Les Cahiers du GERAD ISSN: 0711–2440

An exact approach for the redundancy allocation problem of homogeneous series-parallel multistate systems

M. Ouzineb,
I. El Hallaoui, M. Gendreau

G–2018–99

November 2018

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2018
– Bibliothèque et Archives Canada, 2018

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2018
– Library and Archives Canada, 2018

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

An exact approach for the redundancy allocation problem of homogeneous series-parallel multistate systems

Mohamed Ouzineb [a,b]
Issmail El Hallaoui [a,c]
Michel Gendreau [a,c,d]

[a] *GERAD, Montréal (Québec), Canada, H3T 2A7*

[b] *Institut National de Statistique et d'Economie Appliquée, B.P.:6217 Rabat-Instituts, Madinat Al Irfane, Rabat, Morocco*

[c] *Department of Mathematics and Industrial Engineering, Polytechnique Montréal (Québec) Canada, H3C 3A7*

[d] *CIRRELT, Montréal (Québec), Canada, H3T 1J4*

mohamed.ouzineb@gerad.ca
issmail.el-hallaoui@polymtl.ca
michel.gendreau@polymtl.ca

November 2018
Les Cahiers du GERAD
G–2018–99

Abstract: This paper develops an exact method for the redundancy allocation problem (RAP) for multistate homogeneous series-parallel systems. The problem aims to minimize the linear cost given the nonlinear reliability constraint on the system. We propose a simple 0-1 integer linear programming model and find optimal solutions for the test problems presented in previous research. The system has a finite number of performance levels varying from 0% (complete failure) to 100% (perfect function). Each level has a corresponding state probability. The system reliability is calculated using the universal generating function technique. Because of the complex nature of the problem, it is often solved by heuristics. By using an exact method, we are able to validate the solutions found by heuristics. Moreover, our method solves reasonable instances from the literature in just a few milliseconds.

Keywords: Redundancy allocation, series-parallel systems, multistate systems, universal generating function, 0-1 integer linear programming model

1 Introduction

We consider the redundancy allocation problem (RAP) of a series-parallel system (Figure 1). The system has a finite number of subsystems in series, and the failure of any subsystem implies the failure of the entire system. In each subsystem, multiple redundant components are used in parallel: the subsystem will function if at least one of its components is operational. The failure of a redundant component may however decrease the system performance. Redundant components have a cumulative effect on the overall performance.

The system is designed to achieve reliability and performance. However, while the redundant components contribute to this goal, they also increase the total cost. We wish to select a combination of the components that satisfies the system reliability and/or weight constraints while minimizing the total cost.

The complexity of the problem depends on the application. The RAP is generally an NP-hard combinatorial optimization problem [1]. The model is complex because many factors, such as allowing mixed components or taking into account new demand levels, impact the system reliability and performance. To solve optimally the problem, we have to develop simplifying assumptions (e.g., restricting each subsystem to identical components or limiting each component function to two possible states: good or failed). The components of the system are characterized by their reliability, performance, and cost; they are chosen from the relevant items available in the market. We define the system reliability to be the ability to meet the customer's performance expectation, which is represented as a piecewise cumulative load curve. We apply a universal moment generating function (UMGF) to evaluate the reliability [2, 3].

In recent years, the RAP has been applied to energy production [4], telecommunications design [5], health [6], natural disasters [7], protection [8], and logistics and transportation [9]. The solution approaches include metaheuristics [10, 11, 12, 13]. In [10], the authors apply a combination of space partitioning, genetic algorithms (GAs), and tabu search (TS). The authors in [14] apply a TS–GA algorithm to optimize the nonhomogeneous redundancy of series-parallel multistate systems (MSS). In [15], the authors use space partitioning to solve two design optimization problems: the first is the expansion scheduling of series-parallel MSS, and the second is the RAP for series-parallel binary-state systems.

Other papers on MSS include [16, 17, 18, 19], and there have been several extensive reviews [20, 21, 22, 23]. GAs have been used [3, 24, 25] to find the minimal-cost configuration of a series-parallel MSS under reliability or availability constraints.

Exact optimization techniques are an alternative to metaheuristics. To the best of our knowledge, the existing exact approaches assume that the system has only two possible states: good or failed. This is unrealistic in many applications. For example, [26] uses column generation to solve the RAP for binary series-parallel systems, while [27] uses integer linear programming. In many cases, such as power systems reliability analysis and telecommunication systems reliability analysis, the states range from 0% (complete failure) to 100% (perfect functioning). MSS reliability modeling usually considers a finite set of performance levels. The multistate version of the RAP is more complex and has not been solved using exact methods.

In this paper, we propose a simple 0-1 integer linear programming model that provides an optimal solution for the multistate RAP. We show that this approach is efficient: it is relatively easy to understand and to implement using existing solvers. To the best of our knowledge, this is the first time that the multistate RAP with a UMGF has been solved using an exact method. This is the main contribution of this paper. We examine the performance of our method via extensive computational experiments on benchmark instances.

The remainder of this paper is organized as follows. In Section 2 we provide a description of the RAP for series-parallel MSS. The UMGF method is presented in Section 3, and our solution approach is introduced in Section 4. Section 5 presents the test problems and the results, and Section 6 provides concluding remarks.

