INAE Letters (2019) 4:191–206
https://doi.org/10.1007/s41403-019-00076-8

ORIGINAL ARTICLE

NSGA-II Based Analysis of Fuzzy Multi-objective Reliability–Redundancy Allocation Problem Using Various Membership Functions

Hemant Kumar[1] · **Awadh Pratap Singh**[1] · **Shiv Prasad Yadav**[1]

Received: 30 October 2018 / Accepted: 24 August 2019 / Published online: 11 September 2019

Abstract

In the broadest sense, reliability is a measure of performance of the system under the stated conditions. The reliability–redundancy allocation problem gives a highly reliable system in the presence of optimal redundant components. This design is the most preferred by the design engineer. During the designing phase of the system, all the design data involved in the system are not very precise. Various types of uncertainties such as expert's information character, qualitative statements, vagueness, incompleteness, unclear system boundaries, inability to evaluate the relative importance of the objectives, etc., are typical for many practical problems. Fuzzy set theory is an efficient technique to tackle such types of uncertainties in the system design problem. In this paper, the goals of the fuzzy multi-objective reliability–redundancy allocation problem are specified by various membership functions such as linear, quadratic, parabolic, and hyperbolic. An efficient multi-objective evolutionary algorithm, namely, NSGA-II is employed to solve it. The Pareto-optimal fronts for the various membership functions are shown in both the membership and objective spaces. Fuzzy ranking method then finds the best compromise solution for each membership function. Finally, the performance of membership functions is ranked by the data envelopment analysis by taking cost criteria (cost, weight, and volume) as inputs and benefit criteria (reliability and maximum satisfaction level) as outputs of the system. The effectiveness of the proposed approach is illustrated by a numerical example of the over-speed protection system for a gas turbine. A comparative analysis of the proposed approach is given with the existing approach.

Keywords System reliability · NSGA-II · Membership function · Pareto-optimal front (POF) · Fuzzy ranking method · DEA

List of Symbols

n_j	Number of components in the jth subsystem
r_j	Reliability of each component in the jth subsystem
n	$(n_1, n_2, \dots, n_m)$, the vector of redundancy allocation for the system
r	$(r_1, r_2, \dots, r_m)$, the vector of component reliabilities for the system
m	Total number of subsystems
M	Number of constraints
g_i	ith constraint function, $i = 1, 2, \dots M$
v_j	The volume of each component in the jth subsystem
w_j	The weight of each component in the jth subsystem
γ_j, δ_j	Constants of characteristics factors for each component in the jth subsystem
T	Active operational time
$r_{j,\min}$	The minimum value of reliability of each component in the jth subsystem
c_j	The cost of each component in the jth subsystem
$r_{j,\max}$	The maximum value of reliability of each component in the j th subsystem
$n_{j,\max}$	The maximum number of components in the jth subsystem

✉ Hemant Kumar
hkumar1@ma.iitr.ac.in

Awadh Pratap Singh
awadhiitr.dma2017@iitr.ac.in

Shiv Prasad Yadav
yadavfma@iitr.ac.in

[1] Department of Mathematics, Indian Institute of Technology Roorkee, Roorkee, Uttarakhand 247667, India

R_S, C_S, W_S, V_S	System reliability, system cost, system weight, and system volume, respectively
R	Lower limit on R_S
C	Upper limit on C_S
W	Upper limit on system weight
V	Upper limit on system volume

Introduction

The reliability–redundancy allocation problem (RRAP) is a mixed-integer non-linear programming problem and NP-hard (Chern 1992) for which only approximate solutions have been proposed (Tillman et al. 1977; Sakawa 1978; Dhingra 1992). The increment of redundancies adds more cost, weight, volume, etc., in the system. Therefore, a set of trade-offs is required in multi-objective reliability–redundancy allocation problem (MORRAP) where the number of redundancies is an important decision variable. Heuristic approaches to MORRAP can be viewed in Huang et al. (2006), Tavakkoli et al. (2008), Huang et al. (2009), Garg and Sharma (2013), Sheikhalishahi et al. (2013), Damghani et al. (2013), Liu (2013), Zhang et al. (2014), Kim and Kim (2017) etc.

Multi-objective evolutionary algorithms (MOEAs) (Deb et al. 2002) are popular techniques to handle different types of multi-objective optimization problems (MOOPs) for the purpose of finding multiple solutions (popularly known as Pareto-optimal solutions) in a single simulation run. This group of algorithms conjugates the basic concepts of dominance. MOEAs are able to deal with non-continuous, non-convex, and/or non-linear as well as problems whose objective functions are not explicitly known (Salazar et al. 2006). Taboada et al. (2007) presented a practical solution for multi-objective reliability-based system design using the MOEA and clustering technique. A number of different MOEAs such as MOGA (Fonseca and Fleming 1993), NPGA (Horn et al. 1994), NSGA (Srinivas and Deb 1994), SPEA (Zitzler and Thiele 1998), PAES (Knowles and Corne 1999), MOMGA (Veldhuizen and Lamont 2000), SPEA2 (Zitzler et al. 2001), PESA-II (Corne at al. 2001), NSGA-II (Deb et al. 2002), MOEA/D (Zhang and Li 2007), AGE-II (Wagner and Neumann 2013), etc., have been developed. An ideal multi-objective optimization needs:

1. to get a group of solutions as close as possible to the true Pareto-optimal front, and
2. to maintain these solutions as diverse homogeneous as possible.

The first goal is responsible for the convergence, while the second is for diversity in the solutions set. Many MOEAs and their solution techniques to the MOOP can be viewed in Deb (2001), Konak et al. (2006), Coello et al. (2007), and Zhou et al. (2011).

Elitist non-dominated sorting genetic algorithm (NSGA-II) is one of the popular MOEAs. It comprises a second-generation MOEA which gives much better spread and good convergence near the true Pareto-optimal front compared to other elitist MOEAs such as SPEA and PAES. Simulation results of the constrained NSGA-II give better performance on several non-linear problems (Deb et al. 2002). It has some special property that makes it different from other MOEAs such as elitist strategy, parameter-less sharing approach, crowding distance, classifying the solutions into the fronts, efficient handling constraints, and low computational requirements. Salazar et al. (2006) demonstrated NSGA-II in identifying a set of optimal solutions known as Pareto-optimal front by solving constrained redundancy problems. Wang et al. (2009) solved the MORRAP using NSGA-II and compared their results with single-objective approaches. Safari (2012) proposed a variant of NSGA-II in solving MORRAP. Sharifi et al. (2016) used the NSGA-II algorithm in solving MORRAP for a series–parallel problem and k-out-of-n subsystems with three objectives. Recently, Muhuri et al. (2018) used NSGA-II to solve the MORRAP with interval type-2 fuzzy environment which considers higher order uncertainties in the component. Practically, various types of uncertainties are involved in system design. During the designing phase of the system, the designer does not have a clear idea about the design parameters such as reliability, cost, and weight of the constituent components. As a result, approximate values are considered by guessing. The environmental factors such as improper storage, adverse operating conditions, age etc., affect the reliability of the system. Therefore, all the design data involved in the system are hardly precise. In general, redundant components are found in the different models which contain less information regarding their critical parameters, e.g., failure operating time which is used to evaluate the component's reliability. To cope up these issues, fuzzy optimization techniques (Zimmermann 1996) can be useful during the initial stages of the conceptual design of a system. In the fuzzy decision-making process, the linear membership function is often used by fixing two points as the upper and lower levels of acceptability. In real-world situations, models are built more flexible and adaptable to the human decision-making process. It needs some kind of empirical justification or assumption. Keeping these views in mind, several other (non-linear) shapes for membership functions such as concave, convex shapes need to be analyzed in determining their impact on the overall system design process. The membership function of a fuzzy goal has also been viewed as "a kind of utility function representing the degree of satisfaction or acceptance" (Dhingra and Moskowitz 1991). To choose the best compromise solution in the fuzzy MOOP is another challenge. Liu (2013) proposed an approach with fuzzy programming and DEA method to choose the efficient solution in MORRAP. Recently, Kumar and Yadav (2019) presented NSGA-II based decision-making approach to determine the optimal value of MORRAP.

In this paper, MORRAP is considered under several design constraints. The goals of the problem are specified by the various membership functions to look into the impact on the overall system design process. The best compromise solution in each membership function is obtained by the fuzzy ranking method (Pandiarajan and Babulal 2014) and the overall performance of the various membership functions is then measured by the CCR model (Charnes et al. 1978) by taking three inputs as cost, weight, and volume, and two outputs as reliability and maximum satisfaction level of the system obtained by fuzzy ranking method.

Multi-objective Optimization

This section describes some important facts of the MOOP.

Formulation of the MOOP

In general, an MOOP is defined as follows:

$$\begin{aligned} &\text{Minimize } F(X) = \left[f_1(X), f_2(X), \ldots, f_k(X)\right]^T \\ &\quad \text{subject to } g_i(X) = 0,\ i = 1, 2, \ldots, m_e; \\ &g_i(X) \geq 0,\ i = m_e + 1,\ m_e + 2, \ldots, M; \\ &\qquad x_j^l \leq x_j \leq x_j^u, j = 1, 2, \ldots, n, \end{aligned} \tag{1}$$

where $k \geq 2$ is the number of objectives; m_e is the number of equality constraints; M is the total number of constraints; $X = \left[x_1, x_2, \ldots, x_n\right]^T$ is n dimensional decision vector from the feasible region or decision space Ω defined by the following:

$$\Omega = \left\{ X \in \mathbb{R}^n \middle| \begin{array}{l} g_i(X) \geq 0,\ g_i(X) = 0,\ x_j^l \leq x_j \leq x_j^u \\ j = 1, 2, \ldots, n \end{array} \right\},$$

the feasible region $\Omega \subset \mathbb{R}^n$; objective functions $f_p(X)$, $p = 1, 2, \ldots, k$, where $f_p : \Omega \to \mathbb{R}$ and the constrained functions $g_i(X)$, where $g_i : \Omega \to \mathbb{R}$; the image of the feasible region denoted by $Z \subset \mathbb{R}^k$ and it is called a feasible objective region or objective space which is defined by $Z = \left\{F(X) \in \mathbb{R}^k | X \in \Omega\right\}$. The elements of Z are called objective vectors or criterion vector denoted by $F(X) = \left[f_1(X), f_2(X), \ldots, f_k(X)\right]^T$; x_j^l and x_j^u are the lower and upper bounds of the decision variable x_j, respectively. For every point X in the decision space Ω, there exists a point $F(X)$ in the objective space Z. Therefore, it is a mapping between n-dimensional solution vector and k-dimensional objective vector (see Fig. 1).

