Pitfalls of the Typical Construction of Decision Matrices for Concept Selection

Anoop A. Mullur
Christopher A. Mattson
Achille Messac

Corresponding Author

Prof. Achille Messac
Mechanical, Aerospace, and Nuclear Engineering Department
110 8th Street, JEC-2049
Troy, NY, 12180-3590
USA

Email: messac@rpi.edu
Tel: (518) 276-8145
Fax: (518) 276-6025
www.rpi.edu/~messac

Bibliographical Information

PITFALLS OF THE TYPICAL CONSTRUCTION OF DECISION MATRICES FOR CONCEPT SELECTION

Anoop A. Mullur*, Christopher A. Mattson†, and Achille Messac‡
Rensselaer Polytechnic Institute
Department of Mechanical, Aerospace, and Nuclear Engineering
Multidisciplinary Design and Optimization Laboratory
110 8th Street, Troy, NY 12180-3590

Abstract

The conceptual design phase of the engineering design process can be divided into three general activities (i) function specification, (ii) concept generation, and (iii) concept selection. In this paper we explore the pitfalls of one of the most popular approaches for concept selection – namely decision matrices. Although powerful in many cases, methods based on decision matrices may fail to aid designers in selecting potentially preferable designs. This is simply because decision matrices are based on an inadequate mathematical construct. In this paper, we show that some non-dominated design concepts (in particular, optimal concepts that lie on non-convex regions of the Pareto frontier) do not receive the highest total score when using decision matrices in their typical form; and as a result, may be prematurely eliminated. In practice, it is not uncommon for designers to use information from the decision matrix to classify certain design concepts as undesirable, when actually they may be desirable. This paper exposes significant risks associated with decision matrices, and suggests candidate alternative approaches. Interestingly, we also show that constructing decision matrices using approaches such as compromise programming does not necessarily solve the problem of misrepresenting the desirability of design concepts that lie on non-convex regions of the Pareto frontier.

Lexicon

Concept: A design (or a class of designs) under consideration (also called design solution)

Design criterion: A quantity used to evaluate a concept (also referred to as design objective, or performance measure)

Rating: A numerical value of a design criterion, for a given concept

Design space: A multi-dimensional space whose $i$-th generic dimension is the $i$-th generic design objective; also referred to as the objective space

Total score: An aggregation of the ratings for a concept; also referred to as value of the Aggregate Objective Function (AOF)

Pareto frontier: A set of non-dominated design points (points in design space); each of which may represent a concept

Nadir point: A fictitious point in design space with the worse possible rating for all the design objectives

Ideal point: A fictitious point in design space with the best possible rating for all the design objectives; also referred to as the utopia point

Nomenclature

$\mu_i$: $i$-th generic design objective

$P^j$: $i$-th concept or design point

$P$: Generic concept or design point

$N$: Nadir point

$\mu^j_i$: Rating of $j$-th concept for the $i$-th criterion

$\tilde{\mu}_i$: Rating of a generic concept for the $i$-th criterion

$J^j$: Total score for the $j$-th concept

$\tilde{J}$: Total score for a generic concept

$w$: Vector of weights used to evaluate total scores

1. Introduction

Concept evaluation and selection, hereafter called concept selection, is considered one of the most critical elements of successful engineering design. It is generally accepted that more than 75% of the final product cost and quality are determined during conceptual design.1,2 Various methodologies for conceptual design have been developed by the engineering design community. In this paper, we examine one approach commonly used to perform concept selection – namely decision matrices. More specifically, we examine the pitfalls associated with the typical construction of the decision matrix and propose alternatives that avoid such pitfalls.
There are numerous approaches to concept selection\textsuperscript{3-8}, which include; intuition, feasibility judgment, multivoting, numeric and non-numeric selection charts, pairwise comparisons, decision matrices, and prototype testing. As these names suggest, the level of rigor upon which these critical decisions are made can be tenuous at best.

Figure 1 shows the general objective of the concept selection process, which is to progressively narrow the design concepts down to the most promising concepts in an effective and efficient way. Each circle in Fig. 1 represents a single design concept. Ultimately, as a result of this process, a final concept, or a few concepts are chosen for development. To improve the efficiency of this process, various approaches are often used at different stages of the selection process. For example, consider the leftmost portion of Fig. 1. Here, concept selection begins; and there are many concepts to evaluate. At this stage in the selection process, there are likely to be too many concepts to justify using certain selection approaches, such as prototype testing. Instead, one may prefer such approaches as feasibility judgment. Later in the process however, when only a few promising concepts remain, building and testing prototypes could prove invaluable to the decision making process.

It is well-known that, in practice, decision matrices are broadly used at some point in the selection process.\textsuperscript{8} Concept screening and scoring (see Fig. 1) are popular decision matrix based methods that are often used to narrow a manageable number of concepts down to a select few.\textsuperscript{4} Unfortunately, the typical construction of decision matrices leaves them unable to ensure that promising design concepts are not erroneously eliminated. In this paper, we expose the pitfalls of using decision matrices in their typical construction, and present alternative methods that can be used to avoid such pitfalls.

