Research Annals of Industrial and Systems Engineering

www.raise.reapress.com

Res. Ann. Ind. Syst. Eng. Vol. 2, No. 2 (2025) 119–141.

Paper Type: Original Article

Optimal Design of K-Out-of-N Systems with Mixed Active/Cold-Standby Redundancy Strategy Using NSGAII

Lavan Geramiparvar[1,*], Roya Soltani[1,2], Seyyed Esmaeil Najafi[1]

[1] Department of Industrial Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran; l.geramiparvar@gmail.com; r.soltani@khatam.ac.ir; najafi1515@yahoo.com.

[2] Department of Industrial Engineering, Khatam University, Tehran, Iran; r.soltani@khatam.ac.ir.

Citation:

Received: 14 August 2024 *Revised: 12 November 2024* *Accepted: 05 Januray 2025*	Geramiparvar, L., Soltani, R., & Najafi, S. E. (2025). Optimal design of k-out-of-n systems with mixed active/cold-standby redundancy strategy using NSGAII. *Research annals of industrial and systems engineering*, 2(2), 119-141.

Abstract

Redundancy allocation problem (RAP) is one of the most important issues in reliability engineering to increase the reliability of a system. The design of the redundant systems involves determining the redundancy strategies, redundancy levels, the type of components, and system configuration. The redundancy strategy can be active, standby, or active/standby. This study for the first time provides an approach to determine the optimal configuration in k-out-of-n repairable systems with mixed active/cold standby redundancy strategy to simultaneously maximize system availability and minimize system cost. To deal with this combinatorial problem, a Non-Dominated Sorting Genetic Algorithm II (NSGA-II) is employed. To find the most suitable solutions amongst Pareto solutions, a multi-criteria decision making method, TOPSIS, is employed. To validate the performance of the proposed method for systems with mixed active/standby redundancy strategies, the results are compared with the results of applying the proposed method to systems with active strategy and also systems with cold standby strategy. The results show that both cold standby and mixed redundancy strategies provide high levels of availability. Using mixed redundancy strategy is preferred over the active and cold standby counterparts, because it delivers design flexibility and provides maximum availability. Interestingly, active components used in the proposed mixed strategy do not impose extra costs.

Keywords: Redundancy allocation problem, Active/cold standby, Availability analysis, Non-dominated sorting genetic algorithm, Markov chains, TOPSIS decision making method.

1 | Introduction

The modern industrial systems consist of various components with complex relationships. The continuous operation of these systems is required to achieve the highest level of productivity and profitability. During the recent decades, a considerable attention have been paid to develop maintenance and repair strategies that

Corresponding Author: l.geramiparvar@gmail.com

https://doi.org/10.22105/raise.v2i2.49

allow for managing the inevitable failures of systems and their components [1]. In this context, two approaches have been developed to improve the reliability of the systems; one by increasing the reliability of the existing components, and the other by allocating redundant components. The optimal system design is determined by solving the Redundancy Allocation Problem (RAP). RAP is an NP-hard problem, which determines the most appropriate redundancy strategies and the optimal number of the redundant components given system-level design constraints [2]. The majority of the studies addressing RAP have studied non-repairable systems and focused on reliability as a measure of system's performance. However, in systems with repairable components, analyzing availability could better help to evaluate maintenance management and adopt more efficient strategies [3]. The systems involve subsystems with either standby or active redundancy strategy [4]. The standby strategy improves the system's reliability with minimum depreciation of the components; however, this strategy poses limitations due to complications of failure detection and switching mechanism of the redundant components. On the other hand, the use of active strategy is not always economically feasible due to the high depreciation rate of the components. To overcome these limitations and prevent interruptions in systems' performance, mixed strategies including both active and standby redundancies have been proposed [5], [6]. In this approach, active redundancy is used for critical components of the system, and the other components are used in standby state. During the last decade, the mixed redundancy strategy has increasingly been used to develop novel models for the RAP. Some of the recent studies addressing RAP in systems with mixed redundancy strategy are summarized in *Table 1*. One of the pioneering studies in this field was conducted by Ardakan and Hamadani [5]. They proposed a novel model for RAP with choice of redundancy strategies. The mixed active and cold standby strategy provided higher reliability compared to conventional active or cold strategies. Besides, it delivered more flexibility to design complex systems with improved performance. In 2015, Abouei Ardakan et al. [6] developed a mixed active and cold standby strategy for a series–parallel system with non-repairable components. The proposed non-linear model considered reliability and cost as objective functions. The model allowed for selection of the redundancy strategy in each individual subsystem. Non-Dominated Sorting Genetic Algorithm II (NSGA-II) was implemented for solving the problem. The study showed that mixed strategy outperforms active and standby strategies alone. In another study, Abouei Ardakan et al. [7] addressed RAP in series–parallel systems with mixed active and cold standby strategy. They formulated the problem as a non-linear mixed integer programming model. The problem was subject to cost, weight and volume constraints. They developed a modified version of Genetic Algorithm (GA) to solve the problem. Results revealed that the mixed strategy provide greater reliability than active and standby counterparts. In 2017, Aghaei et al. [8] studied the problem of maximizing reliability in a k-out-of-n series system with non-repairable components. They decision variables were redundancy strategy and the number of components in each subsystem. The problem was solved using a modified version of GA and an exact method based on integer programming. The results revealed that using the proposed approach leads to significant improvements in system reliability. In another study, Gholinezhad and Hamadani [9] developed a new model for reliability optimization of a Series–parallel system with mixed redundancy strategy and component mixing. The proposed model allowed for a subsystem to simultaneously have several active and standby components. The problem was subject to cost and weight constraints. GA was implemented to find the optimal solutions for the problem. It was found that the reliability of subsystems improves by using mixed redundancy strategy and component mixing. In 2019, Ouyang et al. [10] studied RAP with mixed redundancy strategy and heterogeneous components. The aim was to maximize system reliability under cost and volume constraints. They proposed an improved particle swarm optimization algorithm to solve the problem. The study demonstrated that design flexibility provides higher system reliability and has a substantial impact on the optimal system configuration. In another study, Peiravi et al. [11] investigated the reliability optimization problem of a series-parallel systems with a novel redundancy strategy called K-mixed. K-mixed strategy is known as a general form of the mixed strategy in which the starting point of active components is time zero. The main advantage of the K-mixed strategy over the previously introduced mixed strategies is its less sensitivity to switch reliability. The problem was formulated using a mathematical model with two decision variables including the redundancy strategy and the number of components in each subsystem. Results showed that the K-Mixed strategy delivers higher reliability than its mixed complements. In a study conducted by Hadipour at al. [12], redundancy allocation of series-parallel systems with repairable components was considered under mixed active and warm standby strategy. The objective functions were maximizing the minimum of mean time to failure of system and minimizing cost. The problem was formulated using a non-linear integer programming model under a number of system-level constraints. Meta-heuristic Multi Objective Water Flow Algorithm (MOWFA) was used to solve the problem. Although mixed strategy provides higher reliability and lower weight and cost, it requires longer computation time. In 2020, Peiravi et al. [13] proposed a Continuous Time Markov Chain (CTMC) model for reliability optimization problems in series–parallel systems with mixed active and cold standby redundancy strategy. Compared to conventional methods, the proposed model shortened the computation time and found better solutions with higher reliability values. In a study by Gong et al. [14], redundancy allocation in k-out-of-n series systems with mixed warm and cold standby components was investigated. The goal was to maximize system availability and minimize cost. Markov model and simulation studies were utilized to solve the problem. The results indicated that using warm standby components protects the system from sudden failure, but increases the system cost. Since mixed warm and cold standby strategy allows for reducing the risk of system failure with lower costs, it could be considered as a more economically feasible alternative. In 2021, Sadeghi et al. [15] studied reliability optimization for a series-parallel systems with a choice of redundancy strategy and the possibility of using heterogeneous components in each subsystem. The problem was formulated assuming that component time-to-failure follows Erlang distribution. Switch time-to-failure was also assumed to be exponentially distributed. The obtained differential equations were solved by using a CTMC model. The introduced equations could efficiently compute the system reliability. In a study by Chambari et al. [16], RAP in series-parallel systems was formulated using a bi-objective simulation algorithm for maximizing reliability and minimizing cost. The redundancy strategy could be selected among active, cold-standby, mixed, or k-mixed strategies. The study exploited NSGA-II to find Pareto fronts. The results revealed the satisfactory performance of the proposed approach in optimizing system reliability and cost. Recently, Reihaneh et al. [17] proposed an exact Branch-and-Price (BP) algorithm for the RAP in a series–parallel system with mixed redundancy strategy and heterogeneous components. The proposed algorithm could solve the RAP with either active, standby or mixed strategies significantly faster than other heuristics methods. The calculated reliability of the system with mixed strategy was higher than those obtained for standby and active strategies.

Table 1. Literature review of RAP in systems with mixed redundancy strategy.

