# Recent Advances in Optimal Reliability Allocation[1]

Way Kuo and Rui Wan

Department of Industrial and Information Engineering, University of Tennessee, Knoxville, TN 37919, USA

## Acronyms

| | |
|---|---|
| GA | genetic algorithm |
| HGA | hybrid genetic algorithm |
| SA | simulated annealing algorithm |
| ACO | ant colony optimization |
| TS | tabu search |
| IA | immune algorithm |
| GDA | great deluge algorithm |
| CEA | cellular evolutionary approach |
| NN | neural network |
| NFT | near feasible threshold |
| UGF | universal generating function |
| MSS | multi-state system |
| *RB/i/j* | recovery block architecture that can tolerate *i* hardware and *j* software faults |
| *NVP/i/j* | N-version programming architecture that can tolerate *i* hardware and *j* software faults |
| LMSSWS | linear multi-state sliding-window system |
| LMCCS | linear multi-state consecutively connected system |
| ATN | acyclic transmission network |
| WVS | weighted voting system |
| ACCN | acyclic consecutively connected network |
| UMOSA | Ulungu multi-objective simulated annealing |
| SMOSA | Suppapitnarm multi-objective simulated annealing |

---



W. Kuo and R. Wan: *Recent Advances in Optimal Reliability Allocation*, Computational Intelligence in Reliability Engineering (SCI) **39**, 1–36 (2007)
www.springerlink.com 

| | |
|---|---|
| PSA | Pareto simulated annealing |
| PDMOSA | Pareto domination based multi-objective simulated annealing |
| WMOSA | weight based multi-objective simulated annealing |

## Notations

| | |
|---|---|
| $\mathbf{x}_j$ | number of components at subsystem $j$ |
| $\mathbf{r}_j$ | component reliability at subsystem $j$ |
| $n$ | number of subsystems in the system |
| $m$ | number of resources |
| x | $(\mathbf{x}_1,...,\mathbf{x}_n,\mathbf{r}_1,..\mathbf{r}_n)$ |
| $g_i(\mathbf{x})$ | total amount of resource $i$ required for $\mathbf{x}$ |
| $R_S$ | system reliability |
| $C_S$ | total system cost |
| $R_0$ | a specified minimum $R_S$ |
| $C_0$ | a specified minimum $C_S$ |
| $\alpha$ | system user's risk level, $0<\alpha<1$ |
| $t_{\alpha,\mathbf{x}}$ | system percentile life, $t_{\alpha,\mathbf{x}} = \inf\{t \geq 0 : R_S \leq 1-\alpha\}$ |
| $E_S$ | generalized MSS availability index |
| $E_0$ | a specified minimum $E_S$ |
| $S_1$ | the set of optimal solutions of $P_1$ |
| $S_2$ | the set of optimal solutions of $P_2$ |
| $U_i(z)$ | $U$-function of component $i$ |
| $g_{1i}$ | output performance level of component 1, $i = 1,\ldots, I$ |
| $g_{2j}$ | output performance level of component 2, $j = 1,\ldots, J$ |
| $p_{1i}$ | $\lim_{t\to\infty}[\Pr\{g_1(t) = g_{1i}\}]$ |
| $p_{2j}$ | $\lim_{t\to\infty}[\Pr\{g_2(t) = g_{2j}\}]$ |
| $\omega_s$ | $\omega(\cdot)$ of series components |
| $\omega_{s\gamma}$ | $\omega_s$ for Type-$\gamma$ MSS |
| $\omega_s^\tau$ | $\omega_s$ in mode $\tau$ |
| $\omega_p$ | $\omega(\cdot)$ of parallel components |
| $\omega_{p\gamma}$ | $\omega_p$ for Type-$\gamma$ MSS |

| | |
|---|---|
| $\omega_p^\tau$ | $\omega_p$ in mode $\tau$ |
| $G_\tau$ | output performance level of the entire MSS in mode $\tau$ |
| $W_\tau$ | system demand in mode $\tau$ |
| $F_\tau(G_\tau, W_\tau)$ | function representing the desired relation between MSS performance level and demand in mode $\tau$ |
| $\mu_{\tilde{D}}(x)$ | membership function of the fuzzy decision |
| $\mu_{\tilde{f}_i}(x)$ | membership function of $i^{\text{th}}$ fuzzy goal |
| $r_{ij}$ | reliability of component *j* in subsystem *i* |
| $c_{ij}$ | cost of component *j* in subsystem *i* |
| $w_{ij}$ | weight of component *j* in subsystem *i* |
| $\tau_{ij}$ | pheromone trail intensity of (*i*, *j*) |
| $\eta_{ij}$ | problem-specific heuristic of (*i*, *j*), $\eta_{ij} = r_{ij} / (c_{ij} + w_{ij})$ |
| $P_{ij}$ | transition probability of (*i*, *j*), $P_{ij} = \begin{cases} \dfrac{\tau_{ij}(\eta_{ij})^{0.5}}{\sum_{l=1}^{a_i} \tau_{il}(\eta_{il})^{0.5}} & j \in \text{the set of available component choices} \\ 0 & \text{otherwise} \end{cases}$ |
| **T** | MSS operation period |
| $\mathbf{W}^*$ | required MSS performance level |
| *Pry* | probability of failure from related fault between two software versions |
| *Pall* | probability of failure from related fault among all software versions due to faults in specification |

## 1.1 Introduction

Reliability has become an even greater concern in recent years because high-tech industrial processes with increasing levels of sophistication comprise most engineering systems today. Based on enhancing component reliability and providing redundancy while considering the trade-off between system performance and resources, optimal reliability design that aims to determine an optimal system-level configuration has long been an important topic in reliability engineering. Since 1960, many publications have addressed this problem using different system structures, performance measures, optimization techniques and options for reliability improvement.

Refs [45], [93] and [123] provide good literature surveys of the early work in system reliability optimization. Tillman, *et al.* [123] were the first to classify papers by system structure, problem type, and solution methods. Also described and analyzed in [123] are the advantages and shortcomings of various optimization techniques. It was during the 1970s that various heuristics were developed to solve complex system reliability problems in cases where the traditional parametric optimization techniques were insufficient. In their 2000 report, Kuo and Prasad [45] summarize the developments in optimization techniques, along with recent optimization methods such as meta-heuristics, up until that time. This chapter discusses the contributions made to the literature since the publication of [45]. The majority of recent work in this area is devoted to

- multi-state system optimization;
- percentile life employed as a system performance measure;
- multi-objective optimization;
- active and cold-standby redundancy;
- fault-tolerance mechanism;
- optimization techniques, especially ant colony algorithms and hybrid optimization methods.

Based on their system performance, reliability systems can be classified as binary-state systems or multi-state systems. A binary-state system and its components may exist in only two possible states−either working or failed. Binary system reliability models have played very important roles in practical reliability engineering. To satisfactorily describe the performance of a complex system, however, we may need more than two levels of satisfaction−for example, excellent, average, and poor [46]. For this reason, multi-state system reliability models were proposed in the 1970s, although a large portion of the work devoted to MSS optimal design has emerged since 1998. The primary task of multi-state system optimization is to define the relationship between component states and system states.

Measures of system performance are basically of four kinds: reliability, availability, mean time-to-failure and percentile life. Reliability has been widely used and thoroughly studied as the primary performance measure for non-maintained systems. For a maintained system, however, availability, which describes the percentage of time the system really functions, should be considered instead of reliability. Availability is most commonly employed as the performance measure for renewable MSS. Meanwhile, percentile life is preferred to reliability and mean time-to-failure when the system mission time is indeterminate, as in most practical cases.

Some important design principles for improving system performance are summarized in [46]. This chapter primarily reviews articles that address either the provision of redundant components in parallel or the combination of structural redundancy with the enhancement of component reliability. These are called redundancy allocation problems and reliability-redundancy allocation problems, respectively. Redundancy allocation problems are well documented in [45] and [46], which employ a special case of reliability-redundancy allocation problems without exploring the alternatives of component combined improvement. Recently, much of the effort in optimal reliability design has been placed on general reliability-redundancy allocation problems, rather than redundancy allocation problems.

In practice, two redundancy schemes are available: active and cold-standby. Cold-standby redundancy provides higher reliability, but it is hard to implement because of the difficulty of failure detection. Reliability design problems have generally been formulated considering active redundancy; however, an actual optimal design may include active redundancy or cold-standby redundancy or both.

However, any effort for improvement usually requires resources. Quite often it is hard for a single objective to adequately describe a real problem for which an optimal design is required. For this reason, multi-objective system design problem always deserves a lot of attention.

Optimal reliability design problems are known to be NP-hard [8]. Finding efficient optimization algorithms is always a hot spot in this field. Classification of the literature by reliability optimization techniques is summarized in Table 1. Meta-heuristic methods, especially GAs, have been widely and successfully applied in optimal system design because of their robustness and efficiency, even though they are time consuming, especially for large problems. To improve computation efficiency, hybrid optimization algorithms have been increasingly used to achieve an effective combination of GAs with heuristic algorithms, simulation annealing methods, neural network techniques and other local search methods.

This chapter describes the state-of-art of optimal reliability design. Emphasizing the foci mentioned above, we classify the existing literature based on problem formulations and optimization techniques. The remainder of the chapter is organized as follows: Section 2 includes four main problem formulations in optimal reliability allocation; Section 3 describes advances related to those four types of optimization problems; Section 4 summarizes developments in optimization techniques; and Section 5 provides conclusions and a discussion of future challenges related to reliability optimization problems.

