



Multi-objective redundancy allocation problem considering instantaneous availability, reparability, interference factor and load share

Seyed Mohammad Mortazavi¹ · Seyed Hosein Torabi²

Received: 8 February 2019 / Accepted: 24 September 2019 / Published online: 8 October 2019
© Society for Reliability and Safety (SRESA) 2019

Abstract

Redundancy Allocation Problem (RAP) is one of the most well-known and widely applicable optimization problems that has been considered by the researchers and designers over the recent decades. Several factors influence the redundancy allocation decisions in a system. Some of these factors are as follows: availability in repairable systems, redundant system configuration in parallel or standby systems, and time and dependent failures in case of common cause factors (CCF) and load share. These features should be incorporated into the RAP modeling to obtain logical/realistic solutions. In this paper, a RAP in a k -out-of- n system by considering several features of availability, reparability, interference factor, and load sharing is formulated. The Markov chain is used to formulate the RAP and the repair rate is assumed to be dependent on the number of repairmen. In other words, it is assumed that the higher the number of repairmen is, the higher the repair rate and as a result, the higher the availability of the system is; however, in real situations, it is possible that the excessive increase in the number of repairmen decreases the availability due to the repairmen work interference. To resolve this problem, a parameter called interference factor is defined and incorporated into the modeling and calculations for each subsystem. RAP is among the optimization problems that are categorized under NP-hard class of problems. Therefore, NSGA-II algorithm is utilized to solve the proposed model. Finally, a special problem is solved as a sample and the obtained results are discussed.

Keywords Redundancy allocation problem · k -out-of- n · Load share · Interference factor · Instantaneous availability

1 Introduction

Most of the systems are made up of several subsystems, each of which is comprised of some components. Subsystems are connected to each other in series, such that failure of a single subsystem leads to total system failure. Therefore, to avoid total system failure in case of failure of one component, it is better to put more than one component in parallel in each subsystem. Increased number of components in each subsystem leads to increase of system cost, weight, and volume

(Chern (1992)). Hence, system designers should determine number of redundant components in such a way that the system configuration is optimal, which means costs are minimized, availability is maximized, and constraints pertaining to the system weight and volume are taken into account.

Redundancy Allocation Problem (RAP) is one of the most authentic and best-known reliability optimization problems. In such problems, it is posited that several subsystems are connected together in series, such that each subsystem, $i \in \{1, 2, \dots, N\}$, can be constituted of several (n) parallel components (Fig. 1). In RAP, the aim is to select the optimal number of parallel components in each subsystem (Chern 1992). For the RAP to determine the best and most appropriate solution, for system designers, the real system features must be incorporated into RAP modeling and these conditions must be integrated into the modeling. Some of the most important features are: availability (system repair), redundant system configuration (parallel and standby), and time and dependent failures (CCF and Load share).

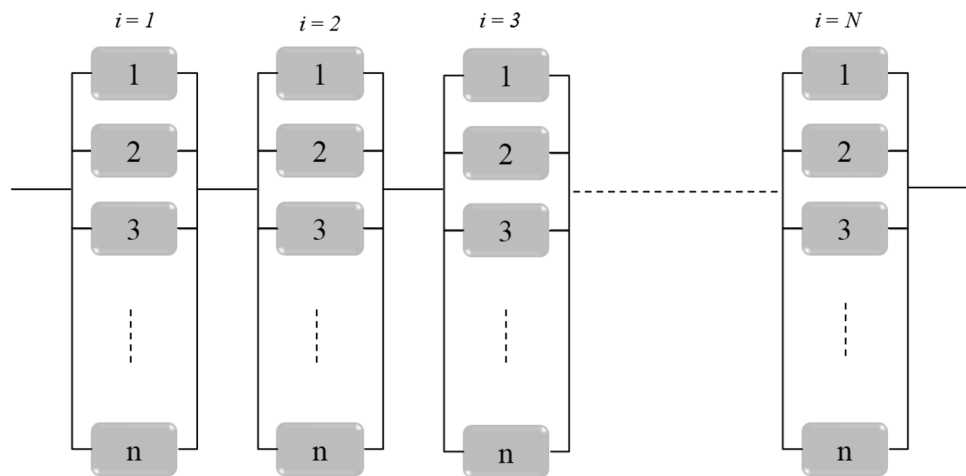
✉ Seyed Mohammad Mortazavi
seyedmohammadmortazavi@gmail.com

Seyed Hosein Torabi
htorabi@hit.edu.cn

¹ Department of Industrial Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran

² Faculty of Aerospace Engineering, Khaje Nasir Toosi University of Technology, Tehran, Iran

Fig. 1 Series parallel system of RAP



The term ‘availability’ refers to a property of systems with repair capability or checking after each period of operation (mission). Integrating the repair in reliability allocation models exerts a significant effect on their outcome. Increased repair rate is accompanied by increased availability and decreased repair rate is accompanied by decreased availability (Lambert et al. 1971). In recent years, the discussion on repair in RAP has received considerable attention. Arabi and Jahromi (2012) modeled the steady-state availability of a system with several redundant subsystems according to the cold standby strategy considering repairable components. They constructed their model using the Markov process. Lins and Droguett (2011) considered the system structure and the number of maintenance groups and provided comprised solutions set in their proposed methodology. This system was subject to imperfect repairs. They presented a Multi-Objective Genetic Algorithm (MOGA) in combination with discrete event simulation to solve the problem. Liu (2015) considered an availability optimization problem with repairable components with different constraints such as weight, volume, the required level of components reliability, and cost. To solve the problem, he developed a redundancy allocation heuristic method by combining four algorithms constituted of tabu search, simulated annealing, non-equilibrium simulated annealing, and the genetic algorithm. He applied the sensitivity analysis for the proposed approach under design limitation. A common categorization of RAP optimization is based on reparability of components in the system and includes reliability optimization and availability optimization. Zoulfaghari et al. (2014a) evaluated a system with configuration for both repairable and non-repairable components concurrently by presenting a Mixed Integer Nonlinear Programming (MINLP) model for the availability optimization problem. To solve their model, they developed a Genetic Algorithm (GA).

Guo et al. (2014) considered a parallel-series system with the repairable components. In their paper, failure rate, repair

rate, and comparative factors were considered as uncertain variables. Also, for solving the Multi-Objective RAP (MORAP), they proposed an efficient Non-Dominated Sorting GA (NSGA-II). Ebrahimipour and Sheikhalishahi (2011) investigated a multi-objective reliability redundancy allocation problem and solved their proposed model to obtain the number of components and reliability of each component in the subsystems. In their paper, parameters were considered as fuzzy numbers. To study effects of the repair strategy, several approaches under various situations were investigated. To increase the system availability, non-identical multi-state components can be added as a redundant component in the parallel configuration in the subsystem. Noureifath et al. (2012) presented a model for investigating redundancy and imperfect preventive maintenance planning optimization in the series–parallel multi-state degraded systems. They utilized Markov process to model and analyze the repair rates. Xie et al. (2014) presented an operational availability maximization model with two decision variables of the component redundancy and the number of spares stocking under cost and physical limitation. In their paper, a single repairable k -out-of- n system under various shut-off rules was modeled by a developed continuous-time Markov chain. Some main parameters of operational availability and spare parts availability were calculated using the model. In the paper by a multi-objective Joint Availability Redundancy Allocation Problem (JARAP) in the series–parallel system was solved using a Simulation-Based Optimization (SBO) method by Attar et al. (2017). With an emphasis on the developed SBO method, they considered random free-distributed time for failures and repair times under several standby conditions. To solve the model, two efficient algorithms were used: a Non-Dominated Sorting Genetic Algorithm (NSGA-II) and a Strength Pareto Evolutionary Algorithm (SPEA2). Khalili-Damghani et al. (2013) proposed a Dynamic Self-Adaptive Multi-Objective Particle Swarm Optimization (DSAMOPSO) method to solve the

binary-state Multi-Objective Reliability Redundancy Allocation Problems (MORAPs). Different properties of their proposed method made it robust and competitive, among other existing methods. A procedure based on an extended version of efficient method and Data Envelopment Analysis (DEA) was proposed to solve binary-state Multi-Objective RAP series–parallel problem by Khalili-Damghani and Amiri (2012). Other innovative ways have been used to solve problems of RAP. Khalili-Damghani et al. (2014) also proposed a Decision Support System (DSS) to efficiently solve the Multiple Objective Decision Making (MODM) problems by producing a Pareto front with a Decision Maker (DM) preferred resolution. The core of proposed DSS is based on Topsis module, modified efficient ε -constraint module, and DEA module. Different applications of metaheuristic methods and MODM methods in RAP can be found in the literature (Ardakan and Rezvan 2018; Yeh 2018; Dolatshahi-Zand and Khalili-Damghani 2015; Li et al. 2010; Taboada et al. 2007; Zio and Bazzo 2011).

