Research Article

Quality and Reliability Engineering International

(wileyonlinelibrary.com) DOI: 10.1002/qre.1545

Published online 3 July 2013 in Wiley Online Library

A Decision Support System for Solving Multi-Objective Redundancy Allocation Problems

Kaveh Khalili-Damghani,[a] Amir-Reza Abtahi[b] and Madjid Tavana[c*†]

The Redundancy Allocation Problem (RAP) is a reliability optimization problem in designing series-parallel systems. The reliability optimization process is intended to select multiple components with appropriate levels of redundancy by maximizing the system reliability under some predefined constraints. Several methods have been proposed to solve the RAPs. However, most of these methods often treat RAP as a single objective problem of maximizing the system reliability (or minimizing the system design cost). We propose a Decision Support System for solving Multi-Objective RAPs. Initially, we use the Technique for Order Performance by Similarity to Ideal Solution method to reduce the multiple objective dimensions of the problem. We then propose an efficient ε-constraint method to generate non-dominated solutions on the Pareto front. Finally, we use a Data Envelopment Analysis model to prune the non-dominated solutions. A benchmark case is presented to assess the performance of the proposed system, demonstrate the applicability of the proposed framework, and exhibit the efficacy of the procedures and algorithms. Copyright © 2013 John Wiley & Sons, Ltd.

Keywords: redundancy allocation problem; multiple objective; decision support system; reliability; ε-constraint method; TOPSIS; data envelopment analysis

1. Introduction

The Redundancy Allocation Problem (RAP) is a well-known and complex reliability design problem that was first introduced by Misra and Ljubojevic.[1] The RAP formulations are characterized by a large combinatorial search space with multiple constraints. The objectives in the RAPs are generally either maximizing the system reliability or minimizing the system cost given a set of system-level reliability, cost, and weight constraints. Solutions to the RAPs are intended to identify the optimal combination of the components and the redundancy levels in the system. However, the RAPs are NP-hard problems[2] because: (i) there are numerous possible combinations for each component; and (ii) for the system to achieve the required reliability level, there is a need to implement redundancy where multiple components are available to continue performing the required tasks in the event of a component failure.

Mathematical programming techniques have been widely used to solve different variations of the RAPs. However, it is often necessary to artificially limit the search space to solutions where only one component can be selected for each sub-system and only the selected component can be used to provide redundancy. However, these restrictions are often unrealistic in the real-life problems where different components, performing the same task, can be used within a system to provide high reliability. A given component in the RAP can have a binary state or a multi-state.[3] In the binary state, a proper structure can be designed by increasing the reliability of the components or supplying parallel redundant components at some stages. These options can be modeled with a mixed integer non-linear programming model. In some other cases, a given component may encounter more than two different levels, ranging from perfectly working to completely failed states. A well-known performance measure for the multi-state system is system utility.[4] In the case of a multi-state system, the status of the system is represented with state distribution. The system utility of a multi-state series-parallel system can also be improved by providing redundancy at each stage, or improving the component state distribution. Series-parallel systems indicate sub-systems in which several components are connected in series, and then in parallel, or sub-systems that several components are connected in parallel, and then in series. A series-parallel system can be improved by four methods[5]: (1) use more reliable components; (2) increase redundant components in parallel; (3) utilize both more reliable components and increase redundant components in parallel; and (4) enable repeatedly the allocation of entire system framework.

[a]Department of Industrial Engineering, South-Tehran Branch, Islamic Azad University, Tehran, Iran
[b]Department of Knowledge Engineering and Decision Sciences, Faculty of Economics Insinuations Management, University of Economic Sciences, Tehran, Iran
[c]Business Systems and Analytics, Lindback Distinguished Chair of Information Systems and Decision Sciences, La Salle University, Philadelphia, PA, 19141, U.S.A.
*Correspondence to: Madjid Tavana, Business Systems and Analytics, Lindback Distinguished Chair of Information Systems and Decision Sciences, La Salle University, Philadelphia, PA, 19141, U.S.A..
†E-mail: tavana@lasalle.edu, URL: http://tavana.us

Different heuristic and meta-heuristic methods have been proposed to solve the RAPs.[5–14] The multi-objective RAPs (MORAPs) are often used in real-life system engineering optimization applications. Konak *et al.*[15] presented an overview and tutorial describing genetic algorithms developed specifically for RAPs with multiple objectives. Different applications of the meta-heuristic RAPs can be found in Taboada *et al.*,[16] Li *et al.*,[17] and Zio and Bazzo.[18] Gen and Yun[19] surveyed the genetic algorithm methods used to solve various reliability optimization problems, including reliability optimization of redundant system, reliability optimization with alternative design, reliability optimization with time-dependent reliability, reliability optimization with interval coefficients, bi-criteria reliability optimization, and reliability optimization with fuzzy goals. Khalili-Damghani and Amiri[20] proposed an efficient version of the ε-constraint method to solve the multi-objective version of the RAPs. They compared the performance of the efficient ε-constraint method with another heuristic procedure called multi-start partial bound enumeration algorithm. Khalili-Damghani *et al.*[21] recently proposed a self-adaptive dynamic multi-objective particle swarm optimization method for solving the RAPs. They also compared the performance of several approaches with benchmark cases.

The complexity of the reliability design problems, especially the multi-objective versions, has prompted the researchers to propose several heuristic and meta-heuristic procedures. Although the meta-heuristic methods have been proven useful in solving different operations research and engineering problems, the existence of commercial operations research software capable of modeling and solving complex real-life problems has had a negative effect on the use of time-consuming procedures for designing customized meta-heuristic methods. The meta-heuristic methods are also often developed to solve single-objective non-linear real-life problems, which are NP-hard in nature; or multi-objective problems, for which one can find non-dominated solutions on the Pareto front.

In many real-life multi-objective problems, Decision Makers (DMs) are interested in achieving high-quality solutions which are as close as possible to a global ideal point and as far as possible from a global nadir point, simultaneously. This is the case in many real-life operations research and engineering problems such as pump facilities, turbines, crude oil extraction, portfolio selection, banking, agriculture, transportation, military, and training. In addition, some real-life problems may also require compromised solutions with high quality and low risk, simultaneously.

Although generating a set of non-dominated solutions with the aforementioned properties (i.e. high quality and low risk, simultaneously) on the Pareto front might be interesting to study, they are not practical in most real-life engineering design problems since the DMs can hardly make a decision when he/she is required to choose from several non-dominated solutions. This is more problematic when the DM has no direct knowledge about the different non-dominated solutions or has no prior/posterior articulation of the preferences on the priority of the objective functions. Simply providing the DMs with a set of non-dominated solutions is not productive and often confuses them. Therefore, a posterior investigation is required after the non-dominated solutions are generated. This posterior investigation could be in the form of a systematic pruning method for eliminating the less attractive non-dominated solutions thus making the decision-making process more practical.

In this paper, we propose a Decision Support System (DSS) for solving Multiple Objective Decision-Making (MODM) problems by generating a set of non-dominated high-quality and low-risk solutions and then applying a subsequent pruning procedure. The proposed DSS is composed of three distinct but related modules. The first module reduces the MODM problem under consideration to an efficient bi-objective problem through the Technique for Order Performance by Similarity to Ideal Solution (TOPSIS) method. The TOPSIS method ensures that all the objectives in the original MODM problem are used to generate high-quality low-risk solutions. The second module generates a set of non-dominated solutions for the bi-objective problem through an efficient ε-constraint method. The proposed efficient ε-constraint method significantly reduces the computational efforts needed to solve the bi-objective problem. The third module prunes the non-dominated solutions through Data Envelopment Analysis (DEA). The proposed DEA model provides the DMs with a manageable number of solutions for selection consideration.