**Table 1.** Reference classification by reliability optimization methods

| | |
|---|---|
| Meta-heuristic Algorithm | |
| ACO | [82], [92], [95], [116] |
| GA | [1], [7], [11], [14], [28], [35], [52], [53], [54], [56], [57], [58], [59], [60], [62], [63], [64], [65], [67], [68], [72], [73], [74], [78], [79], [84], [91], [128] |
| HGA | [33], [34], [48], [49], [50], [114], [115], [121], [122], [132], [133] |
| TS | [41] |
| SA | [1], [118], [124], [131] |
| IA | [9] |
| GDA | [107] |
| CEA | [109] |
| Exact Method | |
| [20], [27], [30], [83], [102], [103], [119], [120] | |
| Max-Min Approach | |
| [51], [104] | |
| Heuristc Method | |
| [105], [129] | |
| Dynamic Programming | |
| [97], [127] | |

## 1.2 Problem Formulations

Among the diversified problems in optimal reliability design, the following four basic formulations are widely covered.

*Problem 1* ($P_1$):

$$
\begin{aligned}
&\max R_S = f(\mathbf{x}) \\
&\text{s.t.} \\
&g_i(\mathbf{x}) \le b_i, \ \text{for } i = 1, \ldots, m \\
&\mathbf{x} \in \mathbf{X}
\end{aligned}
$$

or

$$\min C_S = f(\mathbf{x})$$
$$\text{s.t.}$$
$$Rs \geq R_0$$
$$g_i(\mathbf{x}) \leq b_i, \qquad \text{for } i = 1, \ldots, m$$
$$\mathbf{x} \in \mathbf{X}$$

*Problem 1* formulates the traditional reliability- redundancy allocation problem with either reliability or cost as the objective function. Its solution includes two parts: the component choices and their corresponding optimal redundancy levels.

*Problem 2* ($P_2$):

$$\max t_{\alpha,\mathbf{x}}$$
$$\text{s.t.}$$
$$g_i(t_{\alpha,\mathbf{x}}; \mathbf{x}) \leq b_i, \qquad \text{for } i = 1,\ldots, m$$
$$\mathbf{x} \in \mathbf{X}$$

*Problem 2* uses percentile life as the system performance measure instead of reliability. Percentile life is preferred especially when the system mission time is indeterminate. However, it is hard to find a closed analytical form of percentile life in decision variables.

*Problem 3* ($P_3$):

$$\max E(\mathbf{x}, \mathbf{T}, \mathbf{W}^*)$$
$$\text{s.t.}$$
$$g_i(\mathbf{x}) \leq b_i, \qquad \text{for } i = 1, \ldots, m$$
$$\mathbf{x} \in \mathbf{X}$$

or

$$\min C_s(\mathbf{x})$$
$$\text{s.t.}$$
$$E(\mathbf{x}, \mathbf{T}, \mathbf{W}^*) \geq E_0$$
$$g_i(\mathbf{x}) \leq b_i, \qquad \text{for } i = 1, \ldots, m$$
$$\mathbf{x} \in \mathbf{X}$$

*Problem 3* represents MSS optimization problems. Here, $E$ is used as a measure of the entire system availability to satisfy the custom demand represented by a cumulative demand curve with a known **T** and **W***.

*Problem 4* ($P_4$):

$$\begin{aligned}&\max \quad z=[f_1(\mathrm{x}), f_2(\mathrm{x}),\ldots, f_S(\mathrm{x})]\\&\text{s.t.}\\&g_i(\mathrm{x})\le b_i, \qquad \text{for } i=1,\ldots,m\\&\mathrm{x}\in \mathrm{X}\end{aligned}$$

For multi-objective optimization, as formulated by *Problem 4*, a Pareto optimal set, which includes all of the best possible trade-offs between given objectives, rather than a single optimal solution, is usually identified,.

In all of the above formulations, the resource constraints may be linear or nonlinear or both.

The literature, classified by problem formulation, is summarized in Table 2.

**Table 2.** Reference classification by problem formulation

| | |
|---|---|
| $P_1$ | [1], [2], [3], [9], [12], [13], [15], [20], [27], [30], [35], [36], [41], [48], [49], [50], [79], [81], [83], [95], [97], [100], [101], [102], [109], [119], [120], [121], [122], [124], [127], [129], [132], [133] |
| $P_2$ | [11], [14], [39], [103], [132], [133] |
| $P_3$ | [52], [53], [54], [56], [57], [58], [59], [60], [62], [63], [64], [65], [67], [68], [72], [73], [74], [78], [79], [84], [92], [99], [105] |
| $P_4$ | [7], [16], [28], [91], [106], [114], [115], [116], [118], [131], [132] |

## 1.3 Brief Reviews of Advances in $P_1$-$P_4$

### 1.3.1 Traditional Reliability-Redundancy Allocation Problem ($P_1$)

System reliability can be improved either by incremental improvements in component reliability or by provision of redundancy components in parallel; both methods result in an increase in system cost. It may be advantageous to increase the component reliability to some level and provide redundancy at that level [46], i.e. the tradeoff between these two options must be considered. According to the requirements of the designers, traditional reliability-redundancy allocation problems can be formulated either to maximize system reliability under resource constraints or to minimize the total cost that satisfies the demand on system reliability. These kinds of problems have been well-developed for many different system structures, objective functions, redundancy strategies and time-to-failure distributions. Two important recent developments related to this problem are addressed below.

#### *1.3.1.1 Active and Cold-Standby Redundancy*

$P_1$ is generally limited to active redundancy. A new optimal system configuration is obtained when active and cold-standby redundancies are both involved in the design. A cold-standby redundant component does not fail before it is put into operation by the action of switching, whereas the failure pattern of an active redundant component does not depend on whether the component is idle or in operation. Cold-standby redundancy can provide higher reliability, but it is hard to implement due to the difficulties involved in failure detection and switching.

In Ref [12], optimal solutions to reliability-redundancy allocation problems are determined for non-repairable systems designed with multiple *k*-out-of-*n* subsystems in series. The individual subsystems may use either active or cold-standby redundancy, or they may require no redundancy. Assuming an exponentially distributed component time-to-failure with rate $\lambda_{ij}$, the failure process of subsystem *i* with cold-standby redundancy can be described by a Poisson process with rate $\lambda_{ij} k_i$, while the subsystem reliability with active redundancy is computed by standard binominal techniques.

For series-parallel systems with only cold-standby redundancy, Ref [13] employs the more flexible and realistic Erlang distributed component time-to-failure. Subsystem reliability can still be evaluated through a Poisson process, though $\rho_i(t)$ must be introduced to describe the reliability of the imperfect detection/switching mechanism for each subsystem.

Ref [15] directly extends this earlier work by introducing the choice of redundancy strategies as an additional decision variable. With imperfect switching, it illustrates that there is a maximum redundancy level where cold-standby reliability is greater than, or equal to, active reliability, i.e. cold-standby redundancy is preferable before this maximum level while active redundancy is preferable after that.

All three problems formulated above can be transformed by logarithm transformation and by defining new 0-1 decision variables. This transformation linearizes the problems and allows for the use of integer programming algorithms. For each of these methods, however, no mixture of component types or redundancy strategies is allowed within any of the subsystems.

In addition, Ref [6] investigates the problem of where to allocate a spare in a *k*-out-of-*n*: *F* system of dependent components through minimal standby redundancy; and Ref [110] studies the allocation of one active redundancy when it differs based on the component with which it is to be allocated. Ref [101] considers the problem of optimally allocating a fixed number of *s*-identical multi-functional spares for a deterministic or stochastic mission time. In spite of some sufficiency conditions for optimality, the proposed algorithm can be easily implemented even for large systems.

#### 1.3.1.2 Fault-Tolerance Mechanism

Fault tolerance is the ability of a system to continue performing its intended function in spite of faults. System designs with fault-tolerance mechanisms are particularly important for some computer-based systems with life-critical applications, since they must behave like a non-repairable system within each mission, and maintenance activities are performed only when the system is idle [3].

Ref [2] maximizes the reliability of systems subjected to imperfect fault-coverage. It generalizes that the reliability of such a system decreases with an increase in redundancy after a particular limit. The results include the effect of common-cause failures and the maximum allowable spare limit. The models considered include parallel, parallel-series, series-parallel, *k*-out-of-*n* and *k*-out-of-(2*k*-1) systems.

Similarly to imperfect fault-coverage, Ref [3] later assumes the redundancy configurations of all subsystems in a non-series-parallel system are fixed except the *k*-out-of-*n*: G subsystem being analyzed. The analysis leads to $n^*$, the optimal number of components maximizing the reliability of this subsystem, which is shown to be necessarily greater than, or equal to, the optimal number required to maximize the reliability of the entire system. It also proves that $n^*$ offers exactly the maximal system reliability if the subsystem being analyzed is in series with the rest of the system. These results can even be extended to cost minimization problems.

Ref [87] considers software component testing resource allocation for a system with single or multiple applications, each with a pre-specified reliability requirement. Given the coverage factors, it can also include fault-tolerance mechanisms in the problem formulation. The relationship between the component failure rates of and the cost of decreasing this rate is modeled by various types of reliability-growth curves.

For software systems, Ref [79] presents a UGF & GA based algorithm that selects the set of versions and determines the sequence of their execution, such that the system reliability (defined as the probability of obtaining the correct output within a specified time) is maximized subject to cost constraints. The software system is built from fault-tolerant *NVP* and *RB* components.

All of these optimization models mentioned above have been developed for hardware-only or software-only systems. Ref [124] first considers several simple configurations of fault-tolerant embedded systems (hardware and software) including *NVP/0/1, NVP/1/1,* and *RB/1/1*, where failures of software units are not necessarily statistically independent. A real-time embedded system is used to demonstrate and validate the models solved by a simulated annealing optimization algorithm. Moreover, Ref [80] generally takes into account fault-tolerant systems with series architecture and arbitrary number of hardware and software versions without common cause failures. An important advantage of the presented algorithm lies in its ability to evaluate both system reliability and performance indices.