There are the different structures in a redundant system, including parallel, k -out-of- n , hot standby, warm, and cold standby. The difference between parallel and other structures is the operating condition of all components. Chen and You (2005) considered a series–parallel redundant reliability problem, in which both the multiple component choices of each subsystem and the redundancy levels of every selected component were to be decided simultaneously so as to maximize the system reliability. In another study, Soylu and Ulusoy (2011) studied a bi-objective redundancy allocation problem on a series–parallel system with component level redundancy strategy. Their main aim was to maximize the minimum subsystem reliability, while minimizing the overall system cost. Gen and Yun (2006) introduced several variations of reliability design problem with parallel configurations such as reliability design problems of redundant system, reliability design problems with alternative design, reliability design problems with time-dependent reliability, reliability design problems with interval coefficients, and reliable networks design problems. They have also described various GA-based approaches for the problems. Bhunia et al. (2010) studied reliability stochastic optimization problem in the series–parallel systems with various limitations of resources. For solving the problem, they considered reliability of each component as fuzzy-random quantities with known probability distributions and fuzzy membership functions.

Chambari et al. (2012) presented a model with two objectives: of maximizing the reliability and minimizing the cost of the system in a RAP. In their paper, two redundancy strategies were considered: active and cold standby. They had also effectively solved the model using NSGA-II and MOPSO. Kim and Kim (2017) addressed the Reliability Redundancy Allocation Problem (RRAP) of either active or

cold standby components by considering an optimal redundancy strategy. For modeling purposes, they utilized Markov chains and solved their proposed model using a parallel GA.

There is a limited number of studies on a specific operational time in RAP. Specifically, a specific time implies that systems can or should be operational within a short period of time, i.e., it should respond to a situation instantaneously. For example, consider one-shot systems, which should operate at a specific and very short time. Therefore, it is crucial to consider operation time in RAP modeling. This can be achieved by utilizing Markov chains. Amiri and Ghassemi-Tari (2007) proposed a methodology based on continuous-time Markov chain for analyzing system availability with instantaneous responses. Their proposed method was applicable to series, parallel and k -out-of- n system configurations. An instantaneous availability model for repairable multi-state system (MSS) was studied by Yu et al. (2014). Their model was created from combination of both Markov process and Universal Generating Function (UGF). Xu and Hu (2013) investigated instantaneous availability of a kind of repairable system with preventive maintenance using Markov chain. They also developed a time-dependent solution, which is essential for analyzing the instantaneous availability of the repairable system.

Types of dependent failures in reliability engineering are introduced in (Mortazavi et al. 2016; Mortazavi et al. 2017). Load share and common cause failure (CCF) are among the most important dependent failures. da Costa Bueno (2005) utilized a mathematical expression to allocate spare components in a k -out-of- n system and assumed that components are dependent on each other. Li et al. (2010) studied the heterogeneous redundancy optimization of multi-state series–parallel systems. Their main aim was minimizing costs and finding an optimal level of redundancy. In this regard, they incorporated CCF into their proposed model and concluded that considering CCF results in a different redundancy allocation strategy compared to the case where CCF is absent. Ramirez-Marquez and Coit (2007) formulated a RAP in the presence of CCF and presented three non-linear optimization models categorized based on computational time, degree of similarity to the true system behavior and CCF modeling. After solving models, they concluded that considering CCF significantly influences the outcomes. Arabi and Jahromi (2013) designed a new model for RAP using Markov chain. Two types of decision variables, redundancy level and number of repairmen, are directly calculated in the objective function. In the designed model, load share was also considered, i.e., failure of a component in a subsystem affects the load on others.

To create an applicable model of the RAP, real-world features of the problem should be incorporated. Specifically, as salient matter, the repair rate affects availability of each system. Therefore, in this paper, a parameter named work

interference factor is defined to model influence of the number of repairmen on the repair rate in real-world situations. As previously mentioned, structure of redundant systems can be of different types. One of the most important structures, which is considered the least in RAP, is the k -out-of- n structure. In this structure, the dependent failure decreases availability of the system and it should be considered in reliability analyses. The scarcity of research on RAP in k -out-of- n systems under realistic situations was our main inspiration in this paper, which led to development of a dynamic RAP model considering the realistic assumptions, including time, work interference factor, repairable components, and their dependent and independent failures.

2 Model development of RAP

As explained in Sect. 1, system designers seek to increase system availability; hence, in recent years, many researchers endeavored to increase system availability through various models and techniques. In this section, a model along with its solution algorithm for a RAP in a k -out-of- n system considering load share, reparability, interference factor, and instantaneous response availability will be discussed.

2.1 Problem description and assumptions

Assume a system with m different subsystems connected in series, each has n components with k -out-of- n configuration. In addition, suppose that the failure time of each component in the i^{th} subsystem follows exponential distribution with parameter λ . Exponential distribution is commonly used for its mathematical simplicity and sufficiently realistic description of life time and time to failure (Robinson and Neuts 1989; Rausand and Høyland 2004; Cui and Li 2007). For instance, Çekyay and Özekici (2015) and Kuo et al. (2014) made the assumption that both failure and repair rates follow exponential distribution, considering a k -out-of- n configuration in which failure of k out of n components (in each subsystem) results in the whole system failure.

Components in each subsystem maintain an interdependency of load share type (the subsystems themselves are independent). Load share is a kind of dependent failure in redundant systems in which failure in one redundant system component increases the load on the surviving ones, thus increasing their failure rate. The capacity flow model is a common functional model for computing failure rate of redundant system components with constant failure rate and load share (Pozsgai et al. (2003). Equation (1) presents the formulation of this model for computing failure rate in each subsystem.

$$\lambda_{ij} = \left(\frac{n_i}{n_i - j} \right)^{\gamma_i} \cdot \lambda_{i0}, \quad (1)$$

where, n_i refers to the number of components in the i^{th} subsystem, j is the number of failed components in the subsystem, γ_i is the load factor for the i^{th} subsystem, λ_{i0} represents the initial failure rate for the i^{th} subsystem and λ_{ij} denotes the failure rate of surviving components after failure of j components in the i^{th} subsystem. Load share is common in most redundant systems, including electric generators, water pumps, cable-stayed bridges, CPUs, etc. (Shao and Lamberson 1991).

Components in each subsystem are repairable. Each component can be repaired by a single repairman or more repairmen. Each subsystem may suffer failure in one or several components. In such case, a single repairman or a team of repairmen can repair one failed component at any moment. Here, we assume that each component has a constant repair rate and the repair time follows exponential distribution. In addition, suppose that transfer of repairmen among subsystems is not allowed, i.e., each repairman repairs a specific subsystem and is assigned to that specific subsystem. Each subsystem can have y_i identical repairmen. There is a direct relationship between the number of repairmen and repair rate. More specifically, assume that μ_i represents the repair rate for each component in the i^{th} subsystem and y_i specifies the number of repairmen for the i^{th} subsystem, then the repair rate of this subsystem is obtained by $y_i \times \mu_i$. In real-world situations, due to the work interference among repairmen, increased number of repairmen does not necessarily lead to an increase in repair rate. High degree of work interference reduces repair rate; thus, it diminishes system availability. Equation (2) is a way of modeling the relationship between the number of repairmen and the repair rate, where $\alpha \in (0, 1)$ is the work interference factor. A value close to 1 for the interference factor indicates high work interference among repairmen, while a value close to 0 indicates low work interference.

$$\mu_i \times y_i \times (1 - \alpha_i)^{(y_i-1)}. \quad (2)$$

As an illustration, assume that the repair rate and the interference factor of a subsystem are equal to 0.02 and 0.2, respectively. Table 1 presents repair rate versus number of repairmen. As can be seen, increasing the number of repairmen up to five repairmen leads to increase in repair rate. However, repair rate declines with six or more repairmen due to the work interference. RAP modeling must consider work interference to prevent unnecessary increase in the number of repairmen. In case of high work interference among repairmen, increasing the number of repairmen not only reduces the availability of the system, but imposes higher costs on the systems (increased maintenance costs).

