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), pp. 88–96.
19. K. Ida, M. Gen, and T. Yokota, "System reliability optimization with several failure modes by genetic algorithm," *Proceedings of the 16th International Conference on Computers and Industrial Engineering* (1994), pp. 349–352.
20. L. Painton and J. Campbell, "Genetic algorithms in optimization of system reliability," *IEEE Trans. Reliab.* **44**(2) (1995), pp. 172–178.
21. D. W. Coit and A. E. Smith, "Reliability optimization of series–parallel systems using a genetic algorithm," *IEEE Trans. Reliab.* **45**(2) (1996), pp. 254–260.
22. K. B. Misra and U. Sharma, "Reliability optimization of a system by zero-one programming," *Microelectronics and Reliability* **31**(2/3) (1991), pp. 323–335.
23. K. Nakashima and K. Yamato, "On optimal redundancy of multivalue-output systems," *IEEE Trans. Reliab.* **R-36**(2) (1987).
24. K. B. Misra and U. Sharma, "Reliability optimization of a system by zero-one programming," *Microelectronics and Reliability* **12**(3) (1973), pp. 229–233.
25. G. Nemhauser and L. Wolsey, *Integer and Combinatorial Optimization* (John Wiley & Sons, 1988).
26. G. Parker and R. Rardin, *Discrete Optimization* (Academic Press, 1988).
27. H. Taha, *Integer Programming: Theory, Applications, and Computations* (Academic Press, 1975).
28. K. B. Misra and U. Sharma, "An efficient algorithm to solve integer programming problems arising in system-reliability design," *IEEE Trans. Reliab.* **40**(1) (1991), pp. 81–91.

About the Authors

David W. Coit is an assistant professor of Industrial Engineering at Rutgers University. He has a BSME from Cornell, an MBA from Rensselaer Polytechnic Institute, and M.S. and Ph.D. degrees in Industrial Engineering from the University of Pittsburgh. He previously worked for 12 years at the IIT Research Institute (IITRI), Rome, New York as a reliability engineer and engineering manager. He teaches classes in reliability, applied probability, and optimization. He is currently working on system reliability research projects as part of the Quality and Reliability Engineering (QRE) Center at Rutgers. His interests involve the prediction and optimization of system reliability when there is uncertainty about the reliability of critical components within the system. He is a member of IEEE, IIE and INFORMS.

Jiachen Liu received his B.S. and M.S. degrees in Engineering from Tsinghua University, P. R. China, in 1995 and 1998, respectively. He is currently a Ph.D. student and a research assistant of Industrial Engineering at Rutgers University. He has also worked for Honeywell Inc. His research interests include system optimization, supervisory control, and precision tracking.