### 1.3.2 Percentile Life Optimization Problem ($P_2$)

Many diversified models and solution methods, where reliability is used as the system performance measure, have been proposed and developed since the 1960s. However, this is not an appropriate choice when mission time cannot be clearly specified or a system is intended for use as long as it functions. Average life is also not reliable, especially when the implications of failure are critical or the variance in the system life is high. Percentile life is considered to be a more appropriate measure, since it incorporates system designer and user risk. When using percentile life as the objective function, the main difficulty is its mathematical inconvenience, because it is hard to find a closed analytical form of percentile life in the decision variables.

Ref [11] solves redundancy allocation problems for series-parallel systems where the objective is to maximize a lower percentile of the system time-to-failure (*TTF*) distribution. Component *TTF* has a Weibull distribution with known deterministic parameters. The proposed algorithm uses a genetic algorithm to search the prospective solution-space and a bisection search to evaluate $t'$ in $R(t', \mathbf{x}:\mathbf{k}) = 1-\alpha$. It is demonstrated that the solution that maximizes the reliability is not particularly effective at maximizing system percentile life at any level, and the recommended design configurations are very different depending on the level. Later in the literature, Ref [14] addresses similar problems where Weibull shape parameters are accurately estimated but scale parameters are random variables following a uniform distribution.

Ref [103] develops a lexicographic search methodology that is the first to provide exact optimal redundancy allocations for percentile life optimization problems. The continuous relaxed problem, solved by Kuhn-Tucker conditions and a two-stage hierarchical search, is considered for obtaining an upper bound, which is used iteratively to effectively reduce search space. This algorithm is general for any continuous increasing lifetime distribution.

Three important results are presented in [39] which describe the general relationships between reliability and percentile life maximizing problems.

- $S_2$ equals $S_1$ given $\alpha_t = 1 - R_s(t, \mathbf{x}_t^*)$, where $\mathbf{x}_t^* \in S_1$;
- $S_1$ equals $S_2$ given $t_{\alpha, \mathbf{x}_\alpha^*}$, where $\mathrm{x}_\alpha^* \in S_2$;
- Let $\psi(t)$ be the optimal objective value of $P_1$. For a fixed $\alpha$, $t_\alpha = \inf\{t \geq 0 : \Psi(t) \leq 1 - \alpha\}$ is the optimal objective value of $P_2$.

Based on these results, a methodology for $P_2$ is proposed to repeat solving $P_1$ under different mission times satisfying $\psi(t_0) \geq 1 - \alpha \geq \psi(t_2)$ until $\mathrm{x}_{t=t_0}^* = \mathrm{x}_{t=t_2}^*$ and $t_0 - t_2$ is within a specified tolerance. It is reported to be capable of settling many unsolved $P_2$s using existing reliability-optimization algorithms. Without the necessity of an initial guess, this method is much better than [103] at reaching the exact optimal solution in terms of execution time.

### 1.3.3 MSS Optimization (*P*$_3$)

MSS is defined as a system that can unambiguously perform its task at different performance levels, depending on the state of its components which can be characterized by nominal performance rate, availability and cost. Based on their physical nature, multi-state systems can be classified into two important types: Type I MSS (e.g. power systems) and Type II MSS (e.g. computer systems), which use capacity and operation time as their performance measures, respectively.

In the literature, an important optimization strategy, combining UGF and GA, has been well developed and widely applied to reliability optimization problems of renewable MSS. In this strategy, there are two main tasks:

According to the system structure and the system physical nature, obtain the system UGF from the component UGFs;

Find an effective decoding and encoding technique to improve the efficiency of the GA.

Ref [52] first uses an UGF approach to evaluate the availability of a series-parallel multi-state system with relatively small computational resources. The essential property of the *U*-transform enables the total *U*-function for a MSS with components connected in parallel, or in series, to be obtained by simple algebraic operations involving individual component *U*-functions. The operator $\Omega_\omega$ is defined by (1) - (3).

$$\begin{aligned}\Omega_\omega(U_1(z),U_2(z)) &= \Omega_\omega\left[\sum_{i=1}^{I} p_{1i} z^{g_{1i}}, \sum_{j=1}^{J} p_{2j} z^{g_{2j}}\right] \\ &= \sum_{i=1}^{I}\sum_{j=1}^{J} p_{1i} p_{2j} z^{\omega(g_{1i},g_{2j})}\end{aligned} \tag{1}$$

$$\begin{aligned}&\Omega_\omega(U_1(z),\cdots,U_k(z),U_{k+1}(z),\cdots U_n(z)) \\ &= \Omega_\omega(U_1(z),\cdots,U_{k+1}(z),U_k(z),\cdots U_n(z))\end{aligned} \tag{2}$$

$$\begin{aligned}&\Omega_\omega(U_1(z),\cdots,U_k(z),U_{k+1}(z),\cdots U_n(z)) \\ &= \Omega_\omega(\Omega_\omega(U_1(z),\cdots,U_k(z)),\Omega_\omega(U_{k+1}(z),\cdots U_n(z)))\end{aligned} \tag{3}$$

The function $\omega(\cdot)$ takes the form from (4) - (7).

For Type I MSS,

$$\omega_{s1}(g_1,g_2) = \min(g_1,g_2) \tag{4}$$

$$\omega_{p1}(g_1,g_2) = g_1 + g_2 \tag{5}$$

For Type II MSS,

$$\omega_{s2}(g_1,g_2) = \frac{g_1 g_2}{g_1 + g_2} \tag{6}$$

$$\omega_{p2}(g_1,g_2) = g_1 + g_2 \tag{7}$$

Later, Ref [55] combines importance and sensitivity analysis and Ref [75] extends this UGF approach to MSS with dependent elements. Table 3 summarizes the application of UGF to some typical MSS structures in optimal reliability design.

With this UGF & GA strategy, Ref [84] solves the structure optimization of a multi-state system with time redundancy. TRS can be treated as a Type II MSS, where the system and its component performance are measured by the processing speed. Two kinds of systems are considered: systems with hot reserves and systems with work sharing between components connected in parallel.

**Table 3.** Application of UGF approach

| | |
|---|---|
| Series-Parallel System | [52], [54], [55], [59], [62], [65], [72], [73], [74], [92] |
| Bridge System | [53], [56], [71], [84] |
| LMSSWS | [64], [70], [78] |
| WVS | 58], [66] |
| ATN | [63], [69] |
| LMCCS | [67], [68] |
| ACCN | [61] |

Ref [57] applies the UGF & GA strategy to a multi-state system consisting of two parts:

- RGS including a number of resource generating subsystems;
- MPS including elements that consume a fixed amount of resources to perform their tasks.

Total system performance depends on the state of each subsystem in the RGS and the maximum possible productivity of the MPS. The maximum possible productivity of the MPS is determined by an integer linear programming problem related to the states of the RGS.

Ref [59] develops an UGF & GA strategy for multi-state series-parallel systems with two failure modes: open mode and closed mode. Two optimal designs are found to maximize either the system availability or the proximity of expected system performance to the desired levels for both modes. The function $\omega(\cdot)$ and the conditions of system success for both two modes are shown as follows.

For Type I MSS,

$$\omega_s^O(g_1, g_2) = \omega_s^C(g_1, g_2) = \min(g_1, g_2) \tag{8}$$

$$\omega_p^O(g_1, g_2) = \omega_p^C(g_1, g_2) = g_1 + g_2 \tag{9}$$

$$F_C(G_C, W_C) = G_C - W_C \geq 0 \tag{10}$$

$$F_O(G_O, W_O) = W_O - G_O \geq 0 \tag{11}$$

For Type II MSS,

$$\omega_s^O(g_1, g_2) = \min(g_1, g_2) \tag{12}$$

$$\omega_p^O(g_1, g_2) = \max(g_1, g_2) \tag{13}$$

$$\omega_s^C(g_1, g_2) = \max(g_1, g_2) \tag{14}$$

$$\omega_p^C(g_1, g_2) = \min(g_1, g_2) \tag{15}$$

$$F_C(G_C, W_C) = W_C - G_C \geq 0 \tag{16}$$

$$F_O(G_O, W_O) = W_O - G_O \geq 0 \tag{17}$$

Thus, the system availability can be denoted by

$$A_s(t) = 1 - \Pr\{F_C(G_C(t), W_C) < 0\} - \Pr\{F_O(G_O(t), W_O) < 0\} \tag{18}$$

Later [65, 71] introduces a probability parameter of 0.5 for both modes and [71] even extends this technique to evaluate the availability of systems with bridge structures.

To describe the ability of a multi-state system to tolerate both internal failures and external attacks, survivability, instead of reliability, is proposed in Refs [56, 60, 67, 72-74]. Ref [56] considers the problem of how to separate the elements of the same functionality between two parallel bridge components in order to achieve the maximal level of system survivability, while an UGF &GA strategy in [60] is used to solve the more general survivability optimization problem of how to separate the elements of a series-parallel system under the constraint of a separation cost. Ref [72] considers the problem of finding structure of series-parallel multi-state system (including choice of system elements, their separation and protection) in order to achieve a desired level of system survivability by minimal cost. To improve system's survivability, Ref [73, 74] further applies a multi-level protection to its subsystems and the choice of structure of multi-level protection and choice of protection methods are also included. Other than series-parallel system, Ref [67] provides the optimal allocation of multi-state LCCS with vulnerable nodes. It should be noted that the solution that provides the maximal system survivability for a given demand does not necessarily provide the greatest system expected performance rate and that the optimal solutions may be different when the system operates under different vulnerabilities.

In addition to a GA, Ref [92] presents an ant colony method that combines with a UGF technique to find an optimal series-parallel power structure configuration.