Authors (Year)	System Configuration	System Reparability	Redundancy Strategy	Objective Function(s)	Solution Methods
Ardakan and Hamadani [5]	Series–parallel	Non-repairable	Mixed active and cold standby	Maximizing reliability	Genetic algorithm
Abouei Ardakan et al. [6]	Series–parallel	Non-repairable	Mixed active and cold standby	Maximizing reliability and minimizing cost	NSGAII
Abouei Ardakan et al. [7]	Series–parallel	Non-repairable	Mixed active and cold standby	Maximizing reliability	Genetic algorithm
Aghaei et al. [8]	k-out-of-n series	Non-repairable	Mixed active and cold standby	Maximizing reliability	Genetic algorithm
Gholinezhad and Hamadani [9]	Series–parallel	Non-repairable	Active, cold-standby or mixed	Maximizing availability	Genetic algorithm

Table 1. Continued.

Authors (Year)	System Configuration	System Reparability	Redundancy Strategy	Objective Function(s)	Solution Methods
Ouyang et al. [10]	Series–parallel	Non-repairable	Active, cold-standby or mixed	Maximizing reliability	Stochastic Perturbation Particle Swarm Optimization (SPPSO)
Peiravi et al. [11]	Series–parallel	Non-repairable	K-mixed	Maximizing reliability	Genetic algorithm
Hadipour et al. [12]	Series–parallel	Repairable	Mixed active and warm standby	Maximizing the minimum of mean time to failure of system and minimizing cost	Multi objective water flow algorithm
Peiravi et al. [11]	Series–parallel	Non-repairable	Mixed active and cold standby	Maximizing reliability	Genetic algorithm
Gong et al. [14]	k-out-of-n series	Repairable	Mixed warm and cold standby	Maximizing availability and minimizing cost	Markov model and simulation studies
Sadeghi et al. [15]	Series–parallel	Non-repairable	Active, cold-standby or mixed	Maximizing reliability	Solving differential equation obtained by Continuous-time Markov chain
Chambari et al. [16]	Series–parallel	Non-repairable	Active, cold-standby, mixed, or k-mixed	Maximizing reliability and minimizing cost	NSGAII
Reihaneh et al. [17]	Series–parallel	Non-repairable	Mixed active and cold standby	Maximizing reliability	Branch-and-price algorithm
This study	k-out-of-n series	Repairable	Mixed active and cold standby	Maximizing availability and minimizing cost	NSGAII

The reviews of the preceding studies show that using mixed redundancy strategies improves the system performance and provides design flexibility. From the available studies, it seems that little attention has been paid for solving RAP in k-out-of-n repairable systems with the aim of maximizing the availability. In this study, for the first time, a bi-objective model is proposed to simultaneously maximize the availability and minimize the cost of a k-out-of-n system with mixed redundancy strategy.

The outline of the paper is as follows. In Section 2, the problem is described in details along with model formulation. Section 3 describes the solution methodology used to solve the problem. Computational results are presented in Section 4. Finally, the conclusion of the study is presented in Section 5.

2 | Problem Description

This article addresses RAP in a series–parallel k-out-of-n system with active/cold redundancy strategy. The system studied in this paper is adapted from the system proposed by Coit and Liu [18]. It consists of S subsystems, each of which in a k-out-of-n configuration (*Fig. 1*). The redundant components of each subsystem are configured in either active or cold standby mode.

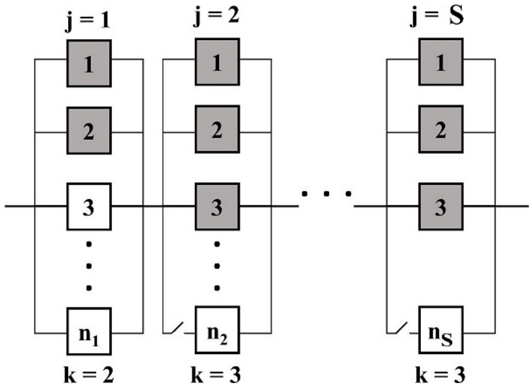

Fig. 1. System consisted of S subsystems in k-out-of-n configuration.

The proposed approach models the problem using a bi-objective formulation to maximize the availability and minimize the cost of the system. As mentioned above, unlike reliability, availability is less studied in the literature. Therefore, finding an exact formula for computing the system availability of a k-out-of-n system is a challenging issue. The challenge is doubled when mixed redundancy strategy comes into account.

A few studies have formulated availability for systems with standby or active redundant components. However, these studies have limitations. In 2002, Wang and Loman [19] proposed a formula to compute the availability in a k-out-of-n system with cold standby components. Although they mentioned that the formula has been obtained by Markov model, the proof of the formula is not given in their article. In 2017, Carpitella et al. [3] proposed a general formula to calculate availability of k-out-of-n systems with active redundant components. To prove the formula, the authors employed Markov chains of a 2-out-of-3 system and generalized the result to k-out-of-n systems. Here, we have used these formulae and adopted them for the system under study. The proof of each formula has been presented in *Appendix 1* and *Appendix 2*.

2.1 | Assumptions

The following assumptions have been considered regarding the active subsystems:

- *The subsystems are in k-out-of-n configuration, meaning that at least k components (k≤n) have to operate concurrently to assure that the subsystem is in the functioning state. Therefore, if (n - k + 1) components fail the subsystem fails and therefore cause the whole system fail.*
- *All of the components are independent, identical, and repairable.*
- *The failure rate (λ) and repair rate (μ) of the components are exponentially distributed.*
- *No constraints are considered for the availability of maintenance crews.*

The assumptions for cold subsystems are as follows:

- *The subsystems are implemented in an (N-1)-out-of-N:G configuration, meaning that each subsystem functions if at least (N-1) components function. Since an exact formula is not available for the availability of k-out-of-n systems with cold standby subsystems, this special configuration is considered.*
- *Although (N-1) components are enough to cover the load, one extra active component is added to decrease the failure rate of the subsystem.*

- *Another component is also added in cold standby. Adding the cold standby component is necessary to keep the system in functioning state all the time. The cold standby component will replace any failed component.*
- *The failed component will be used as the standby component after being repaired.*
- *The components are independent and identical.*
- *The failure rate (λ) and repair rate (μ) of the components follow an exponential distribution.*
- *No load sharing is performed, meaning that changing the load level does not change the failure rate of the components.*
- *Switch is assumed to be perfect.*
- *No failure would occur during the time needed to activate the cold standby component.*

2.2 | Model Formulation

The studying system is subject to weight constraint and the objective is to maximize system's availability, while minimizing the overall cost. Since the subsystems are interconnected in series, the system's cost is obtained from the sum of the active subsystems' cost (CA) and standby subsystems' cost (CS) (i.e. C = CA + CS). The system's availability is calculated by multiplying the subsystems' availabilities (A = AS × AA). The parameters used in the model are presented in *Table 1*.

2.2.1 | Parameters

P: reliability of switch.

τ: time needed to activate the standby component.

λ_{a_j} : failure rate of the active component in standby subsystems.

λ_{b_j} : failure rate of the cold standby component in standby subsystems.

W: maximum allowable system weight.

nmax,j: maximum number of components of each subsystem.

q_j, a_j, b_j, r_j : coefficients of the cost function.

2.2.2 | Decision variables

nj: number of components in subsystem j.

Type: redundancy strategy (in mixed active/cold configuration).

2.2.3 | Cost functions

The cost functions of the subsystems are adapted from the model proposed by Elegbede and Adjallah [20]. In their approach the cost of the subsystems includes failure and repair rates of the components. Thus, the technical performance of the system is taken into account while estimating the cost. For active subsystems the cost function is formulated according to *Eq. (1)*.

$$C_A = \sum_{j \in A} n_j \left(a_j \lambda_j^{r_j} + b_j \mu_j^{q_j} \right), \tag{1}$$

where aj, bj, qj, and rj are real numbers such that: aj, bj, qj > 0 and rj < 0. These coefficients could be obtained from the maintenance databases of organizations.