2 RAP for series-parallel multistate systems

In this section, we present a description of the problem and its standard formulation. We begin with the necessary notation.

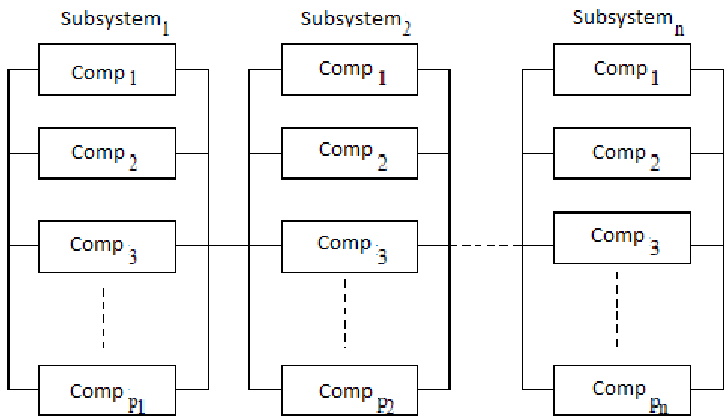

Figure 1: Example of series-parallel system

2.1 Notation

Variables	
X_{ij}	number of components of type j connected in parallel in subsystem i
Parameters	
N	number of series MSS subsystems
m_i	number of component choices available in the market for subsystem i, $i \in \{1, 2, ..., N\}$
$Max(X_{ij})$	maximum X_{ij} allowed (i.e., the upper bound on X_{ij})
X	vector: $(X_{ij})_{1 \leq i \leq N, 1 \leq j \leq m_i}$
R_0	specified minimum system reliability level
R_{ij}	binary-state reliability of component j used in subsystem i
C_{ij}	cost of component j in subsystem i
W_{ij}	nominal performance level of component j in subsystem i
T	MSS operation period
T_k	interval k of T
D_k	required MSS performance for T_k
D	vector: $(D_k)_{1 \leq k \leq K}$
W	system capacity (system performance)
m	MSS state number, $m \in \{1, 2, ..., M\}$, where 1 is the worst and M is the best
W_m	MSS steady-state performance level associated with m

2.2 Description and assumptions

In each subsystem, a number of components are connected in parallel. The subsystems themselves are connected in series, so the MSS and its components can support multiple failures. The capacity or performance of the system is a function of the number and type of components used. Redundancy improves the reliability but increases the total cost. The goal is to select the items to use so that the total cost is minimized, subject to a multistate reliability constraint. For redundancy, we assume that the same type of components are used in each subsystem. That is, for each subsystem, we must determine the component type and the number of redundant components. Each component can have only two states: good or failed.

2.3 Mathematical formulation

The total system cost is the sum of the costs of the components. The cost of $Subsystem_i$ is $\sum_{j=1}^{m_i} C_{ij} X_{ij}$. Thus, the total cost (the objective function) is:

$$C(X) \quad = \quad \sum_{i=1}^{N} \sum_{j=1}^{m_i} C_{ij} X_{ij}. \tag{1}$$

The total cost may be a nonlinear function of X_{ij} to take into account price reductions [3, 28]. The problem can be formulated as follows:

$$\text{minimize} \quad C(X) \tag{2}$$
$$\text{subject to}$$
$$R(X) \geq R_0, \tag{3}$$
$$X_{ij} \in \{0, 1, ...Max(X_{ij})\} \qquad \forall i, \quad 1 \leq i \leq N, 1 \leq j \leq m_i \tag{4}$$

Constraints (3) enforce the reliability limits. Constraints (4) specify that, for each subsystem, the number of connected components is an integer that is at most a prespecified maximum value.

3 Reliability calculation for multistate systems

This section briefly summarizes estimations of MSS reliability. We use the universal z-transform technique [29], which has proven effective for large combinatorial optimization problems [3, 15]. This technique is also called the UMGF or simply the U-function or U-transform.

For MSS, the system capacity (performance) W must be determined and compared to some demand target D to assess the system reliability R. More precisely, R is defined as $Pr(W \geq D)$, and W is based on the performance of its components.

As observed in the Introduction, the demand is represented as a piecewise cumulative load curve. The period T is divided into K intervals where each interval has a duration T_k and a required performance D_k $(k = 1, ..., K)$. The MSS reliability is given by [24, 29]:

$$R(X) \quad = \quad \frac{1}{\sum_{k=1}^{K} T_k} \sum_{k=1}^{K} R^k(X) T_k \tag{5}$$

where $R^k(X) = Pr(W \geq D_k)$, the probability that the system capacity is not lower than D_k for interval k; this depends on X.

3.1 Definition and properties of U-function

We now introduce the U-function and its properties.

Definition 1 *The U-function of a discrete random variable W is a polynomial:*

$$U(z) \quad = \quad \sum_{m=1}^{M} p_m z^{W_m}, \tag{6}$$

where W has M possible values and p_m is the probability that W is equal to W_m.