Our method uniquely combines concepts from MODM. More specifically, we propose a new approach for solving MORAPs by using the TOPSIS method to reduce the multiple objective dimensions of the problem into bi-objective dimensions, an efficient ε-constraint method to guarantee that the non-dominated solutions are generated on the Pareto front with minimum computation requirements, and DEA to prune the non-dominated solutions and help the DM select the most efficient non-dominated solution. The multiple objective dimension reduction, the non-dominated solution generation, and the non-dominated solution pruning considerations distinguish our approach from others in the literature.

Solving MODM problems is not searching for some kind of optimal solution, but rather helping the DMs master the (often complex) data involved in their problem and advance toward a solution.[22] The method proposed in this study was developed after attempting to solve a real-life MORAP. As often happens in applied mathematics, the development of MODM models is dictated by real-life problems. It is therefore not surprising that methods have appeared in a rather diffuse way, without any clear general methodology or basic theory.[23]

We use the proposed DSS to solve a binary-state series-parallel MORAP. We compare the proposed DSS and its efficient ε-constraint component to a benchmark MORAP case to determine the performance of the proposed system, demonstrate the applicability of the proposed framework, and exhibit the efficacy of the procedures and algorithms. We use statistical analysis and show that our proposed system is superior to the *AUGMECON* method proposed by Mavrotas.[24]

The remainder of the paper is arranged as follows. The MODM concepts, including some preliminary definitions and applications of the TOPSIS method and the classical ε-constraint method are presented in Section 2. The proposed DSS framework including the TOPSIS, the efficient ε-constraint, and the DEA modules is presented in Section 3. In Section 4, we discuss the application of the proposed DSS in a binary-state MORAP. The experimental results of the application are presented in Section 5. In Section 6, we discuss our conclusions and future research directions.

2. Multi-objective decision making

Most real-life decision problems involve multiple and conflicting objectives, sometimes subject to certain constraints. MODM is commonly used to solve these problems characterized by multiple and conflicting objective functions such as maximizing performance while simultaneously minimizing fuel consumption of a vehicle over a feasible set of decisions. The goal is to optimize the objective functions, and the DMs choose a solution among a set of efficient solutions since MODM problems rarely have a unique solution.[25]

One of the most challenging problems in MODM applications is related to the identification or approximation of a family of points known as the Pareto-optimal set.[26] Pareto optimality is a measure of efficiency in multi-objective optimization. A large number of methods have been proposed in the literature to generate the Pareto optimal set. These methods vary from simple approaches, requiring very little information, to the methods based on mathematical programming techniques, requiring extensive information on each objective and the preferences of the DMs. There is no 'one size fits all' methodology for MODM problems. A method that works well in theory can fail in practice, and one that works well on some problems may not be suitable for others. A wide range of procedures and methods have been proposed in the literature to solve the MODM problems.[27–30]

Formally, a MODM model considers a vector of decision variables, objective functions, and constraints.[31,32] The DMs attempt to optimize the objective functions. In the absence of a unique optimal solution, the DMs often choose a solution from a set of efficient solutions. Generally, the MODM problem with maximum objective functions can be formulated as (1).

$$(MODM) \quad \begin{cases} \max & f(x) \\ s.t. & x \in S = \{x \in R^n | g(x) \leq b, x \geq 0\} \end{cases} \tag{1}$$

where, $f(x)$ represents the k conflicting objective functions, $g(x) \leq b$ represents m constraints, S is the feasible solution space, and x is an n-vector of the decision variables, $x \in R^n$.

2.1. Dimension reduction with the TOPSIS method

In real-world MODM problems, a serious challenge will arise when the set of efficient solutions is large, and it is difficult to choose a solution. Dimension reduction can help alleviate this problem in MODM. However, one serious downside of the dimension reduction strategy is information loss. The dimension reduction process should involve a systematic selection and elimination process that minimizes the perturbations in the solution space. At the same time, the information loss caused by the dimension reduction strategy should be as little as possible.

The TOPSIS method was initially introduced by Hwang and Yoon.[33] TOPSIS ranks the alternatives according to an algorithmic procedure. Alternatives are sorted in a decreasing order of the Closeness Coefficient (CC) which is calculated with respect to the distance of a given alternative from both positive and negative ideal solutions (NISs) concurrently. Lai *et al.*[34] used the compromise property of the TOPSIS procedure to generate solutions which had the shortest distance from the positive ideal solution (PIS) as well as the longest distance from the NIS. They reduced a k-dimensional objective space to a two-dimensional objective space by a first-order compromise procedure. Although the dominance structure of the solution space may be perturbed through any dimension reduction procedure (including TOPSIS), the TOPSIS method has several interesting and useful properties:

- The compromise property of TOPSIS is effective in generating desired solutions which are far from the NIS and near the PIS, simultaneously. This property is an important advantage in real-world MODM problems where DMs are interested in seeking high-quality solutions and avoiding high-risk solutions, simultaneously.
- All objective functions in TOPSIS effectively influence the generation of the resulting bi-objective problem. No objective is completely omitted, and the structure of the solution space may change as little as possible.
- The relative importance of the objectives in the original MODM problem can easily be controlled through the importance weights and the order of compromise operation which are determined by the DMs.

The TOPSIS method has been widely used for failure mode and effects analysis,[35–37] crisis management,[38,39] and quality control.[40–42]

2.2. Classical ε-constraint method

There are also methods that produce the entire efficient set for a special kind of MODM problems (mostly but not only linear problems). These methods can provide a representative subset of the Pareto set which in most cases is adequate. The ε-constraints method is a one of such techniques applied to MODM proposed by Chankong and Haimes.[43] In this method, the DM chooses one objective out of n to be optimized. The remaining objectives are constrained to be less than or equal to a given target value. In mathematical terms, the DM let $f_j(x), j \in \{1, \ldots, k\}$ be the objective function chosen to be optimized and we have the following problem:

$$\begin{aligned} & P(\varepsilon_j), j \in \{1, \ldots, k\}: \\ & \min\{f_j(x), j \in \{1, \ldots, k\}; f_i(x) \leq \varepsilon_i, \forall i \in \{1, \ldots, k\}, i \neq j; x \in S\}. \end{aligned} \tag{2}$$

where, S is the feasible solution space.

One advantage of the ε-constraint method is that it is able to achieve efficient points on a non-convex Pareto curve. Therefore, as proposed by Steuer,[44] the DM can vary the upper bounds ε_i to obtain the weak Pareto optima. Clearly, this is also a drawback of the ε-constraint method since the DM has to choose appropriate upper bounds for the ε_i values. Moreover, the ε-constraint method is less efficient as the number of the objective functions increases. Mavrotas[24] has proposed a novel version of the ε-constraint method (i.e. the augmented ε-constraint method – AUGMECON) that avoids the production of weakly Pareto optimal solutions and accelerates the selection process by avoiding redundant iterations. Recently, Khalili-Damghani *et al.*[45] customized the AUGMECON method proposed by Mavrotas[24] for real-life large scale multi-objective problems.