We have modified *Eq. (1)* to obtain the cost function of cold standby subsystems according to *Eq. (2)*.

$$C_S = \sum_{j \in S} \left(n_j a_j \lambda_{a_j}^{r_j} + \left(n_j + 1 \right) b_j \mu_j^{q_j} + a_j \lambda_{b_j}^{r_j} \right). \tag{2}$$

Using *Eq. (1)* and *Eq. (2)*, the cost of the system can be expressed according to *Eq. (3)*.

$$C(n)=\sum_{j\in S}\left(n_j a_j \lambda_{a_j}^{r_j}+\left(n_j+1\right) b_j \mu_j^{q_j}+a_j \lambda_{b_j}^{r_j}\right)+\sum_{j\in A} n_j\left(a_j \lambda_j^{r_j}+b_j \mu_j^{q_j}\right). \tag{3}$$

2.2.4 | Availability functions

To calculate the availability of the k-out-of-n active subsystems, the exact formula proposed by Carpitella et al. [3] is used as presented in *Eq. (4)* (see *Appendix 1* for the proof).

$$A_A=\left[\frac{\sum_{i=k}^{n}\binom{n}{i}\mu^{i}\lambda^{(n-i)}}{\sum_{i=k}^{N}\binom{n}{i}\mu^{i}\lambda^{(n-i)}+\binom{n}{k-1}\mu^{(k-1)}\lambda^{(n-k+1)}}\right]. \tag{4}$$

For cold standby subsystems, availability is obtained for (n-1)-out-of-n configuration by adopting the formula suggested by Wang and Loman [19] according to *Eq. (5)* (see *Appendix 2* for the proof).

$$A_S=\frac{\left[1-\left[n_j\left(n_j-1\right)\lambda_{a_j}^2\left(\mu_j+n_j\lambda_{a_j}+\lambda_{b_j}-P\mu_j\right)\right]\right]}{\left[9n_j\lambda_{a_j}\mu_j^2+n_j\left(4n_j-1\right)\lambda_{a_j}^2\mu_j+n_j^2\left(n_j-1\right)\lambda_{a_j}^3+3n_j\left(\lambda_{a_j}\lambda_{b_j}\mu_j-\lambda_{a_j}\mu_j^2P\right)+n_j\left(n_j-1\right)\left(\lambda_{a_j}^2\lambda_{b_j}-P\lambda_{a_j}^2\mu_j\right)+6\mu_j^2\left(\lambda_{b_j}+n_j\lambda_{a_j}\mu_j\tau+\mu_j\right)\right]}. \tag{5}$$

With respect to *Eq. (4)* and *Eq. (5)*, the availability of the studying system is given by *Eq. (6)*.

$$A(n)=\prod_{j\in S}\frac{\left[1-\left[n_j\left(n_j-1\right)\lambda_{a_j}^2\left(\mu_j+n_j\lambda_{a_j}+\lambda_{b_j}-P\mu_j\right)\right]\right]}{\left[9n_j\lambda_{a_j}\mu_j^2+n_j\left(4n_j-1\right)\lambda_{a_j}^2\mu_j+n_j^2\left(n_j-1\right)\lambda_{a_j}^3+3n_j\left(\lambda_{a_j}\lambda_{b_j}\mu_j-\lambda_{a_j}\mu_j^2P\right)+n_j\left(n_j-1\right)\left(\lambda_{a_j}^2\lambda_{b_j}-P\lambda_{a_j}^2\mu_j\right)+6\mu_j^2\left(\lambda_{b_j}+n_j\lambda_{a_j}\mu_j\tau+\mu_j\right)\right]}$$

$$\times\prod_{j\in A}\left[\frac{\sum_{i=k_j}^{n_j}\binom{n_j}{i}\mu_j^{i}\lambda_j^{(n_j-i)}}{\sum_{i=k_j}^{n_j}\binom{n_j}{i}\mu_j^{i}\lambda_j^{(n_j-i)}+\binom{n_j}{k_j-1}\mu_j^{(k_j-1)}\lambda_j^{(n_j-k_j+1)}}\right]. \tag{6}$$

2.2.5 | The proposed mathematical model

The aim of this study is to determine the maximum availability and minimum cost of a k-out-of-n system subject to a constraint on weight of the system. The model is formulated according to *Model (1)*.

$$\max\ A(n),$$

$$\min\ C(n),$$

$$\sum_{j\in A,S} n_j w_j \le W,$$

$$j\in A: n_j\in\left\{k_j, k_j+1, k_j+2, \dots, n_{max,j}\right\},$$

$$j\in S: n_j\in\left\{k_j+1, k_j+2\right\}.$$

The latter two constraints determine the number of components (nj) in active and cold-standby subsystems. As for active subsystems, the configuration is k-out-of-n, therefore kj≤nj. nmax,j is the upper bound for nj and is usually selected based on practical restrictions. On the other hand, the configuration of cold-standby subsystems, is (N-1)-out-of-N:G, which is possible only if nj=kj+1 or nj=kj+2.

3 | Solution Methodology

The methodology to solve the multi-objective optimization problem of this study includes two steps: Finding non-dominated solutions using NSGAII and selecting the best solution.

I. Finding non-dominated solutions

The first step to solve the problem is to find the non-dominated solutions. For this purpose, a NSGA-II is used. Formulated by Deb et al. [21], NSGA-II is a metaheuristic algorithm that follows a parallel search strategy to find the optimal solutions. It uses an elitist approach and specific adaptation assignment rules to determine the rank and the distance of each solution from the others. NSGA-II could provide solutions for any number of objectives with any number of constraints. The best solutions found in each iteration are saved into a Pareto set. The Pareto set of optimal solutions found by NSGA-II is preferable instead of a single solution because it offers more choices to decision-makers. Moreover, NSGA-II preserves the diversity of the solutions using a sharing method to explore different areas of the Pareto front. This prevents stopping at local optimal points and limiting to certain areas of the solution space [22].

The consecutive steps of NSGA-II are as follows:

Step 1. Chromosome definition.

Step 2. Setting the fitness function.

Step 3. Setting the crossover and mutation mechanisms.

Step 4. Setting the algorithm termination condition.

Step 5. Generating the initial population.

Step 6. Creation of the offspring population based on the crossover and mutation mechanisms.

Step 7. Selection of the chromosomes with the largest fitness function as the parents of the next generation.

Step 8. Obtaining the non-dominated Pareto fronts by combining the newly generated population with the previous one and sorting the solutions.

Step 9. Repeating *Steps 6-8* until the termination condition is met [22].

NSGA-II has been extensively used for solving multi-objective problems, including the redundancy-allocation problem [23].

II. Selection of the best solution

Pareto front provides a complete representation of the optimal solution space. Based on the decision criteria, decision makers can choose the most suitable solution among the non-dominated set of solutions. In this study, the best solution is selected by TOPSIS1, which is a multi-criteria decision making method. TOPSIS presents an ordered ranking of the solutions by calculating the distance of each solution from the positive ideal and the negative ideal solutions.

The methodology to find the best solution for RAP in k-out-of-n systems has been described clearly by Carpitella et al. [3]. The steps for selecting the best solution from Pareto front using TOPSIS are as follows:

I. Evaluation of solutions: each solution is given a score. The score of the solution i for the objective function j is shown by g_{ij}. Here, j could be availability or cost.

II. Determining the evaluation criteria and weighting them.

III. Calculating the normalized and the weighted decision matrix using the following equation:

$$U_{ij} = W_j . Z_{ij}, \qquad \text{for all} i, \text{for all} j,$$

where wj is the weight of the criterion j, and zij is the score of the solution i under the criterion j. zij is normalized using the following equation:

[1] Technique for order preference by similarity to ideal solutions

$$Z_{ij} = \frac{g_{ij}}{\sqrt{\sum_{i=1}^{n} g_{ij}^2}}, \text{for all} i, \text{for all} j.$$

IV. Determining the positive (A*) and the negative ideal point (A-) using the following equations:

A*=$(U_1^*, \dots, U_k^* = \{($………………………

where I' and I'' are the sets of criteria that must be maximized and minimized, respectively.

V. Calculating the distance of each solution i from A* and A- using the following equations.

$$S_i^* = \sqrt{\sum_{i=1}^{k} (U_{ij} - U_j^*}.$$

$$S_i^- = \sqrt{\sum_{i=1}^{k} (U_{ij} - U_j^-}.$$

VI. Evaluation of each solution i based on its proximity to A* using the closeness coefficient C_i*

$$C_i^* = \frac{S_i^-}{S_i^- + S_i^*}, \quad 0 \leq C_i^* \leq 1 \text{for all i.}$$

VII. Ranking the Pareto solutions based on closeness coefficients. For two generic solutions i and z, if Ci* ≥ Cz*, solution i is preferred over solution z.

4 | Computational Results

The main purpose of this research is to find the optimal configuration in k-out-of-n systems with active/cold standby redundancy strategy. To investigate the performance of the proposed method, a system composed of 14 subsystems are considered. Each subsystem (j) is characterized by the minimum number of components required to keep the subsystem in the functioning state (kj), the failure rate (λj), the repair rate (μj), the allowable weight of the components (wj), and the type of the redundancy strategy in each subsystem (active or standby) which are summarized in Table 1. Each subsystem can be of active (A) or standby (S) strategy. kj is selected from {1, 2, 3} for each subsystem.

Table 1. Subsystems' data [18].

j	k_j	Strategy	λ_j	μ_j	w_j
1	2	A	0.00532	0.43244	3
2	3	S	0.00818	0.3454	8
3	3	A	0.0133	0.29342	7
4	2	S	0.00741	0.26886	5
5	1	A	0.00619	0.24052	4
6	3	S	0.00436	0.23914	5
7	3	S	0.0105	0.23768	7
8	3	S	0.015	0.23248	4
9	2	S	0.00268	0.20888	8
10	3	S	0.0141	0.19466	6
11	2	S	0.00394	0.19064	5
12	1	S	0.00236	0.1869	4
13	2	A	0.00215	0.06964	5
14	3	S	0.011	0.04167	6

The values of the parameters used in the proposed model are presented in *Table 2*. The maximum number of components of each subsystem (nmax,j) is set to six. The objective is to maximize system availability and minimize system cost given constraint on system weight (W = 170).