The remainder of this paper is organized as follows. Section 2 presents a close examination of the typical construction of the decision matrix, and the pitfalls thereof. In Section 3, we mathematically show how potentially preferable design concepts may never receive the highest total score; and as a result, are unlikely to be selected. In Section 4, a discussion of the practical implications of this weakness is given, along with possible alternatives to the typical construction. In Section 5, examples are provided; and concluding remarks are given in Section 6.

2. Typical decision matrix construction and its pitfalls

In this section, we present an overview of decision matrices, as used in concept selection. The information in this section is intended to (i) show the typical manner in which these matrices are constructed, (ii) expose the risks associated with making design decisions based on these decision matrices and, (iii) lay the groundwork for the mathematical proof presented in Section 3.

2.1 Typical construction of a decision matrix

As shown in Table 1, the typical construction of a decision matrix consists of (i) listing the generic selection criteria (design criteria), $\mu_j$, (ii) their associated weights, $w_j$, and (iii) performance ratings for each concept, $P_j$. Each concept under evaluation is assigned a column of the matrix. Each concept is then rated based on its estimated ability to meet a given criterion, as compared to a reference (datum or benchmark) design. A typical set of ratings is given in Table 2. The weight of each criterion is multiplied by the concept rating, and all weighted ratings are added together to obtain the total score for that concept. This operation is shown in the equation

\[ J^i = \sum_{j=1}^{n} w_j \mu_j^i \]  

where $J^i$ is the total score for the $i$-th concept, $n$ is the number of design criteria, $w_j$ is the weight for the $j$-th criterion, and $\mu_j^i$ is the rating of the $i$-th concept for the $j$-th criterion. Under this framework, concepts with higher total scores are preferred over those with lower total scores.

<table>
<thead>
<tr>
<th>Criteria ($\mu_j$)</th>
<th>Weight ($w_j$)</th>
<th>$P^1$</th>
<th>$P^2$</th>
<th>$P^3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimal Volume ($\mu_1$)</td>
<td>0.6</td>
<td>2</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Minimal Deflection ($\mu_2$)</td>
<td>0.4</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Total Score ($J$)</td>
<td>--</td>
<td>1.6</td>
<td>4.2</td>
<td>3.8</td>
</tr>
<tr>
<td>Concept Rank</td>
<td>--</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1 Typical construction of a decision matrix
Table 2 Typical concept ratings

<table>
<thead>
<tr>
<th>Rating</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Much worse than reference</td>
</tr>
<tr>
<td>2</td>
<td>Worse than reference</td>
</tr>
<tr>
<td>3</td>
<td>Same as reference</td>
</tr>
<tr>
<td>4</td>
<td>Better than reference</td>
</tr>
<tr>
<td>5</td>
<td>Much better than reference</td>
</tr>
</tbody>
</table>

Throughout this paper we refer to the typical construction of the decision matrix. When we do so, we are referring to the construction described by Table 1 and Eq. (1).

There are many variations of this typical construction, such as evaluating a normalized score for each concept. This and other variations provide additional information, but unfortunately do not avoid the pitfalls presented in the following section. Later, we discuss a unique variation of the traditional construction called the Pugh method. Interestingly, the Pugh method is a decision matrix based method that avoids the pitfalls associated with the typical construction shown above.

When designers use the decision matrix based methods as constructed above they are exposed to a significant risk that can greatly impact the decision making process. Furthermore, this risk can occur in the simplest of practical cases. This risk is defined and described in full in the following section.

2.2 Risks of using decision matrix based methods

The risk associated with the typical construction of the decision matrix is rooted in the mathematical construct shown in Eq. (1). This equation shows that decision matrices are typically based on a weighted summation of concept ratings.

We expose the pitfalls of this construction by making an important observation regarding concept selection in general. Within the engineering design process, concept selection can be considered an optimization exercise – the process of choosing the best from among a set under consideration. This observation invites us to consider well-known characteristics of certain popular optimization methods. Numerical optimization methods based on sums of weighted criteria have been outwardly criticized in the multiobjective optimization community. The main limitation of weighted sum methods is their inability to yield solutions that lie on non-convex regions of the Pareto frontier. The critical implication is that weighted sum methods, including a decision matrix of typical construction, could miss potentially preferable design concepts. A development team faced with this situation may eliminate the most promising solution simply because an inadequate formulation dictates the results of the decision matrix based method. In the context of Fig. 1, the preferred design is not allowed to proceed to the prototype and test stages, and consequently does not reach the production stage.

Simply stated, the risk of using decision matrices in their typical form is that design concepts that are actually optimal may appear to be undesirable. Figure 2 depicts a scenario involving concepts that form a non-convex Pareto frontier. (We recall that a Pareto point/design is a feasible design for which no other design is better in every respect.) In Fig. 2, each solid circle represents a concept, and the axes represent the ratings for two design criteria. Such a figure is a representation of the design space. Figure 2 shows the design space for the objectives $\mu_1$ and $\mu_2$. In this case the overall objective is to maximize the total score, and a higher rating reflects better performance.

![Figure 2 Maximization of two objectives](image-url)
form the Pareto frontier. Concept $\hat{P}$ lies on a non-convex Pareto frontier.