3. Proposed DSS

In this section, we present different components of the DSS proposed in this study. First, we introduce an algorithm for the TOPSIS module. We then introduce an efficient ε-constraint method for generating high-quality solutions on the Pareto front. Afterwards, a DEA model is constructed to implement the pruning strategy. Finally, an integrated DSS framework is presented to combine different modules into a cohesive and integrated algorithmic procedure.[46]

3.1. TOPSIS Module (Algorithm I)

Step 1. Solve the single objective optimization problems using the same constraints of the original MODM problem:

$$\{Max\ \ f_i(X), i = 1, 2, \ldots, k; X \in S\} \tag{3}$$

Step 2. Consider the original MODM problem and calculate Z^+ and Z^- vectors (4)–(5) as follows:

$$Z^- = \left(Z_1^-, Z_2^-, \ldots, Z_i^-, \ldots, Z_{k-1}^-, Z_k^-\right) \tag{4}$$

$$Z^+ = \left(Z_1^+, Z_2^+, \ldots, Z_i^+, \ldots, Z_{k-1}^+, Z_k^+\right) \tag{5}$$

where, Z^- is the vector of the optimum values for the single objective problems in the previous step which have been optimized contrary to the directions of the original MODM problem (i.e. NIS). Z^+ is the vector of the optimum values for the single objective problems in the previous step which have been optimized in the same direction of the original MODM problem (i.e. PIS).

Step 3. Using the NIS and the PIS, and the DM's preferences on the relative importance of the objective functions, calculate the distance from the NIS and the distance from the PIS as (6)–(7), respectively.

$$d_p^{PIS} = \left[\sum_{\text{for all min objectives}} \left[W_i \times \frac{\left(f_i(X) - z_k^+\right)}{z_k^- - z_k^+}\right]^p + \sum_{\text{for all max objectives}} \left[W_i \times \frac{\left(z_k^+ - f_i(X)\right)}{z_k^+ - z_k^-}\right]^p\right]^{\frac{1}{p}} \tag{6}$$

$$d_p^{NIS} = \left[\sum_{\text{for all min objectives}} \left[W_i \times \frac{\left(z_k^- - f_i(X)\right)}{z_k^- - z_k^+}\right]^p + \sum_{\text{for all max objectives}} \left[W_i \times \frac{\left(f_i(X) - z_k^-\right)}{z_k^+ - z_k^-}\right]^p\right]^{\frac{1}{p}} \tag{7}$$

where, $\sum_{i=1}^{k} W_i = 1$, and $p = 1, 2, \ldots, \infty$. Note that d_p^{PIS} and d_p^{NIS} are scale independent measures.

Step 4. Considering the DM's preferences on the priority relations of the objective functions, we can solve the following resulting bi-objective Problem (8) using a MODM technique either in a crisp or a fuzzy environment. We refer to the resulting optimization model as the *TOPSIS-based bi-objective problem* throughout the paper.

$$\begin{aligned} &Min && d_p^{PIS} \\ &Max && d_p^{NIS} \\ &s.t. && x \in S \end{aligned} \tag{8}$$

where, S is the feasible space of the original MODM problem.

Figure 1 presents graphic effects of applying the TOPSIS method to a minimization bi-objective problem. As is shown in this figure, TOPSIS applied to a MODM problem produces a rich concentration on the Pareto front. Moreover, some other non-dominated solutions which do not consider cost-based and benefit-based properties are simultaneously ignored. Although the solution diversity on the Pareto front is assumed to be one of the main attributes of MODM procedures, this type of concentration has little effect on the diversity of the solutions. More formally, in real-life MODM problems, designers may be interested in guiding the solutions towards a specific area of the feasible region in which the non-dominated solutions on the Pareto front have high quality and low risk, simultaneously.

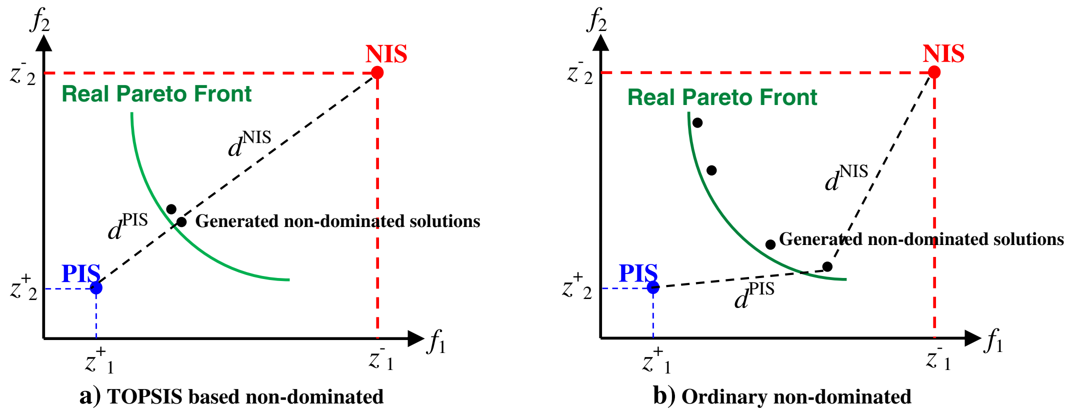

Figure 1. Effect of the TOPSIS on the bi-objective problem.

3.2. *Efficient ε-constraint module*

Re-visiting the MODM problem (1), in the ε-constraint method, one of the objective functions (e.g. $f_j(x)$) is optimized while considering the remaining objective functions as constraints:

$$\max\{f_j(x), j \in \{1,\dots,k\}; f_i(x) \geq \varepsilon_i, \forall i \in \{1,\dots,k\}, i \neq j; x \in S\}. \tag{9}$$

3.2.1. Traditional ε-constraint method The traditional ε-constraint method obtains the efficient solutions of the problem through parametrical variation in the right-hand side (RHS) of the constrained objective functions (i.e. ε_i, $i=1,\dots k, i \neq j$). The ε-constraint method has three points that need attention in its implementation: (i) the calculation of the range of the objective functions over the efficient set, (ii) the guarantee of efficiency of the obtained solution, and (iii) the increased solution time for problems with several (more than two) objective functions. The number of required single optimization problems for a full analysis of the traditional ε-constraint method is equal to $(T+1)^{k-1}$, where k is the number of the objective functions and T is the number of equal intervals of the ranges of the objective functions.

3.2.2. Modified efficient ε-constraint method

Step 1. Calculate the payoff table by using a lexicographic optimization of the objective functions.

Step 2. Divide the ranges of the objective functions to T equal intervals and use the $T+1$ grid points as the RHS values of the constrained objective functions (grid step).

Step 3. Solve the sets of Problem (10) or (11) $T+1$ times.

It is obvious that the optimal solution of Problem (2) or Problem (9) is guaranteed to be an efficient solution only if the value of the slack or the surplus variables of the entire associated $(k-1)$ objective functions' constraints are equal to zero. In order to overcome this problem, the following slack-based models have been proposed:

$$\begin{aligned} &\max f_j(x) + \beta \times (s_1 + \dots + s_{j-1} + s_{j+1} + \dots + s_k) \\ &s.t. \\ &\quad f_i(x) - s_i = \varepsilon_i, \forall i \in \{1,\dots,k\}, i \neq j. \\ &\quad X \in S. \\ &\quad s_i \in \Re^+ \quad \forall i \in \{1,\dots,k\}, i \neq j. \end{aligned} \tag{10}$$

$$\begin{aligned} &\min f_j(x) - \beta \times (s_1 + \dots + s_{j-1} + s_{j+1} + \dots + s_k) \\ &s.t. \\ &\quad f_i(x) + s_i = \varepsilon_i, \forall i \in \{1,\dots,k\}, i \neq j. \\ &\quad X \in S. \\ &\quad s_i \in \Re^+ \quad \forall i \in \{1,\dots,k\}, i \neq j. \end{aligned} \tag{11}$$

where, β is a small number usually between 0.001 and 0.000001.