Besides this primary UGF approach, a few other methods have been proposed for MSS reliability optimization problems. Ref [105] develops a heuristic algorithm RAMC for a Type I multi-state series-parallel system. The availability of each subsystem is determined by a binomial technique, and, thus, the system availability can be obtained in a straightforward manner from the product of all subsystem availabilities without using UGF. Nevertheless, this algorithm can only adapt to relatively simple formulations, including those with only two-state component behavior and no mixing of functionally equivalent components within a particular subsystem.

A novel continuous-state system model, which may represent reality more accurately than a discrete-state system model, is presented in Ref [86]. Given the system utility function and the component state probability density functions, a neural network approach is developed to approximate the objective reliability function of this continuous MSS optimal design problem.

### 1.3.4 Multi-Objective Optimization ($P_4$)

In the previous discussion, all problems were single-objective. Rarely does a single objective with several hard constraints adequately represent a real problem for which an optimal design is required. When designing a reliable system, as formulated by $P_4$, it is always desirable to simultaneously optimize several opposing design objectives such as reliability, cost, even volume and weight. For this reason, a recently proposed multi-objective system design problem deserves a lot of attention. The objectives of this problem are to maximize the system reliability estimates and minimize their associated variance while considering the uncertainty of the component reliability estimations. A Pareto optimal set, which includes all of the best possible trade-offs between the given objectives, rather than a single optimal solution, is usually identified for multi-objective optimization problems.

When considering complex systems, the reliability optimization problem has been modeled as a fuzzy multi-objective optimization problem in Ref [106], where linear membership functions are used for all of the fuzzy goals. With the Bellman & Zadeh model, the decision is defined as the intersection of all of the fuzzy sets represented by the objectives.

$$\mu_{\tilde{D}}(x) = (\mu_{\tilde{f}_1}(x) * \cdots * \mu_{\tilde{f}_i}(x) * \cdots * \mu_{\tilde{f}_m}(x)) \tag{19}$$

The influence of various kinds of aggregators, such as the *product* operator, *min* operator, *arithmetic mean* operator, fuzzy, and a convex combination of the *min* and the *max* operator and $\gamma$-operator on the solution is also studied primarily to learn each advantage over the non-compensatory *min* operator. It was found that in some problems illustrated in this paper, the fuzzy *and* the convex combination of the *min* and the *max* operator yield efficient solutions.

Refs [114, 115] solve multi-objective reliability- redundancy allocation problems using similar linear membership functions for both objectives and constraints. By introducing 0-1 variables and by using an *add* operator to obtain the weighted sum of all the membership functions, the problem is transformed into a bi-criteria single-objective linear programming problem with Generalized Upper Bounding (GUB) constraints. The proposed hybrid GA makes use of the GUB structure and combines it with a heuristic approach to improve the quality of the solutions at each generation.

With a weighting technique, Ref [28] also transfers $P_4$ into a single-objective optimization problem and proposes a GA-based approach whose parameters can be adjusted with the experimental plan technique. Ref [7] develops a multi-objective GA to obtain an optimal system configuration and inspection policy by considering every target as a separate objective. Both problems have two objectives: maximization of the system reliability and minimization of the total cost, subject to resource constraints.

$P_4$ is considered for series-parallel systems, *RB*/1/1, and bridge systems in [16] with multiple objectives to maximize the system reliability while minimizing its associated variance when the component reliability estimates are treated as random variables. For series-parallel systems, component reliabilities of the same type are considered to be dependent since they usually share the same reliability estimate from a pooled data set. The system variance is straightforwardly expressed as a function in the higher moments of the component unreliability estimates [38]. For *RB*/1/1, the hardware components are considered identical and statistically independent, while even independently developed software versions are found to have related faults as presented by the parameters *Prv* and *Pall*. Pareto optimal solutions are found by solving a series of weighted objective problems with incrementally varied weights. It is worth noting that significantly different designs are obtained when the formulation incorporates estimation uncertainty or when the component reliability estimates are treated as statistically dependent. Similarly, [91] utilizes a multi-objective GA to select an optimal network design that balances the dual objectives of high reliability and low uncertainty in its estimation. But the latter exploits Monte Carlo simulation as the objective function evaluation engine.

Ref [131] presents an efficient computational methodology to obtain the optimal system structure of a static transfer switch, a typical power electronic device. This device can be decomposed into several components and its equivalent reliability block diagram is obtained by the minimal cut set method. Because of the existence of unit-to-unit variability, each component chosen from several off-the-shelf types is considered with failure rate uncertainty, which is modeled by a normal, or triangular, distribution. The simulation of the component failure rate distributions is performed using the Latin Hypercube Sampling method, and a simulated annealing algorithm is finally applied to generate the Pareto optimal solutions.

Ref [116] illustrates the application of the ant colony optimization algorithm to solve both continuous function and combinatorial optimization problems in reliability engineering. The single or multi-objective reliability optimization problem is analogized to Dorigo's TSP problem, and a combinatorial algorithm, which includes a global search inspired by a GA coupled with a pheromone-mediated local search, is proposed. After the global search, the pheromone values for the newly created solution are calculated by a weighted average of the pheromone values of the corresponding parent solutions. A trial solution for conducting the local search is selected with a probability proportional to its current pheromone trial value. A two-step strength Pareto fitness assignment procedure is combined to handle multi-objective problems. The advantage of employing the ant colony heuristic for multi-objective problems is that it can produce the entire set of optimal solutions in a single run.

Ref [118] tests five simulated annealing-based multi-objective algorithms – SMOSA, UMOSA, PSA, PDMOSA and WMOSA. Evaluated by 10 comparisons, Measure *C* is introduced to gauge the coverage of two approximations for the real non-dominated set. From the analysis, the computational cost of the WMOSA is the lowest, and it works well even when a large number of constraints are involved, while the PDMOSA consumes more computational time and may not perform very well for problems with too many variables.

## 1.4 Developments in Optimization Techniques

This section reviews recent developments of heuristic algorithms, metaheuristic algorithms, exact methods and other optimization techniques in optimal reliability design. Due to their robustness and feasibility, metaheuristic algorithms, especially GAs, have been widely and successfully applied. To improve computation efficiency or to avoid premature convergence, an important part of this work has been devoted in recent years to developing hybrid genetic algorithms, which usually combine a GA with heuristic algorithms, simulation annealing methods, neural network techniques or other local search methods. Though more computation effort is involved, exact methods are particularly advantageous for small problems, and their solutions can be used to measure the performance of the heuristic or meta-heuristic methods [45]. No obviously superior heuristic method has been proposed, but several of them have been well combined with exact or meta-heuristic methods to improve their computation efficiency.

### 1.4.1 Meta-Heuristic Methods

Meta-heuristic methods inspired by natural phenomena usually include the genetic algorithm, tabu search, simulated annealing algorithm and ant colony optimization method. ACO has been recently introduced into optimal reliability design, and it is proving to be a very promising general method in this field. Ref [1] provides a comparison of meta-heuristics for the optimal design of computer networks.

#### *1.4.1.1 Ant Colony Optimization Method*

ACO is one of the adaptive meta-heuristic optimization methods developed by M. Dorigo for traveling salesman problems in [21] and further improved by him in [22-26]. It is inspired by the behavior of real life ants that consistently establish the shortest path from their nest to food. The essential trait of the ACO algorithm is the combination of *a priori* information about the structure of a promising solution with *posteriori* information about the structure of previously obtained good solutions [88].

Ref [81] first develops an ant colony meta-heuristic optimization method to solve the reliability-redundancy allocation problem for a *k*-out-of-*n*: G series system. The proposed ACO approach includes four stages:

1. Construction stage: construct an initial solution by selecting component $j$ for subsystem $i$ according to its specific heuristic $\eta_{ij}$ and pheromone trail intensity $\tau_{ij}$, which also sets up the transition probability mass function $P_{ij}$.
2. Evaluation stage: evaluate the corresponding system reliability and penalized system reliability providing the specified penalized parameter.
3. Improvement stage: improve the constructed solutions through local search
4. Updating stage: update the pheromone value online and offline given the corresponding penalized system reliability and the controlling parameter for pheromone persistence.

Ref [95] presents an application of the ant system in a reliability optimization problem for a series system, with multi-choice constraints incorporated at each subsystem, to maximize the system reliability subject to the system budget. It also combines a local search algorithm and a specific improvement algorithm that uses the remaining budget to improve the quality of a solution.

The ACO algorithm has also been applied to a multi-objective reliability optimization problem [116] and to the optimal design of multi-state series-parallel power systems [92].

#### 1.4.1.2 Hybrid Genetic Algorithm

GA is a population-based directed random search technique inspired by the principles of evolution. Though it provides only heuristic solutions, it can be effectively applied to almost all complex combinatorial problems, and, thus, it has been employed in a large number of references as shown in Table 1. Ref [29] provides a state-of-the-art survey of GA-based reliability design.

To improve computational efficiency, or to avoid premature convergence, numerous researchers have been inspired to seek effective combinations of GAs with heuristic algorithms, simulation annealing methods, neural network techniques, steepest decent methods or other local search methods. The combinations are generally called hybrid genetic algorithms, and they represent one of the most promising developmental directions in optimization techniques.

Considering a complex system with a known system structure function, Ref [132] provides a unified modeling idea for both active and cold-standby redundancy optimization problems. The model prohibits any mixture of component types within subsystems. Both the lifetime and the cost of redundancy components are considered as random variables, so stochastic simulation is used to estimate the system performance, including the mean lifetime, percentile lifetime and reliability. To speed up the solution process, these simulation results become the training data for training a neural network to approximate the system performance. The trained neural network is finally embedded into a genetic algorithm to form a hybrid intelligent algorithm for solving the proposed model. Later [133] uses random fuzzy lifetimes as the basic parameters and employs a random fuzzy simulation to generate the training data.

