Pergamon

Mathl. Comput. Modelling Vol. 20, No. 3, pp. 107–124, 1994

0895-7177/94 $7.00 + 0.00

0895-7177(94)00117-0

Design Decision Making Based upon Multiple Attribute Evaluations and Minimal Preference Information

P. Sen and J.-B. Yang
Engineering Design Centre, University of Newcastle upon Tyne
Newcastle upon Tyne NE1 7RU, United Kingdom

(Received November 1993; accepted December 1993)

Abstract—In this paper, a design decision problem is treated as a multiple criteria decision making (MCDM) problem. Design selection is based upon multiple attribute evaluations for candidate designs and preference judgments on relative importance of attributes. It is intuitively clear that flexibility and a systematic procedure are important features of a technique for acquisition and representation of preference information. This paper is intended to explore a new technique for assigning weights to attributes through a well-defined iterative procedure using minimal preference information. A semi-submersible design problem is then taken as an example to demonstrate how multiple attribute evaluations for candidate designs can be generated and represented and how relative weights of attributes can be assigned using the new weight assignment technique for the ranking of the generated candidate designs.

Keywords—Multiple criteria decision making, Preference modeling, Weights, Design selection and linear programming.

1. INTRODUCTION

Design selection may be regarded as a decision making process for choosing the most acceptable design from a finite set of candidate designs. Design acceptability is nearly always measured in terms of multiple attributes, such as cost, weight, reliability, safety, and other domain specific technical performance indices, some or all of which may be in conflict in the sense that attractive values of some attributes go hand in hand with poor values for some of the other attributes.

As a basis for making design decisions, a candidate design needs to be separately evaluated by each attribute, either numerically or subjectively. To deal with a design selection problem with a finite number of candidate designs and with multiple conflicting attributes, it is necessary to synthesize the obtained basic evaluations of a design in terms of each attribute, so that any given design can be ranked and scored with respect to the other designs. This paper presents one approach for dealing with this problem.

It is because of the inherent conflict among some or all of the attributes that there generally exists no single feasible candidate design which could simultaneously attain the best values for all the attributes. Otherwise, such a design would naturally be the best and there would be no point in taking multiple attributes into account for selecting a design. Consequently, the preference

This work forms part of a project to develop a general purpose multiple criteria decision support environment supported by the UK Science and Engineering Research Council under Grant No. GR/F 95306.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TeX information of the designer (or decision maker, simply DM) regarding the relative importance of attributes has to be considered for trade-off analysis and it actually constitutes another basis for making design decisions.

Preferences may be represented by relative weights for attributes. Over the years quite a few weight assignment techniques have been proposed. The well-known eigenvector method [1,2] is a widely used technique for weight assignment, which uses pairwise comparisons of attributes to acquire preference information. The method requires exact comparisons for all pairs of attributes. However, it has been realized that many of the required comparisons may be redundant [3] although the redundant information could be used to check the consistency of the preferences. The geometric least square method presented in [3] suggests that much less data than the eigenvector method requires may be sufficient for weight assignment although it provides no bound on how much data would be required to satisfy the DM.

Both methods, however, require exact comparisons in that the DM is required to evaluate how many times one attribute is exactly more important than another. In engineering design, the DM may only be capable of providing a combination of exact and vague pairwise comparisons. For instance, he may assert that one attribute is *at least* twice more important than another. The minimal information trade-off assessment (MITA) method presented in [4,5] can accommodate both exact and vague pairwise comparisons, so that it may be considered to be more flexible for the acquisition and representation of preference information. To assign weights, the MITA method uses only as much preference information as the DM can provide. Unfortunately, it does not either define the minimal information requirement formally or provide a systematic way to guide the DM in preparing his preference information.