The above formulations (10) and (11) of the ε-constraint method produces only efficient solutions. Some consideration about the commensurability may be desirable in the objective functions of problems (10) and (11). So, the objective function of problems (10) and (11) will be as follows:

$$f_j(x) + \beta \times \left(s_1/r_1 + \ldots + s_{j-1}/r_{j-1} + s_{j+1}/r_{j+1} + \ldots + s_k/r_k\right) \tag{12}$$

where, $r_i, i = 1, \ldots, k$ represents the range for objective i which has been calculated from the payoff table of the associated single objective optimization problem of the original MODM problem. It is clear that replacing Problem (10) or (11) in Step 3 of the modified efficient ε-constraint method will allow the AUGMECON method to generate efficient solutions.

3.3. DEA module

Although Models (26)–(28) help to limit the number of generated solutions, the final selection from the resulting non-dominated solutions is still a difficult task. Unfortunately, this problem has been ignored in several multi-objective optimization models.[16,17] Most often, a set of non-dominated solutions or clusters which contain non-dominated solutions are represented to the DMs for final selection. However, choosing among a set of non-dominated solutions becomes somewhat arbitrary in light of the lack of a clear systematic pruning procedure. This is especially problematic when there is no prior articulation of DM's preferences on the priority of the objective functions. We propose a DEA model to prune the generated non-dominated solution when the DM has no prior articulation of the preferences on the objective function priorities.

DEA is a non-parametric method for measuring the efficiency of decision-making units (DMUs), first introduced by Charnes, Cooper, and Rhodes (CCR).[47] The original CCR model was applicable only to problems characterized by constant returns to scale globally. Banker, Charnes, and Cooper [48] later extended the CCR model to accommodate problems that exhibited variable returns to scale.

We consider each non-dominated solution generated by the previous module as a DMU with two inputs (i.e. cost and weight) and one output (i.e. reliability). Figure 2 presents the schematic view of the DMU in this DEA model.

Assuming that there are a set of n DMUs ($DMU_j, j = 1, \ldots, n$) producing s outputs ($y_{rj}, r = 1, \ldots, s$) by consuming m inputs ($x_{ij}, i = 1, \ldots, m$), the following additive model proposed by Charnes *et al.* [48] is to determine whether a given DMU (i.e. a non-dominated solution) is efficient or not:

$$\begin{aligned}
&\max \quad \sum_{i=1}^{m} S_{ip}^{-} + \sum_{r=1}^{s} S_{rp}^{+} \\
&s.t. \\
&\quad \sum_{j=1}^{n} \lambda_j x_{ij} + S_{ip}^{-} = x_{ip}, \quad i = 1, 2, \ldots, m \\
&\quad \sum_{j=1}^{n} \lambda_j y_{rj} - S_{rp}^{+} = y_{rp}, \quad r = 1, 2, \ldots, s \\
&\quad \sum_{j=1}^{n} \lambda_j = 1 \\
&\quad \lambda_j,\ S_{ip}^{-},\ S_{rp}^{+} \geq 0, \quad j = 1, 2, \ldots, n, \quad i = 1, 2, \ldots, m, \quad r = 1, 2, \ldots, s
\end{aligned} \tag{13}$$

where, S_{ip}^{-} and S_{rp}^{+} are the input and output slacks and the DMU_p (i.e. the p-th non-dominated solution in a given set of non-dominated solutions) is efficient under the additive Model (13) if and only if the optimal value of its objective function is zero. Otherwise, Model (13) seeks for the non-dominated solutions that consume lower cost and weight, and produce higher reliability.

3.4. Proposed DSS framework

The proposed DSS framework depicted in Figure 3 is composed of three steps as follows:

Step 1. Apply Module (I) to reduce the dimension of objective space.
Step 2. Apply Module (II) on resultant bi-objective problem of Step (1).
Step 3. Apply Module (III) to prune non-dominated solutions of Step (2).

The proposed DSS has several the following interesting properties: (i) the proposed method reduces the multi-objective functions space into an efficient bi-objective space; (ii) the DM can be confident that high-quality solutions which are simultaneously as close

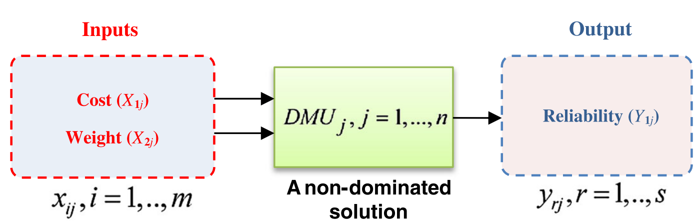

Figure 2. A given non-dominated solution as a DMU.

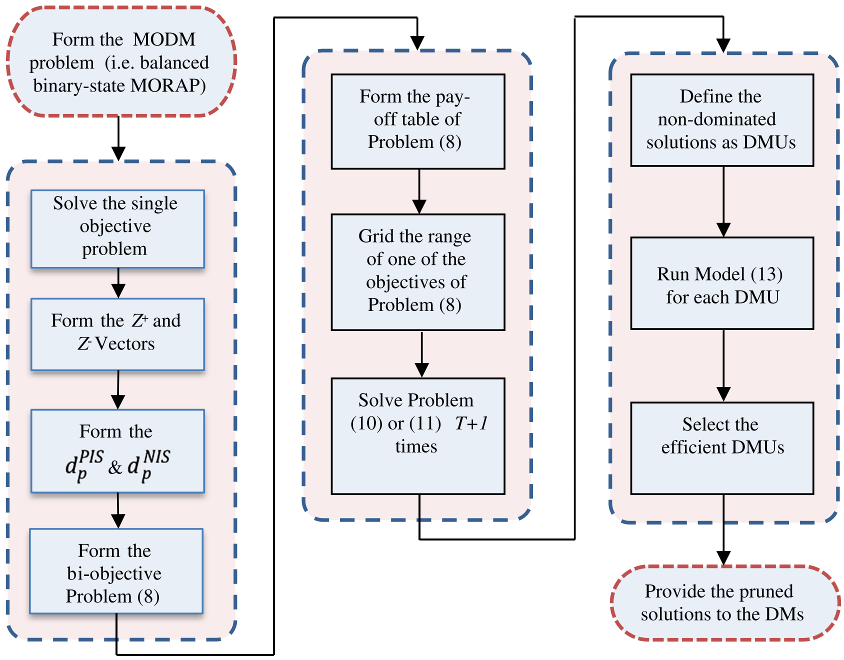

Figure 3. Proposed decision support system.

as possible to the ideal point and as far as possible from the anti-ideal point are generated; (iii) the application of the efficient ε-constraint method guarantees that the non-dominated solutions are generated on the Pareto front with minimum computation requirements; (iv) the calculations of the pay-off table in Problem (8) are more manageable and convenient; (v) the number of required single optimization problems for a full analysis of the efficient ε-constraint method decreases drastically while we are confident that quality solutions are generated on the Pareto front (i.e. a guided search on the Pareto front); and (vi) the DEA module prunes the generated non-dominated solutions and helps the DMs select the most efficient non-dominated solutions considering reliability, cost, and weight considerations.