Ref [48] develops a two-phase NN-hGA in which NN is used as a rough search technique to devise the initial solutions for a GA. By bounding the broad continuous search space with the NN technique, the NN-hGA derives the optimum robustly. However, in some cases, this algorithm may require too much computational time to be practical.

To improve the computation efficiency, Ref [49] presents a NN-flcGA to effectively control the balance between exploitation and exploration which characterizes the behavior of GAs. The essential features of the NN-flcGA include:

- combination with a NN technique to devise initial values for the GA
- application of a fuzzy logic controller when tuning strategy GA parameters dynamically
- incorporation of the revised simplex search method

Later, [50] proposes a similar hybrid GA called f-hGA for the redundancy allocation problem of a series-parallel system. It is based on

- application of a fuzzy logic controller to automatically regulate the GA parameters;
- incorporation of the iterative hill climbing method to perform local exploitation around the near optimum solution.

Ref [33] considers the optimal task allocation strategy and hardware redundancy level for a cycle-free distributed computing system so that the system cost during the period of task execution is minimized. The proposed hybrid heuristic combines the GA and the steepest decent method. Later [34] seeks similar optimal solutions to minimize system cost under constraints on the hardware redundancy levels. Based on the GA and a local search procedure, a hybrid GA is developed and compared with the simple GA. The simulation results show that the hybrid GA provides higher solution quality with less computational time.

#### 1.4.1.3 Tabu Search

Though [45] describes the promise of tabu search, Ref [41] first develops a TS approach with the application of NFT [10] for reliability optimization problems. This method uses a subsystem-based tabu entry and dynamic length tabu list to reduce the sensitivity of the algorithm to selection of the tabu list length. The definition of the moves in this approach offers an advantage in efficiency, since it does not require recalculating the entire system reliability, but only the reliability of the changed subsystem. The results of several examples demonstrate the superior performance of this TS approach in terms of efficiency and solution superiority when compared to that of a GA.

#### *1.4.1.4 Other Meta-Heuristic Methods*

Some other adaptive meta-heuristic optimization methods inspired by activities in nature have also been proposed and applied in optimal reliability design. Ref [9] develops an immune algorithms-based approach inspired by the natural immune system of all animals. It analogizes antibodies and antigens as the solutions and objection functions, respectively. Ref [109] proposes a cellular evolutionary approach combining the multimember evolution strategy with concepts from Cellular Automata [125] for the selection step. In this approach, the parents' selection is performed only in the neighborhood in contrast to the general evolutionary strategy that searches for parents in the whole population. And a great deluge algorithm is extended and applied to optimize the reliability of complex systems in [107]. When both accuracy and speed are considered simultaneously, it is proven to be an efficient alternative to ACO and other existing optimization techniques.

### 1.4.2 Exact Methods

Unlike meta-heuristic algorithms, exact methods provide exact optimal solutions though much more computation complexity is involved. The development of exact methods, such as the branch-and-bound approach and lexicographic search, has recently been concentrated on techniques to reduce the search space of discrete optimization methods.

Ref [119] considers a reliability-redundancy allocation problem in which multiple-choice and resource constraints are incorporated. The problem is first transformed into a bi-criteria nonlinear integer programming problem by introducing 0-1 variables. Given a good feasible solution, the lower reliability bound of a subsystem is determined by the product of the maximal component reliabilities of all the other subsystems in the solution, while the upper bound is determined by the maximal amount of available sources of this subsystem. A branch-and-bound procedure, based on this reduced solution space, is then derived to search for the global optimal solution. Later, Ref [120] even combines the lower and upper bounds of the system reliability, which are obtained by variable relaxation and Lagrangean relaxation techniques, to further reduce the search space.

Also with a branch-and-bound algorithm, Ref [20] obtains the upper bound of series-parallel system reliability from its continuous relaxation problem. The relaxed problem is efficiently solved by the greedy procedure described in [19], combining heuristic methods to make use of some slack in the constraints obtained from rounding down. This technique assumes the objective and constraint functions are monotonically increasing.

Ref [30] presents an efficient branch-and-bound approach for coherent systems based on a 1-neighborhood local maximum obtained from the steepest ascent heuristic method. Numerical examples of a bridge system and a hierarchical series-parallel system demonstrate the advantages of this proposed algorithm in flexibility and efficiency.

Apart from the branch-and-bound approach, Ref [102] presents a partial enumeration method for a wide range of complex optimization problems based on a lexicographic search. The proposed upper bound of system reliability is very useful in eliminating several inferior feasible or infeasible solutions as shown in either big or small numerical examples. It also shows that the search process described in [95] does not necessarily give an exact optimal solution due to its logical flows.

Ref [83] develops a strong Lin & Kuo heuristic to search for an ideal allocation through the application of the reliability importance. It concludes that, if there exists an invariant optimal allocation for a system, the optimal allocation is to assign component reliabilities according to B-importance ordering. This Lin & Kuo heuristic can provide an exact optimal allocation.

Assuming the existence of a convex and differential reliability cost function $C_i(y_{ij})$, $y_{ij} = \log(1 - r_{ij})$ for all component *j* in any subsystem *i*, Ref [27] proves that the components in each subsystem of a series-parallel system must have identical reliability for the purpose of cost minimization. The solution of the corresponding unconstrained problem provides the upper bound of the cost, while a doubly minimization problem gives its lower bound. With these results, the algorithm ECAY, which can provide either exact or approximate solutions depending on different stop criteria, is proposed for series-parallel systems.

### 1.4.3 Other Optimization Techniques

For series-parallel systems, Ref [104] formulates the reliability-redundancy optimization problem with the objective of maximizing the minimal subsystem reliability.

*Problem 5* ($P_5$):

$$\max_{\mathbf{x}_1,\mathbf{x}_2,\ldots} \left(\min_i \left(1 - \prod_j (1 - r_{ij})^{x_{ij}}\right)\right)$$

s.t.

$$g_i(\mathbf{x}) \le b_i, \qquad \text{for } i = 1, \ldots m$$

$$\mathbf{x} \in \mathbf{X}$$

Assuming linear constraints, an equivalent linear formulation of $P_5$ [36] can be obtained through an easy logarithm transformation, and, thus, the problem can be solved by readily available commercial software. It can also serve as a surrogate for traditional reliability optimization problems accomplished by sequentially solving a series of max-min subproblems.

Ref [51] presents a comparison between the Nakagawa and Nakashima method [43] and the max-min approach used by Ramirez-Marquez from the standpoint of solution quality and computational complexity. The experimental results show that the max-min approach is superior to the Nakagawa and Nakashima method in terms of solution quality in small-scale problems, but the analysis of its computational complexity demonstrates that the max-min approach is inferior to other greedy heuristics.

Ref [129] develops a heuristic approach inspired by the greedy method and a GA. The structure of this algorithm includes:

1. randomly generating a specified population size number of minimum workable solutions;
2. assigning components either according to the greedy method or to the random selection method;
3. improving solutions through an inner-system and inter-system solution revision process.

Ref [97] applies a hybrid dynamic programming/ depth-first search algorithm to redundancy allocation problems with more than one constraint. Given the tightest upper bound, the knapsack relaxation problem is formulated with only one constraint, and its solution $f_1(b)$ is obtained by a dynamic programming method. After choosing a small specified parameter $e$, the depth-first search technique is used to find all near-optimal solutions with objectives between $f_1(b)$ and $f_1(b) - e$. The optimal solution is given by the best feasible solution among all of the near-optimal solutions.

Ref [127] also presents a new dynamic programming method for a reliability-redundancy allocation problem in series-parallel systems where components must be chosen among a finite set. This pseudo-polynomial YCC algorithm is composed of two steps: the solution of the subproblems, one for each subsystem, and the global resolution using previous results. It shows that the solutions converge quickly toward the optimum as a function of the required precision.

## 1.5 Comparisons and Discussions of Algorithms Reported in Literature

In this section, we provide a comparison of several heuristic or meta-heuristic algorithms reported in the literature. The compared numerical results are from the GA in Coit and Smith [10], the ACO in Liang and Smith [81], TS in Kulturel-Konak *et al.* [41], linear approximation in Hsieh [36], the IA in Chen and You [9] and the heuristic method in You and Chen [129]. The 33 variations of the Fyffe *et al.* problem, as devised by Nakagawa and Miyazaki [96], are used to test their performance, where different types are allowed to reside in parallel. In this problem set, the cost constraint is maintained at 130 and the weight constraint varies from 191 to 159.

As shown in Table 4, ACO [81], TS [41], IA [36] and heuristic methods [129] generally yield solutions with a higher reliability. When compared to GA [10],

- ACO [81] is reported to consistently perform well over different problem sizes and parameters and improve on GA's random behavior;
- TS [41] results in a superior performance in terms of best solutions found and reduced variability and greater efficiency based on the number of objective function evaluations required;
- IA [9] finds better or equally good solutions for all 33 test problems, but the performance of this IA-based approach is sensitive to value-combinations of the parameters, whose best values are case-dependent and only based upon the experience from preliminary runs. And more CPU time is taken by IAs;
- The best solutions found by heuristic methods [129] are all better than, or as good as, the well-known best solutions from other approaches. With this method, the average CPU time for each problem is within 8 seconds;
- In terms of solution quality, the proposed linear approximation approach [36] is inferior. But it is very efficient and the CPU time for all of the test problems is within one second;
- If a decision-maker is considering the max-min approach as a surrogate for system reliability maximization, the max-min approach [104] is shown to be capable of obtaining a close solution (within 0.22%), but it is unknown whether this performance will continue as problem sizes become larger.

For all the optimization techniques mentioned above, it might be hard to discuss about which tool is superior because in different design problems or even in a same problem with different parameters, these tools will perform variously.