Definition 2 *The reliability R^k is given [29, 24] by the probability*

$$R^k = P[W \geq D_k] \quad = \quad \Phi(U(z) z^{-D_k}), \tag{7}$$

where Φ is a distributive operator defined by

$$\Phi(pz^w) \quad = \quad p1_{[w \geq 0]}. \tag{8}$$

Here $1_{[w \geq 0]}$ is an indicator function that is 1 if $w \geq 0$ and 0 otherwise. We have

$$\Phi(\sum_{m=1}^{M} p_m z^{w_m}) \quad = \quad \sum_{m=1}^{M} \Phi(p_m z^{w_m}). \tag{9}$$

The operator Φ satisfies Ushakov's four properties [29]:

1. $\Phi(pz^w) = pz^w$.
2. $\Phi(p_1 z^{w_1}, p_2 z^{w_2}) = p_1 p_2 z^{f(w_1,w_2)}$, where $f(w_1, w_2)$ is defined according to the system configuration.
3. $\Phi(U_1(z), ..., U_k(z), U_{k+1}(z), ...U_n(z)) = \Phi(\Phi(U_1(z), ..., U_k(z)), \Phi(U_{k+1}(z), ..., U_n(z)))$ for any k.
4. $\Phi(U_1(z), ..., U_k(z), U_{k+1}(z), ...U_n(z)) = \Phi(U_1(z), ..., U_{k+1}(z), U_k(z), ...U_n(z))$ for any k.

We now show that Equations (6)–(9) satisfy $P[W \geq D_k] = \sum_{W_m \geq D_k} p_m$. We have:

$$\begin{aligned} P[W \geq D_k] &= \Phi(U(z) z^{-D_k}) \\ &= \Phi(\sum_{m=1}^{M} p_m z^{W_m - D_k}) \\ &= \sum_{m=1}^{M} \Phi(p_m z^{W_m - D_k}) \\ &= \sum_{m=1}^{M} p_m 1_{[W_m - D_k \geq 0]} \\ &= \sum_{W_m \geq D_k} p_m. \end{aligned}$$

The operator Φ is used here to calculate the polynomial coefficients $U(z)$ by summing every term with $W_m \geq D_k$.

3.2 Series-parallel MSS reliability evaluation using U-functions

The series-parallel MSS reliability is obtained by applying the composition operators consecutively. We first calculate the U-function for a subsystem of components connected in parallel using the operator Φ over the U-function of each component. We then use the U-functions of the subsystems to obtain the reliability of the entire system.

The total performance of the parallel system is the sum of the performance of its components. The function to be used for Φ in Ushakov's second property is $f(w_1, w_2) = w_1 + w_2$. The U-function of $Subsystem_i$ containing X_{ij} parallel components is:

$$U(z) = \Phi(U_1(z), U_2(z), ..., U_{X_{ij}}(z)), \text{ where } f(w_1, w_2, ..., w_{X_{ij}}) = \sum_{e=1}^{X_{ij}} w_e. \tag{10}$$

Therefore, for a pair of components connected in parallel:

$$\Phi(U_1(z), U_2(z)) = \Phi(\sum_{i=1}^{n} P_i z^{w_i}, \sum_{j=1}^{m} Q_j z^{w_j}) = \sum_{i=1}^{n} \sum_{j=1}^{m} P_i Q_j z^{w_i + w_j} \tag{11}$$

where n and m are the numbers of possible performance levels for these components. The operator Φ is simply a product of the individual U-functions. Thus, the U-function of $Subsystem_i$ is:

$$U(z) = \prod_{l=1}^{X_{ij}} U_l(z). \tag{12}$$

We assume that each component has only two states (nominal performance or total failure). For example, let component j in subsystem i have capacity W_{ij} and reliability R_{ij}. Then, $Pr[W = W_{ij}] = R_{ij}$ and $Pr[W = 0] = 1 - R_{ij}$. The UMGF has two terms:

$$U(z) = (1 - R_{ij}) z^0 + R_{ij} z^{W_{ij}}. \tag{13}$$

Given the individual U-function defined in Equation (13), the U-function of $Subsystem_i$ with X_{ij} parallel components is:

$$U_i(z) = \Phi(U_1(z), U_2(z), ..., U_{X_{ij}}(z)) = \prod_{l=1}^{X_{ij}}[(1 - R_{ij})z^0 + R_{ij}z^{W_{ij}}]. \tag{14}$$

Under the assumption that all the components are identical, the U-function becomes:

$$U_i(z) = [(1 - R_{ij})z^0 + R_{ij}z^{W_{ij}}]^{X_{ij}} = \sum_{l=0}^{X_{ij}} \alpha_{il}(X_{ij})z^{lW_{ij}}, \tag{15}$$

$$\alpha_{il}(X_{ij}) = \text{binm}(l, R_{ij}, X_{ij}) = [\frac{X_{ij}!}{l!(X_{ij} - l)!}]R_{ij}^l(1 - R_{ij})^{X_{ij}-l}. \tag{16}$$

To evaluate the probability that $Subsystem_i$ provides a performance level exceeding D_k, the operator Φ is applied to Equation (15) as follows:

$$\Phi(U(z)z^{-D_k}) = \sum_{lW_{ij} \geq D_k} \alpha_{il}(X_{ij}). \tag{17}$$

Using Equation (7),

$$R^k(X) = Pr[W \geq D_k] = \Phi(U(z)z^{-D_k}) = \sum_{lW_{ij} \geq D_k} \alpha_{il}(X_{ij}). \tag{18}$$