4. Application of the proposed DSS

In the binary-state MORAPs, a set of objective functions (i.e. reliability, cost, weight, and volume) are to be optimized considering a set of constraints (i.e. cost, weight, and volume). The basic assumptions for the binary-state MORAPs are as follows:

- All components are assumed to be non-repairable,
- All components are assumed to have two states (i.e. working/fail),
- The functioning and physical properties (i.e. reliability, volume, weight, and cost) of all components are assumed to be known, deterministic, and time independent, and
- The cost of each component is assumed to be time independent.

The following notations are used in the binary-state MORAPs:

m	Number of sub-systems
i	Index of sub-systems, $i=1,2,..,m$
j	Index of components in each sub-systems, $j=1,2,..,n$
r_{ij}	Reliability of component j in sub-system i
c_{ij}	Cost of component j in sub-system i
w_{ij}	Weight of component j in sub-system i
R_s	Overall reliability of the series–parallel system
C_s	Overall cost of the series–parallel system
W_s	Overall weight of the series–parallel system
C_o	Allowed cost of system
W_o	Allowed weight of system
a_i	Number of available component choices for sub-system i
x_{ij}	Quantity of component j used in sub-system i
n_i	Total number of components used in sub-system i
$n_{\max}$	Maximum number of components which can be in parallel
$n_{\min}$	Minimum number of components which can be in parallel

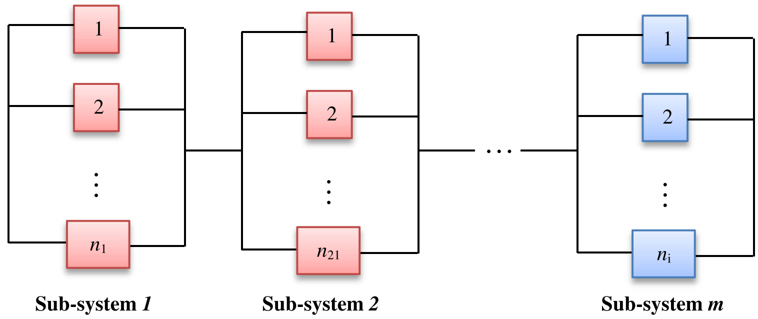

Figure 4. Series-parallel system.

Models (14)–(21) represent a binary-state MORAP in which (14)–(16) are allocated to describe the reliability, cost, and weight objective functions, respectively.

$$Max \quad R_s = \prod_{i=1}^{m}\left(1\text{-}\prod_{j=1}^{a_i}(1\text{-}r_{ij})^{xij}\right) \tag{14}$$

$$Min \quad C_s = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{15}$$

$$Min \quad W_s = \sum_{i=1}^{m}\sum_{j=1}^{n} w_{ij}x_{ij} \tag{16}$$

Table I. Benchmark case data

	Sub-system 1			Sub-system 2			Sub-system 3		
Component	Reliability	Cost	Weight	Reliability	Cost	Weight	Reliability	Cost	Weight
1	0.94	9	9	0.97	12	5	0.96	10	6
2	0.91	6	6	0.86	3	7	0.89	6	8
3	0.89	6	4	0.7	2	3	0.72	4	2
4	0.75	3	7	0.66	2	4	0.71	3	4
5	0.72	2	8	-	-	-	0.67	2	4

Table II. Pay-off of the original MORAP

Ideal calculations	**Reliability**	**Cost**	**Weight**
Reliability	0.9999985	128	105
Cost	0	0	0
Weight	0	0	0
Anti-ideal calculations	**Reliability**	**Cost**	**Weight**
Reliability	0	0	0
Cost	1	284	160
Weight	0.9999998	144	192

Table III. Pay-off of the bi-objective Problem (8)

Ideal calculations	d_p^{PIS}	d_p^{NIS}
d_p^{PIS}	0.2852996	2.714700
d_p^{NIS}	0.2852996	2.714700
Anti-ideal calculations	d_p^{PIS}	d_p^{NIS}
d_p^{PIS}	2.160211	0.8397887
d_p^{NIS}	2.160211	0.8397887

Table IV. Objective values of the generated solutions for the benchmark case

Proposed modified efficient ε-constraint method										
ε_2 values										
-	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
R_s	0.9996531	0.9996531	0.9996531	0.9996531	0.9996531	0.9996531	0.9996531	0.9996531	0.9996531	0.9988790
C_s	68	68	68	68	68	68	68	68	68	62
W_s	54	54	54	54	54	54	54	54	54	48
AUGMECON method										
Epsilon values*										
-	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
R_s	0.9999688	0.9999688	0.9999688	0.9999409	0.9998862	0.9999296	0.9999717	0.9999421	0.9984723	0.9372272
C_s	89	89	89	85	99	84	104	100	72	36
W_s	89	89	89	90	74	95	73	57	38	19

$$s.t.\quad \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \le C_o \tag{17}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} w_{ij}x_{ij} \le W_o \tag{18}$$

$$\sum_{j=1}^{a_i} x_{ij} \le n_{\max},\quad i = 1,2,\ldots,m \tag{19}$$

$$\sum_{j=1}^{a_i} x_{ij} \ge n_{\min},\quad i = 1,2,\ldots,m \tag{20}$$

$$x_{ij} \in Z^{+},\qquad i = 1,2,\ldots,m;\quad j = 1,2,\ldots,n \tag{21}$$

Constraint (17) represents the maximum allowed cost of the system, while Constraint (18) represents the weight of the system. The constraint sets (19) and (20) guarantee the minimum and maximum allowed number of components in each sub-system. The constraint set (21) guarantees that the decision variables are positive integers. Figure 4, represents the schematic view of a series-parallel system.

The application of the efficient ε-constraint method on model (14)–(21) will lead us to model (22)–(25).

$$\text{Max}\ \ R_s + \beta \times (S_2/r_2 + S_3/r_3) \tag{22}$$

s. t.

$$C_s + S_2 = \varepsilon_2,\quad \varepsilon_2 \in \left[C_s^{+}, C_s^{-}\right] \tag{23}$$

$$W_s + S_3 = \varepsilon_3,\quad \varepsilon_3 \in \left[W_s^{-}, W_s^{+}\right] \tag{24}$$

$$X \in S \tag{25}$$

where, r_i, $i = 1,2$ represent the range of the objective i which is calculated from the payoff table of the original binary-state MORAP (i.e. using the value of $C_s^{+}, C_s^{-}, W_s^{+}$, and W_s^{-}). $X \in S$ is the feasible region of the original binary-state MORAP (i.e. Relations (17)-(21)) and β has the same definition as in (10).

Let us consider the binary-state MORAP by maximizing d_P^{PIS} with $P = 1,2,..,\infty$ and equal relative importance for all objectives in models (26)–(28) as follows:

$$\text{Min}\ \ \mathrm{d_P^{PIS}} + \beta \times \left(\frac{S_2}{r_{d_P^{NIS}}}\right) \tag{26}$$

s. t.

$$d_P^{NIS} - S_2 = \varepsilon_2,\quad \varepsilon_2 \in \left[d_P^{NIS^-}, d_P^{NIS^+}\right] \tag{27}$$

$$X \in S \tag{28}$$

where, $d_P^{NIS^-}$ and $d_P^{NIS^+}$ are calculated from the payoff table of the bi-objective Problem (8). $r_{d_P^{NIS}}$ represents the range of the objective d_P^{NIS}. $X \in S$ is the feasible region of the original binary-state MORAP (i.e. Relations (17)–(21)).