It should be noted that the number of components in each subsystem is limited and varies between l_{\min} and l_{\max} . The fact that each subsystem has a k -out-of- n configuration

Table 1 Effect of the number of repairmen on the repair rate

Number of repairmen	Repair rate
1	0.020
2	0.032
3	0.038
4	0.041
5	0.041
6	0.039
7	0.037
8	0.034
9	0.030
10	0.027

necessitates the existence of at least two components in each subsystem, i.e. $l_{min} > 2$. Similarly, the number of repairmen in each subsystem lies in the range $[\eta_{min}, \eta_{max}]$. All components in the subsystem are repairable (All components are fully repaired); therefore, each subsystem requires at least one repairman, i.e. $\eta_{min} > 1$.

Since k and n are decision variables, each subsystem can have a specific configuration; therefore, each system is thought to have different installation costs. Furthermore, since the number of repairmen may vary from one subsystem to another, repairman fees are considered to be different for each subsystem. Moreover, the components function in a binary (not multi-state) manner.

With the above descriptions and assumptions regarding the addressed RAP, model of the current problem can be formulated as follows.

2.2 Notation

See Table 2.

2.3 Mathematical model of RAP

Based on the forgoing assumptions and Table 2 of symbols, the mathematical RAP model is as follows:

$$\text{Max } A(t) = \prod_{i=1}^N A_i(n_i, k_i, y_i, \lambda_{i0}, \lambda_{ij}, \gamma_i, \mu_i, \alpha_i, t), \tag{3}$$

$$\text{Min } \sum_{i=1}^N (c_i n_i + h_i y_i), \tag{4}$$

s.t.,

$$\sum_{i=1}^N w_i n_i \leq W, \tag{5}$$

Table 2 Symbols

$A(t)$	Availability of total system at specific time t
$A_i(t)$	Availability of i th subsystem at specific time t
N	Number of subsystems
λ_{i0}	Initial failure rate of operating component in subsystem i
λ_{ij}	Failure rate of surviving components when j components are failed in subsystem i
γ_i	Load factor in subsystem i
μ_i	Repairing rate for failed component in subsystem i
α_i	Interference factor in subsystem i
c_i	Cost of each component in subsystem i
h_i	Fee of each repairman in subsystem i
w_i	Weight of each component in subsystem i
v_i	Volume of each component in subsystem i
C	Allowed cost of system
W	Allowed weight of system
V	Allowed volume of system
n_i	Number of components used in subsystem i
k_i	Minimum number of components required for subsystem i
y_i	Number of repairmen used in subsystem i
l_{max}	Maximum number of components in each subsystem
l_{min}	Minimum number of components in each subsystem
η_{max}	Maximum number of repairmen in each subsystem
η_{min}	Minimum number of repairmen in each subsystem

$$\sum_{i=1}^N v_i n_i \leq V, \tag{6}$$

$$l_{min} \leq n_i \leq l_{max} \tag{7}$$

$$\eta_{min} \leq y_i \leq \eta_{max}. \tag{8}$$

The two objective functions in the forgoing model are in conflict with each other. The first objective is to maximize total system availability at a specific time t and the second is to minimize the total cost. Since all subsystems are arranged in series, total system availability is determined by multiplying $A_i(t)$ of each subsystem. Hence, it is imperative to compute the $A_i(t)$ of each subsystem prior to computation of $A(t)$. Great number of studies has developed various methods to compute the availability of k -out-of- n systems (Mortazavi et al. 2016; Carpitella et al. 2017; Li et al. 2016). In the present paper, the value of $A_i(t)$, which is a function of number of components (n_i), number of (k_i), number of repairman (y_i), dependent failure rate (λ_{ij}), repair rate (μ_i), interference factor (α_i), and specific time (t), is computed using Markov chain and transition matrix. According to the forgoing discussion, each subsystem can have its specific configuration. For system designers, it is of vital importance to ascertain the proper configuration and number of repairmen for each subsystem.

Figure 2 illustrates the transition diagram for a k -out-of- n system. In this diagram, each state presents the number of failed components in the subsystem. Hence, state 0 denotes absence of failed components in the system (all components are operating) and state n indicates failure of all subsystem components. It is worth mentioning that all components in each subsystem operate at moment 0.

In Fig. 2, each state includes two events: if a component fails, the system enters the next state, and if the failed components are repaired, the system returns to the previous state. As mentioned previously, components in each subsystem maintain an interdependency of load share type. Failure of one component is followed by increased load on others, hence their increased failure rate. Under such circumstance, failure rate varies from one state to the next, which can be computed by Eq. (1). Besides, there are n_i components arranged in series in each subsystem; therefore, transition rate for each state equals $(n_i - j) \times \lambda_{ij}$ (i represents subsystem index and j is the number of failed components in each state).

Repair rate for any state is dependent on the number of repairmen and the interference factor, which is determined by Eq. (2). Due to interference in maintenance activities which increases carelessness and indiscipline, not only increased number of repairmen does not necessarily entail increased repair rate, but also it might even reduce repair rate, resulting in reduced availability for each subsystem.

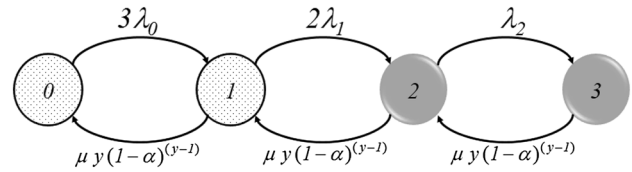


Fig. 3 Transition diagram for 2-out-of-3 configuration

In k -out-of- n systems, if k out of n system components fail, the whole system is considered a failure. Given the transition diagram in Fig. 2, in which each state indicates the number of failed components, the availability of each system is determined as follows:

$$A_i(t) = \sum_{j=0}^{k-1} p_{ij}(t). \tag{9}$$

For instance, Fig. 3 illustrates the transition diagram for a 2-out-of-3 system. Availability of this system is equal to the sum of probabilities of states 0 and 1 (states marked in green).

Computation of failure probabilities requires the conversion of the transition diagram in Fig. 2 into a transition rate matrix named Q here. Q is a square matrix since the number of rows and columns are equal to the number of transition matrix states. For example, Q_{12} shows transition rate from state 1 to state 2. The transition matrix for the transition diagram in Fig. 2 is based on the following:

$$Q = \begin{bmatrix} -n_i\lambda_{i0} & n_i\lambda_{i0} & 0 & \dots & 0 & 0 \\ \mu_i y_i (1 - \alpha_i)^{(y_i-1)} & -((\mu_i y_i (1 - \alpha_i)^{(y_i-1)} + (n_i - 1)\lambda_{i1})) & (n_i - 1)\lambda_{i1} & \dots & 0 & 0 \\ 0 & \mu_i y_i (1 - \alpha_i)^{(y_i-1)} & -((\mu_i y_i (1 - \alpha_i)^{(y_i-1)} + (n_i - 1)\lambda_{i2})) & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \mu_i y_i (1 - \alpha_i)^{(y_i-1)} & -((\mu_i y_i (1 - \alpha_i)^{(y_i-1)} + (n_i - 1)\lambda_{i(n-2})) & (n_i - 1)\lambda_{i(n-2)} & 0 \\ 0 & 0 & 0 & (\mu_i y_i (1 - \alpha_i)^{(y_i-1)}) & -((\mu_i y_i (1 - \alpha_i)^{(y_i-1)} + (n_i - 1)\lambda_{i(n-1})) & (n_i - 1)\lambda_{i(n-1)} \\ 0 & 0 & 0 & 0 & (\mu_i y_i (1 - \alpha_i)^{(y_i-1)}) & -(\mu_i y_i (1 - \alpha_i)^{(y_i-1)}) \end{bmatrix}, \tag{10}$$