If all f_ps and g_is are linear, then the problem is called a multi-objective linear programming problem (MOLPP); otherwise, it is called a multi-objective non-linear programming problem (MONLPP).

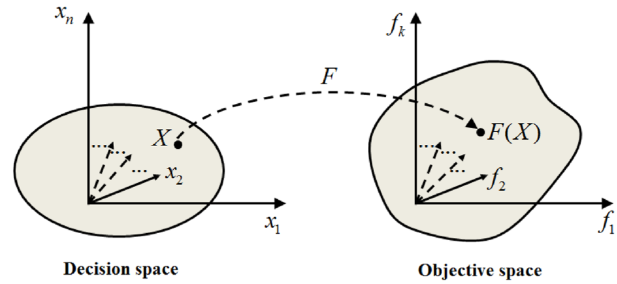

Fig. 1 Multi-objective evaluation mapping

Basic definitions

The concept of optimality in an MOOP depends on *Pareto optimality*. Therefore, the following definitions are defined in terms of Pareto terminology.

Definition 1 Pareto dominance (Coello et al. 2007) A vector $X \in \Omega$ is said to dominate another $Y \in \Omega$ denoted by $X \preceq Y$ iff $f_i(X) \leq f_i(Y) \forall i = 1, 2, \ldots, k$, and there exists at least one $f_j(X) < f_j(Y), j \in \{1, 2, \ldots, k\}, j \neq i$.

Definition 2 Pareto-optimal solution (Coello et al. 2007) A solution vector $X \in \Omega$ is said to be Pareto-optimal solution (Pareto-optimal) iff there does not exist another vector $X' \in \Omega$ which dominates $X \in \Omega$.

Definition 3 Pareto-optimal set (Coello et al. 2007) The Pareto-optimal set is defined as $\mathrm{PS} := \left\{X \in \Omega | \neg\exists X' \in \Omega :\ X' \preceq X\right\}$.

Definition 4 Pareto-optimal front (Coello et al. 2007) The Pareto-optimal front is defined as $\mathrm{PF} := \left\{F(X) = \left[f_1(X), f_2(X), \ldots, f_k(X)\right]^T | X \in \mathrm{PS}\right\}$.

Definition 5 Ideal objective vector (Coello et al. 2007) If a decision vector $X^* = \left[x_1^*, x_2^*, \ldots, x_n^*\right]^T \in \Omega$ is such that $f_i(X^*) = \min_{X \in \Omega} f_i(X)$, $i \in \{1, 2, \ldots, k\}$; then the vector $F(X^*) = \left[f_1(X^*), f_2(X^*), \ldots, f_k(X^*)\right]^T \in Z$ is called an ideal objective vector for an MOOP given by (1) and $X^* = \left[x_1^*, x_2^*, \ldots, x_n^*\right]^T \in \Omega$ is called an ideal vector.

Definition 6 Utopian objective vector (Deb 2001) A utopian objective vector $F(X^{**})$ has each of its components marginally smaller than the ideal objective vector, or $F(X^{**}) = F(X^*) - \in_i$ with $\in_i > 0\, \forall i = 1, 2, \ldots, k$.

Definition 7 Nadir (anti-ideal) objective vector (Deb 2001) Unlike the ideal objective vector which represents the lower bound of each objective, the nadir objective vector F^{nad}

represents the upper bound of each objective in the entire objective space $\mathbb{R}^k$.

Definition 8 Fuzzy set (Zimmermann 1996) Let X be a collection of objects generically denoted by x. A fuzzy set $\widetilde{A}$ in X is a set of ordered pair defined in the form as follows:

$$\widetilde{A} = \left\{ \left(x, \mu_{\widetilde{A}}(x)\right) x \in X \right\},$$

where $\mu_{\widetilde{A}} : X \to [0,1]$ is called the membership function and its function value is known as the grade of membership of x in $\widetilde{A}$.

Mathematical Model of the Problem

RRAP can be understood by a series–parallel system configuration. The general configuration of the series–parallel system is shown in Fig. 2. This system contains several parallels and identical components arrayed in each stage. Generally, redundancy is applied for increasing the system reliability, but this technique gives more complexity in terms of cost, weight, volume, etc., to the system design. Therefore, it is better to solve the multi-objective programming model based problem. Here, twofold design variables are required to determine the optimal design of the system. One is the reliability of each component and other is to select a number of redundant components in each stage. Mathematically, MORRAP is given as follows:

$$\begin{aligned}
&\text{Max. } R_S(r,n) = \prod_{j=1}^{m} \left[1 - \left(1 - r_j\right)^{n_j}\right] \\
&\text{Min. } C_S(r,n) = \sum_{j=1}^{m} C_j\left(r_j\right)\left[n_j + \exp\left(n_j/m\right)\right] \\
&\text{subject to } g_i(r,n) \le 0, i = 1,2,\dots,M \\
&0 \le r_{j,\min} \le r_j \le r_{j,\max} \le 1, \\
&1 \le n_{j,\min} \le n_j \le n_{j,\max} \quad n_j \in \mathbb{Z}^+ \\
&j = 1,2,\dots,m.
\end{aligned} \tag{2}$$

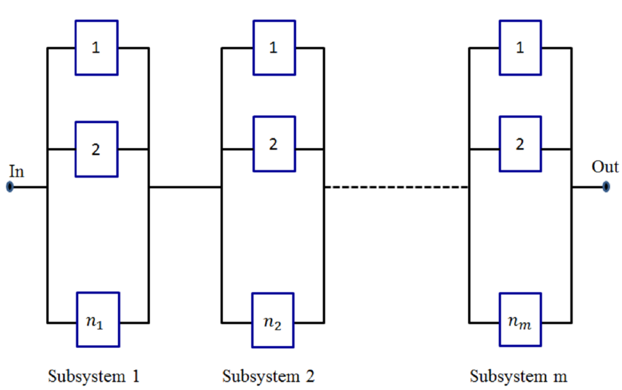

Fig. 2 Series–parallel system configuration

Elitist Non-dominated Sorting Genetic Algorithm (NSGA-II)

Srinivas and Deb (1994) proposed the NSGA which is an early dominance-based MOEA. The purpose of developing NSGA is to find better solutions according to their non-domination levels. NSGA uses the naive and slow (Deb 2001) sorting approach to distribute a population into different non-domination levels, and a sharing function method to maintain the diversity of the population. However, it has high computational complexity $O(kN^3)$, where k is the number of objectives and N is the population size in the non-dominated sorting. Therefore, NSGA is computationally expensive for large population sizes. Moreover, NSGA is a non-elitist approach which affects its convergence rates compared to other MOEAs and it also requires a sharing parameter to calculate the sharing fitness, which ensures the diversity of the population.

Deb et al. (2002) proposed NSGA-II to overcome the drawbacks of NSGA. Specifically, NSGA-II presents a fast non-dominated sorting approach with the worst-case computational complexity $O\left(k(2N)^2\right)$. This approach searches iteratively non-dominated solutions into different fronts.

First, for each solution X in the population, the algorithm calculates two entities:

1. n_X, the number of solutions dominating X,
2. S_X, a set of solutions dominated by X.

The solutions for which $n_X = 0$ belong to the first front. Second, for each member Y in the set S_X, the value of n_Y is reduced by one. If any n_Y is reduced to zero during this stage, the corresponding member Y is put in the second front. The above process is continued with each member in the second front to identify the third front and so on. Furthermore, NSGA-II applies the concept of crowding distance with the worst-case computational complexity $O(k(2N)\log(2N))$. The introduction of crowding distance replaces the fitness sharing approach that requires a sharing parameter to be set by the user.

The crowding-distance value (CD_X) (see Fig. 3) of the X^{th} solution is calculated as follows:

$$d_{Xp} = \frac{f_p^{X+1} - f_p^{X-1}}{f_p^{\max} - f_p^{\min}} \tag{3}$$

$$\mathrm{CD}_X = \sum_{p=1}^{k} d_{Xp}, \tag{4}$$

where f_p^{X+1} and f_p^{X-1} denote the pth objective function of the $(X+1)$th and $(X-1)$th individual (solution), respectively, and $f_p^{\max}$ and $f_p^{\min}$ represent the maximum and minimum values of the pth objective function.

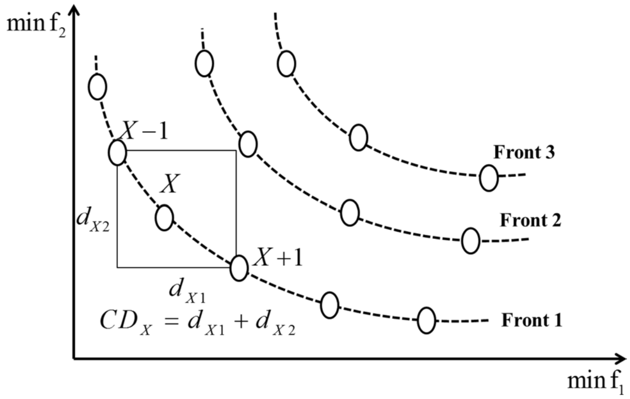

Fig. 3 Fitness evaluation and individual crowding distance estimation

A higher value of crowding-distance gives the lesser crowded region and vice versa (Deb et al. 2002). Therefore, the crowding distance selects solutions located in less-crowded regions after the fast non-dominated sorting procedure, which is extended to an entire POF to maintain the diversity in the solution set. Finally, NSGA-II uses an elitist strategy with the worst-case computational complexity $O(2N\log(2N))$. The elitist strategy (Zitzler et al. 2000; Laumanns et al. 2002) is used to enhance the convergence of an MOEA and avoid the loss of optimal solutions after getting it.