Table V. Structure of the system for the benchmark case

	Proposed modified efficient ε-constraint method														
ε_2	Sub-system 1 component					Sub-system 2 component					Sub-system 3 component				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
0	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.1	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.2	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.3	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.4	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.5	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.6	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.7	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.8	0	0	4	0	0	0	0	8	0	0	0	0	7	0	0
0.9	0	0	4	0	0	0	0	6	0	0	1	0	4	0	0

In real-life problems with k objectives, such as ours which has three conflicting objectives, Models (22)–(25) result in $k-1$ ε-constraints. Therefore, the number of mathematical programming problems for the full analysis of Models (22)–(25) will increase exponentially. We use a benchmark case from the literature to compare the performance of Models (22)–(25) and Models (26)–(28) in the next section.

5. Experimental results

In order to test the proposed DSS, the following well-known benchmark case was selected from the literature.[16–18] This benchmark case consists of sub-systems 1, 2, and 3 with five, four, and five types of components in each sub-system, respectively. The problem involves determining the optimum number of selected component types in each sub-system. The maximum number of components in each sub-system is 8. The benchmark case data used in this study is presented in Table I.

5.1. Results

Table II presents the pay-off table for different objectives. Considering the pay-off values presented in this table, the range for each objective function and relations (17)–(18) can be easily found. Moreover, Z^+ and Z^- vectors can be calculated using (4)–(5). Table III presents the pay-off table for the resulting bi-objective Problem (8).

Table IV presents the objective values of the generated solutions from Models (22)–(25) and Models (26)–(28). Consequently, Table V presents the structure of the benchmark case solution vector for both procedures.

Different p-values for $w_i = 1, 2, 3$ were tested in the proposed DSS. According to the experimental results, the best values for parameters were $P = 1, w_1 = 0.99, w_2 = 0.005,$ and $w_3 = 0.005$. The step-size of the epsilon-based parameters was set equal to 0.01 for both methods. As is shown in Table IV, $R_s, C_s,$ and W_s were not the objective function of the proposed method, and therefore the presented solutions were not non-dominated.

5.2. Comparison index

A comparative performance analysis was conducted between the outputs of the modified efficient ε-constraint method used in the proposed DSS and the AUGMECON method. The CC index of the TOPSIS procedure was selected to evaluate the performance of the aforementioned procedures. This index has been calculated as Equation (29) for all values of ε_2 in both procedures.

$$CC_{ij} = \frac{d^{ij}_{NIS}}{d^{ij}_{PIS} + d^{ij}_{NIS}}, \quad i \in [0, 1], \quad j \in \{1, 2\}. \tag{29}$$

where, CC_{ij} represents the closeness coefficient of procedure j for parameter $\varepsilon_2 = i$. It is obvious that CC_{ij} values are between zero and one. The solutions with high CC_{ij} values are far from the NIS and close to the PIS, simultaneously. Table VI presents the partial (for brevity) CC results of the benchmark case for both procedures (i.e. size 0.1 for ε_2). All values have been generated using the step-size 0.01 for ε_2.

5.3. Analysis of variance (ANOVA) for the comparison index

In this section, the ANOVA experiments are used to compare the performance of the modified efficient ε-constraint method with the performance of the *AUGMECON* method proposed by Mavrotas[24] using the CC_{ij} statistic on the benchmark case. The ANOVA

Table V. Continued

AUGMECON method														
Sub-system 1 component					Sub-system 2 component					Sub-system 3 component				
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	1	2	0	1	1	0	7	0	0	2	2	0	0	1
1	1	2	0	1	1	0	7	0	0	2	2	0	0	1
1	1	2	0	1	1	0	7	0	0	2	2	0	0	1
1	1	1	1	1	1	0	7	0	0	1	2	2	1	0
2	1	0	1	0	3	0	0	0	0	0	2	6	0	0
1	0	2	2	1	1	0	1	5	0	2	1	0	1	1
1	2	2	0	0	3	0	1	0	0	2	1	1	1	0
0	0	5	0	0	3	0	0	0	0	0	1	7	0	0
0	0	4	0	0	2	0	0	0	0	0	0	6	0	0
0	0	2	0	0	1	0	0	0	0	0	0	3	0	0

experiments were conducted for 100 different samples. The number of sample is the direct result of using the 0.01 step-size for ε_2. Table VII presents the results of ANOVA (confidence level for all experiment was set equal to 95%).

As is shown in Table VII, there is sufficient evidence that the means are not equal since the *p*-values are less than the 5% significance level. Therefore, there is a significant difference between the considered metric in the modified efficient ε-constraint method and the *AUGMECON* method at the 95% confidence level. The results presented in Table VII show that the modified efficient ε-constraint method proposed in this study performs better than the *AUGMECON* method.

5.4. Pruning the non-dominated solutions through DEA

In this step, we used the results from the application of Models (26)–(28) on the benchmark case with a step-size of 0.01 for ε_2 and generated 100 non-dominated solutions. Model (13) was run for 10 represented non-dominated solutions (i.e. the proposed designs) of Models (26)–(28), and one design was generated as the final solution. The results are presented in Table VIII.

Generally, since the number of proposed designs is reduced through Model (13), the DM can easily select the preferred design. The last module in the proposed DSS can independently be implemented for any desired number of non-dominated solutions

Table VI. Closeness coefficients of the two procedures for the benchmark case

	Proposed modified efficient ε-constraint method			AUGMECON method		
ε_2	d_{PIS}^{i1}	d_{NIS}^{i1}	CC_{i1}	d_{PIS}^{i2}	d_{NIS}^{i2}	CC_{i2}
0	0.5210335	2.478966	0.82632214	0.7769532	2.223047	0.7410156
0.1	0.5210335	2.478966	0.82632214	0.7769532	2.223047	0.7410156
0.2	0.5210335	2.478966	0.82632214	0.7769532	2.223047	0.7410156
0.3	0.5210335	2.478966	0.82632214	0.7681049	2.231895	0.743965
0.4	0.5210335	2.478966	0.82632214	0.7341220	2.265878	0.7552927
0.5	0.5210335	2.478966	0.82632214	0.7906368	2.209363	0.7364544
0.6	0.5210335	2.478966	0.82632214	0.7464338	2.253566	0.7511887
0.7	0.5210335	2.478966	0.82632214	0.6490456	2.350954	0.7836514
0.8	0.5210335	2.478966	0.82632214	0.4529655	2.547035	0.8490115
0.9	0.4694308	2.530569	0.84352306	0.2884917	2.711508	0.9038361
Mean	0.5158732	2.4841263	0.82804223	0.676066	2.323934	0.7746447

Table VII. Analysis of variance data for the comparison metric CC_{ij}

Source	Degree of freedom	Sum of square	Mean square	F	*p*-value
Proposed method	1	0.16670	0.16670	79.90	0.000
Error	198	0.41312	0.00209		
Total	199	0.57983			
S =0.04568 R-Sq =28.75% R-Sq(adj.) =28.39%					

Table VIII. Pruning the non-dominated solutions with the additive DEA Model (13)

Non-dominated solution (DMU)	Input		Output	S_{ip}^{-}		S_{rp}^{+}	Objective value
	Cost	Weight	Reliability	S_{1p}^{-}	S_{2p}^{-}	S_{1p}^{+}	
DMU_1	68	54	0.9996531	14.00	20.00	0.00	34.00
DMU_2	68	54	0.9996531	0.00	6.00	0.00	6.00
DMU_3	68	54	0.9996531	14.00	20.00	0.00	34.00
DMU_4	68	54	0.9996531	0.00	6.00	0.00	6.00
DMU_5	68	54	0.9996531	14.00	20.00	0.00	34.00
DMU_6	68	54	0.9996531	0.00	6.00	0.00	6.00
DMU_7	68	54	0.9996531	14.00	20.00	0.00	34.00
DMU_8	68	54	0.9996531	0.00	6.00	0.00	6.00
DMU_9	68	54	0.9996531	14.00	14.00	0.00	28.00
DMU_{10}	62	48	0.9988790	0.00	0.00	0.00	0.00

(i.e. 100 non-dominated solutions for step-size 0.01). As is shown in Table VIII, DMU_{10} is the efficient DMU. Therefore, the combination of TOPSIS and the modified efficient ε-constraint method resulted in the generation of non-dominated solutions with minimum cost-based criteria and maximum benefit-based criteria plus. Furthermore, the pruning strategy used in accordance with the additive DEA Model (13) eased the final selection process. The final proposed structure is presented in Figure 5. We should note that the number of non-dominated solutions generated from Model (13) was 10, and only one solution in this case was selected after the application of the pruning procedure. However, several non-dominated solutions may be selected as the efficient DMU after the pruning procedure depending on the size of the problem and the values of the input and output data in the final DEA model.