**Table 4.** Comparison of several algorithms in the literature. Each for the test problems form [96]

| W | System Reliability | | | | | |
|---|---|---|---|---|---|---|
| | GA [10] | ACO [81] | TS [41] | Hsieh[36] | IA [9] | Y&C [129] |
| 191 | 0.98670 | 0.9868 | 0.98681 | 0.98671 | 0.98681 | 0.98681 |
| 190 | 0.98570 | 0.9859 | 0.98642 | 0.98632 | 0.98642 | 0.98642 |
| 189 | 0.98560 | 0.9858 | 0.98592 | 0.98572 | 0.98592 | 0.98592 |
| 188 | 0.98500 | 0.9853 | 0.98538 | 0.98503 | 0.98533 | 0.98538 |
| 187 | 0.98440 | 0.9847 | 0.98469 | 0.98415 | 0.98445 | 0.98469 |
| 186 | 0.98360 | 0.9838 | 0.98418 | 0.98388 | 0.98418 | 0.98418 |
| 185 | 0.98310 | 0.9835 | 0.98351 | 0.98339 | 0.98344 | 0.98350 |
| 184 | 0.98230 | 0.9830 | 0.98300 | 0.9822 | 0.9827 | 0.98299 |
| 183 | 0.98190 | 0.9822 | 0.98226 | 0.98147 | 0.98221 | 0.98226 |
| 182 | 0.98110 | 0.9815 | 0.98152 | 0.97969 | 0.98152 | 0.98152 |
| 181 | 0.98020 | 0.9807 | 0.98103 | 0.97928 | 0.98103 | 0.98103 |
| 180 | 0.97970 | 0.9803 | 0.98029 | 0.97833 | 0.98029 | 0.98029 |
| 179 | 0.97910 | 0.9795 | 0.97951 | 0.97806 | 0.97951 | 0.97950 |
| 178 | 0.97830 | 0.9784 | 0.97840 | 0.97688 | 0.97821 | 0.97840 |
| 177 | 0.97720 | 0.9776 | 0.97747 | 0.9754 | 0.97724 | 0.97760 |
| 176 | 0.97640 | 0.9765 | 0.97669 | 0.97498 | 0.97669 | 0.97669 |
| 175 | 0.97530 | 0.9757 | 0.97571 | 0.9735 | 0.97571 | 0.97571 |
| 174 | 0.97435 | 0.9749 | 0.97479 | 0.97233 | 0.97469 | 0.97493 |
| 173 | 0.97362 | 0.9738 | 0.97383 | 0.97053 | 0.97376 | 0.97383 |
| 172 | 0.97266 | 0.9730 | 0.97303 | 0.96923 | 0.97303 | 0.97303 |
| 171 | 0.97186 | 0.9719 | 0.97193 | 0.9679 | 0.97193 | 0.97193 |
| 170 | 0.97076 | 0.9708 | 0.97076 | 0.96678 | 0.97076 | 0.97076 |
| 169 | 0.96922 | 0.9693 | 0.96929 | 0.96561 | 0.96929 | 0.96929 |
| 168 | 0.96813 | 0.9681 | 0.96813 | 0.96415 | 0.96813 | 0.96813 |
| 167 | 0.96634 | 0.9663 | 0.96634 | 0.96299 | 0.96634 | 0.96634 |
| 166 | 0.96504 | 0.9650 | 0.96504 | 0.96121 | 0.96504 | 0.96504 |
| 165 | 0.96371 | 0.9637 | 0.96371 | 0.95992 | 0.96371 | 0.96371 |
| 164 | 0.96242 | 0.9624 | 0.96242 | 0.9586 | 0.96242 | 0.96242 |
| 163 | 0.96064 | 0.9606 | 0.95998 | 0.95732 | 0.96064 | 0.96064 |
| 162 | 0.95912 | 0.9592 | 0.95821 | 0.95555 | 0.95919 | 0.95919 |
| 161 | 0.95803 | 0.9580 | 0.95692 | 0.9541 | 0.95804 | 0.95803 |
| 160 | 0.95567 | 0.9557 | 0.9556 | 0.95295 | 0.95571 | 0.95571 |
| 159 | 0.95432 | 0.9546 | 0.95433 | 0.9508 | 0.95457 | 0.95456 |

Generally, if computational efficiency is of most concern to designer, linear approximation or heuristic methods can obtain competitive feasible solutions within a very short time (few seconds), as reported in [36, 129]. The proposed linear approximation [36] is also easy to implement with any LP software. But the main limitation of those reported approaches is that the constraints must be linear and separable.

Due to their robustness and feasibility, meta-heuristic methods such as GA and recently developed TS and ACO could be successfully applied to almost all NP-hard reliability optimization problems. However, they can not guarantee the optimality and sometimes can suffer from the premature convergence situation of their solutions because they have many unknown parameters and they neither use a prior knowledge nor exploit local search information. Compared to traditional meta-heuristic methods, a set of promising algorithms, hybrid GAs [33-34, 48-50, 132-133], are attractive since they retain the advantages of GAs in robustness and feasibility but significantly improve their computational efficiency and searching ability in finding global optimum with combining heuristic algorithms, neural network techniques, steepest decent methods or other local search methods.

For reliability optimization problems, exact solutions are not necessarily desirable because it is generally difficult to develop exact methods for reliability optimization problems which are equivalent to methods used for nonlinear integer programming problems [45]. However, exact methods may be particularly advantageous when the problem is not large. And more importantly, such methods can be used to measure the performance of heuristic or meta-heuristic methods.

## 1.6 Conclusions and Discussions

We have reviewed the recent research on optimal reliability design. Many publications have addressed this problem using different system structures, performance measures, problem formulations and optimization techniques.

The systems considered here mainly include series-parallel systems, $k$-out-of-$n$: $G$ systems, bridge networks, $n$-version programming architecture, recovery block architecture and other unspecified coherent systems. The recently introduced *NVP* and *RB* belong to the category of fault tolerant architecture, which usually considers both software and hardware.

Reliability is still employed as a system performance measure in a majority of cases, but percentile life does provide a new perspective on optimal design without the requirement of a specified mission time. Availability is primarily used as the performance measure of renewable multi-state systems whose optimal design has been emphasized and well developed in the past 10 years. Optimal design problems are generally formulated to maximize an appropriate system performance measure under resource constraints, and more realistic problems involving multi-objective programming are also being considered.

When turning to optimization techniques, heuristic, meta-heuristic and exact methods are significantly applied in optimal reliability design. Recently, many advances in meta-heuristics and exact methods have been reported. Particularly, a new meta-heuristic method called ant colony optimization has been introduced and demonstrated to be a very promising general method in this field. Hybrid GAs may be the most important recent development among the optimization techniques since they retain the advantages of GAs in robustness and feasibility but significantly improve their computational efficiency.

Optimal reliability design has attracted many researchers, who have produced hundreds of publications since 1960. Due to the increasing complexity of practical engineering systems and the critical importance of reliability in these complex systems, this still seems to be a very fruitful area for future research.

Compared to traditional binary-state systems, there are still many unsolved topics in MSS optimal design including

- using percentile life as a system performance measure;
- involving cold-standby redundancy;
- nonrenewable MSS optimal design;
- applying optimization algorithms other than GAs, especially hybrid optimization techniques.

From the view of optimization techniques, there are opportunities for improved effectiveness and efficiency of reported ACO, TS, IA and GDA, while some new meta-heuristic algorithms such as Harmony Search Algorithm and Particle Swarm Optimization may offer excellent solutions for reliability optimization problems. Hybrid optimization techniques are another very promising general developmental direction in this field. They may combine heuristic methods, NN or some local search methods with all kinds of meta-heuristics to improve computational efficiency or with exact methods to reduce search space. We may even be able to combine two meta-heuristic algorithms such as GA and SA or ACO.

The research dealing with the understanding and application of reliability at the nano level has also demonstrated its attraction and vitality in recent years. Optimal system design that considers reliability within the uniqueness of nano-systems has seldom been reported in the literature. It deserves a lot more attention in the future. In addition, uncertainty and component dependency will be critical areas to consider in future research on optimal reliability design.

## Acknowledgement

This work was partly supported by NSF Projects DMI-0429176. It is republished with the permission of IEEE Transactions on Systems, Man, Cybernetics, Part A: Systems and Humans.

## References

[1] Altiparmak F, Dengiz B, Smith AE (2003) Optimal design of reliable computer networks: a comparison of meta heuristics. Journal of Heuristics 9: 471-487

[2] Amari SV, Dugan JB, Misra RB (1999) Optimal reliability of systems subject to imperfect fault-coverage. IEEE Transactions on Reliability 48: 275-284

[3] Amari SV, Pham H, Dill G (2004) Optimal design of k-out-of-n: G subsystems subjected to imperfect fault-coverage. IEEE Transactions on Reliability 53: 567-575

[4] Andrews JD, Bartlett LA (2005) A branching search approach to safety system design optimization. Reliability Engineering & System Safety 87: 23-30

[5] Aneja YP, Chandrasekaran R, Nair KPK (2004) Minimal-cost system reliability with discrete-choice sets for components. IEEE Transactions on Reliability 53:71-76

[6] Bueno VC (2005) Minimal standby redundancy allocation in a k-out-of-n: F system of dependent components. European Journal of Operation Research 165: 786-793

[7] Busacca PG, Marseguerra M, Zio E (2001) Multi-objective optimization by genetic algorithms: application to safety systems. Reliability Engineering & System Safety 72: 59-74

[8] Chern MS (1987) On the computational complexity of reliability redundancy allocation in a series system. Operation Research Letter SE-13: 582-592

[9] Chen TC, You PS (2005) Immune algorithms-based approach for redundant reliability problems with multiple component choices. Computers in Industry 56: 195-205

[10] Coit DW, Smith AE (1996) Reliability optimization of series-parallel systems using a genetic algorithm. IEEE Transactions on Reliability 45: 254-260