Deb et al. (2002) proposed constraint dominance-based binary tournament selection method in constraint handling procedure. A search space (decision space) is divided by the constraints in two regions—feasible and infeasible. Accordingly, a solution X is called a constrained-dominate to a solution Y if

- X is feasible and Y is infeasible.
- X and Y are infeasible, but X contains a smaller overall constraint violation.
- X and Y are feasible, but X dominates Y.

The pseudo code of the NSGA-II is given as:

1: Initialize randomly population P_0
2: Compute fitness values of individuals in P_0
3: Perform non-dominated sorting on P_0
4: Apply binary tournament selection on P_0
5: Generate child population Q_0
6: Apply recombination and mutation
7: **While** $t < t_{max}$ (max no. of generation) **do**
8: Generate $R_t = P_t \cup Q_t$
9: Perform non-dominated sorting on R_t
10: Copy individuals from non-dominated fronts P_{t+1}
11: Apply binary tournament selection on P_{t+1}
12: Generate child population Q_{t+1}
13: Apply recombination and mutation
14: **End while**
15: Return

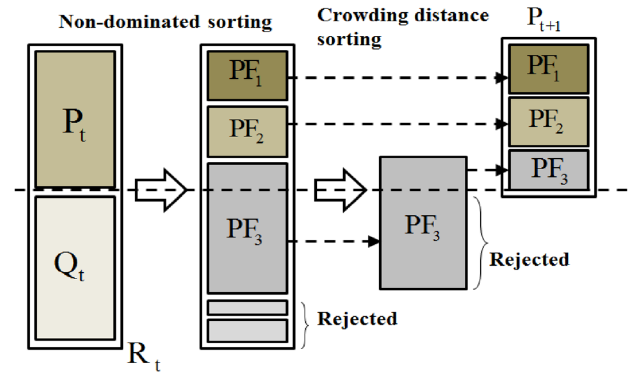

Fig. 4 An evaluation cycle of the NSGA-II algorithm

In Fig. 4, an evaluation cycle of the NSGA-II is shown. First, an offspring Q_t of size N is obtained using the genetic operators such as selection, recombination, and mutation. A combined population R_t of size 2 N is then formed which consists of the current population P_t and the offspring population Q_t. Using fast non-dominated sorting, R_t is divided into different fronts $\mathrm{PF}_1, \mathrm{PF}_2, \ldots, \mathrm{PF}_n$. Let the number of solutions in each front PF_i be N_i. Next, we choose members for the new population P_{t+1} from the front PF_1 to PF_{t-1}, noting that $N_1 + N_2 + \cdots + N_t > N$ and $N_1 + N_2 + \cdots + N_{t-1} \le N$. Afterwards, to get the exactly N population members in P_{t+1}, we sort the solutions in front PF_t using the crowding distance sorting procedure and choose the best solutions to fill empty slot in the new population P_{t+1}. This process is continued until the termination condition is satisfied.

Data Envelopment Analysis (DEA)

DEA is a data-oriented approach for evaluating the performance of decision-making units (DMUs). A complete ranking system is developed for evaluating the best DMU. An efficient design indicator E_k is defined for the kth DMU as follows:

$$E_k = \frac{v_1 y_{1k} + v_2 y_{2k} + \cdots + v_s y_{sk}}{u_1 x_{1k} + u_2 x_{2k} + \cdots + u_m x_{mj}}. \tag{5}$$

Let us consider n DMUs to evaluate the performance of kth DMU. Each DMU has m inputs and s outputs. Then, the CCR model (Charnes et al. 1978) is given in fractional form as follows:

$$\begin{aligned}&\text{Max. } E_k\\ &\text{subject to } 0 \le \frac{v_1 y_{1j} + v_2 y_{2j} + \cdots + v_s y_{sj}}{u_1 x_{1j} + u_2 x_{2j} + \cdots + u_m x_{mj}} \le 1; \\ &j = 1, 2, \ldots, n;\ u_1, u_2, \ldots, u_s \ge 0;\ v_1, v_2, \ldots, v_m \ge 0,\end{aligned} \tag{6}$$

where u_is and v_is are weights corresponding to inputs x_is and y_is, respectively. x_{ij} is the observed amount of the i th input of the j th DMU and y_{rj} is the observed amount of the r th output of the j th DMU. The general form of CCR Model can be written in linear form as follows:

$$\text{Max. } E_k = \sum_{r=1}^{s} v_r y_{rk}$$

subject to

$$\begin{aligned} &\sum_{i=1}^{m} u_i x_{ik} = 1; \\ &\sum_{r=1}^{s} v_r y_{rj} \le \sum_{i=1}^{m} u_i x_{ij},\ j = 1, 2, \ldots, n; \\ &u_i \ge 0,\ v_r \ge 0,\ i = 1, 2, \ldots, m;\ r = 1, 2, \ldots, s. \end{aligned} \tag{7}$$

The dual form of the linear model is given as follows:

$$\begin{aligned} &\text{Min. } \theta_k \\ &\text{subject to } s_{ik}^- = \theta_k x_{ik} - \sum_{j=1}^{n} x_{ij}\lambda_{jk} \ge 0, i = 1, 2, \ldots, m; \\ &s_{rk}^+ = \sum_{j=1}^{n} y_{rj}\lambda_{jk} - y_{rk} \ge 0, r = 1, 2, \ldots, s; \\ &\lambda_{jk} \ge 0, j = 1, 2, \ldots, n. \end{aligned} \tag{8}$$

θ_k is unrestricted in sign, s_{ik}^- is called the ith input excesses and s_{rk}^+ is called the rth output shortfalls.

To discover the possible input excesses and output shortfalls, the following two phases are considered.

1. The efficiency of the CCR model is the optimal value of dual form denoted by θ_k^*.
2. The value of θ_k^* is used in the following problem:

$$\begin{aligned} &\text{Max. } w = es^- + es^+ \\ &\text{subject to} \\ &s_{ik}^- = \theta_k x_{ik} - \sum_{j=1}^{n} x_{ij}\lambda_{jk} \ge 0, i = 1, 2, \ldots, m; \\ &s_{rk}^+ = \sum_{j=1}^{n} y_{rj}\lambda_{jk} - y_{rk} \ge 0, r = 1, 2, \ldots, s; \\ &\lambda_{jk} \ge 0, j = 1, 2, \ldots, n; \\ &e = (1, 1, \ldots, 1). \end{aligned} \tag{9}$$

Any DMU is said to be fully efficient if $\theta_k^* = 1$. There are possibilities to have more than one DMUs with efficiency score 1. To tackle this issue, super-efficiency model (Noura et al. 2011) in DEA can be effective. The following steps are given in this context:

1. Choose upper and lower limits for each input and output among efficient DMUs as follows:

$$\begin{aligned} &E = \left\{ j | E_j^* = 1 \right\}. \\ &x_i^{*u} = \underbrace{\text{Max}}_{j \in E} \left| x_{ij} \right|, \quad i = 1, 2, \ldots, m; \\ &x_i^{*l} = \underbrace{\text{Min}}_{j \in E} \left| x_{ij} \right|, \quad i = 1, 2, \ldots, m; \\ &y_r^{*u} = \underbrace{\text{Max}}_{j \in E} \left| y_{rj} \right|, \quad r = 1, 2, \ldots, s; \\ &y_r^{*l} = \underbrace{\text{Min}}_{j \in E} \left| y_{rj} \right|, \quad r = 1, 2, \ldots, s. \end{aligned}$$

2. Indicate aspiration levels for each input and output. The utility inputs–outputs regarding the definition of sets as $D_i^-, D_i^+, D_o^-, D_o^+$

$$\begin{aligned} &\bar{x} = x_i^{*l}, \quad \forall i\left(i \in D_i^-\right); \\ &\bar{x} = x_i^{*u}, \quad \forall i\left(i \in D_i^+\right); \\ &\bar{y} = y_r^{*l}, \quad \forall i\left(i \in D_o^-\right); \\ &\bar{y} = y_r^{*u}, \quad \forall i\left(i \in D_o^+\right). \end{aligned}$$

3. In this step, we calculate (d_{ij}, d_{rj}) for each DMU$_j$ s.t. $j \in E$ as follows:

$$\begin{aligned} &d_{ij} = \frac{x_{ij}}{\bar{x}_i + \gamma} \forall i \in D_i^+; \\ &d_{ij} = \frac{\bar{x}_i}{x_{ij} + \gamma} \forall i \in D_i^-; \\ &d_{rj} = \frac{y_{rj}}{\bar{y}_r + \gamma} \forall i \in D_r^+; \\ &d_{rj} = \frac{\bar{y}_r}{y_{rj} + \gamma} \forall i \in D_r^-. \end{aligned}$$

Here, γ is a small and non-zero number which prevents division by zero. So, D_j can be defined as: $D_j = \sum_{i \in I} d_{ij} + \sum_{r \in R} d_{rj}$; $I = D_i^+ \cup D_i^-$; $R = D_r^+ \cup D_r^-$. It is noticed that the larger D_j is more successful DMU$_j$ in given proposed objectives for input–output. Thus, it is possible to rank efficient DMUs with higher D_j.

Proposed Methodology

The proposed methodology is given step by step as follows.

Step 1 Find the upper and lower limits on R_S and C_S.

The upper and lower limits on R_S and C_S are obtained by solving each objective one by one as follows:

$$\begin{aligned} &\text{Max./Min. } R_S(r,n) \\ &\text{subject to } g_i(r,n) \le 0,\ i = 1,2,\ldots,M \\ &0 \le r_{j,\min} \le r_j \le r_{j,\max} \le 1, \\ &1 \le n_{j,\min} \le n_j \le n_{j,\max},\ n_j \in \mathbb{Z}^+,\ j = 1,2,\ldots,m \end{aligned} \tag{10}$$

$$\begin{aligned} &\text{Max./Min. } C_S(r,n) \\ &\text{subject to } g_i(r,n) \le 0,\ i = 1,2,\ldots,M \\ &0 \le r_{j,\min} \le r_j \le r_{j,\max} \le 1, \\ &1 \le n_{j,\min} \le n_j \le n_{j,\max},\ n_j \in \mathbb{Z}^+,\ j = 1,2,\ldots,m. \end{aligned} \tag{11}$$

Step 2 Construct the membership functions.