6. Conclusions and future research directions

In this paper, a DSS has been proposed to efficiently solve the MODM problems by producing a Pareto front with a DM preferred resolution. The core of proposed DSS is based on three main modules: A TOPSIS module, a modified efficient ε-constraint module, and a DEA module. Initially, the multi-objective space of the MODM problem was reduced to a bi-objective space using TOPSIS. This resulted in high-quality and low-risk solutions. Moreover, all the objective functions in the original MODM problem had an impact on designing the proposed bi-objective problem. The solutions of the bi-objective problem were far from the NIS and close to the PIS, simultaneously while the domination structure of the original problem met the least perturbations. We then used a modified efficient ε-constraint method to generate non-dominate solutions with a predefined and arbitrary resolution on the Pareto front of the aforementioned bi-objective problem. The ε-constraint method was customized for large-scale and real-life MODM problems. Only one ε-constraint type was added to the original constraints of the MODM problem with *k* different objectives while this number is equal to $k-1$ in the existing procedures in the literature. This narrows the solution space and reduces the computational efforts needed to solve the problem. Nevertheless, it cannot pinpoint the optimum solution. A DEA model was constructed to prune and select a solution form the non-dominated solutions generated using the modified efficient ε-constraint method.

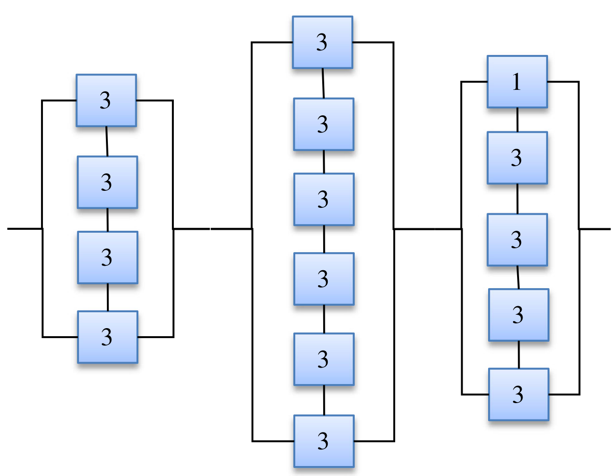

Figure 5. Final proposed structure.

We used the framework proposed in this study to solve a binary-state multi-objective reliability redundancy allocation series-parallel problem (which is an NP-Hard problem). A well-known binary-state MORAP was selected from the literature as a benchmark case. The proposed DSS was used to solve this benchmark case. The result from the modified efficient ε-constraint method was compared with the results from the *AUGMECON* method. The statistical analysis revealed that the proposed DSS can efficiently generate solutions with higher quality.

The DSS proposed in this study has been developed as a general engineering design tool and can be customized and used to design and develop various real-life operations and manufacturing systems in the public and private sectors of the economy.

Acknowledgement

The authors would like to thank the anonymous reviewers and the editor for their insightful comments and suggestions.

References

1. Misra KB, Ljubojevic MD. Optimal reliability design of a system: A new look. *IEEE Transaction on Reliability* 1973; **22**(5):255–258.
2. Chern MS. On the computational complexity of reliability redundancy allocation in a series system. *Operations Research Letters* 1992; **11**(5):309–315.
3. Kuo W, Zuo MJ. Optimal Reliability Modeling: Principles and Applications. John Wiley and Sons: Hoboken, 2003.
4. Aven T. On performance-measures for multistate monotone systems. *Reliability Engineering and System Safety* 1993; **41**(3):259–266.
5. Juang Y-Sh, Lin Sh-Sh, Kao H-P. A knowledge management system for series-parallel availability optimization and design. *Expert Systems with Applications* 2008; **34**(2):181–193.
6. Calabria R, Pulcini G, Rapone M. Optimal reliability and maintainability allocation in manufacturing systems. *Quality and Reliability Engineering International* 1995; **11**(3):183–188.
7. Shelokar PS, Jayaraman VK, Kulkarni BD. Ant algorithm for single and multiobjective reliability optimization problems. *Quality and Reliability Engineering International* 2002; **18**(6):497–514.
8. Liang YC, Smith AE. An ant colony optimization algorithm for the redundancy allocation problem (RAP). *IEEE Transactions on Reliability* 2004; **53**(3):417–423.
9. Kuo W, Wen R. Recent advances in optimal reliability allocation. *Studies in Computational Intelligence* 2007; **39**:1–36.
10. Yeh W-Ch. A two-stage discrete particle swarm optimization for the problem of multiple multi-level redundancy allocation in series systems. *Expert Systems with Applications* 2009; **36**(5):9192–9200.
11. Wang Z, Tang K, Yao X. A Memetic Algorithm for Multi-Level Redundancy Allocation. *IEEE Transactions on Reliability* 2010; **59**(4):754–765.
12. Ahmadizar F, Soltanpanah H. Reliability optimization of a series system with multiple-choice and budget constraints using an efficient ant colony approach. *Expert Systems with Applications* 2011; **38**(4):3640–3646.
13. Zou D, Gao L, Li S, Wu J. An effective global harmony search algorithm for reliability problems. *Expert Systems with Applications* 2011; **38**(4):4642–4648.
14. Wang Y, Li L. Heterogeneous Redundancy Allocation for Series-Parallel Multi-State Systems Using Hybrid Particle Swarm Optimization and Local Search. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 2012; **42**(2):464–474.
15. Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering and System Safety* 2006; **91**(9):992–1007.
16. Taboada H, Baheranwala F, Coit DW, Wattanapongsakorn N. Practical solutions for multi-objective optimization: An application to system reliability design problems. *Reliability Engineering and System Safety* 2007; **92**(3):314–322.
17. Li Z, Liao H, Coit DW. A two-stage approach for multi-objective decision making with applications to system reliability optimization. *Reliability Engineering and System Safety* 2009; **94**(2):1585–1592.
18. Zio E, Bazzo R. A clustering procedure for reducing the number of representative solutions in the Pareto Front of multi-objective optimization problems. *European Journal of Operational Research* 2011; **210**(3):624–634.
19. Gen M, Yun Y. Soft computing approach for reliability optimization: State-of-the-art survey. *Reliability Engineering and System Safety* 2006; **91**(5):1008–1026.
20. Khalili-Damghani K, Amiri M. Solving binary-state multi-objective reliability redundancy allocation series-parallel problem using a hybrid efficient epsilon-constraint constraint, multi-start partial bound enumeration algorithm, and DEA. *Reliability Engineering and System Safety* 2012; **103**:35–44.
21. Khalili-Damghani K, Abtahi A-R, Tavana M. A new multi-objective particle swarm optimization method for solving reliability redundancy allocation problems. *Reliability Engineering and System Safety* 2013; **111**:58–75.
22. Roy B. Decision-aid and decision making. *European Journal of Operational Research* 1990; **45**(2-3):324–331.
23. Vincke P. Multicriteria Decision Aid. John Wiley: New York, 1992.
24. Mavrotas G. Effective implementation of the ε-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computations* 2009; **213**(2):455–465.
25. Zhang JLG, Ruan D, Wu F. Multi-objective Group Decision Making Methods, Software and Applications with Fuzzy Set Techniques. Imperial College Press: London, 2007.
26. Ehrgott M. Multicriteria Optimization (2nd edn). Springer: New York, 2005.
27. Hwang CL, Masud ASM. Multiple Objective Decision Making Methods and Applications. Springer-Verlag: New York, 1979.
28. Shelokar PS, Jayaraman VK, Kulkarni BD. Ant algorithm for single and multiobjective reliability optimization problems. *Quality and Reliability Engineering International* 2002; **18**(6):497–514.
29. Lins ID, Moura MDC, Zio E, Droguett EL. A particle swarm-optimized support vector machine for reliability prediction. *Quality and Reliability Engineering International* 2012; **28**(2):141–158.
30. Ardakani MK, Wulff SS. An overview of optimization formulations for multiresponse surface problems. *Quality and Reliability Engineering International* 2013; **29**(1):3–16.
31. Ehrgott M, Wiecek MM. Multiobjective programming. In Multiple Criteria Decision Analysis: State of the Art Surveys, Figueira J, Greco S, Ehrgott M (eds). Springer Science + Business Media: New York, 2005; 667–722.
32. Kahraman C, Kaya I. Fuzzy multiple objective linear programming. In Fuzzy Multi-Criteria Decision Making: Theory and Applications with Recent Developments, Kahraman C (ed.). Springer Science + Business Media: New York, 2008; 325–338.
33. Hwang CL, Yoon K. Multiple Attribute Decision Making: Methods and Applications. Springer-Verlag: Heidelbeg, 1981.