Table 2. Parameters used in the model [19], [20].

Parameters and Their Value
$P = 0.95$
$\tau = 10\min = 0.16\text{hr}$
λ_{a_j}
$\lambda_{b_j} = \frac{\lambda_{a_j}}{10} = \frac{\lambda_j}{10}$
$r_j = -0.8\times(0.4, 0.2, 1, 0.8, 1.2, 0.9, 1.4, 1.1, 0.5, 1.3, 0.7, 0.6, 0.1, 0.3)$
$q_j = 0.85\times(0.4, 0.2, 1, 0.8, 1.2, 0.9, 1.4, 1.1, 0.5, 1.3, 0.7, 0.6, 0.1, 0.3)$
$a_j = 0.01\times(1, 4, 3, 2, 5, 7, 8, 6, 10, 9, 13, 11, 12, 14)$
$b_j = 0.1\times(0.4, 0.2, 1, 0.8, 1.2, 0.9, 1.4, 1.1, 0.5, 1.3, 0.7, 0.6, 0.1, 0.3)$
W = 170
$n_{max,j} = 6$

To solve the proposed model, NSGA-II is coded in the MATLAB®. The program was run on an Intel Core i5-480M @ 2.67GHz CPU with 4 GB of RAM. Briefly, each solution was encoded using a 3 × 14 chromosome (*Fig. 2*). The first and the second rows of the solution represent the subsystems' number and the relevant redundancy strategies, respectively. The third row represents the number of the allocated components for each subsystem.

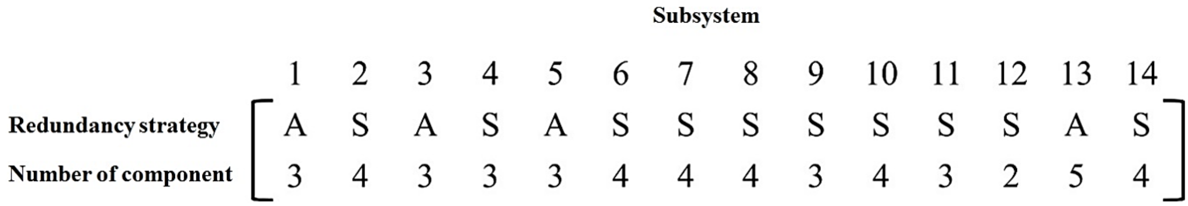

Fig. 2. Encoding solution as a chromosome representation.

The initial population was randomly generated from a population size of N = 100. Parent chromosomes were selected randomly from the initial population. To generate offsprings, the max-min crossover operator was implemented with crossover rate of 0.9. The crossover operator uses a binary 3 × 14 random matrix to exchange the respective values between parents (*Fig. 3*).

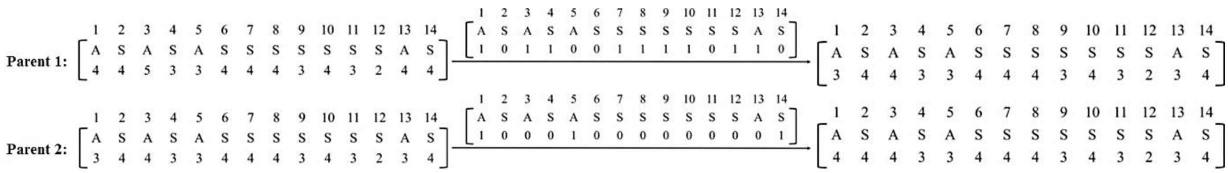

Fig. 3. Example of crossover operator.

To diversify the new population, the values within each solution matrix were changed randomly using a mutation operator. The mutation rate (pm) is calculated as $\frac{1}{d}$, where d is the number of decision variables [21]. Since d = 28, pm was set to 0.036. An example of a mutation operator is shown in *Fig. 4*.

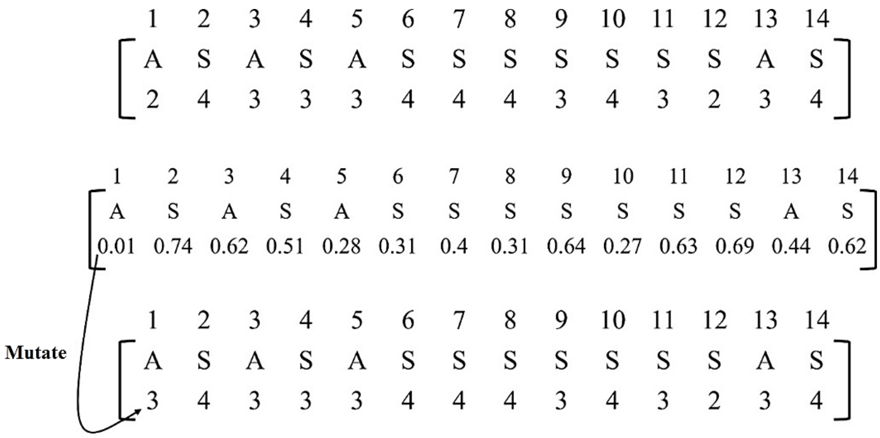

Fig. 4. Example of mutation operator.

The maximum iterations of 100 is determined as the stopping condition to terminate NSGA-II. The parameters used for the NSGA-II procedure are summarized in *Table 3*.

Table 3. Parameters used for NSGA-II.

Parameters	Values
Initial population size	100
Crossover rate	0.9
Mutation rate	0.07, for active and cold standby strategies 0.036, for mixed strategy
Number of iterations	100

To validate the performance of the proposed method for systems with mixed active/standby redundancy strategies, the results are compared with the results of applying the proposed method to systems with active strategy and also systems with, standby strategy. After obtaining the Pareto set of solutions, TOPSIS decision making method is employed to rank and prioritize the non-dominated solutions, based on their distances from the best and the worst solutions. Each solution is assigned a score which indicates the ranking index of the solution. The results are shown in *Tables 4-6* respectively for active, standby and active/standby strategies.

Table 4. The non-dominated solutions and their rankings for the system with Active redundancy strategy.

Solution no.	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}	n_{13}	n_{14}	Availability	Cost	Score	Ranking
1	5	4	3	3	3	4	4	4	3	4	3	2	4	4	0.004961113	356.3485589	0.062847	21
2	4	4	5	3	4	4	4	4	3	4	3	2	4	4	0.002774079	369.4173493	0.0034832	26
3	3	4	3	3	2	4	4	4	3	4	3	2	3	4	0.022741748	296.2183862	0.58167	14
4	5	4	4	3	4	4	4	4	3	4	3	2	5	4	0.159832517	284.3562346	0.93477	8
5	5	4	3	3	3	4	4	4	3	4	3	2	4	4	0.03216577	293.1393938	0.11382	18
6	3	4	3	3	3	4	4	4	3	4	3	2	3	4	0.014801531	313.8243713	0.0005057	29
7	5	4	4	3	4	4	4	4	3	4	3	2	5	4	0.004589343	360.1591692	0.00097424	27
8	4	4	3	3	3	4	4	4	3	4	3	2	4	4	0.00239325	410.930783	0.065631	20
9	4	4	4	3	3	4	4	4	3	4	3	2	3	4	0.048927022	290.0468643	0.40038	15
10	4	4	3	3	4	4	4	4	3	4	3	2	4	4	0.008099859	349.1110222	0.27254	16
11	4	4	3	3	3	4	4	4	3	4	3	2	4	4	0.002396776	397.2479114	0.93992	7
12	3	4	3	3	2	4	4	4	3	4	3	2	3	4	0.012869836	319.1139118	0.00066835	28
13	5	4	4	3	4	4	4	4	3	4	3	2	5	4	0.008507105	348.312472	0.79877	12
14	4	4	3	3	2	4	4	4	3	4	3	2	4	4	0.011668925	329.9013223	0.89025	10
15	5	4	4	3	3	4	4	4	3	4	3	2	4	4	0.003507032	363.7020475	1	1
16	3	4	4	3	3	4	4	4	3	4	3	2	3	4	0.165962072	276.4396376	0.98134	4
17	4	4	4	3	4	4	4	4	3	4	3	2	4	4	0.0147853	314.2326161	0.1158	17
18	4	4	4	3	4	4	4	4	3	4	3	2	4	4	0.018269144	300.2418354	0.97124	5
19	4	4	4	3	4	4	4	4	3	4	3	2	4	4	0.010235303	335.9900947	0.095463	19
20	5	4	4	3	3	4	4	4	3	4	3	2	4	4	0.016371118	304.9915864	0.59981	13
21	5	4	3	3	2	4	4	4	3	4	3	2	4	4	0.025940409	294.6103099	0.030749	25
22	4	4	5	3	4	4	4	4	3	4	3	2	4	4	0.020804506	298.436918	0.042357	23
23	5	4	4	3	3	4	4	4	3	4	3	2	4	4	0.008530924	341.7675242	0.058414	22
24	5	4	3	3	3	4	4	4	3	4	3	2	4	4	0.015579595	308.9060543	1.86E-07	31
25	4	4	4	3	3	4	4	4	3	4	3	2	3	4	0.002668704	371.792114	0.98372	3
26	5	4	3	3	3	4	4	4	3	4	3	2	4	4	0.00349395	365.5207432	0.99736	2
27	3	4	3	3	2	4	4	4	3	4	3	2	2	4	0.042400681	291.9224892	0.94581	6
28	4	4	3	3	3	4	4	4	3	4	3	2	4	4	0.16242731	281.0080597	0.86917	11
29	5	4	4	3	4	4	4	4	3	4	3	2	5	4	0.009343502	340.7591308	0.91944	9
30	4	4	5	3	2	4	4	4	3	4	3	2	5	4	0.01096824	332.7567868	3.81E-05	30
31	5	4	4	3	3	4	4	4	3	4	3	2	4	4	0.012156797	325.7044897	0.039267	24