This paper explores a new technique for weight assignment, which also uses exact and/or vague pairwise comparisons of attributes for preference acquisition. It adopts an iterative procedure to assign weights, which is composed of two main steps. First of all, it generates an initial weight assignment based on minimum number of complete pairwise comparisons which may represent the DM's initial overall preference structure. A linear programming model is proposed to facilitate the assignment. Then the initially assigned weights may be revised if the DM is not satisfied with them and if he can provide more useful information. In the procedure, the consistency and determinacy of the given comparisons are iteratively checked and numerically measured so that the DM can clearly judge the quality of the given preference information and the assigned weights. To implement the iterative procedure, a goal programming model is explored.

A semi-submersible design synthesis problem is then discussed, which was originally modeled as a vector nonlinear programming problem [6]. Candidate designs are generated using interactive MCDM techniques. This problem is used to demonstrate how a design synthesis problem may be dealt with using a multiple attribute decision making (MADM) method and how the new method could be implemented to assign relative weights.

2. MULTIPLE ATTRIBUTE DESIGN SELECTION

2.1. Description of Design Decision Problems

Design selection usually deals with a finite set of candidate designs and generally multiple attributes representing technical and economical performance of a design. Candidate designs may be defined explicitly or implicitly. In the latter case, certain techniques need to be used to generate candidate designs. For instance, feasible designs may be defined by a knowledge-based system and then generated using heuristics [4]. If a design problem is formulated as a vector mathematical programming problem, a finite number of efficient designs may then be generated using interactive decision making methods [6–9].

One of the simplest structures for describing a design selection problem in terms of multiple attribute evaluations may be a decision matrix in which each design is explicitly defined and numerically evaluated by attributes. Table 1 shows a decision matrix for evaluating m candidate designs in terms of n attributes, where y_j is the j^{th} attribute, a_i the i^{th} alternative design, and y_{ij} stands for the numerical value of attribute j for evaluating design i ($i = 1, \ldots, m; j = 1, \ldots, n$). The attributes in Table 1 are assumed to be quantitative. Without loss of generality, all attributes in Table 1 are assumed to be for maximization.

Table 1. The decision matrix.

Alternative designs	Attributes			
	y_1	y_2	$\cdots$	y_n
a_1	y_{11}	y_{12}	$\cdots$	y_{1n}
a_2	y_{21}	y_{22}	$\cdots$	y_{2n}
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
a_m	y_{m1}	y_{m2}	$\cdots$	y_{mn}

The decision matrix provides a basis for various MADM methods to make decision analysis. More complex design selection problems with both qualitative and quantitative attributes and with subjective judgments with uncertainty may eventually be transformed into a decision matrix in which y_j may represent an aggregated attribute and y_{ij} may only be a scaled value without any unit. If, for example, there are both quantitative and qualitative attributes in a design selection problem, the qualitative attributes may be quantified for each design by a rule-based system [10] or by means of hierarchical multiple factor analysis and reasoning with uncertain decision knowledge [11–13]. If a design space is defined by a knowledge-based system, candidate designs may be generated using heuristics and evaluated by attributes which are scored and scaled (to the range $[0 \quad 1]$, say) in accordance with expert evaluations [4]. If a design space is defined by mathematical constraints, (interactive) multiple objective decision making (MODM) methods may be used to generate efficient designs which are evaluated by objective functions [6,8,9,14].

In addition to the decision matrix, another widely used structure for describing a design selection problem is pairwise comparison matrices. In a pairwise comparison matrix, each pair of candidate designs are subjectively compared with respect to an attribute [2]. The main difference between a pairwise comparison matrix and a decision matrix may be that the former is composed of pure subjective judgments and the latter contains numerical data though subjective judgments with uncertainty may also be accommodated in a generalized decision matrix [11–13].

If one of the candidate designs attains the best values for all the attributes, it is of course the best design. Unfortunately, such a "best" design hardly ever exists in a design selection problem. So, compromise among attributes is always necessary. In other words, the "best" design is only the best compromise design which is determined by not only the multiple attribute evaluations but also the DM's preferences. Relative weights of attributes are widely used to represent the DM's preference information. The following subsection outlines a MADM method, which uses relative weights and a decision matrix to generate the best compromise design. The next section explores a new weight assignment technique based on a systematic procedure for acquisition of minimal preference information.