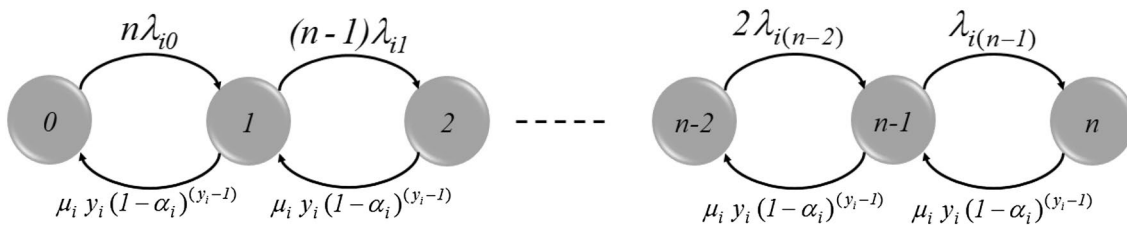


Fig. 2 Transition diagram for k -out-of- n configuration

Amiri and Ghassemi-Tari (2007) developed an equation to compute system availability (Eq. (11)). Using Markov chain and transition matrix, they determined time-based availability functions. Using Eq. (11), the availability functions of a k -out-of- n system versus time can be determined. The obtained matrix is denoted by $P(t)$. Due to memory-less property of exponential distribution, the probability of failing n components at time t which is denoted by $P_n(t)$ depends only on probability of failing n components at time $t=0$ denoted by $P_n(0)$.

$$P(t) = p_{ij}(t) = e^{Q \times t}, \tag{11}$$

where it is assumed that all n components function at moment $t=0$. $P_n(0)$ is a row vector; all arrays of which are zero except the first array which is one. Equation (12) demonstrates the probability of occurrence of the n th failure at moment t , which is determined by the multiplication, $P_n(0) \times P(t)$.

$$P_n(t) = P_n(0) \times e^{Q \times t}. \tag{12}$$

Availability of each k -out-of- n subsystem can be computed through Eq. (9) and the total availability is obtained by multiplication of $A_i(t)$ of each subsystem.

3 NSGA II

Solving multi-objective redundancy allocation problem (MORAP) using exact techniques is very difficult and it has been proved that the MORAP is a NP-hard problem (Alavi et al. 2017a, b). Many researchers have used metaheuristic algorithms that have shown a great efficiency in solving the MORAPs (Eshraghniaye Jahromi and Feizabadi 2017). Genetic Algorithm (GA) is an intelligent population-based evolutionary metaheuristic algorithm. Deb et al. (2002) introduced a modified version of Non-Dominated Sorting Genetic Algorithm (NSGA) called NSGA-II.

In the NSGA II algorithm, with addition of two necessary operators, the single-objective GA is converted into a multi-objective algorithm, which offers a set of best solutions known as the Pareto front, rather than only the best solution. NSGA II is highly efficient in finding the optimal Pareto front and many researchers have applied NSGA II to optimize their problems. The two necessary operators are:

1. The operator which assigns an excellence criterion to the population members based on non-dominated sorting. The concept of domination sort is used where certain conditions are followed, if solution from objective 1 dominates the solutions from objective 2.
2. The crowding distance which maintains solution diversity among solutions with equal ranks.

If two solutions are of the same rank, the solution with larger crowding distance is selected. Large average crowding distance will result in better diversity in the population (Alikar et al. 2017)

In the following, steps of the algorithm are presented (Farrokhi-Asl et al. 2017):

Step 1 Initialization

An initial population is generated randomly

Step 2 Evaluation

Values of fitness function for each individual are calculated. Individuals are compared based on times of non-dominated and a rank is assigned to each chromosome. Rank one is the best level, rank two is the next best level, and so on (Finding non-dominated solutions as the first front).

Step 3 Density estimation

The average distance of two points on each side of this point is estimated and a crowding distance is specified.

Step 4 Selection

Parents are selected for participating in reproduction

Step 5 Crossover operator

Crossover operator is applied to create offspring from parents for a predetermined percentage of individuals selected

Step 6 Mutation operator

Mutation operator is applied to create individuals from a predetermined percentage of individuals selected

Step 7 Replacement

Old set of solutions and newly created solutions are merged to create a new population. The new population is sorted using the non-domination criterion with respect to elitism and crowding distance.

Step 8 Individuals with the non-domination level 1 are specified as Pareto solutions.

Step 9 These steps are repeated until a stopping condition is met

In this paper, to solve the proposed multi-objective problem, NSGA-II algorithm is applied to obtain the optimal solution.

3.1 Individual representation

An effective definition of the chromosome can help find a better result quickly. These chromosomes are converted to

	Subsystem
Number of component (n)	[4 3 2 5]
Number of K (k)	[2 2 1 3]
Repair man (r)	[2 2 5 3]

Fig. 4 Chromosome definition

meaningful chromosomes to display solutions of the model (Farrokhi-Asl et al. 2017).

In the proposed GA, the solution encoding chromosome is presented as a $3 \times n$ matrix, where n is the number of subsystems and by considering the bounds of variables, random permutation of numbers as a chromosome is created. The first row represents number of components in each subsystem; the elements of the second row illustrate number of k component in a k -out-of- n system, and final row represents number of repairmen. Figure 4 presents a chromosome structure considered for this problem with $n=4$.

Defining the initial population is a primary and important part of any metaheuristic algorithm. In this paper, population is generated legally and randomly. Initial population size is considered 100.

3.2 Constraint handling

To handle the constraints of the MORAP, a strict method called ‘remove infeasible individual’ is considered. To manage the limitations, infeasible individual solutions are eliminated after production. In this method, solutions are produced regardless of limitations and are investigated afterward in terms of feasibility so that the infeasible solutions are eliminated from the population. This method is, in fact, a strict penalty allocation method with very high penalties.

3.3 Selection

To select parents, the tournament selection is applied to find the top solutions for the next generation in which k chromosomes are selected randomly. These chromosomes are compared based on two criteria. The first criterion is the rank of the selected solutions; the one with the least front rank is chosen. Second, if rank of the solutions is equal, the crowding distance is compared and the one with the higher crowding distance will be selected (Alikar et al. 2017; Eshraghniaye Jahromi and Feizabadi 2017).

3.4 Crossover operators

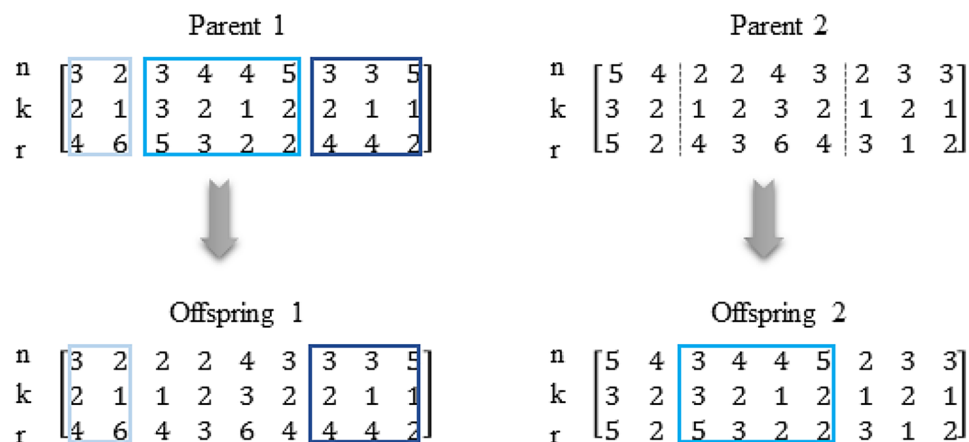
In the crossover operator, a new solution is produced by combining the information of two or more parents. Combining chromosomes to produce new chromosomes (offspring) prevents premature convergence and helps to conduct an exhaustive investigation of the solution space. The popular crossover methods are: (1) one-point crossover, (2) two-point crossover, (3) uniform crossover. In this paper, the two-point method is employed. After selecting parents, two random integers between 1 and the chromosome length (number of variables) are selected. Parents are divided into three distinct parts by these two integers (Tavana et al. 2016). Offspring is produced by swapping the mid-part of parent’s chromosomes and other parts of parents are without change. The resulting chromosomes are the offspring. Figure 5 depicts the crossover accomplished in the algorithm. After crossover operator, replacement strategy is adopted so that parents can be replaced by their corresponding offspring. The cross rate of 0.8 is assumed here.