The risk associated with using decision matrices in their typical weighted sum form is that no concept within the triangle (Fig. 2) can receive the highest total score – even though it is a non-dominated concept (Pareto). This is because any point inside the triangle will form a non-convex Pareto frontier with points $P^1$ and $P^2$. Therefore it is plausible that, to a designer, concept $\hat{P}$ will appear undesirable when in fact it may be desirable. The danger here is that by appearing undesirable, concept $\hat{P}$ is likely to be eliminated. This situation would be particularly deleterious if concept $\hat{P}$ would have otherwise been the preferred design.

In the following section, we prove mathematically that indeed no point within the prescribed triangle (polyhedral in $n$ dimensions) will receive the highest score while using the typical construction of the decision matrix.

It is important at this point to discuss a popular decision matrix based method that, because of its particular properties, does not directly suffer from the problem described above. This popular method is the Pugh method [7] (closely related to concept screening [4]). The Pugh method has the same formulation as that described by Eq. (1) and Table 1, with the added requirement that all weights must be equal to one. That is, no one selection criterion has more or less importance than any other criterion. In this respect, the Pugh method deliberately avoids the use of weights. In fact, Pugh himself argues that weights are misleading in nature [7]. A different and important item to note regarding the Pugh method is the rating structure. There are only three levels: +1, 0, and -1, depending on whether the concept is better than, equal to, or worse than a reference concept for some criterion. Let us consider the bi-objective case, from which similar conclusions can be made for the $n$-objective case. An interesting observation can be made: (i) By requiring all weights to be equal, solutions (concepts) are forced to lie either on convex regions of the Pareto frontier or on dominated regions of the design space, and (ii) the effective slope of the objective function $J$, with respect to each criterion, is fixed at -1. This is shown in Fig. 3. Using the Pugh method, concepts can be represented only by points A through I. This discrete space is markedly different from the continuous space of Fig. 2.

Consider Quadrant I of Fig. 3. As in Fig. 2, $\mu_1$ and $\mu_2$, are design criteria; and the overall objective is to maximize the concept ratings. If Concept A is part of the evaluation then it always receives the highest score ($J = 1$). If Concept A does not exist, then Concepts B and C both receive the highest score ($J = 1$), while Concepts D through I are always dominated by Concepts A, B, and C.

It is important to note that, while the Pugh method does not suffer from the pitfalls of the typical decision matrix construction, it does contain less information than other decision matrix based methods such as concept scoring [4]. It contains less information because there is no description of how selection criteria may actually relate to each other – it assumes all criteria are equally important. In the later stages of concept selection such an assumption may prove ill-advised.

We now proceed to develop a mathematical development that shows how decision matrix based methods fail to identify some potentially preferable concepts.

3. Mathematical proof

In this section, we present a mathematical proof that exposes the pitfalls of using decision matrices as they are typically formed. In Section 3.1 we describe the basic outline of the proof. In Section 3.2, notations and assumptions for the proof are presented, while the proof itself is given in Section 3.3.

3.1 Proof: Approach

To illustrate the basic outline this proof, consider Fig. 2, where the points labeled $\bar{P}$, $P^1$, and $P^2$ represent candidate design concepts. The coordinates of the points are the ratings of the concepts for the criteria represented by $\mu_1$ and $\mu_2$. Point $P^N$ is the nadir point, a point with minimum possible rating for each criterion. In the two dimensional case of Fig. 2, the triangle defined by $P^1$, $P^2$, and $P^N$ represents an area of the design space inside which any generic concept, $\bar{P}$, will not receive the highest score.
even though it is a non-dominated concept. To extend this discussion to \( n \) dimensions, we similarly examine \( n \) criteria and \( n+1 \) concepts including \( P^i (i=1,\ldots,n) \) and \( \tilde{P} \).

Concept \( P^i \) is defined as having the maximum value of the rating for the \( i-th \) criterion, represented on axis \( \mu_i \). Any generic concept, \( \tilde{P} \) that lies within the polyhedral formed by \( P^i (i=1,\ldots,n) \) and \( P^N \), will not receive the highest score, even though it is a non-dominated concept. In the following proof, we show that a generic concept, \( \tilde{P} \), will never receive the highest total score, when all the concepts \( P^i (i=1,\ldots,n) \) and \( \tilde{P} \) are evaluated using the typical decision matrix formulation shown in Table 1 and Eq. (1).

In this proof we consider only the comparison of non-dominated, or Pareto, concepts. As a result, each concept can be considered a solution to a well-posed optimization problem. So, throughout the remainder of the paper, we also refer to a concept as a solution. Again, a Pareto solution is one for which any improvement in one objective results in the worsening of at least one other objective. In order to compare only those concepts that are Pareto optimal, we eliminate all dominated solutions using a **Global Pareto Filter**. The global Pareto filter is designed to examine the entire set of concepts, and to remove non-Pareto and locally Pareto solutions; leaving only the global Pareto solutions. Details regarding various filtering techniques can be found in Ref. [15].

The following proposition clearly states the objective of this proof.

**Proposition:** Using decision matrices of typical construction, the total score for concept \( \tilde{P} \), which lies on a non-convex region of the Pareto frontier defined by \( \tilde{P} \) and \( P^i (i=1,\ldots,n) \), is never the highest among all the concepts under evaluation.

### 3.2 Notations and Assumptions

Before we present the mathematical proof, we state the following assumptions and define the following notations.