Figure 5 shows various membership functions such as linear, quadratic, parabolic, and hyperbolic to the objective functions R_S and C_S. These membership functions are defined as follows.

1. Linear membership function (Kumar and Yadav 2017, 2019)

   This function is a strictly decreasing concave and convex function. It is defined as follows:

$$\mu_{\tilde{R}_S} = \begin{cases} 0, & R_S \le R_S^l \\ \left(R_S - R_S^l\right)/\left(R_S^u - R_S^l\right), & R_S^l \le R_S \le R_S^u \\ 1, & R_S \ge R_S^u \end{cases} \tag{12}$$

$$\mu_{\tilde{C}_S} = \begin{cases} 1, & C_S \le C_S^l \\ \left(C_S^u - C_S\right)/\left(C_S^u - C_S^l\right), & C_S^l \le C_S \le C_S^u \\ 0, & C_S \ge C_S^u \end{cases}. \tag{13}$$

2. Quadratic membership function (Garg et al. 2014b)

   This function is a convex function and it is defined as follows:

$$\mu_{\tilde{R}_S} = \begin{cases} 0, & R_S \le R_S^l \\ \left(\left(R_S - R_S^l\right)/\left(R_S^u - R_S^l\right)\right)^2, & R_S^l \le R_S \le R_S^u \\ 1, & R_S \ge R_S^u \end{cases} \tag{14}$$

$$\mu_{\tilde{C}_S} = \begin{cases} 1, & C_S \le C_S^l \\ \left(\left(C_S^u - C_S\right)/\left(C_S^u - C_S^l\right)\right)^2, & C_S^l \le C_S \le C_S^u \\ 0, & C_S \ge C_S^u \end{cases} \tag{15}$$

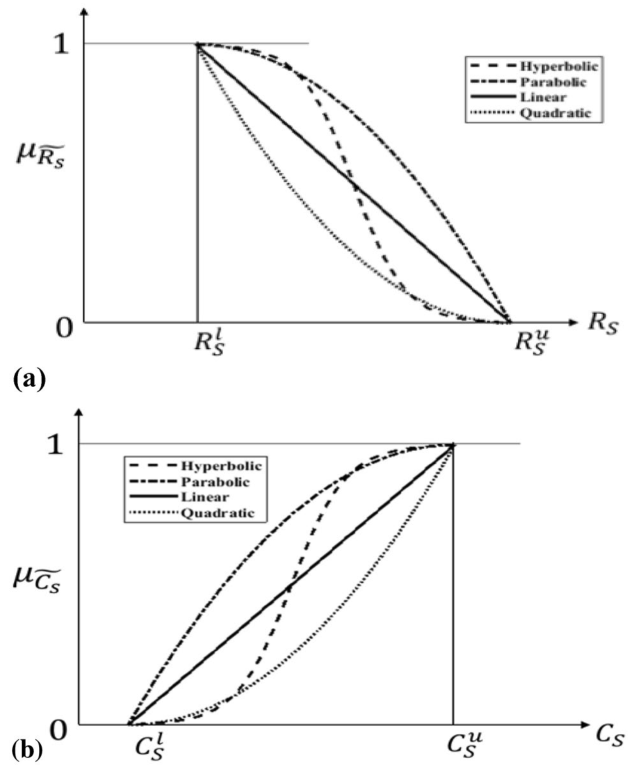

Fig. 5 **a** Monotonically increasing membership functions for system reliability; **b** monotonically decreasing membership functions for system cost

3. Parabolic membership function (Singh and Yadav 2017).

   This function is a concave function and it is defined as follows:

$$\mu_{\tilde{R}_S} = \begin{cases} 0, & R_S \le R_S^l \\ 1 - \left(\left(R_S^u - R_S\right)/\left(R_S^u - R_S^l\right)\right)^2, & R_S^l \le R_S \le R_S^u \\ 1, & R_S \ge R_S^u \end{cases} \tag{16}$$

$$\mu_{\tilde{C}_S} = \begin{cases} 1, & C_S \le C_S^l \\ 1 - \left(\left(C_S - C_S^l\right)/\left(C_S^u - C_S^l\right)\right)^2, & C_S^l \le C_S \le C_S^u \\ 0, & C_S \ge C_S^u \end{cases}. \tag{17}$$

4. Hyperbolic membership function (Dhingra and Moskowitz 1991; Singh and Yadav 2017).

   This function is a convex in one part of the objective function and concave in the remaining part. "If the decision-maker is worse off with respect to a goal, then he/she tends to have a higher marginal rate of satisfaction with respect to that goal. A convex shape captures that behavior in the membership function. Similarly, if the decision-maker is better-off with respect to a goal, then he/she tends to have a smaller marginal rate of satisfaction. This type of behavior is modeled using the concave shape of the membership function" (Singh and Yadav 2017). It is defined as follows:

$$\mu_{\widetilde{R}_S} = \begin{cases} 0, & R_S \le R_S^l \\ \frac{1}{2}\tanh\left(\left(R_S - \frac{R_S^l + R_S^u}{2}\right)\alpha_1\right) + \frac{1}{2}, & R_S^l \le R_S \le R_S^u \;; \quad \alpha_1 = \dfrac{6}{R_S^u - R_S^l} \\ 1, & R_S \ge R_S^u \end{cases} \tag{18}$$

$$\mu_{\widetilde{C}_S} = \begin{cases} 1, & C_S \le C_S^l \\ \frac{1}{2}\tanh\left(\left(\frac{C_S^l + C_S^u}{2} - C_S\right)\alpha_2\right) + \frac{1}{2}, & C_S^l \le C_S \le C_S^u \;; \quad \alpha_2 = \dfrac{6}{C_S^u - C_S^l}. \\ 0, & C_S \ge C_S^u \end{cases} \tag{19}$$

Step 3 Reformulate the problem as a fuzzy MOOP.

The mathematical model of the problem is reformulated as follows:

$$\begin{aligned} &\text{Maximize}\left(\mu_{\widetilde{R}_S}, \mu_{\widetilde{C}_S}\right)\text{or} \\ &\text{Minimize}\left(-\mu_{\widetilde{R}_S}, -\mu_{\widetilde{C}_S}\right) \\ &\text{subject to } g_i(r, n) \le 0,\ i = 1, 2, \dots, M \\ &0 \le r_{j,\min} \le r_j \le r_{j,\max} \le 1, \\ &1 \le n_{j,\min} \le n_j \le n_{j,\max},\ n_j \in \mathbb{Z}^+,\ j = 1, 2, \dots, m. \end{aligned} \tag{20}$$

Theorem 1 *The Pareto-optimal solution of the fuzzy MOOP* (20) *satisfies the MOOP* (2).

Proof Let R^* be a Pareto-optimal solution of the fuzzy MOOP (20). Then, by definition of Pareto-optimal solution, we get

$\nexists R \in \Omega$ (feasible region), such that $-\mu_{\widetilde{R}_S}(R) \le -\mu_{\widetilde{R}_S}(R^*)$ and $-\mu_{\widetilde{C}_S}(R) < -\mu_{\widetilde{C}_S}(R^*)$

$\Leftrightarrow \nexists R \in \Omega$, such that $-h_1\left[R_S(R)\right] \le -h_1\left[R_S(R^*)\right]$ and $-h_2\left[C_S(R)\right] < -h_2\left[C_S(R^*)\right]$

$\Leftrightarrow \nexists R \in \Omega$, such that $h_1\left[R_S(R)\right] \ge h_1\left[R_S(R^*)\right]$ and $h_2\left[C_S(R)\right] > h_2\left[C_S(R^*)\right]$

$\Leftrightarrow \nexists R \in \Omega$, such that $R_S(R) \ge R_S(R^*)$ and $C_S(R) < C_S(R^*)$ (since h_1 is monotonically increasing and h_2 is monotonically decreasing function).

$\Leftrightarrow \nexists R \in \Omega$, such that $-R_S(R) \le -R_S(R^*)$ and $C_S(R) < C_S(R^*)$

$\Leftrightarrow R^* \in \Omega$ is a Pareto-optimal solution of the MOOP given by (2).

Similarly, we can prove by taking the second objective C_S first.

Step 4 Find the Pareto-optimal solutions (Pareto-optimal front).

In this step, the given model of the problem is encoded in MATLAB. All information about optimization part such as no. of objectives, design variables, design constraints, and the NSGA-II parameters such as population size (N), maximum no. of generation ($t_{\max}$), crossover probability (p_c), mutation probability (p_m), and distribution indices (Deb 2001) for SBX crossover (η_c) and polynomial mutation (η_m) are collected. To find a well-spread and well-converged set of solutions, a rigorous experimentation and tuning of parameters need to be exercised. After getting the Pareto-optimal fronts in terms of membership values of the objective functions, we are able to find the Pareto-optimal fronts in terms of the objective values.

Step 5 Find the best compromise solution.

Fuzzy ranking method (Pandiarajan and Babulal 2014) ranks the multiple solutions as per their degree of satisfaction. The best compromise solution achieves the maximum satisfaction level as follows:

$$\mu_{\text{best}} = \max_P \left[\min\left\{\mu_{\widetilde{R}_S}, \mu_{\widetilde{C}_S}\right\}\right], \tag{21}$$

where P is the number of obtained Pareto-optimal solutions.

Step 6 Rank the DMUs obtained by the various membership functions.