2.2. Solution of Design Decision Problems

Many methods for dealing with multiple attribute decision making problems have been proposed, some of which may be applied to treat design selection problems. The choice of a particular method for a design problem depends upon the data structure representing the problem. For instance, a design selection problem may be solved using the AHP method [2] if it is represented by pairwise comparison matrices [2,15]. If a problem is represented by a decision matrix, it may be dealt with using a range of techniques, dependent upon other characteristics of the problem and the DM's preferences.

The simple additive utility function method is perhaps one of the simplest for dealing with a design selection problem represented by a decision matrix [1]. Suppose $u(y(a))$ is an additive utility function, $u_i(y_i(a))$ is the marginal utility function of an attribute y_i, and the normalized weight vector of attributes is defined by

$$W = [\, w_1 \quad \dots \quad w_n \,]^{\top}, \tag{1}$$

where w_i is the relative weight of an attribute y_i. Then the utility of an alternative design a_r may be calculated by

$$u(a_r) = \sum_{i=1}^{n} w_i u_i \left(y_i(a_r) \right), \tag{2}$$

or more simply by

$$u(a_r) = \sum_{i=1}^{n} w_i y_i(a_r). \tag{3}$$

$u(a_r)$ is normally scaled to $[\, 0 \quad 1\,]$. A design a_r is then ranked based upon the value of $u(a_r)$. A larger value of $u(a_r)$ means that a_r is more favorable.

However, this simple method assumes that all attributes in the attribute set $\{\, y_1 \quad \dots \quad y_n \,\}$ are preferentially independent of one another. That is, each attribute may be evaluated regardless of the states of the other attributes. In addition to the preferential independence assumption, it is further assumed in (3) that the marginal utility function of an attribute is linear and that one attribute can be directly offset by other attributes. These three assumptions are rather strict and may not always be satisfied in practice. Hence, other techniques have also been developed which are more flexible but also more elaborate.

If the marginal utility functions of attributes in a selection problem are all monotonous, either nonincreasing or nondecreasing, for example, the CODASID method [16] may be used to treat the problem. The CODASID method is based on comprehensive concordance and discordance analyses and is composed of a well-refined information aggregation and synthesis procedure where the DM may set up a veto threshold value for each attribute. The simplified version of the procedure is composed of two main steps. First of all, the multi-attribute evaluation information, contained in the decision matrix, and the preference information, capsulated in the weights, are aggregated so that the following Judgment-Evaluation (J-E) matrix can be constructed.

Table 2. The J-E matrix.

a	$pc(a)$	$ec(a)$	$d(a)$
a_1	pc_1	ec_1	d_1
a_2	pc_2	ec_2	d_2
$\dots$	$\dots$	$\dots$	$\dots$
a_m	pc_m	ec_m	d_m

where $pc(a)$ and $ec(a)$ are two aggregated benefit attributes, $d(a)$ is an aggregated cost attribute, and

$$pc_r = pc(a_r) = \sum_{\substack{j=1 \\ j \neq r}}^{m} pc_{rj} - \sum_{\substack{j=1 \\ j \neq r}}^{m} pc_{jr}, \qquad r = 1, \dots, m, \tag{4}$$

$$ec_r = ec(a_r) = \sum_{\substack{j=1 \\ j \neq r}}^{m} ec_{rj} - \sum_{\substack{j=1 \\ j \neq r}}^{m} ec_{jr}, \qquad r = 1, \dots, m, \tag{5}$$

$$d_r = d(a_r) = \sum_{\substack{j=1 \\ j \neq r}}^{m} d_{rj} - \sum_{\substack{j=1 \\ j \neq r}}^{m} d_{jr}, \qquad r = 1, \ldots, m, \tag{6}$$