**Assumptions and observations:**

1. On a rating scale, a higher value is better, so that a concept with a higher total score is preferred over a concept with a lower total score.
2. Concepts \( \tilde{P} \) and \( P^i (i=1,\ldots,n) \) are non-dominated concepts.
3. When evaluating the total scores, the same set of weights is used for all concepts.
4. Point \( \tilde{P} \) lies inside the polyhedral formed by \( P^i (i=1,\ldots,n) \) and \( P^N \) (nadir point). Since \( \tilde{P} \) is non-dominated, it lies on a non-convex region of the Pareto frontier.
5. There are \( n+1 \) concepts under evaluation.

**Notation and Definitions:**

- The coordinates of \( P^i \) are given by 
  \[ P^i = \left[ \mu^i_1, \mu^i_2, \ldots, \mu^i_n \right]^T = \mu^i \]  
  \[ i = 1, \ldots, n \]  
  \[ (2) \]
- The coordinates of \( \tilde{P} \) are denoted by 
  \[ \tilde{P} = \left[ \tilde{\mu}_1, \tilde{\mu}_2, \ldots, \tilde{\mu}_n \right]^T = \tilde{\mu} \]  
  \[ (3) \]
- The coordinates of \( P^N \) are given by 
  \[ P^N = \left[ \mu^N_1, \mu^N_2, \ldots, \mu^N_n \right]^T = \mu^N \]  
  \[ (4) \]
- The nadir point is defined as having the minimum/worst of the coordinate values of all the points, \( P^i (i=1,\ldots,n) \) so that 
  \[ \mu^N_k = \min(\mu^i_1, \mu^i_2, \ldots, \mu^i_n) \text{ for } k = 1, \ldots, n \]  
  \[ (5) \]
- The set of weights, used to calculate the total score of the concepts, is denoted by 
  \[ w = \left[ w_1, w_2, \ldots, w_n \right]^T \]  
  \[ (6) \]
- The total score for the \( i-th \) concept, \( P^i \), based on Eq. (1), can be given by 
  \[ J^i = \sum_{j=1}^{n} w_j \mu^i_j = w^T P^i \]  
  \[ (7) \]
- The total score for the generic concept, \( \tilde{P} \), can be given by 
  \[ \tilde{J} = \sum_{j=1}^{n} w_j \tilde{\mu}_j = w^T \tilde{P} \]  
  \[ (8) \]
- The total score for the nadir point, \( P^N \), can be given by 
  \[ J^N = \sum_{j=1}^{n} w_j \mu^N_j = w^T P^N \]  
  \[ (9) \]

(The nadir point itself is not a concept, but its total score is used as a reference in Part 2 of the proof)

### 3.3 Proving the Proposition

To prove the Proposition stated above we show that we must always have 
\[ \tilde{J} < \max\left( J^1, J^2, \ldots, J^n \right) \]  
\[ (10) \]
We present the proof in three parts. In Part 1, we develop a general expression for the total score of Concept \( \tilde{P} \) in terms of the total scores of concepts \( P^i (i=1,\ldots,n) \) and point \( P^N \). In Part 2, we show that the total score of Point \( P^N \) is always smaller than the total score for any concept \( P^i (i=1,\ldots,n) \). In Part 3, we use the result obtained from Part 2 to show that the total score for Concept \( \tilde{P} \) is always
smaller than the highest of the total scores for Concepts $P^i (i = 1, \ldots, n)$. That is, we show that Eq. (10) is indeed true.

**Proof Part 1 – Expression for total score of $\tilde{P}$**

We develop a general expression for the total score of concept $\tilde{P}$ in terms of the expressions for the total scores of concepts $P^i (i = 1, \ldots, n)$ and point $P^N$.

In two dimensions, we let point $\tilde{P}$ lie inside the triangle formed by the points $P^1$, $P^2$ and $P^N$ (see Fig. 2). The coordinates of this point are given by

$$\tilde{P} = \alpha_1 P^1 + \alpha_2 P^2 + \alpha_N P^N$$

(11)

Extending this to $n$ dimensions, we obtain

$$\tilde{P} = \alpha_1 P^1 + \alpha_2 P^2 + \ldots + \alpha_n P^n + \alpha_N P^N$$

(12)

where $\alpha_1, \alpha_2, \ldots, \alpha_n$ and $\alpha_N$ are positive numbers with the relationship

$$\left(\sum_{i=1}^{n} \alpha_i\right) + \alpha_N = 1$$

(13)

We also impose Eq. (14) in order to restrict our discussion to only those points that lie inside the polyhedral, formed by concepts $P^i (i = 1, \ldots, n)$ and point $P^N$, while the points that lie on the boundary of the polyhedral are not considered.

$$\left(\sum_{i=1}^{n} \alpha_i\right) \neq 1$$

(14)

For simplicity, Eq. (12) can be written as

$$\tilde{P} = \overline{P} \alpha$$

(15)

where

$$\overline{P} = \begin{bmatrix} P^1 & \ldots & P^n & P^N \end{bmatrix}$$

(16)

$$\alpha = \begin{bmatrix} \alpha_1 & \ldots & \alpha_n & \alpha_N \end{bmatrix}^T$$