In this step, DEA is implemented to rank the efficiency of the optimal results given by the various membership functions in the form of DMUs. The CCR model (Charnes et al. 1978) is applied for efficiency and then the super-efficiency model (Noura et al. 2011) for resolving the issue of more than one DMUs having equal efficiencies. There are four DMUs as follows:

$$\text{DMU}_1 \text{ Linear: } \mu_{\text{best}}^{\text{L}} = \max_P \left[\min\left\{\mu_{\tilde{R}_S}, \mu_{\tilde{C}_S}\right\}\right]$$
$$\text{DMU}_2 \text{ Quadratic: } \mu_{\text{best}}^{\text{Q}} = \max_P \left[\min\left\{\mu_{\tilde{R}_S}, \mu_{\tilde{C}_S}\right\}\right]$$
$$\text{DMU}_3 \text{ Parabolic: } \mu_{\text{best}}^{\text{P}} = \max_P \left[\min\left\{\mu_{\tilde{R}_S}, \mu_{\tilde{C}_S}\right\}\right]$$
$$\text{DMU}_4 \text{ Hyperbolic: } \mu_{\text{best}}^{\text{H}} = \max_P \left[\min\left\{\mu_{\tilde{R}_S}, \mu_{\tilde{C}_S}\right\}\right].$$

An Illustrative Example

Let us consider a four-stage over-speed protection system (see Fig. 6) for a gas turbine (Dhingra 1992). The over-speed detection is continuously provided by the electrical and mechanical systems. When an over-speed occurs, it is necessary to cut off the fuel supply by closing the four control valves (V1–V4). The control system is modeled as a four-stage series system. All components have a constant failure rate in the system. The objective is to determine the optimal design variables r_j (component reliability) and n_j (no. of the redundant components) at stage j, such that maximization of the system reliability (R_S) and minimization of the system cost (C_S) are achieved simultaneously. In addition, several design constraints such as minimum requirements for reliability of the system, the overall cost of the system, the total permissible volume of the system, and maximum allowable weight of the system are considered in this example. The mathematical formulation of the over-speed protection system is given as follows:

$$\text{Max. } R_S(r,n) = \prod_{j=1}^{4}\left[1-\left(1-r_j\right)^{n_j}\right] \tag{22}$$

$$\text{Min. } C_S(r,n) = \sum_{j=1}^{4} c_j\left(r_j\right)\left[n_j + \exp\left(n_j/4\right)\right] \tag{23}$$

$$\text{subject to } V_S = \sum_{j=1}^{4} v_j n_j^2 \le V \tag{24}$$

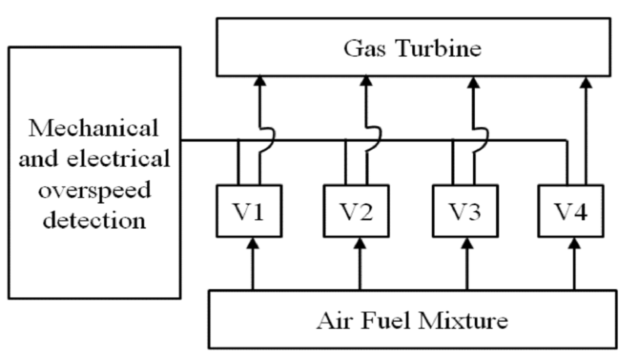

Fig. 6 A schematic diagram of the over-speed protection system

$$W_S = \sum_{j=1}^{4} w_j n_j \exp\left(n_j/4\right) \le W \tag{25}$$

$$\prod_{j=1}^{4}\left[1-\left(1-r_j\right)^{n_j}\right] \ge R \tag{26}$$

$$\sum_{j=1}^{4} c_j\left(r_j\right)\left[n_j + \exp\left(n_j/4\right)\right] \le C \tag{27}$$

$$0.5 \le r_j \le 1-10^{-6},\ 1 \le n_j \le 10,\ n_j \in \mathbb{Z}^+,\ j = 1,2,3,4. \tag{28}$$

The cost of the j th component $c_j\left(r_j\right)$ is assumed to be an increasing function of r_j (conversely, a decreasing function of the component failure rate) in the form:

$$c_j\left(r_j\right) = \gamma_j/\lambda_j^{\delta_j}, \tag{29}$$

where γ_j and δ_j are constants of characteristics factors for each component at the j th stage or subsystem. This formula can be found in Kumar et al. (2009).

Each component of the system has a constant failure rate λ_j that follows an exponential distribution. The reliability of each component is given by the following:

$$r_j(T) = \int_T^{\infty} \lambda_j e^{-\lambda_j T}\mathrm{d}T = e^{-\lambda_j T}. \tag{30}$$

From (29) and (30), we have

$$c_j\left(r_j\right) = \gamma_j\left[-T/\ln\left(r_j\right)\right]^{\delta_j}. \tag{31}$$

The parameters δ_j and γ_j give the physical features (shaping and scaling factor) of the cost–reliability curve of each component in the j th subsystem and the factor $\exp\left(n_j/4\right)$ accounts for the additional cost due to the interconnection between the parallel components (Kuo and Prasad 2000; Wang et al. 2009).

Computational Results and Discussion

This section describes and analyzes the results obtained by the proposed approach.

Parameter Settings

The overall process is implemented in the MATLAB (R2017a) on Intel(R) Core(TM) i3-2370M CPU @ 2.40 GHz with 4 GB RAM. The integer variables n_j are initially treated as real variables, but during the evaluation of the objective functions, the real values are transformed to the nearest integer values. MATLAB optimization tool-box function namely "fmincon" (Coleman et al. 1999) is used to determine the optimal values to each of the objective functions with given constraints. The design data for the given example are listed in Table 1. The optimal results for the single-objective optimization problem (SOOP) are given in Table 2 and compared it with other heuristic approaches. The parameters settings of the NSGA-II and other MOEAs such as PESA-II and SPEA2 are given as follows:

NSGA-II: $N=40$, $t_{\max}=100$, $p_c=0.9$, $p_m=0.1$, $\eta_c=10$, $\eta_m=100$.

PESA-II (Corne et al. 2001): hyper-grid size $=10\times 10$; $p_c=0.8$, $p_m=0.2$, $\eta_c=10$, $\eta_m=100$, archive size $=40$; $t_{\max}=100$.

SPEA2 (Zitzler et al. 2001): $N=40$; $p_c=0.9$, $p_m=0.1$; and $\eta_c=10$, $\eta_m=100$, archive size $=40$; $t_{\max}=100$.

Simulation Results

After setting the parameters, a simulation run for the given problem is shown to non-fuzzy approach fuzzy approach in Fig. 7. In Figs. 8 and 9, the simulation results of the proposed approach have been shown to both the membership

Table 1 Design data for the given example

Subsystem	$10^5\gamma_j$	δ_j	v_j	w_j	R	C	W	V	T
1	1.0	1.5	1	6	0.75	400	500	250	1000 h
2	2.3	1.5	2	6					
3	0.3	1.5	3	8					
4	2.3	1.5	2	7					

Table 2 Optimum solutions for the SOOP

	MATLAB optimization function "fmincon" (Coleman et al. 1999)		GA (Goldberg 1989)		PSO (Kennedy and Eberhart 1995)		Dhingra (1992)	
	Design variables	Attributes	Design variables	Attributes	Design variables	Attributes	Design variables	Attributes
Max R_S	(0.91,403, 5)	$R_S=0.99994$	(0.92472, 5)	$R_S=0.99810$	(0.91513, 6)	$R_S=0.9996$	(0.81604, 6)	$R_S=0.99961$
	(0.86723, 6)	$C_S=399.52$	(0.87732, 3)	$C_S=396.95$	(0.87883, 4)	$C_S=393.92$	(0.80309, 6)	$C_S=399.94$
	(0.94282, 3)	$W_S=439.02$	(0.83430, 6)	$W_S=434.05$	(0.94312, 5)	$W_S=442.30$	(0.98364, 3)	$W_S=495.65$
	(0.87349, 5)	$V_S=174$	(0.91449, 4)	$V_S=183$	(0.87959, 4)	$V_S=175$	(0.80373, 5)	$V_S=185$
Min C_S	(0.5, 4)	$R_S=0.7541$	(0.5, 4)	$R_S=0.7608$	(0.53447, 4)	$R_S=0.7615$	(0.5, 4)	$R_S=0.7604$
	(0.5, 4)	$C_S=20.30$	(0.5, 4)	$C_S=20.62$	(0.5, 4)	$C_S=20.71$	(0.5, 4)	$C_S=20.72$
	(0.54536, 5)	$W_S=314.55$	(0.53150, 5)	$W_S=314.55$	(0.51812, 5)	$W_S=314.55$	(0.59251, 5)	$W_S=314.55$
	(0.5, 3)	$V_S=141$	(0.51457, 3)	$V_S=141$	(0.5, 3)	$V_S=141$	(0.5, 3)	$V_S=141$