$$d_{rj} = \sum_{k \in D_{rj}} \frac{|\tilde{p}_{rk} - \tilde{p}_{jk}|}{S_d}, \tag{7}$$

$$pc_{rj} = \sum_{k \in C_{rj}} \frac{\omega_k}{S_p}, \tag{8}$$

$$ec_{rj} = \sum_{k \in C_{rj}} \frac{|\overline{p}_{rk} - \overline{p}_{jk}|}{S_e}, \tag{9}$$

$$S_p = \sum_{k=1}^{n} \omega_k, \quad S_e = \sum_{k=1}^{n} \max_{r,j \in R_a} \{|\overline{p}_{rk} - \overline{p}_{jk}|\}, \quad S_d = \sum_{k=1}^{n} \max_{r,j \in R_a} \{|\tilde{p}_{rk} - \tilde{p}_{jk}|\},$$

$$0 \leq pc_{rj}, ec_{rj}, d_{rj} \leq 1 \quad \text{and} \quad R_a = \{1, \ldots, m\},$$

$$C_{rj} = \{k \mid p_{rk} \geq p_{jk}, k = 1, \ldots, n\}; \quad D_{rj} = \{k \mid p_{rk} < p_{jk}, k = 1, \ldots, n\}, \tag{10}$$

$$\tilde{p}_{rj} = w_j \overline{p}_{rj}; \quad \overline{p}_{rj} = \frac{y_{rj} - y_j^{\min}}{y_j^{\max} - y_j^{\min}}, \qquad r = 1, \ldots, m; \quad j = 1, \ldots, n, \tag{11}$$

$$y_j^{\max} = \max\{y_{1j} \quad y_{2j} \quad \cdots \quad y_{mj}\}; \quad y_j^{\min} = \min\{y_{1j} \quad y_{2j} \quad \cdots \quad y_{mj}\}. \tag{12}$$

Then, synthesize the information contained in the J-E matrix into the following relative closeness indices $u(a_r)$, $r = 1, \ldots, m$

$$u(a_r) = \frac{s_r^-}{s_r^- + s_r^*} \qquad 0 \leq u(a_r) \leq 1, \quad r = 1, \ldots, m; \quad u(a^-) = 0, \quad u(a^*) = 1, \tag{13}$$

where

$$s_r^* = \sqrt{(\tilde{pc}(a_r) - \tilde{pc}(a^*))^2 + (\tilde{ec}(a_r) - \tilde{ec}(a^*))^2 + \left(\tilde{d}(a_r) - \tilde{d}(a^*)\right)^2}, \quad r = 1, \ldots, m, \tag{14}$$

$$s_r^- = \sqrt{(\tilde{pc}(a_r) - \tilde{pc}(a^-))^2 + (\tilde{ec}(a_r) - \tilde{ec}(a^-))^2 + \left(\tilde{d}(a_r) - \tilde{d}(a^-)\right)^2}, \quad r = 1, \ldots, m, \tag{15}$$

$$\tilde{pc}(a^*) = \max\{\tilde{pc}(a_1) \quad \tilde{pc}(a_2) \quad \cdots \quad \tilde{pc}(a_m)\}, \tag{16.1}$$

$$\tilde{ec}(a^*) = \max\{\tilde{ec}(a_1) \quad \tilde{ec}(a_2) \quad \cdots \quad \tilde{ec}(a_m)\}, \tag{16.2}$$

$$\tilde{d}(a^*) = \min\{\tilde{d}(a_1) \quad \tilde{d}(a_2) \quad \cdots \quad \tilde{d}(a_m)\}, \tag{16.3}$$

$$\tilde{pc}(a^-) = \min\{\tilde{pc}(a_1) \quad \tilde{pc}(a_2) \quad \cdots \quad \tilde{pc}(a_m)\}, \tag{17.1}$$

$$\tilde{ec}(a^-) = \min\{\tilde{ec}(a_1) \quad \tilde{ec}(a_2) \quad \cdots \quad \tilde{ec}(a_m)\}, \tag{17.2}$$