Table 5. The non-dominated solutions and their rankings for the system with Cold standby redundancy strategy.

Solution No.	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}	n_{13}	n_{14}	Availability	Cost	Score	Ranking
1	2	4	5	3	4	4	4	4	3	4	3	2	2	4	0.9999323	275.9337	0.00025705	9
2	2	4	4	3	4	4	4	4	3	4	3	2	4	4	0.9998109	276.28299	0.00017107	10
3	4	4	4	3	4	4	4	4	3	4	3	2	4	4	0.9999278	275.86483	0.0009582	7
4	4	4	5	3	3	4	4	4	3	4	3	2	4	4	0.9999322	275.79059	0.15132	3
5	4	4	3	3	4	4	4	4	3	4	3	2	3	4	0.9998662	276.23365	0.32505	2
6	3	4	3	3	3	4	4	4	3	4	3	2	4	4	0.999847	283.13657	0.029869	4
7	5	4	4	3	3	4	4	4	3	4	3	2	4	4	0.9998381	286.05393	0.00065797	8
8	3	4	4	3	2	4	4	4	3	4	3	2	4	4	0.9998908	279.31801	0.0010879	6
9	2	4	3	3	2	4	4	4	3	4	3	2	4	4	0.9998931	276.10374	0.00014497	11
10	4	4	4	3	5	4	4	4	3	4	3	2	4	4	0.9998109	276.28299	1.34E-06	12
11	4	4	4	3	3	4	4	4	3	4	3	2	4	4	0.9998996	275.76436	0.02701	5
12	3	4	3	3	2	4	4	4	3	4	3	2	3	4	0.9999294	275.48646	1.01E-11	13
13	4	4	4	3	2	4	4	4	3	4	3	2	3	4	0.9998585	279.15101	1	1

Table 6. The non-dominated solutions and their rankings for the system with Mixed Active/Standby redundancy strategy.

Solution no.	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}	n_{11}	n_{12}	n_{13}	n_{14}	Availability	Cost	Score	Ranking
	A	S	A	S	A	S	S	S	S	S	S	S	A	S				
1	3	4	3	3	3	4	4	4	3	4	3	2	4	4	0.9992602	308.6204072	0.33768	9
2	3	4	4	3	4	4	4	4	3	4	3	2	2	4	0.9992667	290.0468643	0.9552	3
3	4	4	4	3	3	4	4	4	3	4	3	2	4	4	0.9992646	304.6815084	1.27E-12	26
4	5	4	3	3	4	4	4	4	3	4	3	2	5	4	0.9992602	334.4468611	0.0012899	17
5	4	4	5	3	4	4	4	4	3	4	3	2	5	4	0.9993462	274.0947286	0.1107	12
6	4	4	4	3	3	4	4	4	3	4	3	2	3	4	0.9992905	276.6418922	0.70063	7
7	4	4	5	3	4	4	4	4	3	4	3	2	5	4	0.9992615	293.2435169	0.95894	2
8	3	4	4	3	3	4	4	4	3	4	3	2	3	4	0.9992605	318.4947254	0.090293	13
9	4	4	4	3	4	4	4	4	3	4	3	2	5	4	0.9992601	347.5178319	4.72E-05	25
10	3	4	4	3	5	4	4	4	3	4	3	2	2	4	0.9993206	291.6847811	0.00025334	23
11	3	4	5	3	5	4	4	4	3	4	3	2	4	4	0.9993615	274.5957306	0.075958	14
12	3	4	4	3	3	4	4	4	3	4	3	2	2	4	0.9993416	275.2452196	0.00045201	22
13	4	4	3	3	2	4	4	4	3	4	3	2	4	4	0.9992667	290.4551092	0.0065206	15
14	4	4	3	3	2	4	4	4	3	4	3	2	3	4	0.9995961	275.6235848	0.0004931	21
15	4	4	5	3	4	4	4	4	3	4	3	2	4	4	0.9992919	279.5973006	1	1
16	5	4	5	3	4	4	4	4	3	4	3	2	4	4	0.9994733	275.6901197	0.00057902	20
17	4	4	5	3	3	4	4	4	3	4	3	2	4	4	0.9992602	334.9293495	0.38818	8
18	5	4	5	3	3	4	4	4	3	4	3	2	4	4	0.9993612	275.8204651	0.72757	6
19	4	4	3	3	2	4	4	4	3	4	3	2	3	4	0.9992606	306.6484891	0.75508	5
20	2	4	3	3	3	4	4	4	3	4	3	2	3	4	0.9992612	319.6452164	0.00014511	24
21	2	4	4	3	3	4	4	4	3	4	3	2	2	4	0.9992602	320.8748883	0.11787	11
22	2	4	3	3	3	4	4	4	3	4	3	2	3	4	0.9994954	274.9687215	0.77098	4
23	3	4	5	3	4	4	4	4	3	4	3	2	4	4	0.9992642	293.7494563	0.004283	16
24	2	4	3	3	4	4	4	4	3	4	3	2	3	4	0.9992601	321.6171345	0.00089525	19
25	3	4	3	3	3	4	4	4	3	4	3	2	3	4	0.9992889	278.6138103	0.0010502	18
26	3	4	4	3	3	4	4	4	3	4	3	2	3	4	0.9993612	276.2287099	0.12386	10

The solutions with highest scores for systems with active, cold standby, and mixed redundancy strategies are showed in *Fig. 5* and *Table 7*. The results indicate that using the active redundancy strategy leads to low levels of availability. In other words, constrained cost and weight do not allow to achieve high availability with active configuration. In contrast, both cold-standby and mixed redundancy strategies provide more than 0.999 availability. The total cost of the system with cold and mixed configuration is 279.15101 and 279.5973006, respectively. The data demonstrate that both strategies perform well in providing maximum availability and minimum cost. Moreover, despite the use of active components, the system cost with mixed strategy remains within the range of the cost of system with cold standby strategy.