[11] Coit DW, Smith AE (1998) Redundancy allocation to maximize a lower percentile of the system time-to-failure distribution. IEEE Transactions on Reliability 47: 79-87
[12] Coit DW, Liu J (2000) System reliability optimization with k-out-of-n subsystems. International Journal of Reliability, Quality and Safety Engineering 7: 129-142
[13] Coit DW (2001) Cold-standby redundancy optimization for nonrepairable systems. IIE Transactions 33: 471-478
[14] Coit DW, Smith AE (2002) Genetic algorithm to maximize a lower-bound for system time-to-failure with uncertain component Weibull parameters. Computer & Industrial Engineering 41: 423-440
[15] Coit DW (2003) Maximization of system reliability with a choice of redundancy strategies. IIE Transactions 35: 535-543
[16] Coit DW, Jin T, Wattanapongsakorn N (2004) System optimization with component reliability estimation uncertainty: a multi-criteria approach. IEEE Transactions on Reliability 53: 369-380
[17] Cui L, Kuo W, Loh HT, Xie M (2004) Optimal allocation of minimal & perfect repairs under resource constraints. IEEE Transactions on Reliability 53: 193-199
[18] Dimou CK, Koumousis VK (2003) Competitive genetic algorithms with application to reliability optimal design. Advances in Engineering Software 34: 773-785
[19] Djerdjour M (1997) An enumerative algorithm framework for a class of nonlinear programming problems. European Journal of Operation Research 101: 101-121
[20] Djerdjour M, Rekab K (2001) A branch and bound algorithm for designing reliable systems at a minimum cost. Applied Mathematics and Computation 118: 247-259
[21] Dorigo M (1992) Optimization learning and natural algorithm. Ph.D. Thesis, Politecnico di Milano, Italy
[22] Dorigo M, Maniezzo V, Colorni A (1996) Ant system: optimization by a colony of cooperating agents. IEEE Transactions on System, Man, and Cybernetics-part B: Cybernetics 26: 29-41
[23] Dorigo M, Gambardella LM (1997) Ant colonies for traveling salesman problem. BioSystem 43: 73-81
[24] Dorigo M, Gambardella LM (1997) Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary computation 1: 53-66
[25] Dorigo M, Caro G (1999) The ant colony optimization meta heuristic. In: Corne D, Dorigo M, Glover F (eds.) New Ideas in Optimization. McGraw-Hill, pp 11-32.
[26] Dorigo M, Caro GD, Gambardella LM (1999) Ant algorithm for discrete optimization. Artificial Life 5: 137-172
[27] Elegbede C, Chu C, Adjallah K, Yalaoui F (2003) Reliability allocation through cost minimization. IEEE Transactions on Reliability 52: 106-111

[28] Elegbede C, Adjallah K (2003) Availability allocation to repairable systems with genetic algorithms: a multi-objective formulation Reliability Engineering and System Safety 82: 319-330
[29] Gen M, Kim JR (1999) GA-based reliability design: state-of-the-art survey. Computer & Industrial Engineering 37: 151-155
[30] Ha C, Kuo W (2006) Reliability redundancy allocation: an improved realization for nonconvex nonlinear programming problems. European Journal of Operational Research 171: 24-38
[31] Ha C, Kuo W (accepted) Multi-paths iterative heuristic for redundancy allocation: the tree heuristic. IEEE Transactions on Reliability
[32] Ha C, Kuo W (2005) Multi-path approach for Reliability-redundancy allocation using a scaling method. Journal of Heuristics, 11: 201-237
[33] Hsieh CC, Hsieh YC (2003) Reliability and cost optimization in distributed computing systems. Computer & Operations Research 30: 1103-1119
[34] Hsieh CC (2003) Optimal task allocation and hardware redundancy policies in distributed computing system. European Journal of Operational Research 147: 430-447
[35] Hsieh YC, Chen TC, Bricker DL (1998) Genetic algorithm for reliability design problems. Microelectronics Reliability 38: 1599-1605
[36] Hsieh YC (2002) A linear approximation for redundant reliability problems with multiple component choices. Computer & Industrial Engineering 44: 91-103
[37] Huang J, Zuo MJ, Wu Y (2000) Generalized multi-state k-out-of-n: G systems. IEEE Transactions on Reliability 49: 105-111
[38] Jin T, Coit DW (2001) Variance of system-reliability estimates with arbitrarily repeated components. IEEE Transactions on Reliability 50: 409-413
[39] Kim KO, Kuo W (2003) Percentile life and reliability as a performance measures in optimal system design. IIE Transactions 35: 1133-1142
[40] Kulturel-Kotz S, Lai CD, Xie M (2003) On the effect of redundancy for systems with dependent components. IIE Transactions 35: 1103-1110
[41] Kulturel-Konak S, Smith AE, Coit DW (2003) Efficiently solving the redundancy allocation problem using tabu search. IIE Transactions 35: 515-526
[42] Kumar UD, Knezevic J (1998) Availability based spare optimization using renewal process. Reliability Engineering and System Safety 59: 217-223
[43] Kuo W, Hwang CL, Tillman FA (1978) A note on heuristic method for in optimal system reliability. IEEE Transactions on Reliability 27: 320-324
[44] Kuo W, Chien K, Kim T (1998) Reliability, Yield and Stress Burn-in. Kluwer
[45] Kuo W, Prasad VR (2000) An annotated overview of system-reliability optimization. IEEE Transactions on Reliability 49: 487-493
[46] Kuo W, Prasad VR, Tillman FA, Hwang CL (2001) Optimal Reliability Design: Fundamentals and Applications. Cambridge University Press
[47] Kuo W, Zuo MJ (2003) Optimal Reliability Modeling: Principles and Applications. John Wiley & Sons
[48] Lee CY, Gen M, Kuo W (2002) Reliability optimization design using hybridized genetic algorithm with a neural network technique. IEICE Transactions

on Fundamentals of Electronics, Communications and Computer Sciences: E84-A: 627-637
[49] Lee CY, Yun Y, Gen M (2002) Reliability optimization design using hybrid NN-GA with fuzzy logic controller. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E85-A: 432-447
[50] Lee CY, Gen M, Tsujimura Y (2002) Reliability optimization design for coherent systems by hybrid GA with fuzzy logic controller and local search. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E85-A: 880-891
[51] Lee H, Kuo W, Ha C (2003) Comparison of max-min approach and NN method for reliability optimization of series-parallel system. Journal of System Science and Systems Engineering 12: 39-48
[52] Levitin G, Lisnianski A, Ben-Haim H, Elmakis D (1998) Redundancy optimization for series-parallel multi-state systems. IEEE Transactions on Reliability 47: 165-172
[53] Levitin G, Lisnianski A (1998) Structure optimization of power system with bridge topology. Electric Power Systems Research 45: 201-208
[54] Levitin G, Lisnianski A (1999) Joint redundancy and maintenance optimization for multistate series-parallel systems. Reliability Engineering and System Safety 64: 33-42
[55] Levitin G, Lisnianski A (1999) Importance and sensitivity analysis of multistate systems using the universal generating function method. Reliability Engineering and System Safety 65: 271-282
[56] Levitin G, Lisnianski A (2000) Survivability maximization for vulnerable multi-state systems with bridge topology. Reliability Engineering and System Safety 70: 125-140
[57] Levitin G (2001) Redundancy optimization for multi-state system with fixed resource-requirements and unreliable sources. IEEE Transactions on Reliability 50: 52-59
[58] Levitin G, Lisnianski A (2001) Reliability optimization for weighted voting system. Reliability Engineering and System Safety 71: 131-138
[59] Levitin G, Lisnianski A (2001) Structure optimization of multi-state system with two failure modes. Reliability Engineering and System Safety 72: 75-89
[60] Levitin G, Lisnianski A (2001) Optimal separation of elements in vulnerable multi-state systems. Reliability Engineering and System Safety 73: 55-66
[61] Levitin G (2001) Reliability evaluation for acyclic consecutively connected network with multistate elements. Reliability Engineering and System Safety 73: 137-143
[62] Levitin G, Lisnianski A (2001) A new approach to solving problems of multi-state system reliability optimization Quality and Reliability Engineering International 17: 93-104
[63] Levitin G (2002) Optimal allocation of multi-state retransmitters in acyclic transmission networks. Reliability Engineering and System Safety 75: 73-82
[64] Levitin G (2002) Optimal allocation of elements in a linear multi-state sliding window system. Reliability Engineering and System Safety 76: 245-254