Finally, from Equation (5), the subsystem reliability $R_i(X)$ is:

$$R_i(X) = \frac{1}{\sum_{k=1}^{K} T_k} \sum_{k=1}^{K} R^k(X)T_k = \frac{1}{\sum_{k=1}^{K} T_k} \sum_{k=1}^{K} \left(\sum_{lW_{ij} \geq D_k} \alpha_{il}(X_{ij}) \right) T_k. \tag{19}$$

If N subsystems are connected in series, the system reliability $R(X)$ is the product of the subsystem reliabilities [3, 28]:

$$R(X) = \prod_{i=1}^{N} R_i(X). \tag{20}$$

4 Formulation and linearization

The RAP for a series-parallel MSS has a nonlinear reliability constraint: $R(X) \geq R_0$. Using Equation (20), we obtain

$$\prod_{i=1}^{N} R_i(X) \geq R_0, \tag{21}$$

or equivalently

$$\sum_{i=1}^{N} log(R_i(X)) \geq log(R_0). \tag{22}$$

Using Equation (19), we obtain

$$\sum_{i=1}^{N} log(\frac{1}{\sum_{k=1}^{K} T_k} \sum_{k=1}^{K} \left(\sum_{lW_{ij} \geq D_k} \alpha_{il}(X_{ij}) \right) T_k) \geq R_0. \tag{23}$$

Let Y_{ijp} be a decision variable such that

$$Y_{ijp} = \begin{cases} 1 & \text{if component type } j \text{ is used } p \text{ times in subsystem } i; \\ 0 & \text{otherwise.} \end{cases}$$

The problem can then be reformulated as a linear 0-1 integer program:

$$(P2) \qquad \text{minimize } \; C(Y) = \sum_{i=1}^{N} \sum_{j=1}^{m_i} \sum_{p=1}^{Max(X_{ij})} pC_{ij}Y_{ijp} \tag{24}$$

s.t.

$$\sum_{i=1}^{N} \sum_{j=1}^{m_i} \sum_{p=1}^{Max(X_{ij})} a_{ijp}Y_{ijp} \geq \overline{R}_0, \tag{25}$$

$$\sum_{j=1}^{m_i} \sum_{p=1}^{Max(X_{ij})} Y_{ijp} = 1 \qquad \forall i, \;\; 1 \leq i \leq s, \tag{26}$$

$$Y_{ijp} \in \{0,1\} \qquad \forall i, \;\; 1 \leq i \leq s \;, \; \forall j, \;\; 1 \leq j \leq m_i, \;\; \forall p, \;\; 1 \leq p \leq Max(X_{ij}) \tag{27}$$

where

$$\overline{R}_0 = \log(R_0) \text{and}$$

$$a_{ijp} = \log \left\lceil \frac{1}{\sum_{k=1}^{K} T_k} \sum_{k=1}^{K} \left(\sum_{lW_{ij} \geq D_k} \alpha_{il}(p) \right) T_k \right\rceil$$

with $\lceil x \rceil$ being the smallest integer greater than or equal to x.

To solve (P2), we can use a standard 0-1 integer programming solver. There are many readily available packages such as IBM ILOG CPLEX, LINDO, and Xpress. For very large instances, a specialized algorithm can be used.

5 Test problems and numerical results

We use four design optimization problems (benchmarks) from the literature and two new instances to investigate our algorithm for (P2). These problems do not allow component mixing.

5.1 Notation for benchmarks

We denote each benchmark by *99xa-(b/c)-d*, where $99x$ indicates the first three characters of the first author's name in the paper where the instance was introduced; a is the number of subsystems connected in series; (b/c) means that the number of component types ranges from b to c; and d is the number of levels in the cumulative load demand curve. The first three problems, lis4-(7/11)-4, lev5-(4/9)-4, and lev4-(4/6)-3, were solved by GA and TS in [25, 3, 24, 28]. The fourth, ouz6-(4/11)-4, was solved by TS in [28], and the last two are new instances constructed as follows:

- ouz9-(4/9)-4: merge of lev5-(4/9)-4 and lev4-(4/6)-3;
- ouz15-(4/11)-4: merge of lev5-(4/9)-4, lev4-(4/6)-3, and ouz6-(4/11)-4.

The algorithm is implemented in C^{++} using IBM ILOG CPLEX. The tests were performed on an Intel Core i7 at 2.8 GHz with 8 GB of RAM, running Linux.

5.2 Benchmark data and new instances

Table 1 gives a brief description of the four benchmarks. The new instances ouz9-(4/9)-4 and ouz15-(4/11)-4 are larger: there are nine subsystems for the first problem and fifteen subsystems for the second, with 4 to 11 component types. We set the reliability index R_0 to 0.975, 0.98, and 0.99. Tables E11 and F12 present the data, and the parameters of the piecewise cumulative load demand curve are given in Table B7.

Table 1: Information for each benchmark

Problem	Information
lis4-(7/11)-4	Data (reliability, cost, nominal performance, and parameters of the cumulative load demand curve) are given in Tables A4 and A5 in A. Four subsystems with 7–11 component types.
lev5-(4/9)-4	Data are given in Tables B6 and B7 in B. Five subsystems with 4–9 component types.
lev4-(4/6)-3	Data are given in Tables C8 and C9 in C. Four subsystems with 4–6 component types.
ouz6-(4/11)-4	Data are given in Tables D10 and B7. Six subsystems with 4–11 component types.