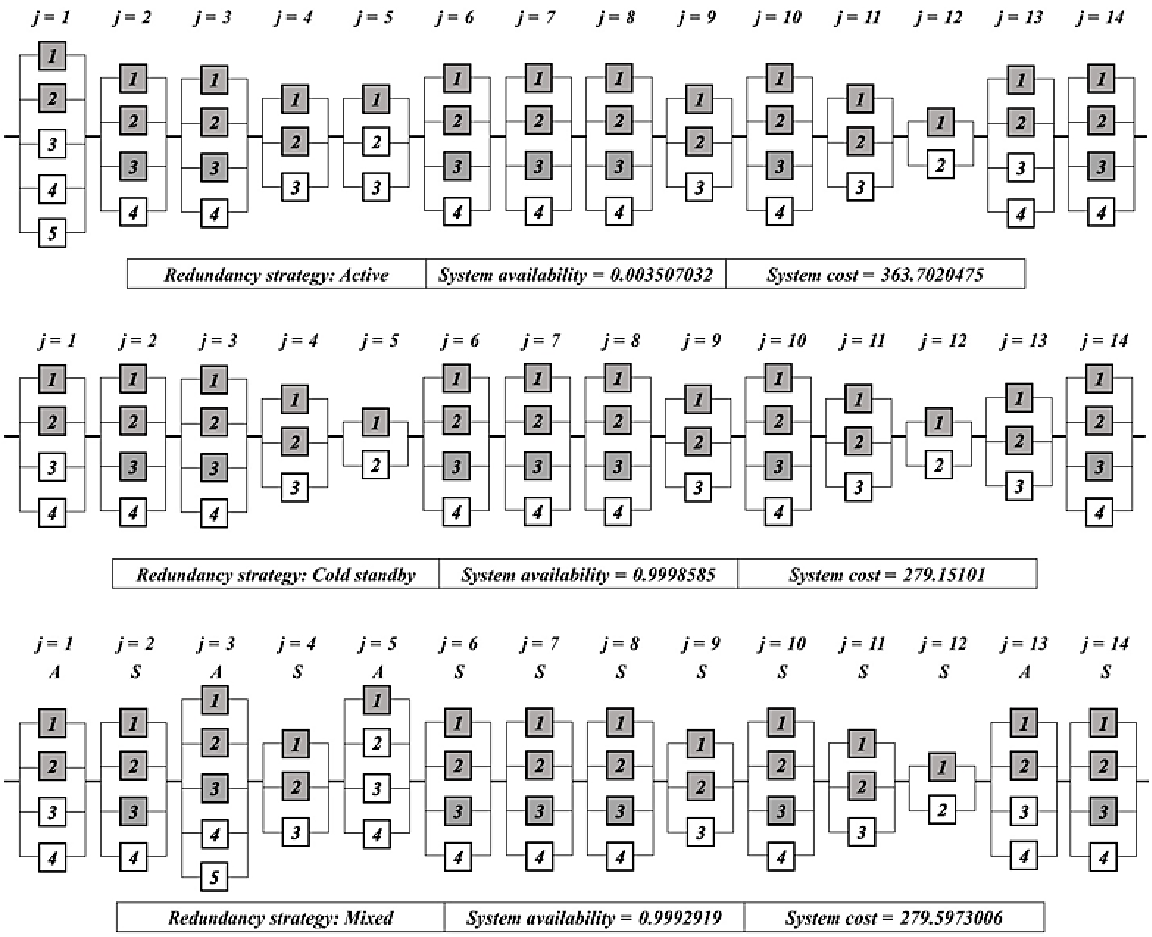

Fig. 5. Solutions with highest scores for systems with Active, Cold standby, and Mixed redundancy strategies.

Table 7. Comparison of the solutions with highest scores for systems with active, cold standby, and mixed redundancy strategies.

j	Active n_j	Cold Standby n_j	Mixed n_j	Type
1	5	4	4	A
2	4	4	4	S
3	4	4	5	A
4	3	3	3	S
5	3	2	4	A
6	4	4	4	S
7	4	4	4	S
8	4	4	4	S
9	3	3	3	S
10	4	4	4	S
11	3	3	3	S
12	2	2	2	S
13	4	3	4	A
14	4	4	4	S
Availability	0.003507032	0.9998585	0.9992919	
Cost	363.7020475	279.15101	279.5973006	

5 | Conclusion

The present study provided an approach to find the optimal configuration in k-out-of-n repairable systems with active and cold redundancy strategies. The purpose of this study was to maximize system availability while minimizing system cost. Using Markov chains, an exact formula was proposed to calculate the availability of systems with active and cold standby subsystems. An adapted formula was also proposed to calculate system cost. NSGA-II was used to deal with the proposed bi-objective model. To choose the most suitable solution, the obtained Pareto solutions, which met both objectives, were ranked by TOPSIS method. To validate the performance of the proposed method for systems with mixed active/standby redundancy strategies, the results are compared with the results of applying the proposed method to systems with active strategy and also systems with cold standby strategy. The results show that both cold standby and mixed redundancy strategies provide high levels of availability with minimum cost.

The following aspects are suggested for future studies:

I. Investigating the proposed method for multi-state systems.

II. Consideration of uncertainties in model parameters.

III. Selection of maintenance strategies considering the system's reparability.

IV. Implementation of hybrid meta-heuristic algorithms to solve the proposed problem.

Acknowledgements

The authors would like to appreciate Dr Alireza Balavand's help for encoding the model in MATLAB.

Conflicts of Interest

None.

References

[1] Garg, H., & Sharma, S. P. (2013). Multi-objective reliability-redundancy allocation problem using particle swarm optimization. *Computers and industrial engineering*, *64*(1), 247–255. https://doi.org/10.1016/j.cie.2012.09.015

[2] Coit, D. W., & Smith, A. E. (1995). *Optimization approaches to the redundancy allocation problem for series-parallel systems* [presentation]. Fourth industrial engineering research conference proceedings (pp. 342–349). https://www.eng.auburn.edu/~smithae/files/davelp.pdf

[3] Carpitella, S., Certa, A., Izquierdo, J., & La Fata, C. M. (2018). K-out-of-n systems: An exact formula for the stationary availability and multi-objective configuration design based on mathematical programming and TOPSIS. *Journal of computational and applied mathematics*, *330*, 1007–1015. https://doi.org/10.1016/j.cam.2017.01.006

[4] Coit, D. W., & Konak, A. (2006). Multiple weighted objectives heuristic for the redundancy allocation problem. *IEEE transactions on reliability*, *55*(3), 551–558. https://doi.org/10.1109/TR.2006.879654

[5] Ardakan, M. A., & Hamadani, A. Z. (2014). Reliability optimization of series-parallel systems with mixed redundancy strategy in subsystems. *Reliability engineering & system safety*, *130*, 132–139. https://doi.org/10.1016/j.ress.2014.06.001

[6] Abouei Ardakan, M., Zeinal Hamadani, A., & Alinaghian, M. (2015). Optimizing bi-objective redundancy allocation problem with a mixed redundancy strategy. *ISA transactions*, *55*, 116–128. https://doi.org/10.1016/j.isatra.2014.10.002

[7] Abouei Ardakan, M., Sima, M., Zeinal Hamadani, A., & Coit, D. W. (2016). A novel strategy for redundant components in reliability--redundancy allocation problems. *IIE transactions (institute of industrial engineers)*, *48*(11), 1043–1057. https://doi.org/10.1080/0740817X.2016.1189631

[8] Aghaei, M., Zeinal Hamadani, A., & Abouei Ardakan, M. (2017). Redundancy allocation problem for k-out-of-n systems with a choice of redundancy strategies. *Journal of industrial engineering international*, *13*(1), 81–92. https://doi.org/10.1007/s40092-016-0169-3

[9] Gholinezhad, H., & Zeinal Hamadani, A. (2017). A new model for the redundancy allocation problem with component mixing and mixed redundancy strategy. *Reliability engineering and system safety*, *164*, 66–73. https://doi.org/10.1016/j.ress.2017.03.009

[10] Ouyang, Z., Liu, Y., Ruan, S. J., & Jiang, T. (2019). An improved particle swarm optimization algorithm for reliability-redundancy allocation problem with mixed redundancy strategy and heterogeneous components. *Reliability engineering and system safety*, *181*, 62–74. https://doi.org/10.1016/j.ress.2018.09.005

[11] Peiravi, A., Karbasian, M., Ardakan, M. A., & Coit, D. W. (2019). Reliability optimization of series-parallel systems with K-mixed redundancy strategy. *Reliability engineering and system safety*, *183*, 17–28. https://doi.org/10.1016/j.ress.2018.11.008

[12] Hadipour, H., Amiri, M., & Sharifi, M. (2019). Redundancy allocation in series-parallel systems under warm standby and active components in repairable subsystems. *Reliability engineering and system safety*, *192*, 106048. https://doi.org/10.1016/j.ress.2018.01.007

[13] Peiravi, A., Ardakan, M. A., & Zio, E. (2020). A new Markov-based model for reliability optimization problems with mixed redundancy strategy. *Reliability engineering and system safety*, *201*, 106987. https://doi.org/10.1016/j.ress.2020.106987

[14] Gong, M., Liu, H., & Peng, R. (2020). Redundancy allocation of mixed warm and cold standby components in repairable K-out-of-N systems. *Proceedings of the institution of mechanical engineers, part o: journal of risk and reliability*, *234*(5), 696–707. https://doi.org/10.1177/1748006X20921306

[15] Sadeghi, M., Roghanian, E., Shahriari, H., & Sadeghi, H. (2021). Reliability optimization for non-repairable series-parallel systems with a choice of redundancy strategies and heterogeneous components: erlang time-to-failure distribution. *Proceedings of the institution of mechanical engineers, part o: journal of risk and reliability*, *235*(3), 509–528. https://doi.org/10.1177/1748006X20952575

[16] Chambari, A., Azimi, P., & Najafi, A. A. (2021). A bi-objective simulation-based optimization algorithm for redundancy allocation problem in series-parallel systems. *Expert systems with applications*, *173*, 114745. https://doi.org/10.1016/j.eswa.2021.114745

[17] Reihaneh, M., Abouei Ardakan, M., & Eskandarpour, M. (2022). An exact algorithm for the redundancy allocation problem with heterogeneous components under the mixed redundancy strategy. *European journal of operational research*, *297*(3), 1112–1125. https://doi.org/10.1016/j.ejor.2021.06.033

[18] Coit, D. W., & Liu, J. (2000). System reliability optimization with k-out-of-n subsystems. *International journal of reliability, quality and safety engineering*, *7*(2), 129–142. https://doi.org/10.1142/S0218539300000110