3.5 Mutation operators

The mutation operator helps to move toward a new point in the solution space. Mutation provides access to the solution space areas, a possibility which is not offered by the crossover operator. The main purpose of applying the mutation operator is to enhance diversity and avoid being trapped in local optimization (Zoulfaghari et al. 2014b; Amiri et al. 2013).

In this paper, adaptive feasible mutation method is used. This method does a great search for the solution space widely and produces the new generation by generating directions. This method shows improvement in performance throughout the evolution steps and mutates individuals in a constrained optimization. The following steps clearly describe the proposed method (Kumar 2010; Tavana et al. 2016).

Fig. 5 Example of two-point crossover



1. Select chromosome randomly for applying mutation.
2. Generate mutation direction vector and initial step size randomly.
3. Generate mutated individual by considering direction vector and step size.
 A direction vector is generated. According to the vector, the step size is added to the amount of the selected gene for mutation or the step size is subtracted from the amount of the selected gene.
4. Check mutated individual within constraints and bounds.
5. Continue, if individual is feasible, produce mutated individual; else decrease the step size.

If the generated mutated individual is located on an infeasible space, algorithm automatically reduces the amount of step and generates another mutated individual using direction vector. The process repeats to obtain a feasible individual.

3.6 Stopping condition

The proposed algorithm is stopped after a specified iteration that provides a stable Pareto front for problem. The number of iterations is considered 1000 in the current problem.

4 Discussion

In this section, the model is validated through a numerical example. Table 3 demonstrates that eight subsystems (For example, electronic components) interconnected in series are assumed; each of which has a *k*-out-of-*n* configuration. Both *k* and *n* are regarded as decision variables. System designers can build a separate *k*-out-of-*n* configuration for each

subsystem. Furthermore, considering the presence of load share between the components of each subsystem, different configurations can be built for each subsystem. Each subsystem configuration can have different availability; therefore, selection of an appropriate and optimal configuration leads to establishment of a system with high availability. In addition, since the model presented in the present paper is time-dependent, an appropriate mission time can be determined for the system. Table 3 presents the initial failure rate, repair rate, interference factor, costs of each component, repairman costs, weight of each component, and volume of each component.

The maximum permissible weight for the whole system is approximately 130,000 ($W = 130,000$ kg) and the maximum permissible volume for the whole system is about 110,000 ($V = 110,000$ m³). Table 4 demonstrates the maximum and minimum numbers of components and repairmen for each subsystem. The minimum number of components in each subsystem is assumed to be two (because, the *k*-out-of-*n* configuration of the subsystems necessities the presence of at least two components in each subsystem). Furthermore, since all the subsystems are repairable, each subsystem requires at least one repairman. The numerical example presented in this section is solved using NSGA II. The initial population size is considered 50; number of iterations is 200; and number of function evaluations is 11,250.

The model has been solved in three different times: $t = 0.1$ h, $t = 0.5$ h, and $t = 1$ h. Figure 6 illustrates the first Pareto front for the aforementioned model with two objective functions of costs and availability at specified times. In this illustration, the red arrow represents the worst solution and the green arrow represents the best solution. Figure 6 clearly demonstrates the diminution of availability over time. For instance, consider a cost of 9000 monetary units. Given

Table 3 Parameters for each subsystem

Parameters	Sub 1	Sub 2	Sub 3	Sub 4	Sub 5	Sub 6	Sub 7	Sub 8
λ_{i0}	0.4	0.6	0.5	0.2	0.1	0.7	0.1	0.3
μ_i	0.6	0.8	0.62	0.4	0.22	0.9	0.4	0.7
α_i	0.1	0.2	0.5	0.12	0.14	0.2	0.3	0.2
c_i	265	163	338	152	248	340	390	110
h_i	165	59	137	163	370	289	144	63
w_i	67	17	52	44	54	54	74	100
v_i	62	36	34	150	800	20	40	89

Table 4 Maximum and minimum number of components and repairmen in each subsystem

Parameters	Sub 1	Sub 2	Sub 3	Sub 4	Sub 5	Sub 6	Sub 7	Sub 8
l_{\max}	10	15	17	7	9	9	6	8
l_{\min}	2	2	2	2	2	2	2	2
η_{\max}	9	11	8	7	3	3	7	2
η_{\min}	1	1	1	1	1	1	1	1

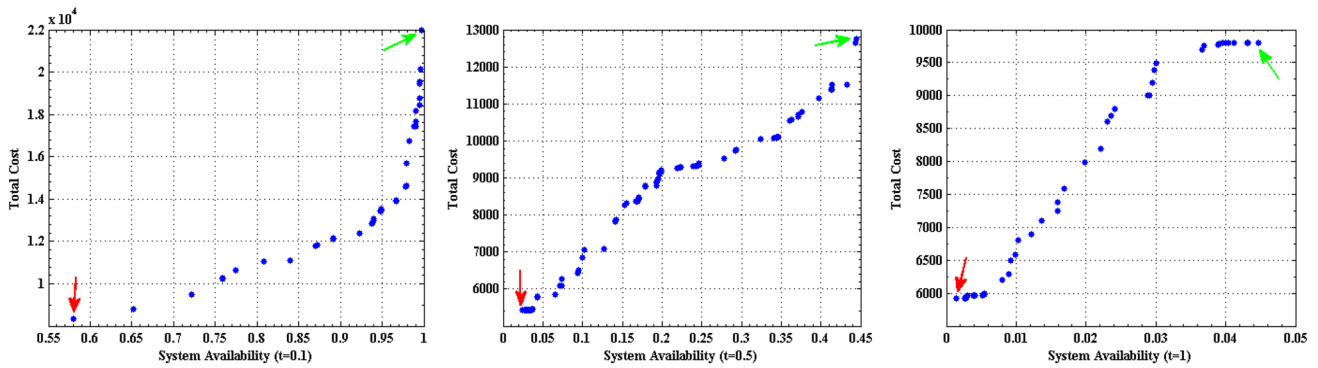


Fig. 6 RAP Pareto front

Table 5 Minimum and maximum points of Pareto front based on availability

Pareto front	Worst solution	Best solution
$t=0.1$	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 8 & 3 & 1 & 2 & 1 & 2 & 2 & 1 \end{bmatrix}$ <p>Availability = 0.58026, Cost = 8353.85</p>	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 4 & 5 & 5 & 5 & 4 & 4 & 5 & 5 \\ 3 & 4 & 4 & 4 & 2 & 3 & 3 & 3 \\ 7 & 6 & 1 & 5 & 2 & 2 & 5 & 1 \end{bmatrix}$ <p>Availability = 0.99669, Cost = 21,983.8</p>
$t=0.5$	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 3 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$ <p>Availability = 0.2439, Cost = 5442.91</p>	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 3 & 3 & 5 & 3 & 3 & 5 & 5 & 5 \\ 2 & 2 & 4 & 2 & 2 & 4 & 3 & 4 \\ 3 & 3 & 4 & 2 & 1 & 2 & 3 & 1 \end{bmatrix}$ <p>Availability = 0.44519, Cost = 12,757.5</p>
$t=1$	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 4 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$ <p>Availability = 0.01355, Cost = 5924.36</p>	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 3 & 3 & 3 & 3 & 2 & 3 & 3 & 3 \\ 2 & 2 & 1 & 2 & 1 & 2 & 2 & 2 \\ 7 & 1 & 1 & 1 & 2 & 1 & 2 & 1 \end{bmatrix}$ <p>Availability = 0.04472, Cost = 9808</p>

this amount, the whole system availability is approximately 72% at $t=0.1$, 19.6% at $t=0.5$, and 3% at $t=1$. Figure 6 also shows that increased availability leads to increase in the whole system costs.