5.3 Results

Table 2 gives the six optimal solutions. Computational times are reported in Table 2 in the column "CPU". They did not exceed 2 ms. The second column gives the settings for R_0, and the third and fourth columns contain the optimal reliability and cost. The fifth and sixth columns contain the component type and the number of components used in each subsystem, e.g., the first instance uses one type-11 component in subsystem 1, one type-7 component in subsystem 2, four type-2 components in subsystem 3, and five type-3 components in subsystem 4.

Table 2: Optimal solutions obtained by the exact method

Problem	R_0	$R(X)$	$C(X)$ (\$M)	Component type	Number of components	CPU (sec)
lis4-(7/11)-4	0.910	0.9136	14.886	11,7,2,3	1,1,4,5	0.01
	0.920	0.9202	15.075	10,7,2,3	1,1,4,5	0.02
	0.940	0.9421	17.805	1,7,5,2	5,1,3,5	0.02
	0.950	0.9518	20.049	1,3,2,2	5,2,5,5	0.02
	0.960	0.9604	21.155	10,5,2,3	1,3,4,5	0.02
	0.970	0.9711	21.907	10,3,5,3	1,3,3,5	0.02
	0.980	0.9815	22.656	10,3,2,2	1,3,5,5	0.02
	0.990	0.9911	24.305	1,3,5,2	5,3,3,5	0.01
	0.999	0.9992	26.952	1,3,2,3	6,4,6,6	0.02
lev5-(4/9)-4	0.975	0.9774	16.450	2,3,2,7,2	2,2,3,3,1	0.02
	0.980	0.9808	16.520	2,5,2,7,2	2,6,3,3,1	0.02
	0.990	0.9937	17.050	2,3,2,7,4	2,2,3,3,3	0.02
lev4-(4/6)-3	0.900	0.9102	5.986	4,3,1,5	1,2,3,2	0.02
	0.960	0.9609	7.303	2,3,1,5	2,3,3,2	0.02
	0.990	0.9917	8.328	1,3,1,2	3,3,3,5	0.01
ouz6-(4/11)-4	0.975	0.9790	11.241	3,1,2,2,3,4	4,4,5,7,2,1	0.02
	0.980	0.9802	11.369	3,1,2,2,3,4	4,5,5,8,2,1	0.03
	0.990	0.9902	12.764	3,1,2,2,3,4	4,4,4,8,2,2	0.02
ouz9-(4/9)-4	0.975	0.9756	25.1930	2,5,2,7,3,1,3,1,1	2,6,3,3,2,3,3,3,5	0.03
	0.980	0.9844	25.3780	2,3,2,7,4,1,3,1,2	2,2,3,3,3,3,3,3,5	0.02
	0.990	0.9904	25.6620	2,5,2,7,4,1,3,1,2	2,6,3,3,3,3,3,4,5	0.02
ouz15-(4/11)-4	0.975	0.9750	38.0030	2,5,2,7,4,1,1,1,1,3,1,2,2,3,4	2,6,3,4,3,3,7,4,5,4,5,5,8,2,1	0.05
	0.980	0.9802	38.3930	2,5,2,7,3,1,3,1,2,3,1,2,2,3,4	2,6,3,3,3,3,4,5,4,4,4,7,2,2	0.06
	0.990	0.9902	39.4110	2,5,2,7,3,1,3,1,1,3,1,2,2,3,4	2,6,3,3,3,3,4,5,5,5,5,8,2,2	0.05

Table 3 lists the cost obtained by the existing methods, mostly based on GAs. In most cases, the metaheuristics [28, 3, 24] give optimal or quasi-optimal solutions.

Table 3: Comparison of methods: optimal values are in bold font

Problem	R_0	Exact method cost	GA cost
lis4-(7/11)-4	0.910	**14.886**	15.852
	0.920	**15.075**	16.035
	0.940	**17.805**	**17.805**
	0.950	**20.049**	**20.049**
	0.960	**21.155**	**21.155**
	0.970	**21.907**	**21.907**
	0.980	**22.656**	**22.656**
	0.990	**24.305**	**24.305**
	0.999	**26.952**	**26.952**
lev5-(4/9)-4	0.975	**16,450**	**16.450**
	0.980	**16.520**	**16.520**
	0.990	**17.050**	17.095
lev4-(4/6)-3	0.900	**5.986**	6.348
	0.960	**7.303**	7.571
	0.990	**8.328**	**8.328**
ouz6-(4/11)-4	0.975	**11.241**	**11.241**
	0.980	**11.369**	11.608
	0.990	**12.764**	**12.764**

6 Conclusion

We have proposed an integer linear programming formulation for the multistate version of the RAP. We have found the optimal solution for each instance proposed in [28] and shown that the metaheuristics proposed in [28, 3, 24] are effective for this problem. In the future, it would be interesting to extend the method to nonhomogenous series-parallel MSSs.