(17)

In Eq. (16), $\overline{P}$ is a matrix in which the $i$-th generic column consists of the ratings associated with the $i$-th generic concept. From Eqs. (2) and (6), and from the expression for the total score given in Eq. (7), we recall that the total score for any concept $P^i (i = 1, \ldots, n)$ is given by

$$J^i = w^T P^i$$

(18)

Let

$$\overline{J} = \begin{bmatrix} J^1 & \ldots & J^n & J^N \end{bmatrix}$$

(19)

Combining Eqs. (7) and (16), we obtain

$$\overline{J} = w^T \overline{P}$$

(20)

Thus, using Eqs. (15) and (20), the total score for $\tilde{P}$ is given as

$$\tilde{J} = w^T \tilde{P} = w^T \overline{P} \alpha = \overline{J} \alpha$$

(21)

Thus,

$$\tilde{J} = J^1 \alpha_1 + J^2 \alpha_2 + \ldots + J^n \alpha_n + J^N \alpha_N$$

(22)

This expression gives the total score for a general concept $\tilde{P}$ in terms of the total scores of $P^i (i = 1, \ldots, n)$, and $P^N$. This relationship is used in Part 3 of the proof.

**Proof Part 2 – Total score of nadir point is the smallest**

Here, we show that the total score for $P^N$ is always smaller than the total scores for concepts $P^i (i = 1, \ldots, n)$. In other words, we show that $J^N < J^i$ where $i = 1, \ldots, n$. This part of the proof is used in the development of Part 3 of the proof.

Let us assume that, of $P^i (i = 1, \ldots, n)$, the $s$-th concept, $P^s$, has the minimum total score. We then show that the total score of concept $P^N$ is less than this minimum value. That is, we show that $J^N < J^s = \min_{i} (J^i)$, where $i \in (1, \ldots, n)$, and the quantity $J^s$ denotes the total score of concept $P^s$.

From Eq. (5) and Eq. (9), we can write the expression for the total score of the nadir point as

$$J^N = \sum_{j=1}^{m} w_j \min(\mu_j^i)$$

(23)

for $i = 1, \ldots, n$. Consider the case where $J^s$ is, as much as possible, smaller than the total scores of all other concepts. This will happen when

$$\mu_j^i = \min(\mu_j^1, \mu_j^2, \ldots, \mu_j^n) \text{ for all } j$$

(24)

Interestingly, Equation (24) is the definition of the nadir point (see Eq. (5)). In other words, concept $P^s$ and the nadir point have the same coordinates in the design space. As a result,

$$J^s = J^N$$

(25)

This also means that concept $P^s$ is dominated by all other concepts $P^i (i = 1, \ldots, n)$. However, this violates our assumption (Assumption 2) that all concepts are non-dominated. Therefore, for $J^s$ to be minimum among all the concepts under consideration and also satisfy Assumption 2, we must have $\mu_j^i > \min(\mu_j^1, \mu_j^2, \ldots, \mu_j^n)$ for all $j$. This leads to the conclusion that

$$J^N < J^s$$

(26)

Therefore, $J^N < \min_{i} (J^i), i = 1, \ldots, n$, which means,

$$J^N < J^i \quad \text{for } i = 1, \ldots, n$$

(27)
We have shown that point $P^N$ has the smallest total score when compared with the total scores of concepts $P^i$ $(i = 1, n)$. 

Proof Part 3 – Total score of $\tilde{P}$ is never the highest

Here we make use of the expression for the total score of concept $P$, obtained in Part 1 of the proof (Eq. (22)) and show that it is always smaller than the maximum total score for the concepts $P^i$ $(i = 1, n)$. For convenience, we rewrite Eq. (22) as

$$\tilde{J} = J^i \alpha^i + J^2 \alpha^2 + \ldots + J^n \alpha^n + J^N \alpha^N$$

In Part 2 we showed that $J^N$ is the minimum of all $J$ (see Eq. (27)). The important point here is that, as a result, $J^N$ can never be the maximum. We let $J^k = \max(J^1, J^2, \ldots, J^n)$. Then all the total scores can be represented as $J^i = J^k - p^k$, $i = 1, n$, where

$$p^k = \begin{cases} 0, & \text{if } i = k \\ > 0, & \text{if } i \neq k \end{cases}$$

Also, from Part 2, $J^N = J^k - p^N$ where $p^N > 0$. Substituting these values in the expression for $\tilde{J}$ (Eq. (28)), we obtain

$$\tilde{J} = \sum_{i=1}^{n} \alpha^i (J^i - p^N) + \alpha^N (J^k - p^N)$$

From Eq. (13), $\sum_{i=1}^{n} \alpha^i + \alpha^N = 1$. Therefore,

$$\tilde{J} = J^k - \left( \sum_{i=1}^{n} \alpha^i p^i + \alpha^N p^N \right)$$

or

$$\tilde{J} = J^k - R$$

where

$$R = \sum_{i=1}^{n} \alpha^i p^i + \alpha^N p^N + \alpha^k p^k$$

where $R \geq 0$, since all $\alpha^i$s and $p^i$s are greater than or equal to zero. Note that, in Eq. (34), the summation of $\alpha^i p^i$ is from $i = 1, n$ except $i = k$. The term $\alpha^k p^k$ is separated from the summation for reasons discussed in the following.