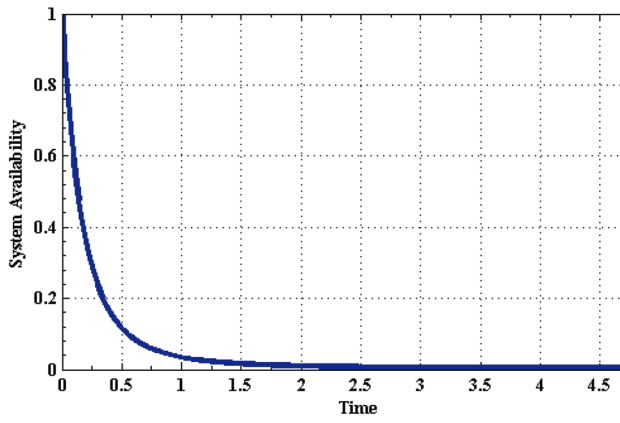
Table 5 demonstrates six different configurations in the Pareto front diagram and presents the data relating to the best and worst degree of availability for each Pareto front (green arrow and red arrow). These tables specify the values of n , y , and k for each subsystem at $t=0.1$, $t=0.5$, and $t=1$. More components are required to reach maximum availability. For example, 37 components are required in the best configuration at $t=0.1$, 32 are required at $t=0.5$, and only 23 at $t=1$ to reach the best availability. Availability decreases over the time; hence, this decrease makes it possible to reach the best availability with a smaller number of components.

As the number of components decreases, fewer repairmen are required. This is presented in Table 5 for the three states (i.e. $t=1$, $t=2$, and $t=3$).

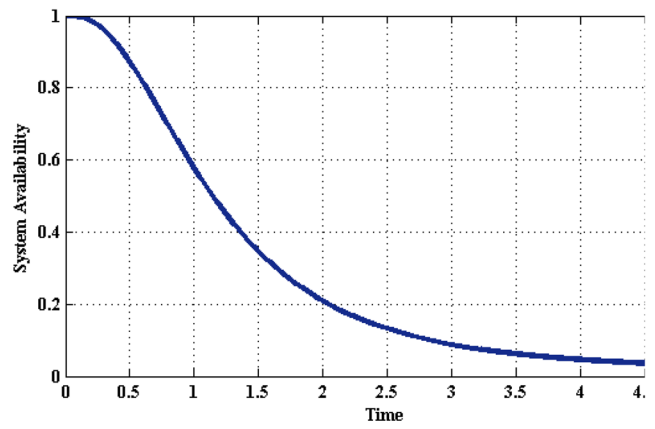
Figure 7 illustrates the diagrams for all the six configurations in Table 5. As illustrated in all the diagrams, system availability decreases over time. In other words, all systems, except system (b), fail over time (systems hardly

resist failure). For system b, if $t=0.1$, then availability equals 0.9974. The availability of system (b) reduces over time, but the system resists failure up to $t=3.5$. System (b) is in a stable state at $t=3.5$ and maintains an availability of approximately 0.062 (still resisting failure). This system can be used as an optimization system with very short operation time.

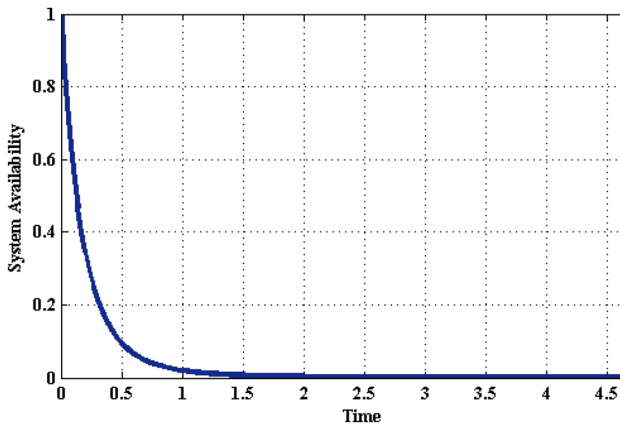
In the light of the forgoing explanations, it is illogical to select systems (a), (c), (d), (e), or (f) as optimizations systems, because these systems undergo failure prior to reaching the stable state. Availability of the systems (c) and (d) at $t=0.5$ is 0.0947 and 0.7695, respectively. Availability of the system (d) might appear logical, but this system also undergoes failure prior to reaching the stable state. This is the case for systems (e) and (f) as well. As regards systems (d) and (f), if short-time operation is desirable, system (d) can be more effective than system (f). Besides, the costs of system (d) are lower than those of system (f). However, as mentioned previously, for short-time operation, system (b) is the most efficient of all. In real-world settings, designers sometimes need to build new systems with very short operation time and short-time operation. Therefore, they must



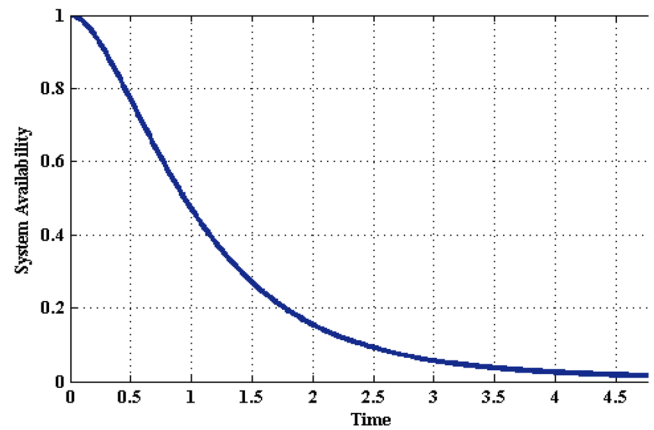
(a) Worst Solution of Pareto Front $t=0.1$ based on availability



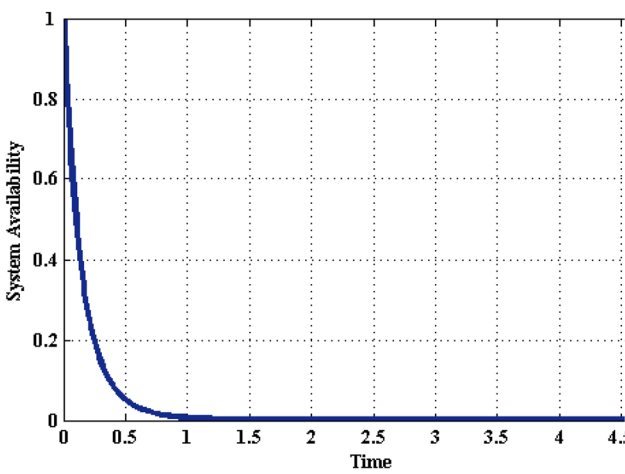
(b) Best Solution of Pareto Front $t=0.1$ based on availability



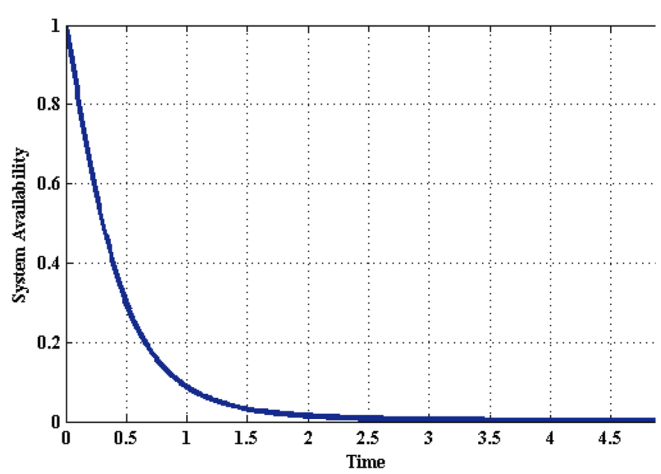
(c) Worst Solution of Pareto Front $t=0.5$ based on availability