Appendix A Problem 1

Table A4: Data for components available in the market [25]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
1	1	0.990	1.117	25
	2	0.996	1.310	25
	3	0.995	1.903	35
	4	0.961	1.640	35
	5	0.993	2.122	50
	6	0.957	1.910	50
	7	0.942	1.722	50
	8	0.991	2.591	72
	9	0.951	2.001	72
	10	0.986	3.284	100
	11	0.979	3.095	100
2	1	0.967	4.010	40
	2	0.914	3.450	40
	3	0.960	4.350	55
	4	0.953	4.840	78
	5	0.920	4.210	78
	6	0.950	5.800	93
	7	0.948	6.550	110
3	1	0.967	0.636	25
	2	0.952	0.448	35
	3	0.973	0.868	35
	4	0.972	0.964	50
	5	0.949	0.678	50
	6	0.988	1.096	50
	7	0.966	1.358	72
	8	0.954	1.298	72
	9	0.945	1.810	100

Table A4: Data for components available in the market [25]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} (\$M)	Nominal performance W_{ij} (%)
4	1	0.987	0.614	12.5
	2	0.985	0.883	25
	3	0.961	0.745	25
	4	0.980	0.963	30
	5	0.958	0.885	30
	6	0.974	1.338	45
	7	0.982	1.445	45

Table A5: Parameters of the cumulative load demand curve [25]

$D_k(\%)$	100	80	50	20
$T_k(h)$	4260	800	1200	2496

Appendix B Problem 2

Table B6: Data for components available in the market [3]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} (\$M)	Nominal performance W_{ij} (%)
1	1	0.980	0.590	120
	2	0.977	0.535	100
	3	0.982	0.470	85
	4	0.978	0.420	85
	5	0.983	0.400	48
	6	0.920	0.180	31
	7	0.984	0.220	26
2	1	0.995	0.205	100
	2	0.996	0.189	92
	3	0.997	0.091	53
	4	0.997	0.056	28
	5	0.998	0.042	21
3	1	0.971	7.525	100
	2	0.973	4.720	60
	3	0.971	3.590	40
	4	0.976	2.420	20
4	1	0.977	0.180	115
	2	0.978	0.160	100
	3	0.978	0.150	91
	4	0.983	0.121	72
	5	0.981	0.102	72
	6	0.971	0.096	72
	7	0.983	0.071	55
	8	0.982	0.049	25
	9	0.977	0.044	25
5	1	0.984	0.986	128
	2	0.983	0.825	100
	3	0.987	0.490	60
	4	0.981	0.475	51

Table B7: Parameters of the cumulative load demand curve [3]

$D_k(\%)$	100	80	50	20
$T_k(h)$	4203	788	1228	2536

Appendix C Problem 3

Table C8: Data for components available in the market [24]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
1	1	0.970	0.520	50
	2	0.964	0.620	80
	3	0.980	0.720	80
	4	0.969	0.890	100
	5	0.960	1.020	150
2	1	0.967	0.516	20
	2	0.914	0.916	50
	3	0.960	0.967	50
	4	0.953	1.367	75
3	1	0.959	0.214	60
	2	0.970	0.384	90
	3	0.959	0.534	180
	4	0.960	0.614	200
	5	0.970	0.783	200
	6	0.960	0.813	240
4	1	0.989	0.683	25
	2	0.979	0.645	25
	3	0.980	0.697	30
	4	0.960	1.190	70
	5	0.980	1.260	70

Table C9: Parameters of the cumulative load demand curve [24]

$D_k(\%)$	100	80	40
$T_k(h)$	20	30	50

Appendix D Problem 4

Table D10: Data for components available for Problem 4 [28]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
1	1	0.932	1.590	27.3
	2	0.998	0.515	27.7
	3	0.983	0.225	49.8
	4	0.927	3.220	52.5
	5	0.959	4.020	62.0
	6	0.955	4.270	66.4
	7	0.984	3.670	84.6
	8	0.918	4.630	90.7
	9	0.939	1.010	97.0
	10	0.988	0.779	124
	11	0.984	3.130	129
2	1	0.989	0.050	35.9
	2	0.923	1.290	44.7
	3	0.900	0.204	51.4
	4	0.946	2.220	63.2
	5	0.917	0.872	68.8
	6	0.962	1.830	81.8
	7	0.994	0.294	82.0
	8	0.984	2.810	115
3	1	0.931	3.620	34.7
	2	0.950	0.475	41.0

Table D10: Data for components available for Problem 4 [28]

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
	3	0.911	1.170	41.4
	4	0.956	0.793	43.6
	5	0.966	3.740	48.6
	6	0.992	4.590	59.6
	7	0.929	1.740	66.2
	8	0.968	1.720	91.9
	9	0.901	1.300	121
4	1	0.915	2.490	25.1
	2	0.908	0.078	28.8
	3	0.928	1.370	50.2
	4	0.944	4.470	129
5	1	0.908	1.550	34.9
	2	0.980	4.920	64.3
	3	0.964	2.650	108
	4	0.924	4.720	126
6	1	0.965	3.220	24.8
	2	0.927	2.890	47.3
	3	0.986	3.410	58.8
	4	0.983	1.920	107
	5	0.991	4.510	120
	6	0.954	4.580	125