Consider the case where $R = 0$. This is possible only if all products in Eq. (34) are identically zero, since all $\alpha^i$s and $p^i$s are greater than or equal to zero. Also from Eq. (29), $p^k = 0; p^i (i \neq k) > 0; p^N > 0$. Therefore, $R = 0$ only if the following relations are true.

(i) $\alpha^i = 0$, for all $i$ except $i = k$

(ii) $\alpha^N = 0$.

Using this information in Eq. (13), we can conclude that $\alpha^i = 1$ with $\sum_{i=1}^{n} \alpha^i = 1$. However, this is contradictory to the condition of Eq. (14), as we consider only those points that lie inside the polyhedral formed by points $P^i$ $(i = 1, n)$ and $P^N$. Therefore, we must have $R > 0$. Using this information in Eq. (33), we conclude that $\tilde{J} < J^k$. As we assumed $J^k$ to be the maximum of all the $J^i$s, we can conclude that

$$\tilde{J} < \max(J^1, J^2, \ldots, J^n)$$

Thus, we have shown that that a concept that lies on the non-convex region of the Pareto frontier will never receive the maximum score. In the following section, we discuss some of the alternatives to the typical construction of decision matrices, which can help overcome its deficiencies. Important related issues are discussed in Refs. (9), (10), and (12).

4. Alternatives to the typical construction of decision matrices

In this section, we consider possible modifications that can be made to the typical construction of the decision matrix, so that the pitfalls discussed in previous sections can be avoided. Specifically, we modify the approach for obtaining the total score for each concept. As discussed previously, decision matrices are typically formed such that the total score is a weighted sum. In this section, we consider compromise programming9,16 as an alternative to the weighted sum approach.

It has been discussed in the literature9 that compromise programming overcomes some of the deficiencies of the weighted sum method, in that it can yield solutions that lie on non-convex regions of the Pareto frontier. Accordingly, a seemingly logical extension to the weighted sum approach for concept selection is the compromise programming method represented as

$$\max J^i = \sum_{j=1}^{n} w_j (\mu_j^i)^m$$

or

$$\min J^i = \sum_{j=1}^{n} w_j (\mu_j^i)^m$$

where $m$ is an even integer greater than or equal to 2. When $m = 2$, the compromise programming approach is
also known as the weighted square sum method. Equation (36) is an expression for the total score of a concept using the compromise programming method. This construction can be compared to the formulation shown in Eq. (1), where a weighted sum approach is used. The difference between Eq. (1) and Eq. (36a) is that in Eq. (36a) the ratings are raised to the power $m$.

Table 3 Compromise programming effectiveness for minimization case

<table>
<thead>
<tr>
<th>Order</th>
<th>Convex</th>
<th>Non-convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>2</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>$m$</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>$\infty$</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

Table 4 Compromise programming effectiveness for maximization case

<table>
<thead>
<tr>
<th>Order</th>
<th>Convex</th>
<th>Non-convex</th>
<th>Convex</th>
<th>Non-convex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>2</td>
<td>?</td>
<td>N</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>?</td>
<td>N</td>
<td>Y</td>
<td>?</td>
</tr>
<tr>
<td>$m$</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>$\infty$</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

Note:
‘Y’: works – yields solution
‘N’: does not work – does not yield solution
‘?’: depends on the order of the AOF and convexity of Pareto Frontier

We note that in general, decision matrices are used in the context of maximization – maximization of performance measures. With this note in mind, we will nevertheless explore whether the pitfalls of the conventional decision matrix method can be avoided by using minimization, rather than maximization. We also recall that the total score of a concept is also the value of the Aggregate Objective Function (AOF). The construction of the Aggregate Objective Function shown in Eq. (36a) is such that the contours of $J$, when plotted in a two dimensional design space ($n = 2$), are centered at the origin. If the compromise programming method is used as shown in Eq. (36a) (see Fig. 4a), then an important observation can be made: Regardless of the value of $m$, compromise programming does not generally yield solutions on non-convex regions of the Pareto frontier (see Table 4 (columns 2 and 3), and Fig. 4a). This is in stark contrast to the minimization problem, where compromise programming yields solutions on the non-convex regions of the Pareto frontier. If we replace the weighted sum in Eq. (1) by a compromise programming formulation, then the concepts that lie on the non-convex regions of the Pareto frontier will continue to appear undesirable. Thus, using the traditional compromise programming approach will not avoid the pitfalls of the decision matrix based methods.

Figure 4 (a) Maximization about origin, (b) Maximization about utopia point

At this point it is helpful to examine the differences between the two basic optimization procedures; namely, maximization and minimization. Tables 3 and 4 help depict these differences. We consider progressively larger values of the order, $m$, of the Aggregate Objective Function (AOF). We then compare the two optimization procedures in terms of their ability (or lack thereof) to yield solutions on the non-convex regions of the Pareto frontier, using compromise programming. Table 3 shows the effectiveness of compromise programming to yield solutions for the
minimization case, for various orders of the AOF, and Table 4 shows the effectiveness for the maximization case.