(d) Best Solution of Pareto Front $t=0.5$ based on availability



(e) Worst Solution of Pareto Front $t=1$ based on availability

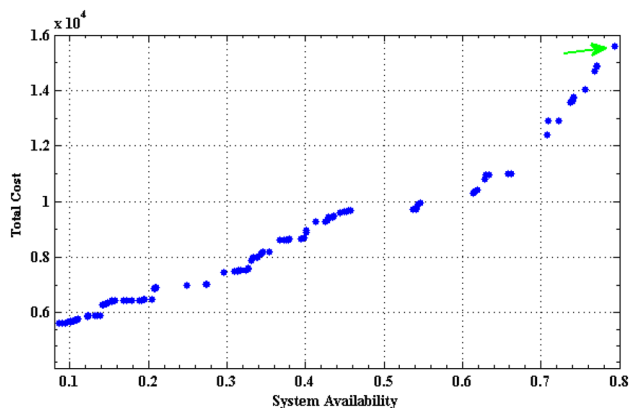


(f) Best Solution of Pareto Front $t=1$ based on availability

Fig. 7 System availability for systems presented in Table 5

Table 6 Maximum point of Pareto front of Fig. 8 based on availability

$t=0.1$	$\begin{bmatrix} n \\ k \\ y \end{bmatrix} = \begin{bmatrix} 4 & 6 & 4 & 5 & 6 & 5 & 5 & 6 \\ 3 & 4 & 3 & 3 & 3 & 4 & 2 & 5 \\ 6 & 8 & 6 & 5 & 2 & 2 & 5 & 1 \end{bmatrix}$
Availability = 0.7936, Cost = 15,584.91	

**Fig. 8** RAP Pareto front for $\alpha_i = 0$

account for the effect of the passage of time on RAP. The best mission time which can be considered for the system is $t=0.1$. Considering the existence of dependent failure (e.g. load share) in most redundant systems, it is recommended that these systems are not given long operation time, since their availability diminishes substantially over time.

As mentioned before, increased number of repairmen does not necessarily lead to increase in availability, and, in case of work interference among the repairman, an extreme increase in the number of repairmen does not result in increase in repair rate. Results presented in Table 5 demonstrate that the first subsystem has the lowest interference factor; therefore, it receives the greatest number of repairmen in all states ($t=0.1$, $t=0.5$, $t=1$). In contrast, the third subsystem has the highest interference factor, and hence it receives the lowest number of repairmen. To better understand the effect of the interference factor in the model presented in this paper, a special sensitivity analysis is carried out on this parameter. For instance, assuming that the interference factor is zero for all the subsystems (no work interference among the repairmen), the model is solved again at $t=0.1$. Table 6 presents the best solution for the model (Other parameters are assumed to be according to Table 3 and the model solution conditions are assumed to be identical). Due to the lack of work interference among the repairmen, the number of repairmen in relation to the best solution in Table 5 ($t=0.1$) has increased (from 29 to 35). Moreover, the configuration of each subsystem has changed and the best availability at

$t=0.1$ has decreased (from 99 to 79%) in comparison to Table 5. Hence, it is very important to incorporate the work interference among repairmen into RAP, since it exerts various effects on model solution results. Figure 8 illustrates the Pareto front for the model without considering the work interference.

5 Conclusion

In the present study, a RAP is presented for the k -out-of- n system with reparability, time, load share, and interference factor. In this regard, the Markov chain has been used to account for these features and efforts have been made to integrate real conditions into the optimization model. The first objective function in the optimization model is defined to create the system availability function by the use of the transition rate. Using Markov chain equation, this function is obtained in terms of time; considering the mission time in RAP modeling is very important, as it affects the number of redundant systems. Moreover, there are certain systems designed for a very short time interval (such as one-shot systems). In these systems, the time interval of the mission is very sensitive and has a significant impact on the performance of the system. Additionally, reparability is an important feature that should be considered in creating an availability function. The repair rate is directly related to the number of repairmen. Typically, with the increase in the number of repairmen, the repair rate also increases, but in real conditions, this is not always the case and it is possible that as number of the repairmen increases, the interactions between them also increase and this affects the repair rate. To address such a situation, the interference factor has been defined and is integrated in the optimization model. Another feature that has been investigated in this paper for RAP is dependent failures. One of the most important dependent failures is the load share. This type of dependency makes availability extremely vulnerable in redundant systems. To take the load share into account, we used the capacity flow model.

To reduce the system costs, the second function has been created. System costs include component costs and the repairmen fees. Besides, as the problem is NP-hard, the genetic algorithm (NSGA II) has been selected for solving the optimization model and finally the results were analyzed in accordance with different times ($t=0.1, 0.5, 1$).

Future studies can focus on developing methods for solving and choosing the system configurations (strategy selection). Other evolutionary methods can also be used to solve the model in terms of uncertainty and to analyze the solutions. On the other hand, there are different configurations for redundancy systems (such as cold standby, warm standby, etc.) through which the configuration can be

accessed with the highest availability and the lowest cost by adding a decision variable to the optimization model. It is also suggested that other possible distributions (except exponential distribution) be considered for calculations.