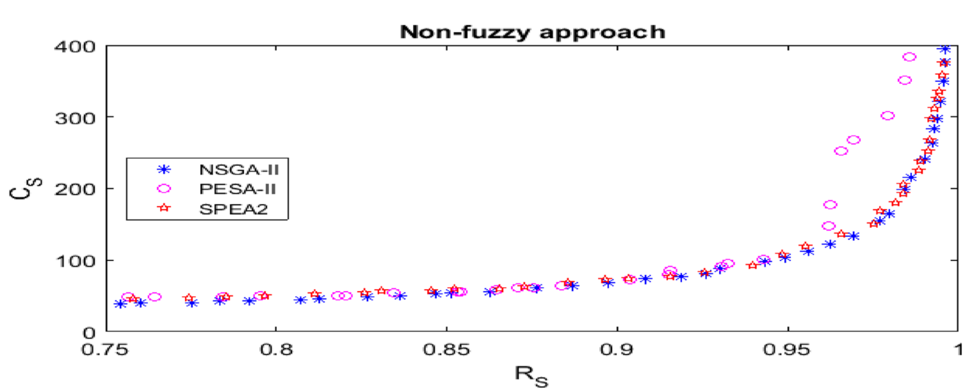

Fig. 7 The POFs based on the non-fuzzy approach in a single simulation run

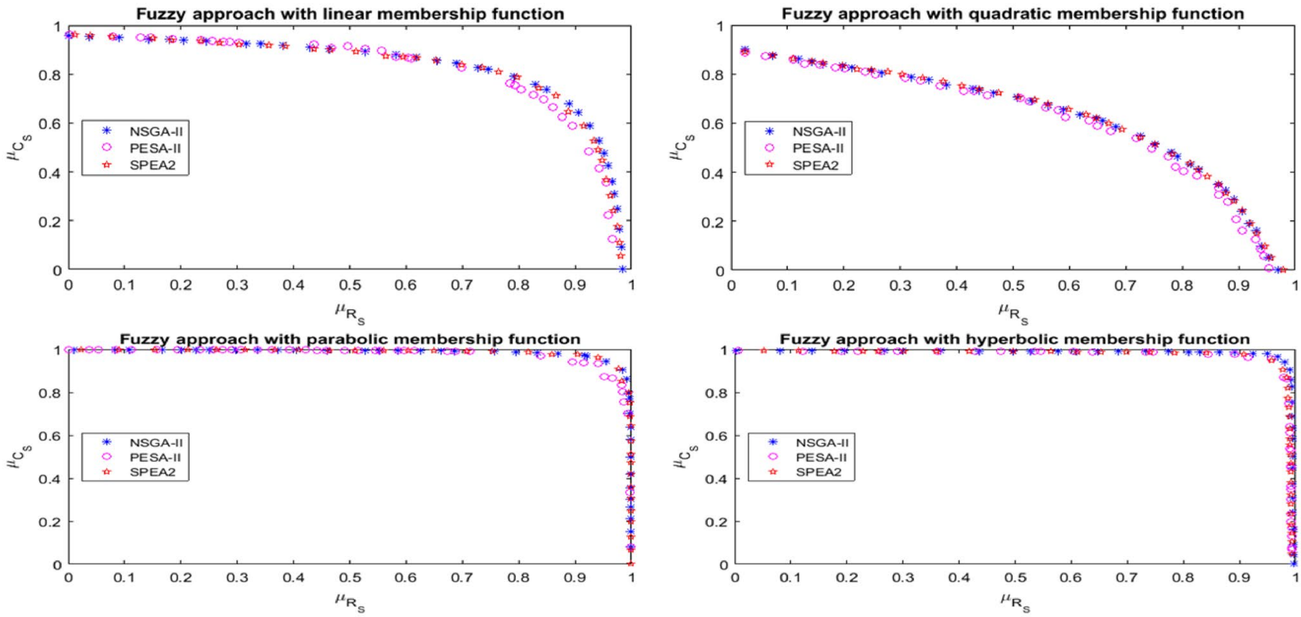

Fig. 8 The POFs in the membership grades space using the various membership functions

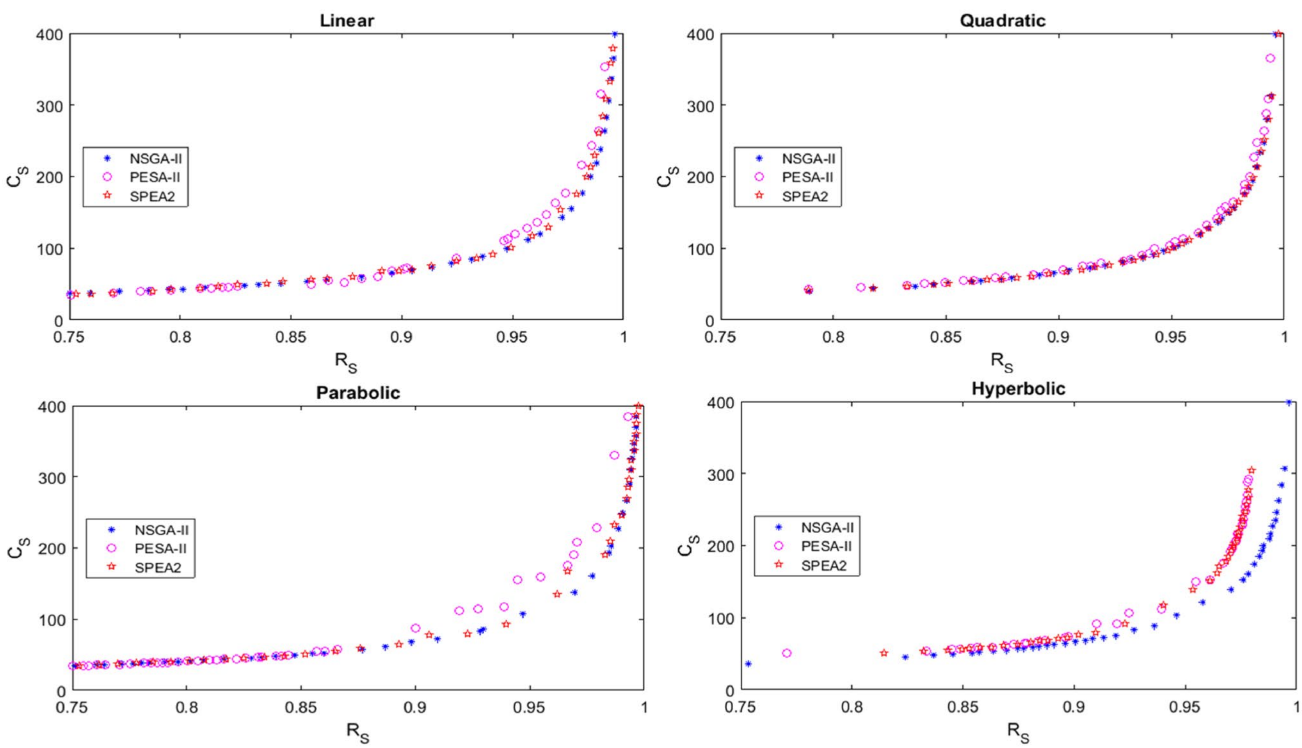

Fig. 9 The POFs in the objective space on the basis of the various membership functions

and objective spaces respectively. NSGA-II is shown comparatively with other elitist MOEAs such as PESA-II and SPEA2. Figure 10 shows a box-plot comparison of R_S and C_S between non-fuzzy and fuzzy approach.

The Best DMU

The best trade-off of optimal solutions for each of the membership functions is obtained by the fuzzy ranking method in Table 3. Figure 11 shows the optimal values obtained by the

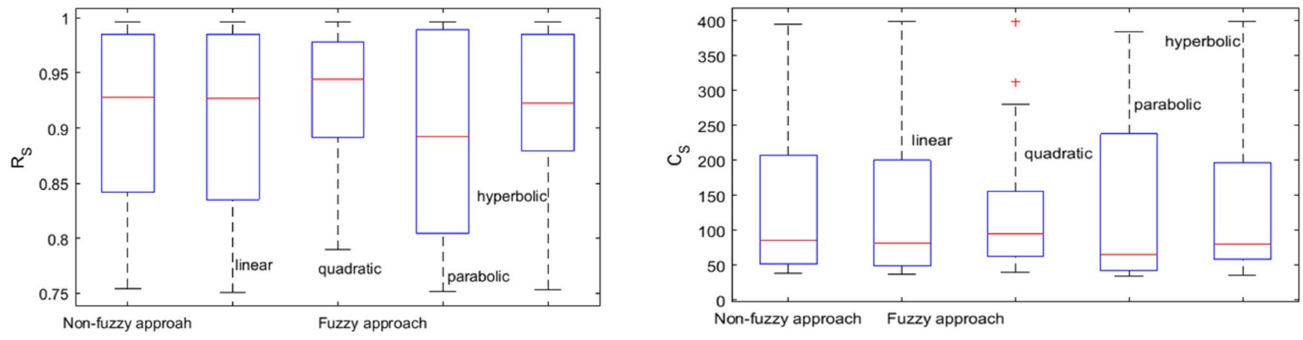

Fig. 10 Box-plot comparison between the fuzzy and non-fuzzy approach

Table 3 The best compromise solution for each membership function

	r_1	n_1	r_2	n_2	r_3	n_3	r_4	n_4	R_S	C_S	W_S	V_S	μ_{best}
Linear	0.81986	5	0.75042	3	0.67214	3	0.78987	4	0.947628	99.28	269.74	102	0.7907
Quadratic	0.78327	5	0.67480	3	0.67243	4	0.85639	3	0.951174	100.85	274.26	109	0.6207
Parabolic	0.71126	5	0.72219	4	0.64577	3	0.83937	4	0.947256	108.18	296.87	116	0.9461
Hyperbolic	0.82737	5	0.67367	4	0.65271	4	0.80692	4	0.972764	103.69	333.05	137	0.9666

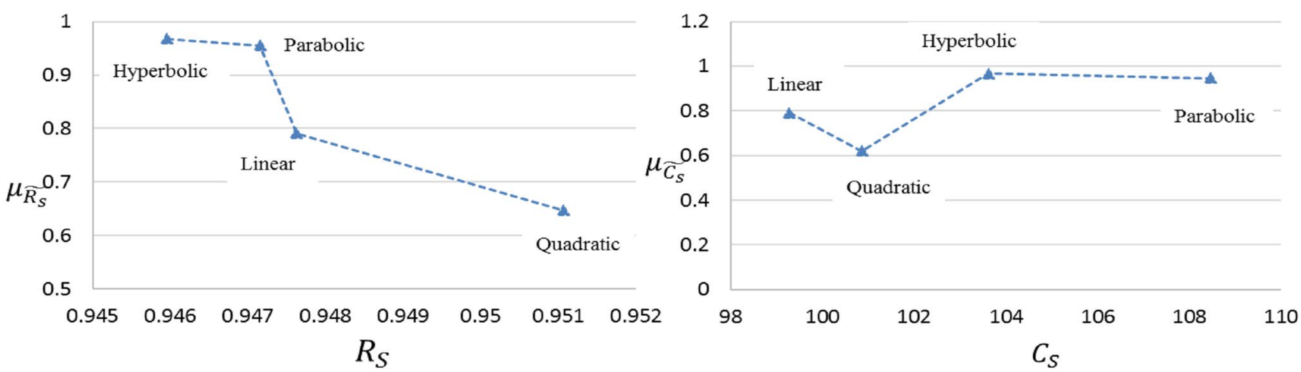

Fig. 11 The optimal values of R_S and C_S on the basis of the various membership functions

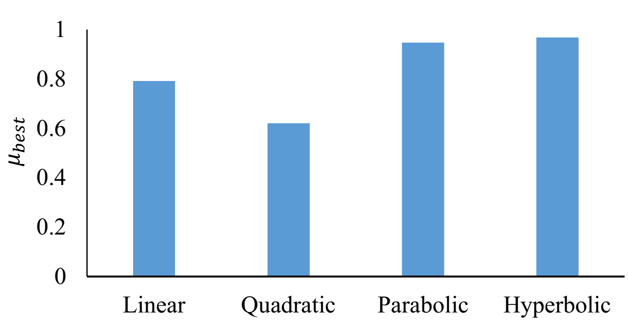

Fig. 12 Maximum satisfaction level achieved by the various membership functions

various membership functions in R_S and C_S. In Fig. 12, the maximum satisfaction levels achieved by the various membership functions are compared. After that, DEA is implemented to rank the DMUs obtained by the various membership functions. Table 4 gives the ranking to each DMU. It is based on the CCR model that is implemented in DEAP solver software (Coelli 1996). The efficiencies are obtained in this model as $\text{DMU}_1 = 1$, $\text{DMU}_2 = 0.988$, $\text{DMU}_3 = 1$, and $\text{DMU}_4 = 1$. To resolve it, super-efficiency model concept (Noura et al. 2011) is used, and finally, it is ranked as $\text{DMU}_4 > \text{DMU}_3 > \text{DMU}_1 > \text{DMU}_2$.