Figure 5 shows the contours of the aggregate objective function of a minimization problem. Here, we wish to minimize the design objectives $\mu_1$ and $\mu_2$. A smaller value of the AOF ($J$) is preferred. The contours are centered at the origin or about the utopia point. We note that similar results are achieved if the contours are centered at various other points. However, an exhaustive discussion of these possibilities would be of little use to our discussion. We focus primarily on two points: the origin and the utopia point. For a minimization or maximization about the utopia point, we use the AOF given by

$$\max J^i = \sum_{j=1}^{n} w_j (\mu^*_j - \mu_j^i)^m \quad (37a)$$

$$\min J^i = \sum_{j=1}^{n} w_j (\mu^*_j - \mu_j^i)^m \quad (37b)$$

where, $\mu_j^*$ are the coordinates of the utopia point and $m \geq 2$. We refer to this formulation as the modified compromise programming approach to concept selection.

The AOF that results in the contours shown in Fig. 5 is given by Eqs. (36b) or (37b). Points $P^1$, $P^2$, and $P^3$ form a Pareto frontier, such that $P^1$ lies on a non-convex region. By examining Fig. 5 (minimization), it can be seen that compromise programming can yield solutions on non-convex regions of the Pareto frontier, depending on the order of the AOF, $m$. Table 3 provides important information regarding the capabilities of the minimization approach. For details about the relationship between Pareto frontier orders and the values of $m$, see Ref. 9 where this general problem is comprehensively studied.

![Image](https://via.placeholder.com/150)

**Figure 5** Minimization about utopia point, or origin

Figure 4(a) shows the contours of a problem in which we wish to maximize the design objectives $\mu_1$ and $\mu_2$. In this case, a higher value of the AOF ($J$) is preferred. The contours are centered at the origin (Eq. (36a)). From Table 4, we see that in the third column (maximization about the origin) all the entries are ‘N’ for non-convex regions. That is, compromise programming will not yield solutions on the non-convex region of the Pareto frontier for any value of $m$ in Eq. (36). We can see from Fig. 4(a), that the maximization will not stop at point $P^i$, but will continue until it reaches either point $P^1$ or $P^2$.

To be able to maximize (as in concept selection, where a concept with higher ratings is preferred) as well as to obtain points on non-convex regions of the Pareto frontier, we need to use the modified compromise programming approach to concept selection.

It is important to note that the value of the contour that is closer to the utopia point (Fig. 4b) is smaller; and the contour values increase as the contours move farther away from the utopia point. Equation (37b) gives the expression for obtaining the total score of a concept with the modified approach. However, in this case, the concept with the smallest total score is preferred. This is because such a concept will satisfy the primary objective, which is to maximize the criteria (ratings). Figure 4(b) shows the contours of $J$ for the formulation shown in Eq. (37b). As we can see, if a point such as $P$ lies on the non-convex region of the Pareto frontier, it is possible to obtain that point as a solution, if the contours are centered at the utopia point. Table 4 provides more details in this regard.

The important difference between the formulations presented in Eq. (36a) and Eq. (37b) for the maximization case, is that, with Eq. (36a), the most desirable score for a concept is the highest score, while with Eq. (37b), the most desirable score for a concept is the lowest.

We now conclude that if compromise programming is to be used, then the formulation given in Eq. (37b) needs to be used in order to ensure that we can obtain solutions on non-convex regions of the Pareto frontier. In other words, non-dominated concepts that lie on non-convex regions of the Pareto frontier will have an opportunity of being selected.

We discuss a concept selection problem in the next section, which will (i) illustrate the risks associated with using the weighted sum approach, and (ii) demonstrate how the modified approach, involving the use of compromise programming, can be effective under certain circumstances.
5. Concept selection example

In this section, an example is given wherein the selection of an aircraft is carried out. Table 5 contains data for each proposed aircraft. Data for the first four aircraft is from Refs. [17-19]. An additional concept, Aircraft A, is added to better illustrate our approach.

An airline carrier is planning to purchase an aircraft for its operations. Four types of aircraft have been identified as potential alternatives. The criteria that are used to compare the aircraft are: (1) passenger capacity, (2) cruise range, and (3) cruise speed. A decision matrix based concept selection approach is used.

To begin the selection process, we remove all dominated concepts from the proposed set using a global Pareto filter. Aircraft A330-200 is dominated and is therefore eliminated from further consideration. Table 6 shows the ratings given to the above aircraft. The ratings are based on a five point scale, where a higher value reflects a better performance.

Table 5 Data for different aircraft

<table>
<thead>
<tr>
<th>Aircraft</th>
<th>Criteria</th>
<th>B777-200</th>
<th>B747-200</th>
<th>A330-200</th>
<th>A340-200</th>
<th>Aircraft A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td></td>
<td>0.84</td>
<td>0.85</td>
<td>0.85</td>
<td>0.86</td>
<td>0.855</td>
</tr>
<tr>
<td>Cruise range</td>
<td></td>
<td>8820</td>
<td>6900</td>
<td>6650</td>
<td>8000</td>
<td>7500</td>
</tr>
<tr>
<td>No. of passengers</td>
<td></td>
<td>301</td>
<td>366</td>
<td>253</td>
<td>239</td>
<td>300</td>
</tr>
</tbody>
</table>

The ratings in Table 5 are such that no concept dominates or is dominated by any other concept. In other words, all the concepts are globally Pareto optimal. Also, the rating structure shown in Table 6 is such that the concept represented by Aircraft A lies on a non-convex region of the Pareto frontier.