References

- Alavi SR, Rahmati M, Ziaei-Rad S (2017a) A new approach to design safe-supported HDD against random excitation by using optimization of rubbers spatial parameters. *Microsyst Technol* 23:2023–2032
- Alavi SR, Rahmati M, Ziaei-Rad S (2017b) Optimization of passive control performance for different hard disk drives subjected to shock excitation. *J Central South Univ* 24:891–899
- Alikar N, Mousavi SM, Raja Ghazilla RA, Tavana M, Olugu EU (2017) Application of the NSGA-II algorithm to a multi-period inventory-redundancy allocation problem in a series-parallel system. *Reliab Eng Syst Safety* 160:1–10. <https://doi.org/10.1016/j.res.2016.10.023>
- Amiri M, Ghassemi-Tari F (2007) A methodology for analyzing the transient availability and survivability of a system with repairable components. *Appl Math Comput* 184:300–307
- Amiri M, Abtahi A-R, Khalili-Damghani K (2013) Solving a generalised precedence multi-objective multi-mode time-cost-quality trade-off project scheduling problem using a modified NSGA-II algorithm. *Int J Serv Oper Manag* 14:355–372
- Arabi AAY, Jahromi AE (2012) Developing a new model for availability of a series repairable system with multiple cold-standby subsystems and optimization using simulated annealing considering redundancy and repair facility allocation. *Int J Syst Assur Eng Manag* 3:310–322
- Arabi AAY, Jahromi AE (2013) Availability optimization of a series system with multiple repairable load sharing subsystems considering redundancy and repair facility allocation. *Int J Syst Assur Eng Manag* 4:262–274
- Ardakan MA, Rezvan MT (2018) Multi-objective optimization of reliability–redundancy allocation problem with cold-standby strategy using NSGA-II. *Reliab Eng Syst Saf* 172:225–238
- Attar A, Raissi S, Khalili-Damghani K (2017) A simulation-based optimization approach for free distributed repairable multi-state availability-redundancy allocation problems. *Reliab Eng Syst Saf* 157:177–191
- Bhunia AK, Sahoo L, Roy D (2010) Reliability stochastic optimization for a series system with interval component reliability via genetic algorithm. *Appl Math Comput* 216:929–939
- Carpitella S, Certa A, Izquierdo J, La Fata CM (2017) k -out-of- n systems: an exact formula for the stationary availability and multi-objective configuration design based on mathematical programming and TOPSIS. *J Comput Appl Math* 330:1007–1015
- Çekyay B, Özekici S (2015) Reliability, MTTF and steady-state availability analysis of systems with exponential lifetimes. *Appl Math Model* 39:284–296
- Chambari A, Rahmati SHA, Najafi AA (2012) A bi-objective model to optimize reliability and cost of system with a choice of redundancy strategies. *Comput Ind Eng* 63:109–119
- Chen T-C, You P-S (2005) Immune algorithms-based approach for redundant reliability problems with multiple component choices. *Comput Ind* 56:195–205
- Chern M-S (1992) On the computational complexity of reliability redundancy allocation in a series system. *Oper Res Lett* 11:309–315
- Cui L, Li H (2007) Analytical method for reliability and MTTF assessment of coherent systems with dependent components. *Reliab Eng Syst Saf* 92:300–307
- da Costa Bueno V (2005) Minimal standby redundancy allocation in a k -out-of- n : F system of dependent components. *Eur J Oper Res* 165:786–793
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6:182–197. <https://doi.org/10.1109/4235.996017>
- Dolatshahi-Zand A, Khalili-Damghani K (2015) Design of SCADA water resource management control center by a bi-objective redundancy allocation problem and particle swarm optimization. *Reliab Eng Syst Saf* 133:11–21
- Ebrahimipour V, Shekhalishahi M Application of multi-objective particle swarm optimization to solve a fuzzy multi-objective reliability redundancy allocation problem. In: *Systems Conference (SysCon), 2011 IEEE International, 2011*, pp 326–333
- Eshraghniae Jahromi A, Feizabadi M (2017) Optimization of multi-objective redundancy allocation problem with non-homogeneous components. *Comput Ind Eng* 108:111–123. <https://doi.org/10.1016/j.cie.2017.04.009>
- Farrokhi-Asl H, Tavakkoli-Moghaddam R, Asgarian B, Sangari E (2017) Metaheuristics for a bi-objective location-routing-problem in waste collection management. *J Indus Prod Eng* 34:239–252. <https://doi.org/10.1080/21681015.2016.1253619>
- Gen M, Yun Y (2006) Soft computing approach for reliability optimization: state-of-the-art survey. *Reliab Eng Syst Saf* 91:1008–1026
- Guo J, Wang Z, Zheng M, Wang Y (2014) Uncertain multiobjective redundancy allocation problem of repairable systems based on artificial bee colony algorithm. *Chin J Aeronaut* 27:1477–1487
- Khalili-Damghani K, Amiri M (2012) Solving binary-state multi-objective reliability redundancy allocation series-parallel problem using efficient epsilon-constraint, multi-start partial bound enumeration algorithm, and DEA. *Reliab Eng Syst Saf* 103:35–44
- Khalili-Damghani K, Abtahi A-R, Tavana M (2013) A new multi-objective particle swarm optimization method for solving reliability redundancy allocation problems. *Reliab Eng Syst Saf* 111:58–75
- Khalili-Damghani K, Abtahi AR, Tavana M (2014) A decision support system for solving multi-objective redundancy allocation problems. *Qual Reliab Eng Int* 30:1249–1262
- Kim H, Kim P (2017) Reliability–redundancy allocation problem considering optimal redundancy strategy using parallel genetic algorithm. *Reliab Eng Syst Saf* 159:153–160
- Kumar R (2010) System and method for the use of an adaptive mutation operator in genetic algorithms. Google Patents
- Kuo C-C, Sheu S-H, Ke J-C, Zhang ZG (2014) Reliability-based measures for a retrial system with mixed standby components. *Appl Math Model* 38:4640–4651
- Lambert BK, Walvekar AG, Hirmas JP (1971) Optimal redundancy and availability allocation in multistage systems. *IEEE Trans Reliab* 20(3):182–185
- Li C-y, Chen X, Yi X-s, Tao J-y (2010) Heterogeneous redundancy optimization for multi-state series–parallel systems subject to common cause failures. *Reliab Eng Syst Saf* 95:202–207
- Li J, Chen G, Li J, Wang R (2016) Availability evaluation and design optimization of multi-state weighted k -out-of- n systems. In: *Prognostics and system health management conference (PHM-Chengdu), 2016, IEEE*, pp 1–6
- Lins ID, Drogue EL (2011) Redundancy allocation problems considering systems with imperfect repairs using multi-objective genetic algorithms and discrete event simulation. *Simul Model Pract Theory* 19:362–381
- Liu G-S (2015) Combination methods to solve the availability–redundancy optimisation problem for repairable parallel–series systems. *Int J Syst Sci* 46:2240–2257

- Mortazavi SM, Karbasian M, Goli S (2016) Evaluating MTTF of 2-out-of-3 redundant systems with common cause failure and load share based on alpha factor and capacity flow models. *Int J Syst Assur Eng Manag*. <https://doi.org/10.1007/s13198-016-0553-9>
- Mortazavi SM, Mohamadi M, Jouzdani J (2017) MTBF evaluation for 2-out-of-3 redundant repairable systems with common cause and cascade failures considering fuzzy rates for failures and repair: a case study of a centrifugal water pumping system. *J Indus Eng Int* 14(2):281–291
- Nourelfath M, Châtelet E, Nahas N (2012) Joint redundancy and imperfect preventive maintenance optimization for series–parallel multi-state degraded systems. *Reliab Eng Syst Saf* 103:51–60
- Pozsgai P, Neher W, Bertsche B (2003) Models to consider load-sharing in reliability calculation and simulation of systems consisting of mechanical components. In: *Reliability and maintainability symposium, 2003. Annual, 2003 IEEE*, pp 493–499
- Ramirez-Marquez JE, Coit DW (2007) Optimization of system reliability in the presence of common cause failures. *Reliab Eng Syst Saf* 92:1421–1434
- Rausand M, Høyland A (2004) *System reliability theory: models, statistical methods, and applications*, vol 396. Wiley, New York
- Robinson DG, Neuts MF (1989) An algorithmic approach to increased reliability through standby redundancy. *IEEE Trans Reliab* 38:430–435
- Shao J, Lamberson LR (1991) Modeling a shared-load k -out-of- n : G system. *IEEE Trans Reliab* 40(2):205–209
- Soylu B, Ulusoy SK (2011) A preference ordered classification for a multi-objective max–min redundancy allocation problem. *Comput Oper Res* 38:1855–1866
- Taboada HA, Baheranwala F, Coit DW, Wattanapongsakorn N (2007) Practical solutions for multi-objective optimization: an application to system reliability design problems. *Reliab Eng Syst Saf* 92:314–322
- Tavana M, Khalili-Damghani K, Di Caprio D, Oveisi Z (2016) An evolutionary computation approach to solving repairable multi-state multi-objective redundancy allocation problems. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-016-2676-y>
- Xie W, Liao H, Jin T (2014) Maximizing system availability through joint decision on component redundancy and spares inventory. *Eur J Oper Res* 237:164–176
- Xu H, Hu W (2013) Modelling and analysis of repairable systems with preventive maintenance. *Appl Math Comput* 224:46–53
- Yeh W-C (2018) A novel boundary swarm optimization method for reliability redundancy allocation problems. *Reliab Eng Syst Saf*. <https://doi.org/10.1016/j.res.2018.02.002>
- Yu H, Yang J, Mo H (2014) Reliability analysis of repairable multi-state system with common bus performance sharing. *Reliab Eng Syst Saf* 132:90–96
- Zio E, Bazzo R (2011) A clustering procedure for reducing the number of representative solutions in the Pareto Front of multiobjective optimization problems. *Eur J Oper Res* 210:624–634
- Zoufaghari H, Hamadani AZ, Ardakan MA (2014a) Bi-objective redundancy allocation problem for a system with mixed repairable and non-repairable components. *ISA Trans* 53:17–24
- Zoufaghari H, Zeinal Hamadani A, Abouei Ardakan M (2014b) Bi-objective redundancy allocation problem for a system with mixed repairable and non-repairable components. *ISA Trans* 53:17–24. <https://doi.org/10.1016/j.isatra.2013.08.002>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.