Table 4 Ranking from the DEA model

DMU	Linear (1)	Quadratic (2)	Parabolic (3)	Hyperbolic (4)
Reliability	0.947628	0.951174	0.947256	0.972764
Cost	99.28	100.85	108.18	103.69
Weight	269.74	274.26	296.87	333.05
Volume	102	109	116	137
Satisfaction level	0.7907	0.6207	0.9461	0.9666
Efficiency	1	0.988	1	1
Aspiration level	1.9977	–	2.1369	2.1667
Ranking	3	4	2	1

A Comparative Study with Existing Approach

The proposed approach is compared with other approaches applied to the same problem. Liu (2013) solved this problem by converting into a single-objective fuzzy non-linear programming problem. A heuristic method is developed to get a set of satisfactory solutions. To rank the satisfactory solutions, DEA model is used by considering criteria of reliability, cost, volume, and weight. However, an MOOP is preferred to obtain a set of optimal solutions popularly known as Pareto-optimal solutions and the best Pareto-optimal solution is then chosen by some higher level information involved in the problem. The proposed approach simultaneously optimizes the membership functions instead of objective functions and gets multiple solutions in a single simulation run. Figure 13 shows the box-plot comparison of the proposed approach and Liu (2013) by taking the same range of limits in the linear membership function. The proposed approach gives well-distributive solutions in the same conditions.

Garg et al. (2014a) solved this problem by developing a model in fuzzy environment with the assumption that the reliability of each component is a triangular fuzzy number. To solve the problem, the developed fuzzy model is converted into crisp model using expected values of fuzzy numbers and taking into account the preference of decision-maker regarding cost and reliability goals. However, the proposed approach does not require any kind of aggregate operators and various membership functions give the desirability functions to the decision-maker in choosing the best compromise solution according to his/her own's interest. Figure 14 shows the box-plot comparison between linear and non-linear (sigmoidal shape) membership functions. The proposed approach uses MOEA technique rather than heuristic approaches such as GA (Goldberg 1989), and PSO (Kennedy and Eberhart 1995), so it covers a larger search space where multiple solutions are generated in a single simulation run. It gives more information about the characteristics of the solutions.

Wang et al. (2009) solved the MORRAP using NSGA-II in the crisp environment. In a similar environment, Damghani et al. (2013) used multi-objective particle swarm optimization (MOPSO). Both these approaches do not reflect the real-life situations where uncertainty is an inherent character in the system design problem. Recently, Muhuri et al. (2018) solved the MORRAP with interval type-2 fuzzy set

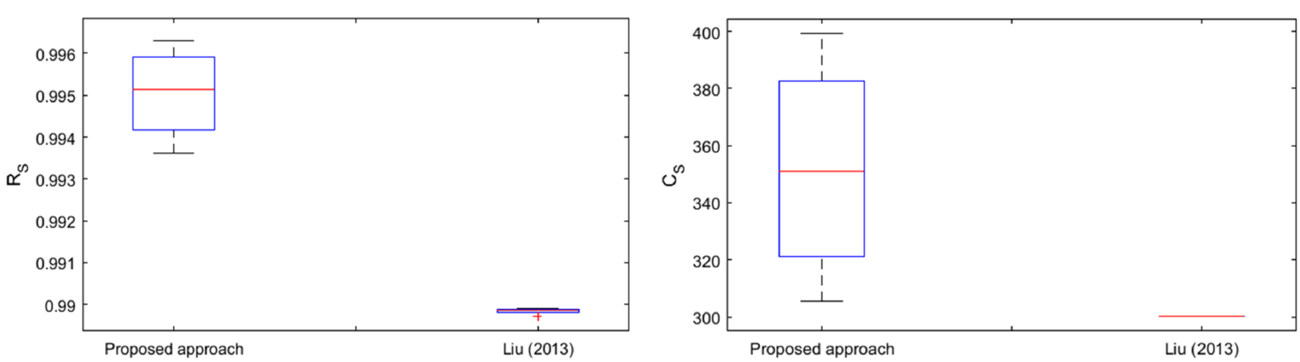

Fig. 13 The box-plot comparison with Liu (2013)

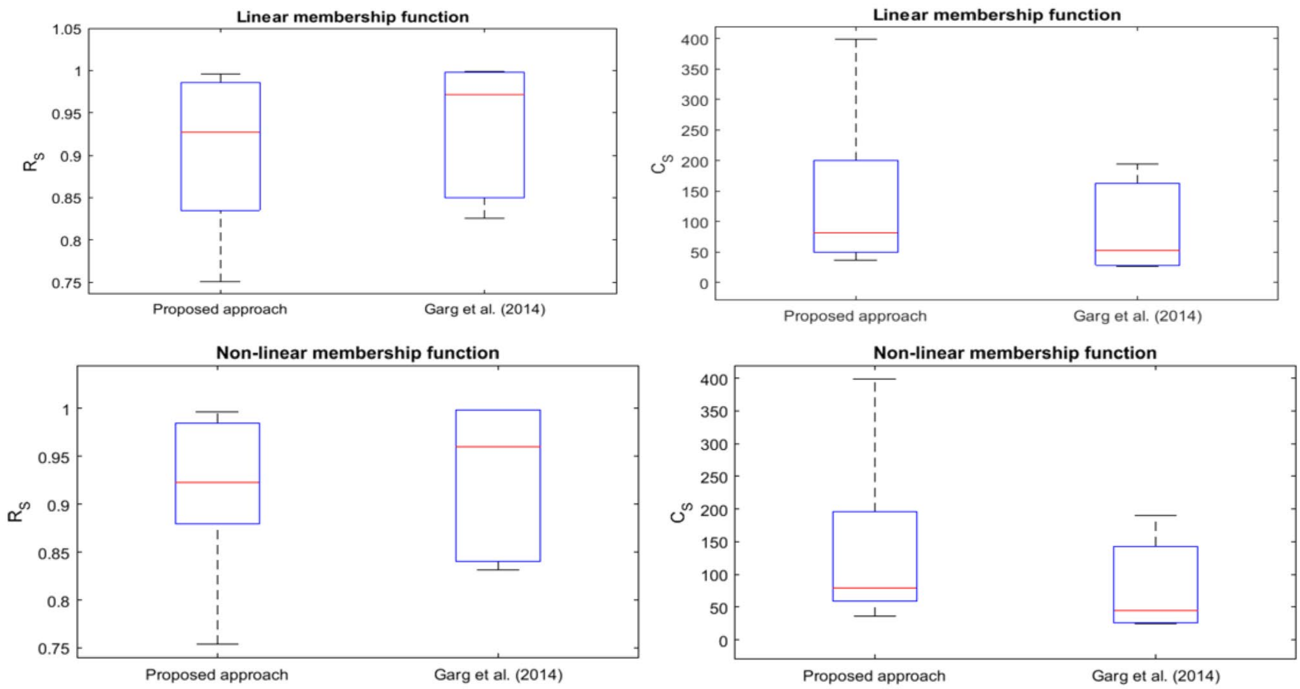

Fig. 14 The box-plot comparison with Garg et al. (2014a)

which considers higher order uncertainties in the component parameters and addressed it using NSGA-II. The comparison between the Muhuri et al. (2018) and the proposed approach is given as follows.

Muhuri et al. (2018)	Proposed approach
This type of modeling is suitable for those situations where the end points of the component or objective are ambiguous	The present approach finds the end points of each objective first and then models the given problem as a fuzzy MOOP
It does not show the POF in the membership space. In fact, this type of modeling creates difficulty to show the POF in the membership space	The present approach finds the POF in the objective space as well as its membership space
Only linear membership function is considered	However, a real-world situation demands models to be more flexible and adaptable to the human decision-making process as well as some kind of empirical justification or assumption. Keeping these views in mind, various membership functions are considered
It does not give the best solution	It finds the best "trade-off" or compromise solution

Conclusions

In this piece of work, MOEA approach is used to solve MORRAP in a fuzzy environment where the goals of the objectives are specified by various membership functions such as linear, quadratic, parabolic, and hyperbolic. A numerical example of over-speed protection system with conflicting objectives such as maximization of system reliability and minimization of system cost is considered simultaneously under several design constraints such as minimum requirements for reliability of the system, the overall cost of the system, the total permissible volume of the system, and maximum allowable weight of the system. Fuzzy ranking method is used to obtain the best compromise solution. Finally, DEA model is used to rank the results for the various membership functions in the form of DMUs.

The experiments performed by the proposed approach are concluded as follows:

- MOEA technique namely NSGA-II is shown with other MOEAs which are capable in finding a complete picture of Pareto-optimal solutions in a single simulation run, where a decision-maker gets more information such as non-dominated and their characteristics.