[19] Wang, W., & Loman, J. (2002). Reliability/availability of k-out-of-n system with m cold standby units. *Annual reliability and maintainability symposium. 2002 proceedings (Cat. No. 02CH37318)* (pp. 450–455). IEEE. https://doi.org/10.1109/RAMS.2002.981684

[20] Elegbede, C., & Adjallah, K. (2003). Availability allocation to repairable systems with genetic algorithms: A multi-objective formulation. *Reliability engineering and system safety*, *82*(3), 319–330. https://doi.org/10.1016/j.ress.2003.08.001

[21] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, *6*(2), 182–197. https://doi.org/10.1109/4235.996017

[22] Safari, J. (2012). Multi-objective reliability optimization of series-parallel systems with a choice of redundancy strategies. *Reliability engineering and system safety*, *108*, 10–20. https://doi.org/10.1016/j.ress.2012.06.001

[23] Amiri, M., & Khajeh, M. (2016). Developing a bi-objective optimization model for solving the availability allocation problem in repairable series–parallel systems by NSGA II. *Journal of industrial engineering international*, *12*(1), 61–69. https://doi.org/10.1007/s40092-015-0128-4

Supplementary Material

Appendix 1

Availability formula in in k-out-of-n systems with active components

Few studies have offered general formula to calculate availability. Carpitella et al. [3] have proposed the following formula to calculate availability of k-out-of-n systems with active redundant components.

$$A_A = \left[\frac{\sum_{i=k}^{n} \binom{n}{i} \mu^i \lambda^{(n-i)}}{\sum_{i=k}^{N} \binom{n}{i} \mu^i \lambda^{(n-i)} + \binom{n}{k-1} \mu^{(k-1)} \lambda^{(n-k+1)}} \right].$$

As the article shows, Markov chains have been used to obtain the formula for a 2-out-of-3 system. Then, the formula has been generalized to k-out-of-n systems. Here, we have proved the formula for an (n-1)-out-of-n system:

Proof: consider the following Markov graph (*Fig. A1*) of an (n-1)-out-of-n system. 0 and 1 represent the functioning states, and 2 is the failure state. In 0, all the n components are available.

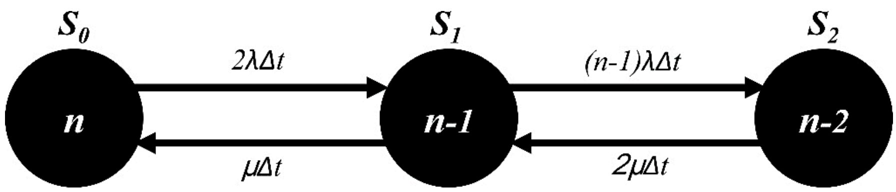

Fig. A1. Markov graph of an (n-1)-out-of-n system.

The possible transitions of the system between the states are summarized in *Table A1*. After Δt, the system will be in S_0 state if one of the following scenarios occurs:

I. It has previously been in S0 and has remained in this state.

II. It has moved from S1 to S0.

Table A1. Transition matrix of an (n-1)-out-of-n system.

		Future States		
		S_0	S_1	S_2
Present States	S_0	$1-n\lambda\Delta t$	$n\lambda\Delta t$	0
	S_1	$\mu\Delta t$	$1-\mu\Delta t-(n-1)\lambda\Delta t$	$(n-1)\lambda\Delta t$
	S_2	0	$2\mu\Delta t$	$1-2\mu\Delta t$

According to the transition matrix:

$$P_0(t+\Delta t) = (1-n\lambda\Delta t)\, P_0(t) + \mu\Delta t\, P1(t).$$

It is clear that the time derivative of the probability of the system being in S0 is equal to the probability of entering S0 minus the probability of exiting it:

$$\frac{dP_0(t)}{dt} = -n\lambda P_0(t) + \mu\, P_1(t). \quad \textbf{(S1-1)}$$

Similarly, based on the transition matrix:

$$P_1(t+\Delta t) = n\lambda\Delta t\, P_0(t) + (1-\mu\Delta t-(n-1)\lambda\Delta t)\, P_1(t) + 2\mu\Delta t\, P_2(t).$$

Therefore, for S1:

$$\frac{dP_1(t)}{dt} = n\lambda P_0(t) - \mu P_1(t) - (n-1)\lambda P_1(t) + 2\mu P_2(t). \quad \textbf{(S1-2)}$$

Also, for S2:

$$P_2(t+\Delta t) = (n-1)\lambda\Delta t\, P_1(t) + (1-2\mu\Delta t)\, P_2(t).$$

and

$$\frac{dP_2(t)}{dt} = (n-1)\lambda P_1(t) - 2\mu P_2(t). \quad \textbf{(S1-3)}$$

The time-dependent availability can be calculated by solving the above differential equations (S1-1, S1-2, and S1-3). The stationary availability is calculated over long enough periods of time (t→∞). In this case, since the probability of the subsystem being in any of the possible states, Pi (t), takes a constant value, therefore:

$$\lim_{t\to\infty}\frac{dP_i(t)}{dt} = 0. \quad \textbf{(S1-1)}$$

As a result, the above-mentioned probabilities will be time-independent functions, and:

$$\lim_{t\to\infty}\frac{dP_0(t)}{dt} = -n\lambda P_0(\infty) + \mu\, P_1(\infty) = 0. \quad \textbf{(S1-4)}$$

$$\lim_{t\to\infty}\frac{dP_1(t)}{dt} = n\lambda P_0(\infty) - \mu P_1(\infty) - (n-1)\lambda P_1(\infty) + 2\mu P_2(\infty) = 0. \quad \textbf{(S1-5)}$$

$$\lim_{t\to\infty}\frac{dP_2(t)}{dt} = (n-1)\lambda P_1(\infty) - 2\mu P_2(\infty) = 0. \quad \textbf{(S1-6)}$$

The total probability of the system in different possible states is equal to one:

$$P0 + P1 + P2 = 1. \quad \textbf{(S1-7)}$$

Equation (S1-4) suggests that:

$$P_1 = \frac{N\lambda}{\mu} P_0. \quad \textbf{(S1-8)}$$

From equations (S1-6) and (S1-6) we conclude:

$$P_2 = \frac{N(N-1)\lambda^2}{2\mu^2} P_0. \quad \textbf{(S1-9)}$$

By placing (S1-8) and (S1-9) in (S1-7), the following equation is obtained:

$$P_0 + \frac{N\lambda}{\mu} P_0 + \frac{N(N-1)\lambda^2}{2\mu^2} P_0 = 1.$$

As the result;

$$P_0 = \frac{2\mu^2}{2\mu^2 + 2\mu N\lambda + N(N-1)\lambda^2}. \quad \textbf{(S1-10)}$$

Similarly, the following is obtained:

$$P_1 = \frac{2\mu N\lambda}{2\mu^2 + 2\mu N\lambda + N(N-1)\lambda^2}. \quad \textbf{(S1-11)}$$

Since the subsystem is only available in P0 and P1, therefore:

$$A_S = P_0 + P_1. \quad \textbf{(S1-12)}$$

By placing (S1-10) and (S1-11) in (S1-12), the stationary availability of the subsystem is obtained as follows:

$$A_S = \frac{2\mu^2 + 2\mu N\lambda}{2\mu^2 + 2\mu N\lambda + N(N-1)\lambda^2}. \quad \textbf{(S1-13)}$$

The obtained formula can be generalized to obtain stationary availability in active redundancy subsystems with k-out-of-n configuration. The equation is as follows:

$$A_A = \left[\frac{\sum_{i=k}^{n}\binom{n}{i}\mu^i\lambda^{(n-i)}}{\sum_{i=k}^{N}\binom{n}{i}\mu^i\lambda^{(n-i)} + \binom{n}{k-1}\mu^{(k-1)}\lambda^{(n-k+1)}}\right]. \quad \textbf{(S1-14)}$$

Appendix 2. Availability formula in in k-out-of-n systems with cold standby components

To the best of our knowledge, no precise formula has been proposed to calculate the availability of k-out-of-n systems with cold standby components. In a study by Wang and Loman [19] a formula has been proposed to obtain the availability of (n-1)-out-of-n:G systems with cold redundant components. Although the authors have mentioned that the formula is obtained by solving the Markov model, but they have not proved the formula. We prove the formula by using the Markov chain and the transition matrix.

Proof: consider the following Markov graph (*Fig. A2*) of an (n-1)-out-of-n:G system. 0-3 represent the functioning states, and 4 is the failure state.