[65] Levitin G (2002) Optimal series-parallel topology of multi-state system with two failure modes. Reliability Engineering and System Safety 77: 93-107
[66] Levitin G (2002) Evaluating correct classification probability for weighted voting classifiers with plurality voting. European Journal of Operational Research 141: 596-607
[67] Levitin G (2003) Optimal allocation of multi-state elements in linear consecutively connected systems with vulnerable nodes. European Journal of Operational Research 150: 406-419
[68] Levitin G (2003) Optimal allocation of multistate elements in a linear consecutively-connected system. IEEE Transactions on Reliability 52: 192-199
[69] Levitin G (2003) Reliability evaluation for acyclic transmission networks of multi-state elements with delays. IEEE Transactions on Reliability 52: 231-237
[70] Levitin G (2003) Linear multi-state sliding-window systems. IEEE Transactions on Reliability 52: 263-269
[71] Levitin G (2003) Reliability of multi-state systems with two failure-modes. IEEE Transactions on Reliability 52: 340-348
[72] Levitin G, Lisnianski A (2003) Optimizing survivability of vulnerable series-parallel multi-state systems. Reliability Engineering & System Safety 79: 319-331
[73] Levitin G (2003) Optimal multilevel protection in series-parallel systems. Reliability Engineering and System Safety, 81: 93-102
[74] Levitin G, Dai Y, Xie M, Poh KL (2003) Optimizing survivability of multi-state systems with multi-level protection by multi-processor genetic algorithm. Reliability Engineering and System Safety. 82: 93-104
[75] Levitin G (2004) A universal generating function approach for the analysis of multi-state systems with dependent elements. Reliability Engineering and System Safety 84: 285-292
[76] Levitin G (2004) Consecutive k-out-of-r-from-n system with multiple failure criteria. IEEE Transactions on Reliability 53: 394-400
[77] Levitin G (2004) Reliability optimization models for embedded systems with multiple applications. IEEE Transactions on Reliability 53: 406-416
[78] Levitin G (2005) Uneven allocation of elements in linear multi-state sliding window system. European Journal of Operational Research 163: 418-433
[79] Levitin G (2005) Optimal structure of fault-tolerant software systems. Reliability Engineering and System Safety 89: 286-295
[80] Levitin G (accepted) Reliability and performance analysis of hardware-software systems with fault-tolerant software components. Reliability Engineering and System Safety
[81] Liang YC, Smith AE (2004) An ant colony optimization algorithm for the redundancy allocation problem. IEEE Transactions on Reliability 53: 417-423
[82] Lin F, Kuo W, Hwang F (1999) Structure importance of consecutive-k-out-of-n systems. Operations Research Letters 25: 101-107
[83] Lin F, Kuo W (2002) Reliability importance and invariant optimal allocation. Journal of Heuristics 8: 155-172

[84] Lisnianski A, Levitin G, Ben-Haim H (2000) Structure optimization of multi-state system with time redundancy. Reliability Engineering and System Safety 67: 103-112
[85] Lisianski A, Levitin G (2003) Multi-state System Reliability, Assessment, Optimization and Applications. World Scientific
[86] Liu PX, Zuo MJ, Meng MQ (2003) Using neural network function approximation for optimal design of continuous-state parallel-series systems. Computers & Operations Research 30: 339-352
[87] Lyu MR, Rangarajan S, Moorsel APA (2002) Optimal allocation of test resources for software reliability growth modeling in software development. IEEE Transactions on Reliability 51: 183-192
[88] Maniezzo V, Gambardella LM, De Luigi F. (2004) Ant Colony Optimization. In: Onwubolu GC, Babu BV (eds) New Optimization Techniques in Engineering. Springer-Verlag Berlin Heidelberg, pp 101-117
[89] Marseguerra M, Zio E (2000) Optimal reliability/availability of uncertain systems via multi-objective genetic algorithms. 2000 Proceedings Annual Reliability and Maintainability Symposium, pp 222-227
[90] Marseguerra M, Zio E (2004) System design optimization by genetic algorithm. IEEE Transactions on Reliability 53: 424-434
[91] Marseguerra M, Zio E, Podofillini L, Coit DC (2005) Optimal design of reliable network systems in presence of uncertainty. IEEE Transactions on Reliability 54: 243-253
[92] Massim Y, Zeblah A, Meziane R, Benguediab M, Ghouraf A (2005) Optimal design and reliability evaluation of multi-state series-parallel power systems. Nonlinear Dynamics 40: 309-321
[93] Misra KB (1986) On optimal reliability design: a review. System Science 12: 5-30
[94] Misra KB (1991) An algorithm to solve integer programming problems: an efficient tool for reliability design. Microelectronics and Reliability 31: 285-294
[95] Nahas N, Nourelfath M (2005) Ant system for reliability optimization of a series system with multiple-choice and budget constraints. Reliability Engineering and System Safety 87: 1-12
[96] Nakagawa Y, Miyazaki S (1981) Surrogate constraints algorithm for reliability optimization problems with two constraints. IEEE Transactions on Reliability R-30: 175-180
[97] Kevin YKNG, Sancho NGF (2001) A hybrid dynamic programming/depth-first search algorithm with an application to redundancy allocation. IIE Transactions 33: 1047-1058
[98] Nordmann L, Pham H (1999) Weighted voting systems. IEEE Transactions on Reliability 48: 42-49
[99] Nourelfath M, Dutuit Y (2004) A combined approach to solve the redundancy optimization problem for multi-state systems under repair policies. Reliability Engineering and System Safety 84: 205-213

[100] Prasad VR, Raghavachari M (1998) Optimal allocation of interchangeable components in a series-parallel system. IEEE Transactions on Reliability 47: 255-260
[101] Prasad VR, Kuo W, Kim KO (2000) Optimal allocation of s-identical multi-functional spares in a series system. IEEE Transactions on Reliability 48: 118-126
[102] Prasad VR, Kuo W (2000) Reliability optimization of coherent systems. IEEE Transactions on Reliability 49: 323-330
[103] Prasad VR, Kuo W, Kim KO (2001) Maximization of a percentile life of a series system through component redundancy allocation IIE Transactions 33: 1071-1079
[104] Ramirez-Marquez JE, Coit DW, Konak A (2004) Redundancy allocation for series-parallel systems using a max-min approach. IIE Transactions 36: 891-898
[105] Ramirez-Marquez JE, Coit DW (2004) A heuristic for solving the redundancy allocation problem for multi-state series-parallel system. Reliability Engineering and System Safety 83: 341-349
[106] Ravi V, Reddy PJ, Zimmermann HJ (2000) Fuzzy global optimization of complex system reliability. IEEE Transactions on Fuzzy Systems 8: 241-248
[107] Ravi V (2004) Optimization of complex system reliability by a modified great deluge algorithm. Asia-Pacific Journal of Operational Research 21: 487-497
[108] Ren Y, Dugan JB (1998) Design of reliable system using static & dynamic fault trees. IEEE Transactions on Reliability 47: 234-244
[109] Rocco CM, Moreno JA, Carrasquero N (2000) A cellular evolution approach applied to reliability optimization of complex systems. 2000 Proceedings Annual Reliability and Maintainability Symposium, pp 210-215
[110] Romera R, Valdés JE, Zequeira RI (2004) Active-redundancy allocation in systems. IEEE Transactions on Reliability 53: 313-318
[111] Royset JO, Polak E (2004) Reliability-based optimal design using sample average approximations. Probabilistic Engineering Mechanics 19: 331-343
[112] Royset JO, Kiureghian AD, Polak E (2001) Reliability-based optimal design of series structural systems. Journal of Engineering Mechanics 127: 607-614
[113] Ryoo HS (2005) Robust meta-heuristic algorithm for redundancy optimization in large-scale complex systems. Annals of Operations Research 133: 209-228
[114] Sasaki M, Gen M (2003) A method of fuzzy multi-objective nonlinear programming with GUB structure by Hybrid Genetic Algorithm. International Journal of Smart Engineering Design 5: 281-288
[115] Sasaki M, Gen M (2003) Fuzzy multiple objective optimal system design by hybrid genetic algorithm. Applied Soft Computing 2: 189-196
[116] Shelokar PS, Jayaraman VK, Kulkarni BD (2002) Ant algorithm for single and multiobjective reliability optimization problems. Quality and Reliability Engineering International 18: 497-514

[117] Stocki R, Kolanek K, Jendo S, Kleiber M (2001) Study on discrete optimization techniques in reliability-based optimization of truss structures. Computers and Structures 79: 2235-2247
[118] Suman B (2003) Simulated annealing-based multi-objective algorithm and their application for system reliability. Engineering Optimization 35: 391-416
[119] Sung CS, Cho YK (1999) Branch-and-bound redundancy optimization for a series system with multiple-choice constraints. IEEE Transactions on Reliability 48: 108-117
[120] Sung CS, Cho YK (2000) Reliability optimization of a series system with multiple-choice and budget constraints. European Journal of Operational Research 127: 158-171
[121] Taguchi T, Yokota T, Gen M (1998) Reliability optimal design problem with interval coefficients using hybrid genetic algorithms. Computers & Industrial Engineering 35: 373-376
[122] Taguchi T, Yokota T (1999) Optimal design problem of system reliability with interval coefficient using improved genetic algorithms. Computers & Industrial Engineering 37: 145-149
[123] Tillman FA, Hwang CL, Kuo W (1977) Optimization techniques for system reliability with redundancy- a review. IEEE Transactions on Reliability R-26: 148-152
[124] Wattanapongsakorn N, Levitan SP (2004) Reliability optimization models for embedded systems with multiple applications. IEEE Transactions on Reliability 53: 406-416
[125] Wolfram S (1984) Cellular automata as models of complexity. Nature, 3H
[126] Yalaoui A, Chu C, Châtelet E (2005) Reliability allocation problem in a series-parallel system. Reliability Engineering and System Safety 90: 55-61
[127] Yalaoui A, Châtelet E, Chu C (2005) A new dynamic programming method for reliability & redundancy allocation in a parallel-series system. IEEE Transactions on Reliability 54: 254-261
[128] Yang JE, Hwang MJ, Sung TY, Jin Y (1999) Application of genetic algorithm for reliability allocation in nuclear power plants. Reliability Engineering and System Safety 65: 229-238
[129] You PS, Chen TC (2005) An efficient heuristic for series-parallel redundant reliability problems. Computer & Operations Research 32: 2117-2127
[130] Yun WY, Kim JW (2004) Multi-level redundancy optimization in series systems. Computer and Industrial Engineering 46: 337-346
[131] Zafiropoulos EP, Dialynas EN (2004) Reliability and cost optimization of electronic devices considering the component failure rate uncertainty. Reliability Engineering and System Safety 84: 271-284
[132] Zhao R, Liu B (2003) Stochastic programming models for general redundancy-optimization problems. IEEE Transactions on Reliability 52: 181-191
[133] Zhao R, Liu B (2004) Redundancy optimization problems with uncertainty of combining randomness and fuzziness. European Journal of Operation Research 157: 716-735