$$\tilde{d}(a^-) = \max\{\tilde{d}(a_1) \quad \tilde{d}(a_2) \quad \cdots \quad \tilde{d}(a_m)\}, \tag{17.3}$$

$$\tilde{pc}(a_r) = \rho_1 \overline{pc}(a_r); \quad \tilde{ec}(a_r) = \rho_2 \overline{ec}(a_r); \quad \tilde{d}(a_r) = \rho_3 \overline{d}(a_i), \qquad r = 1, \ldots, m, \tag{18}$$

$$\rho_1 = \rho_2 = 0.25; \quad \rho_3 = 0.5, \tag{19}$$

$$\overline{pc}(a_r) = \frac{pc(a_r)}{\sqrt{\sum_{j=1}^{m} pc^2(a_j)}}; \quad \overline{ec}(a_r) = \frac{ec(a_r)}{\sqrt{\sum_{j=1}^{m} ec^2(a_j)}}; \quad \overline{d}(a_r) = \frac{d(a_r)}{\sqrt{\sum_{j=1}^{m} d^2(a_j)}}, \quad r = 1, \ldots, m. \tag{20}$$

Obviously, $u(a_r) \in [0 \quad 1]$. A large value of $u(a_r)$ indicates that a_r is more favorable as it is simultaneously closer to the ideal point s_r^* and further from the negative ideal point s_r^- in the J-E space.

3. A PREFERENCE MODELING PROCEDURE USING MINIMAL INFORMATION

3.1. Review of Some Existing Procedures

It may be noted that the relative weights of attributes play an essential role in preference acquisition and representation in many MADM methods. Several techniques have been developed for weight assignment, such as the simple direct assignment method [1], the eigenvector method [2], the geometric least square method [3] and the MITA method specific for engineering design [4,5]. In this subsection, a few weight assignment techniques are reviewed to place the new method to be developed in context.

The eigenvector method is a widely used tool for assigning weights. In this method, the DM is supposed to judge the relative importance of one attribute over another. Suppose a_{ij} is the pairwise comparison of the i^{th} attribute and the j^{th} attribute, reflecting the relative importance of attribute i over attribute j. In general, a_{ij} is estimated by the DM, based on certain standards for pairwise comparison [2,9,15], and a_{ij} may not necessarily be equal to w_i/w_j where w_i is the weight of y_i as defined in (1). If we define a judgment matrix as $A = (a_{ij})_{n\times n}$, $\lambda_{\max}$ as the largest eigenvalue of A and W' as the normalized eigenvector of A with respect to $\lambda_{\max}$, then

$$AW' = \lambda_{\max} W'. \tag{21}$$

$W' = [\, w_1' \quad \dots \quad w_n' \,]^{\mathrm{T}}$ can be obtained by solving (21) and used as the approximation of W [1,2].

In the eigenvector method, the judgment matrix is assumed to be reciprocal, i.e., $a_{ij} = 1/a_{ji}$, and $n(n-1)/2$ pairwise comparisons have to be made. It is argued that much of the data may be redundant [3], and this can lead to inconsistency. The Geometric Least Square (GLS) method requires much less data [3]. For design selection, a weight assignment technique may be more favorable if it only requires the DM to provide as much significant preference information as necessary through a systematic procedure.

The pairwise comparisons required by the eigenvector and the GLS methods are all exact ones, that is, the DM is required to evaluate how many times one attribute is exactly more important than another. The DM may not always prefer to or be able to provide such exact comparisons. A combination of exact and vague but practical pairwise comparisons may be the best that can be provided. Suppose "*COST*" and "*FLEXIBILITY*" are two attributes, for example, the DM may describe the relative importance of the attributes using the following statements which are either exact or vague [4],

(i) *COST is the most important attribute (vague);*
(ii) *it is better for the OPERATING COST to remain low at the expense of a higher INITIAL COST (vague);*
(iii) *COST is twice as important as FLEXIBILITY (exact);*
(iv) *high FLEXIBILITY and low COST are equally important (exact); and*
(v) *COST is at least as important as FLEXIBILITY (vague).*