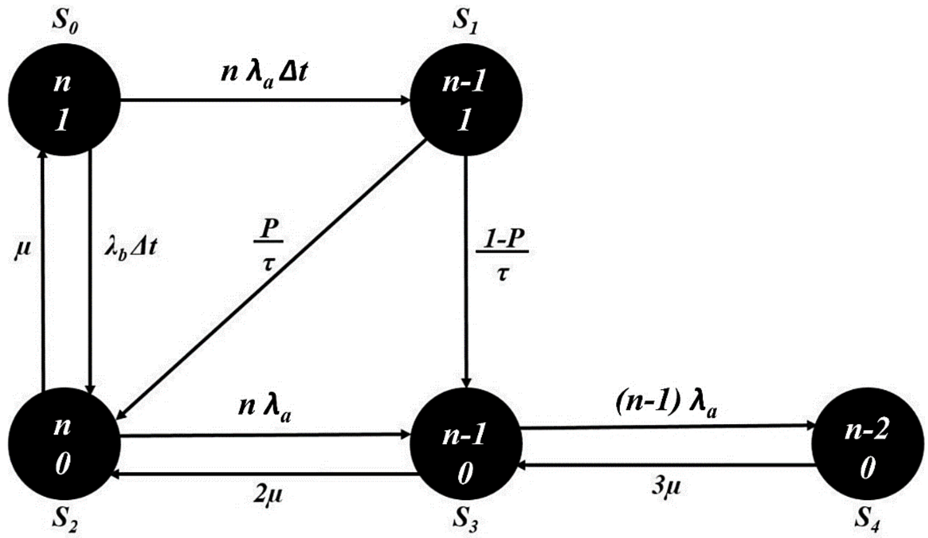

Fig. A2. Markov graph of an (n-1)-out-of-n:G system.

Table A2 shows the transitions of the system between the states.

Table A2. Transition matrix of an (n-1)-out-of-n:G system.

		Future States				
		S_0	**S_1**	**S_2**	**S_3**	**S_4**
Present states	S_0	$1-n\lambda_a\Delta t-\lambda_b\Delta t$	$n\lambda_a\Delta t$	$\lambda_b\Delta t$	0	0
	S_1	0	$1-\frac{P}{\tau}\Delta t-\frac{1-P}{\tau}\Delta t$	$\frac{P}{\tau}\Delta t$	$\frac{1-P}{\tau}\Delta t$	0
	S_2	$\mu\Delta t$	0	$1-\mu\Delta t-n\lambda_a\Delta t$	$n\lambda_a\Delta t$	0
	S_3	0	0	$2\mu\Delta t$	$1-(n-1)\lambda_a\Delta t-2\mu\Delta t$	$(n-1)\lambda_a\Delta t$
	S_4	0	0	0	$3\mu\Delta t$	$1-3\mu\Delta t$

According to the transition matrix:

$$P_0(t+\Delta t) = (1-n\lambda_a\Delta t-\lambda_b\Delta t)\, P_0(t) + \mu\Delta t\, P_2(t),$$

$$\lim_{\Delta t\to 0}\frac{P_0(t+\Delta t)-P_0(t)}{\Delta t} = -n\lambda_a P_0(t) - \lambda_b P_0(t) + \mu\Delta t\, P_2(t). \quad \textbf{(S2-1)}$$

$$P_1(t+\Delta t) = n\lambda_a\Delta t\, P_0(t) + \left(1-\frac{P}{\tau}\Delta t-\frac{1-P}{\tau}\Delta t\right) P_1(t),$$

$$\lim_{\Delta t\to 0}\frac{P_1(t+\Delta t)-P_1(t)}{\Delta t} = n\lambda_a P_0(t) - \frac{1}{\tau}P_1(t). \quad \textbf{(S2-2)}$$

$$P_2(t+\Delta t) = \lambda_b\Delta t\, P_0(t) + \frac{P}{\tau}\Delta t\, P_1(t) + (1-\mu\Delta t-n\lambda_a\Delta t)\, P_2(t) + 2\mu\Delta t\, P_3(t),$$

$$\lim_{\Delta t\to 0}\frac{P_2(t+\Delta t)-P_2(t)}{\Delta t} = \lambda_b P_0(t) + \frac{P}{\tau}P_1(t) - \mu P_2(t) - n\lambda_a P_2(t) + 2\mu P_3(t), \quad \textbf{(S2-3)}$$

$$P_3(t+\Delta t) = \frac{1-P}{\tau}\Delta t\, P_1(t) + n\lambda_a\Delta t\, P_2(t) + (1-(n-1)\lambda_a\Delta t-2\mu\Delta t)\, P_3(t) + 3\mu\Delta t\, P_4(t),$$

$$\lim_{\Delta t\to 0}\frac{P_3(t+\Delta t)-P_3(t)}{\Delta t} = \frac{1-P}{\tau}P_1(t) + n\lambda_a P_2(t) - (n-1)\lambda_a P_3(t) - 2\mu P_3(t) + 3\mu P_4(t). \quad \textbf{(S2-4)}$$

$$P_4(t+\Delta t) = (n-1)\lambda_a\Delta t\, P_3(t) + (1-3\mu\Delta t)\, P_4(t),$$

$$\lim_{\Delta t\to 0}\frac{P_4(t+\Delta t)-P_4(t)}{\Delta t} = (n-1)\lambda_a P_3(t) - 3\mu P_4(t). \quad \textbf{(S2-5)}$$

The time-dependent availability of the subsystem can be obtained by solving the *Eqs. (S2-1)-(S2-5)*. However, the stationary availability is calculated when $t\to\infty$. In this case, the probability of the system being in any of the possible states (P_i) takes on a fixed value, therefore:

$$\lim_{t\to\infty}\frac{dP_i(t)}{dt} = 0.$$

As a result, Pi(t) functions will be time-independent and *Eq. (S2-1)* will be as follows:

$$-N\lambda_a P_0(t) - \lambda_b P_0(t) + \mu\Delta t\, P_2(t) = 0.$$

Therefore,

$$P_2 = \frac{N\lambda_a+\lambda_b}{\mu}\, P_0. \quad \textbf{(S1-6)}$$

In a similar way, it follows from *Eq. (S2-2)*:

$$P_1 = N\lambda_a \tau P_0. \tag{S1-7}$$

It also follows from *Eq. (S2-3)*:

$$P_3 = \frac{N^2\lambda_a^2 + \mu N\lambda_a + \mu\lambda_b + N\lambda_a\lambda_b - \mu\lambda_b - PN\mu\lambda_a}{2\mu^2} P_0. \tag{S1-8}$$

Similarly, it results from *Eq. (S2-4)*:

$$P_4 = \frac{(N-1)\lambda_a\,(N^2\lambda_a^2 + \mu N\lambda_a + \mu\lambda_b + N\lambda_a\lambda_b - \mu\lambda_b - PN\mu\lambda_a)}{6\mu^3} P_0. \tag{S1-9}$$

The total probability of the possible states is equal to one:

$$P_0 + P_1 + P_2 + P_3 + P_4 = 1.$$

By placing *Eq*s. *(S2-6)-(S2-9)* in the latter Equation, the following relation is obtained:

$$\frac{6\mu^3 + 6\mu^3 N\lambda_a\tau + 6\mu^2(N\lambda_a + \lambda_b) + 3\mu(N^2\lambda_a^2 + \mu N\lambda_a + \mu\lambda_b + N\lambda_a\lambda_b - \mu\lambda_b - PN\mu\lambda_a) + (N-1)\lambda_a(\mu N\lambda_a + \mu\lambda_b + N^2\lambda_a^2 + N\lambda_a\lambda_b - \mu\lambda_b - PN\mu\lambda_a)}{6\mu^3} P_0 = 1.$$

Which results in:

$$P_0 = \frac{6\mu^3}{6\mu^3 + 6\mu^3 N\lambda_a\tau + 6\mu^2(N\lambda_a + \lambda_b) + 3\mu(N^2\lambda_a^2 + \mu N\lambda_a + N\lambda_a\lambda_b - PN\mu\lambda_a) + (N-1)\lambda_a(\mu N\lambda_a + N^2\lambda_a^2 + N\lambda_a\lambda_b - PN\mu\lambda_a)}. \tag{S1-10}$$

In the system of the study, availability is obtained as follows:

$$A_S = P_0 + P_1 + P_2 + P_3.$$

By placing *Eq. (S2-10)* in *Eqs. (S2-6)-(S2-9)* and rewriting the latter equation, we conclude:

$$A_S = \frac{6\mu^3 + 6\mu^3 N\lambda_a\tau + 6\mu^2(N\lambda_a + \lambda_b) + 3\mu(N^2\lambda_a^2 + \mu N\lambda_a + N\lambda_a\lambda_b - PN\mu\lambda_a)}{6\mu^3 + (N-1)N^2\lambda_a^3 + N(N-1)\lambda_a^2\lambda_b - (N-1)NP\mu\lambda_a^2 + 4N^2\lambda_a^2\mu - N\lambda_a^2\mu + 6\mu^3 N\lambda_a\tau + 9N\mu^2\lambda_a + 6\mu^2\lambda_b + 3N\mu\lambda_a\lambda_b - 3\mu^2 N\lambda_a P}. \tag{S1-11}$$