- The proposed approach is free from any kind of aggregation. It deals with purely multi-objective manner.
- The trade-offs between the system reliability and system cost are shown in both membership space and objective space.
- Simulation results of fuzzy and non-fuzzy approach are comparatively analyzed. It shows a fuzzy approach gives the flexibility to the decision-maker in setting the desired goals.
- The best compromise solutions are obtained for the various membership functions.
- DEA ranks the DMUs as Hyperbolic > Parabolic > Linear > Quadratic.
- Hyperbolic membership function gives the best result to the decision-maker.
- The proposed approach gives flexibility to the decision-maker in choosing the membership function to his/her's own interests.

Acknowledgements The authors gratefully acknowledge the Ministry of Human Resource and Development (MHRD), Govt. of India, New Delhi for providing the financial grant. The authors also thank to anonymous reviewers for giving valuable suggestions.

References

Charnes A, Cooper WW, Rhodes E (1978) Measuring the efficiency of decision making units. Eur J Oper Res 2(6):429–444

Chern MS (1992) On the computational complexity of reliability redundancy allocation in a series system. Oper Res Lett 11(5):309–315

Coelli T (1996) A guide to DEAP version 2.1: a data envelopment analysis (computer) program. Centre for efficiency and productivity analysis (CEPA)-working papers no. 8/96, Dept. of Econometrics, University of New England Armidale, NSW, 2351, Australia. http://www.owlnet.rice.edu/~econ380/DEAP.PDF

Coello CAC, Lamont GB, Van Veldhuizen DA (2007) Evolutionary algorithms for solving multi-objective problems. Springer-Verlag, New York

Coleman T, Branch MA, Grace A (1999) Optimization toolbox for use with MATLAB: User's guide. The Math Works, Inc. https://instruct.uwo.ca/engin-sc/391b/downloads/optim_tb.pdf

Corne DW, Jerram NR, Knowles JD, Martin J (2001) PESA-II : Region-based selection in evolutionary multiobjective optimization. In: Proceeding GECCO'01 proceedings of the 3rd annual conference on genetic and evolutionary computation, pp 283–290

Damghani KK, Abtahi AR, Tavana M (2013) A new multiobjective particle swarm optimization method for solving reliability redundancy allocation problems. Rel Eng Syst Saf 111:58–75

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, New York

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

Dhingra AK (1992) Optimal apportionment of reliability and redundancy in series systems under multiple objectives. IEEE Trans Reliab 41(4):576–582

Dhingra AK, Moskowitz H (1991) Application of fuzzy theories to multiple objective decision making in system design. Eur J Oper Res 53:348–361. https://doi.org/10.1016/0377-2217(91)90068-7

Fonseca CM, Fleming PJ (1993) Genetic algorithms for multiobjective optimization: formulation, discussion, and generalization, genetic algorithms. In: Proceedings of the ICGA-93: Fifth International Conference on Genetic Algorithms, 17-22 July 1993, San Mateo, pp 416–423. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.48.9077&rep=rep1&type=pdf

Garg H, Sharma SP (2013) Multi-objective reliability–redundancy allocation problem using particle swarm optimization. Comput Ind Eng 64:247–255. https://doi.org/10.1016/j.cie.2012.09.015

Garg H, Rani M, Sharma SP, Vishwakarma Y (2014a) Bi-objective optimization of the reliability–redundancy allocation problem for series-parallel system. J Manuf Syst 33(3):335–347

Garg H, Rani M, Sharma SP (2014b) Intuitionistic fuzzy optimization technique for solving multi-objective reliability optimization problems in interval environment. Expert Syst Appl 41:3157–3167

Goldberg DE (1989) Genetic algorithms for search, optimization, and machine learning. Addison-Wesley, Reading

Horn J, Nafpliotis N, Goldberg DE (1994) A niched Pareto genetic algorithm for multiobjective optimization. Proc First IEEE Conf Evol Comput IEEE World Congr Comput Intell 1:82–87. https://doi.org/10.1109/ICEC.1994.350037

Huang HZ, Qu J, Zuo MJ (2006) A new method of system reliability multi-objective optimization using genetic algorithms. RAMS' 06 Annual Reliability Maintainability Symposium IEEE 278–283. https://doi.org/10.1109/rams.2006.1677387

Huang HZ, Jian Q, Ming JZ (2009) Genetic-algorithm-based optimal apportionment of reliability and redundancy under multiple objectives. IIE Trans 41(4):287–298. https://doi.org/10.1080/07408170802322994

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95 - International conference on neural networks, Perth, Australia, 27 November–1 December 1995. IEEE, pp 1942–1948

Kim H, Kim P (2017) Reliability–redundancy allocation problem considering optimal redundancy strategy using parallel genetic algorithm. Rel Eng Syst Saf 159:153–160

Knowles J, Corne D (1999) The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation Proceedings of the 1999 congress on evolutionary computation. IEEE Press, Piscataway, pp 98–105. https://doi.org/10.1109/cec.1999.781913

Konak A, Coit DW, Smith AE (2006) Multi-objective optimization using genetic algorithms: a tutorial. Rel Eng Syst Saf 91:992–1007

Kumar H, Yadav SP (2017) NSGA-II based fuzzy multi-objective reliability analysis. Int J Syst Assur Eng Manag 8:817–825. https://doi.org/10.1007/s13198-017-0672-y

Kumar H, Yadav SP (2019) NSGA-II based decision-making in fuzzy multi-objective optimization of system reliability. In: Deep K, Jain M, Salhi S (eds) Decision science in action. Asset analytics (performance and safety management). Springer, Singapore, pp 105–117.

Kumar R, Izui K, Yoshimura M, Nishiwaki S (2009) Optimal multilevel redundancy allocation in series and series-parallel systems. Comput Ind Eng 57:169–180. https://doi.org/10.1016/j.cie.2008.11.008

Kuo W, Prasad VR (2000) An annotated overview of system-reliability optimization. IEEE Trans Reliab 49(2):176–187. https://doi.org/10.1109/24.877336

Laumanns M, Thiele L, Deb K (2002) Zitzler E (2002) Combining convergence and diversity in evolutionary multiobjective optimization. Evol Comput 10(3):263–282

Liu CM (2013) Fuzzy programming and data envelopment analysis for improving redundancy-reliability allocation problems. Int J Phys Sci 8(15):635–646

Muhuri PK, Ashraf Z, Lohani QMD (2018) Multi-objective reliability–redundancy allocation problem with interval type-2 fuzzy uncertainty. IEEE Trans Fuzzy Syst 26(3):1339–1355. https://doi.org/10.1109/TFUZZ.2017.2722422

Noura AA, Lotfi FH, Jahanshahloo GR, Rashidi SF (2011) Super-efficiency in DEA by effectiveness of each unit in society. Appl Math Lett 24:623–626

Pandiarajan K, Babulal CK (2014) Fuzzy ranking based non-dominated sorting genetic algorithm-II for network overload alleviation. Arch Electr Eng 63:367–384. https://doi.org/10.2478/aee-2014-0027

Safari J (2012) Multi-objective reliability optimization of series-parallel systems with a choice of redundancy strategies. Reliab Eng Syst Saf 108:10–20. https://doi.org/10.1016/j.ress.2012.06.001

Sakawa M (1978) Multiobjective optimization by the surrogate worth trade-off method. IEEE Trans Reliab R-27:311–314

Salazar D, Rocco CM, Galván BJ (2006) Optimization of constrained multiple-objective reliability problems using evolutionary algorithms. Reliab Eng Syst Saf 91:1057–1070. https://doi.org/10.1016/j.ress.2005.11.040

Sharifi M, Guilani PP, Shahriari M (2016) Using NSGA-II algorithm for a three objective redundancy allocation problem with k-out-of-n sub-systems. J Optim Indl Eng 19:87–95

Sheikhalishahi M, Ebrahimipour V, Shiri H, Zaman H, Jeihoonian M (2013) A hybrid GA–PSO approach for reliability optimization in redundancy allocation problem. Int J Adv Manuf Technol 68:317–338. https://doi.org/10.1007/s00170-013-4730-6

Singh SK, Yadav SP (2017) Intuitionistic fuzzy multi-objective linear programming problem with various membership functions. Ann Oper Res. https://doi.org/10.1007/s10479-017-2551-y

Srinivas N, Deb K (1994) Multiobjective optimization using nondominated sorting in genetic algorithms. Evol Comput 2(3):221–248

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. https://doi.org/10.1016/j.ress.2006.04.014

Tavakkoli MR, Safari J, Sassani F (2008) Reliability optimization of series-parallel systems with a choice of redundancy strategies using a genetic algorithm. Reliab Eng Syst Saf 93(4):550–556

Tillman FA, Hwang CL, Kuo W (1977) Determining component reliability and redundancy for optimum system reliability. IEEE Trans Reliab R-26:162–165

Veldhuizen DV, Lamont GB (2000) Multiobjective optimization with messy genetic algorithms. In: Proc 2000 ACM Symp Appl Comput, pp 470–476

Wagner M, Neumann F (2013) A fast approximation-guided evolutionary multi-objective algorithm. In: Proc of the 15th Ann Conf on Genetic and Evolutionary Computation, pp 687–694

Wang Z, Chen T, Tang K, Yao X (2009) A multi-objective approach to redundancy allocation problem in parallel-series systems. In: 2009 IEEE Congress on Evolutionary Computation, pp 582–589. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4982998

Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11:712–731

Zhang E, Wu Y, Chen Q (2014) A practical approach for solving multi-objective reliability redundancy allocation problems using extended bare-bones particle swarm optimization. Rel Eng Syst Saf 127:65–76

Zhou A, Qu BY, Li H, Zhao SZ, Suganthan PN, Zhang QZ (2011) Multiobjective evolutionary algorithms: a survey of the state of the art. Swarm Evolut Comput 1(1):32–49

Zimmermann HJ (1996) Fuzzy set theory and its applications. Kluwer Academic Publishers, Boston. ISBN 0-7923-9624-3

Zitzler E, Thiele L (1998) An evolutionary algorithm for multiobjective optimization: the strength Pareto approach. Tech Rep No. 43. Comput Eng Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Zurich. https://doi.org/10.3929/ethz-a-004288833

Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: empirical results. Evolut Comput 8(2):173–195

Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength Pareto evolutionary algorithm. Tech Rep No. 35. Comput Eng Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, pp 1–21. https://doi.org/10.3929/ethz-a-004284029