34. Lai Y-J, Liu T-Y, Hwang C-L. TOPSIS for MODM. *European Journal of Operational Research* 1996; **76**(3):486–500.
35. Braglia M, Frosolini M, Montanari R. Fuzzy TOPSIS approach for failure mode, effects and criticality analysis. *Quality and Reliability Engineering International* 2003; **19**(5):425–443.
36. Kutlu AC, Ekmekçioğlu M. Fuzzy failure modes and effects analysis by using fuzzy TOPSIS-based fuzzy AHP. *Expert Systems with Applications* 2012; **39**(1):61–67.
37. Song W, Ming X, Wu Z, Baoting Z. A rough TOPSIS approach for failure mode and effects analysis in uncertain environments. *Quality and Reliability Engineering International* 2013, Article first published online: 7 FEB 2013, DOI: 10.1002/qre.1500.
38. Ölçer Aİ, Majumder J. A case-based decision support system for flooding crises onboard ships. *Quality and Reliability Engineering International* 2006; **22**(1):59–78.
39. Ju Y, Wang A. Emergency alternative evaluation under group decision makers: A method of incorporating DS/AHP with extended TOPSIS. *Expert Systems with Applications* 2012; **39**(1):1315–1323.
40. He Y, Tang X, Chang W. Technical decomposition approach of critical to quality characteristics for Product design for Six Sigma. *Quality and Reliability Engineering International* 2010; **26**(4):325–339.
41. Awasthi A, Chauhan SS, Omrani H, Panahi A. A hybrid approach based on SERVQUAL and fuzzy TOPSIS for evaluating transportation service quality. *Computers and Industrial Engineering* 2011; **61**(3):637–646.
42. Büyüközkan G, Çifçi G. A combined fuzzy AHP and fuzzy TOPSIS based strategic analysis of electronic service quality in healthcare industry. *Expert Systems with Applications* 2012; **39**(3):2341–2354.
43. Chankong V, Haimes Y. Multi-Objective decision making theory and methodology. Elsevier Science: New York, 1983.
44. Steuer RE. Multiple Criteria Optimization, Theory, Computation and Application. John Wiley: New York, 1986.
45. Khalili-Damghani K, Tavana M, Sadi-Nezhad S. An integrated multi-objective framework for solving multi-period project selection problems. *Applied Mathematics and Computations* 2012; **219**(6):3122–3138.
46. Charnes A, Cooper WW, Rhodes E. Measuring the efficiency of decision-making units. *European Journal of Operational Research* 1978; **2**(6):429–444.
47. Banker RD, Charnes A, Cooper WW. Some models for estimating technical and scale efficiencies in data envelopment analysis. *Management Science* 1984; **30**(9):1078–1092.
48. Charnes A, Cooper WW, Seiford LM, Stutz J. A multiplicative model for efficiency analysis. *Socio-Economic Planning Sciences* 1982; **16**(5):223–224.

Authors' biographies

Kaveh Khalili-Damghani is an assistant professor of Industrial Engineering at the Islamic Azad University, South Tehran Branch in Iran. He holds an MSc and PhD in Industrial Engineering and a PhD in Industrial Management. His research interests include soft computing, fuzzy sets and systems, meta-heuristic methods, multicriteria decision making, data envelopment analysis, reliability optimization, and applied operations research. He is an Associate Editor for several journals and has published more than 40 papers in academic journals such as *Information Sciences, Annals of Operations Research, Expert Systems with Applications, Computers and Industrial Engineering, International Journal of Advanced Manufacturing Technology, Applied Soft Computing, International Journal of Applied Decision Sciences*, and *Reliability Engineering and System Safety*.

Amir-Reza Abtahi is an assistant professor of Industrial Management at the Economic Sciences University in Tehran, Iran. He holds an MSc in industrial Management and a PhD in Industrial Engineering. His research interests include soft computing, artificial neural networks, meta-heuristic methods, project scheduling, reliability optimization, and multi-criteria decision making. He has published several papers in academic journals such as *Expert System with Application, Reliability Engineering and System Safety*, International Journal of Applied Decision Science, and *International Journal of Services and Operations Management*.

Madjid Tavana is a professor of Business Systems and Analytics and the Lindback Distinguished Chair of Information Systems and Decision Sciences at La Salle University, where he served as Chairman of the Management Department and Director of the Center for Technology and Management. He is a Distinguished Research Fellow at Kennedy Space Center, Johnson Space Center, Naval Research Laboratory at Stennis Space Center, and Air Force Research Laboratory. He was recently honored with the prestigious Space Act Award by NASA. He holds a MBA, PMIS, and PhD in Management Information Systems and received his Post-Doctoral Diploma in Strategic Information Systems from the Wharton School at the University of Pennsylvania. He is the Editor-in-Chief of *Decision Analytics, International Journal of Applied Decision Sciences, International Journal of Management and Decision Making, International Journal of Strategic Decision Sciences*, and *International Journal of Enterprise Information Systems*. He has published several books and over one hundred research papers in academic journals such as *Information Sciences, Decision Sciences, Information Systems, Interfaces, Annals of Operations Research, Advances in Space Research, Omega, Information and Management, Knowledge-Based Systems, Expert Systems with Applications, European Journal of Operational Research, Journal of the Operational Research Society, Computers and Operations Research, Energy Economics, Applied Soft Computing*, and *Energy Policy*.