Table 6 Ratings for proposed aircraft concepts

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Wt.</th>
<th>B777-200</th>
<th>B747-200</th>
<th>A340-200</th>
<th>Aircraft A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>$w_1$</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Cruise range</td>
<td>$w_2$</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>No. of passengers</td>
<td>$w_3$</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Total Score</td>
<td></td>
<td>$J^1$</td>
<td>$J^2$</td>
<td>$J^3$</td>
<td>$J^4$</td>
</tr>
</tbody>
</table>

We evaluate the above decision matrix using three different methods for obtaining the total score, namely: the weighted sum (WS), weighted square sum (WSS) ($m = 2$), and compromise programming (CP). For each approach, we vary the weights $w_1$, $w_2$, and $w_3$ uniformly between 0 and 1 with an increment of 0.1, in order to ascribe each criterion a different relative importance. This process is carried out to evaluate the probabilities of selecting the different aircraft. For the compromise programming approach, the total score is evaluated using the two different formulations presented in Eq. (36a) and Eq. (37b). With the formulation given in Eq. (36a) a higher total score for a concept is more desirable, while with the formulation in Eq. (37b), a lower score is more desirable.

The aircraft that receives the most desirable total score is selected each time. The probability $H^i$ that the $i$-th concept receives the most desirable score is calculated as given in Eq. (38), based on the total number of selections, $s$, and the number of times, $c^i$, the $i$-th concept receives the most desirable total score.

$$H^i = \frac{c^i}{s}$$  \hspace{1cm} (38)

The results are given in Table 7 and Table 8.

Table 7 shows the results using the formulation given in Eq. (31) for the WS method, and Eq. (36) for the compromise programming method. It can be seen that the probability of Aircraft A receiving the highest score is zero, with all methods. Therefore, it is not likely that Aircraft A will be chosen, not only when using the weighted sum method, but also when using the compromise programming method. This exposes a potential risk of using the decision matrix methods for concept selection. Aircraft A is not a dominated concept, but it may appear undesirable to a designer, because it lies on a non-convex region of the Pareto frontier. The decision matrix gives a false representation of the desirability of Aircraft A to the designer, which might lead him/her decide that Aircraft A is undesirable because it never receives the highest score, whereas in reality, it might even be the most suitable choice. In practice one would choose only one set of weights to evaluate the decision matrix and make a decision based on the evaluation. The results in Table 7 show that regardless of which set of weights one uses, Aircraft A will never receive the most desirable score.

We now repeat the same process of evaluating the probabilities of selecting different concepts. We use the formulation as given in Eq. (37b), which represents a compromise programming formulation with the contours centered at the utopia point instead of the origin (Fig. 4(b)). The results are shown in Table 8.
Table 7 Probability of concepts receiving most desirable total score with contours centered at origin

<table>
<thead>
<tr>
<th>Method</th>
<th>B777-200</th>
<th>B747-200</th>
<th>A340-200</th>
<th>Aircraft A</th>
</tr>
</thead>
<tbody>
<tr>
<td>WS</td>
<td>0.33</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
</tr>
<tr>
<td>WSS</td>
<td>0.33</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
</tr>
<tr>
<td>CP($m = 4$)</td>
<td>0.33</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 8 shows that the probability of Aircraft A obtaining the most desirable total score has increased (for the WSS and CP methods). This would not have been the case if we had used the formulation in Eq. (36a), where the contours are centered at the origin (Fig. 4(a)). Interestingly, using the formulation as shown in Eq. (37b), the probability of Aircraft A receiving the most desirable total score is higher than any other aircraft with the compromise programming approach ($m = 4$).

Table 8 Probability of concepts receiving most desirable total score with contours centered at utopia point

<table>
<thead>
<tr>
<th>Method</th>
<th>B777-200</th>
<th>B747-200</th>
<th>A340-200</th>
<th>Aircraft A</th>
</tr>
</thead>
<tbody>
<tr>
<td>WS</td>
<td>0.33</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
</tr>
<tr>
<td>WSS</td>
<td>0.33</td>
<td>0.29</td>
<td>0.29</td>
<td>0.09</td>
</tr>
<tr>
<td>CP($m = 4$)</td>
<td>0.23</td>
<td>0.14</td>
<td>0.13</td>
<td>0.49</td>
</tr>
</tbody>
</table>

6. Conclusion

In this paper we have shown that methods based on decision matrices pose a certain risk if used in their typical form. The risk is that the decision matrix may present certain concepts as less-desirable when in fact they may be preferred by the designer. More specifically, the decision matrix is unable to give some concepts the most desirable score, and therefore these concepts are not likely to be selected. Importantly, it was also shown that the traditional compromise programming approach is unable to ascribe to some concepts the most desirable score. We showed how a modified compromise programming method can be used to allow concepts in non-dominated regions to receive the most desirable score. Using the proposed modified compromise programming approach, the risks discussed in this paper are avoided.

Acknowledgments

This research was partially supported by the National Science Foundation Grant number DMI-0196243.

References