Appendix E Problem 5

Table E11: Data for components available for Problem 5

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
1	1	0.980	0.590	120
	2	0.977	0.535	100
	3	0.982	0.470	85
	4	0.978	0.420	85
	5	0.983	0.400	48
	6	0.920	0.180	31
	7	0.984	0.220	26
2	1	0.995	0.205	100
	2	0.996	0.189	92
	3	0.997	0.091	53
	4	0.997	0.056	28
	5	0.998	0.042	21
3	1	0.971	7.525	100
	2	0.973	4.720	60
	3	0.971	3.590	40
	4	0.976	2.420	20
4	1	0.977	0.180	115
	2	0.978	0.160	100
	3	0.978	0.150	91
	4	0.983	0.121	72
	5	0.981	0.102	72
	6	0.971	0.096	72
	7	0.983	0.071	55
	8	0.982	0.049	25
	9	0.977	0.044	25
5	1	0.984	0.986	128
	2	0.983	0.825	100
	3	0.987	0.490	60
	4	0.981	0.475	51

Table E11: Data for components available for Problem 5

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} (\$M)	Nominal performance W_{ij} (%)
6	1	0.970	0.520	50
	2	0.964	0.620	80
	3	0.980	0.720	80
	4	0.969	0.890	100
	5	0.960	1.020	150
7	1	0.967	0.516	20
	2	0.914	0.916	50
	3	0.960	0.967	50
	4	0.953	1.367	75
8	1	0.959	0.214	60
	2	0.970	0.384	90
	3	0.959	0.534	180
	4	0.960	0.614	200
	5	0.970	0.783	200
	6	0.960	0.813	240
9	1	0.989	0.683	25
	2	0.979	0.645	25
	3	0.980	0.697	30
	4	0.960	1.190	70
	5	0.980	1.260	70

Appendix F Problem 6

Table F12: Data for components available for Problem 6

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} (\$M)	Nominal performance W_{ij} (%)
1	1	0.980	0.590	120
	2	0.977	0.535	100
	3	0.982	0.470	85
	4	0.978	0.420	85
	5	0.983	0.400	48
	6	0.920	0.180	31
	7	0.984	0.220	26
2	1	0.995	0.205	100
	2	0.996	0.189	92
	3	0.997	0.091	53
	4	0.997	0.056	28
	5	0.998	0.042	21
3	1	0.971	7.525	100
	2	0.973	4.720	60
	3	0.971	3.590	40
	4	0.976	2.420	20
4	1	0.977	0.180	115
	2	0.978	0.160	100
	3	0.978	0.150	91
	4	0.983	0.121	72
	5	0.981	0.102	72
	6	0.971	0.096	72
	7	0.983	0.071	55
	8	0.982	0.049	25
	9	0.977	0.044	25
5	1	0.984	0.986	128
	2	0.983	0.825	100
	3	0.987	0.490	60
	4	0.981	0.475	51

Table F12: Data for components available for Problem 6

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
6	1	0.970	0.520	50
	2	0.964	0.620	80
	3	0.980	0.720	80
	4	0.969	0.890	100
	5	0.960	1.020	150
7	1	0.967	0.516	20
	2	0.914	0.916	50
	3	0.960	0.967	50
	4	0.953	1.367	75
8	1	0.959	0.214	60
	2	0.970	0.384	90
	3	0.959	0.534	180
	4	0.960	0.614	200
	5	0.970	0.783	200
	6	0.960	0.813	240
9	1	0.989	0.683	25
	2	0.979	0.645	25
	3	0.980	0.697	30
	4	0.960	1.190	70
	5	0.980	1.260	70
10	1	0.932	1.590	27.3
	2	0.998	0.515	27.7
	3	0.983	0.225	49.8
	4	0.927	3.220	52.5
	5	0.959	4.020	62.0
	6	0.955	4.270	66.4
	7	0.984	3.670	84.6
	8	0.918	4.630	90.7
	9	0.939	1.010	97.0
	10	0.988	0.779	124
	11	0.984	3.130	129
11	1	0.989	0.050	35.9
	2	0.923	1.290	44.7
	3	0.900	0.204	51.4
	4	0.946	2.220	63.2
	5	0.917	0.872	68.8
	6	0.962	1.830	81.8
	7	0.994	0.294	82.0
	8	0.984	2.810	115
12	1	0.931	3.620	34.7
	2	0.950	0.475	41.0
	3	0.911	1.170	41.4
	4	0.956	0.793	43.6
	5	0.966	3.740	48.6
	6	0.992	4.590	59.6
	7	0.929	1.740	66.2
	8	0.968	1.720	91.9
	9	0.901	1.300	121
13	1	0.915	2.490	25.1
	2	0.908	0.078	28.8
	3	0.928	1.370	50.2
	4	0.944	4.470	129
14	1	0.908	1.550	34.9
	2	0.980	4.920	64.3
	3	0.964	2.650	108
	4	0.924	4.720	126

Table F12: Data for components available for Problem 6

$Subsystem_i$	Component type j	R_{ij}	Cost C_{ij} ($M)	Nominal performance W_{ij} (%)
15	1	0.965	3.220	24.8
	2	0.927	2.890	47.3
	3	0.986	3.410	58.8
	4	0.983	1.920	107
	5	0.991	4.510	120
	6	0.954	4.580	125

References

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

[2] I. Ushakov, Optimal standby problems and a universal generating function, Sov. J. Computing System Science 25(4) (1987) 79–82.

[3] G. Levitin, A. Lisnianski, H. Ben-Haim, D. Elmakis, Structure optimization of power system with different redundant elements, Electric Power Systems Research 43(1) (1997) 19–27.

[4] L. Zhitao, T. CherMing, L. Feng, A reliability-based design concept for lithium-ion battery pack in electric vehicles, Reliability Engineering & System Safety 134 (2015) 169–177.

[5] S. Olli, The effect of introducing increased-reliability-risk electronic components into 3rd generation telecommunications systems, Reliability Engineering & System Safety 89(2) (2005) 208–218.

[6] A. O. Ikenna, T. Longbin, Reliability analysis and optimisation of subsea compression system facing operational covariate stresses, Reliability Engineering & System Safety 156 (2016) 159–174.

[7] W. Yezhou, C. Chen, W. Jianhui, Research on resilience of power systems under natural disasters–a review, IEEE Transactions on Power Systems 31(2) (2016) 1604–1613.

[8] H. Liisa, P. Urho, K. Mikko, J. Jussi, A method for analysing the reliability of a transmission grid, Reliability Engineering & System Safety 93(2) (2008) 277–287.

[9] J. Xiuhong, D. Fuhai, T. Heng, W. Xuedong, Optimization of reliability centered predictive maintenance scheme for inertial navigation system, Reliability Engineering & System Safety 140 (2015) 208–217.

[10] M. Ouzineb, M. Nourelfath, M. Gendreau, A heuristic method for non-homogeneous redundancy optimization of series-parallel multi-state systems, Journal of Heuristics 17(1) (2011) 1–22.

[11] Y. Liang, Y. Chen, Redundancy allocation of series-parallel systems using a variable neighborhood search algorithm, Reliability Engineering and System Safety 92 (2007) 323–331.

[12] N. Nahas, M. Nourelfath, D. Ait-Kadi, Coupling ant colony and the degraded ceiling algorithm for the redundancy allocation problem of series-parallel systems, Reliability Engineering and System Safety 92(2) (2007) 211–222.

[13] M. Agarwal, R. Gupta, Homogeneous redundancy optimization in multi-state series-parallel systems: A heuristic approach, IIE Transactions 39(3) (2007) 277–289.

[14] N. Nahas, M. Nourelfath, M. Gendreau, Selecting machines and buffers in unreliable assembly/disassembly manufacturing networks, International Journal of Production Economics 154 (2014) 113–126.

[15] M. Ouzineb, M. Nourelfath, M. Gendreau, An efficient heuristic for reliability design optimization problems, Computers & Operations Research 37(2) (2010) 223–235.

[16] D. Ait-Kadi, M. Nourelfath, Availability optimization of fault-tolerant systems, in: International Conference on Industrial Engineering and Production Management (IEPM'2001), Quebec, Canada, 2001.

[17] R. Billinton, R. Allan, Reliability Evaluation of Power Systems, Pitman, 1990.

[18] J. Murchland, Fundamental concepts and relations for reliability analysis of multi-state systems, Reliability and Fault Tree Analysis, SIAM, Philadelphia: ed. R. Barlow, J. Fussell, 1975.

[19] M. Nourelfath, D. Ait-Kadi, I. Soro, Availability modeling and optimization of reconfigurable manufacturing systems, Journal of Quality in Maintenance Engineering 9(3) (2003) 284–302.

[20] G. Levitin, Universal generating function in reliability analysis and optimization, Springer-Verlag, 2005.

[21] G. Levitin, A. Lisnianski, A new approach to solving problems of multi-state system reliability optimization, Quality and Reliability Engineering International 47(2) (2001) 93–104.

[22] A. Lisnianski, G. Levitin, Multi-state System Reliability: Assessment, Optimization and Applications, World Scientific, 2003.

[23] I. Ushakov, G. Levitin, A. Lisnianski, Multi-state system reliability: From theory to practice, in: Proceedings of 3rd International Conference on Mathematical Methods in Reliability (MMR), Trondheim, Norway, 2002, pp. 635–638.

[24] G. Levitin, A. Lisnianski, H. Ben-Haim, D. Elmakis, Redundancy optimization for series-parallel multi-state systems, IEEE Transactions on Reliability 47(2) (1998) 165–172.

[25] A. Lisnianski, G. Levitin, H. Ben-Haim, D. Elmakis, Power system structure optimization subject to reliability constraints, Electric Power Systems Research 39(2) (1996) 145–152.

[26] L. Zia, D. Coit, Redundancy allocation for series-parallel systems using a column generation approach, IEEE Transactions on Reliability 59 (2010) 706–717.

[27] A. Billionnet, Redundancy allocation for series-parallel systems using integer linear programming, IEEE Transactions on Reliability 57(3) (2008) 507–516.

[28] M. Ouzineb, M. Nourelfath, M. Gendreau, Tabu search for the redundancy allocation problem of homogenous series parallel multi-state systems, Reliability Engineering and System Safety 93(8) (2008) 1257–1272.

[29] I. Ushakov, Universal generating function, Sov. J. Computing System Science 24(5) (1986) 118–129.