In the minimal information trade-off assessment (MITA) method [4,5], set inclusion is used to define the information represented by these statements. Actually, these preference statements provide information which may be transformed into linear equality or inequality constraints on the weights. The transformation is based on the assumption that the statement $y_i R_{ij} y_j$ for attributes y_i, y_j, and preference relation R_{ij} holds if and only if $c_i w_i \Delta_r c_j w_j$ for some real numbers

c_i and c_j and $\Delta_r \in \{<, >, =, \leq, \geq\}$. Under this assumption, the "more important than" relation for attributes is a "greater than" relation on the weights; "equivalence" relation for attributes is an "equality" relation on the weights, and the "at least as important as" relation for attributes is a "greater than or equal to" relation on the weights. The "c times more important than" relation for attributes is a "greater than" relation on the weights where the weight of the less important attribute is multiplied by c. This transformation provides a set of mappings from preference statements into weight constraints.

The MITA method searches for a specific value for the weight vector W as the solution to the following mathematical programming problem

$$\min H(W) = \sum_{i=1}^{n} w_i \log \frac{w_i}{w_i'},$$
$$\text{subject to } W \in \Lambda \quad \sum_{i=1}^{n} w_i = 1, \tag{22}$$

where Λ is the feasible space defined by the weight constraints, in which W must lie. $W' = [w_1' \ \dots \ w_n']^{\mathrm{T}}$ is *a prior* preference structure or a uniform preference structure if no prior structure exists. The above model as defined by (22) is a nonlinear programming problem with a nonlinear objective function and generally linear constraints. The objective function $H(W)$ in (22) is the relative entropy, measuring the amount of information given by the preference structure W relative to W'. It is argued that relative entropy satisfies some inferential properties [4]. For instance, if no preference information is provided by the DM (i.e., Λ is empty), equal weight on each attribute may be used by the experts, that is, W is given by $w_i = 1/n, i = 1, \dots, n$. If a constraint on W is given, for example, $w_j \geq 2w_i$ for $i \neq j$, then $w_j = 2/(n+1)$ and $w_i = 1/(n+1)$ for $i = 1, \dots, n$ and $i \neq j$ may be used.

Obviously, exact pairwise comparisons for attributes required by the AHP and the GLS methods can be transformed into linear equality constraints on weights. Thus, the MITA method provides greater flexibility in acquiring and representing preference information as constraints on weights in Λ may be exact or vague. This flexibility may be appreciated by the DM who prefers to provide a combination of exact and vague comparisons. In the MITA method, however, no systematic procedure is designed to acquire preference statements from the DM. In other words, the DM may be allowed to provide preference statements for attributes in a random manner. It may be better if the DM can follow a systematic yet flexible way to present preference statements, in which the consistency of the statements can be checked. Moreover, when the statements are transformed into constraints on weights which constitute Λ, the determinacy of Λ should be measured as well. It is to address the aforementioned considerations that a new weight assignment technique will be explored in this section. Problems such as consistency and determinacy will be explored so that the DM can check and may improve the consistency and determinacy of his preference statements.

3.2. Minimal Preference Information

Pairwise comparisons (exact or vague) for attributes are easy to understand and thus are quite likely to be available. In the proposed method, preference information is also acquired through pairwise comparisons. The new method uses a systematic procedure to acquire and represent preference information, such that relative weights for attributes can be initially assigned on the basis of a comparison set composed of minimal number of exact and/or vague complete pairwise comparisons. Then the minimum comparison set may be revised or extended if the DM is not satisfied with the initial weight assignment and if he can provide more information. The new method is therefore called the MInimal PAirwise Comparison (MIPAC) method. By minimum and complete, we mean that the number of pairwise comparisons for attributes is minimized under the restrictions that each attribute is at least compared with one other attribute and that all comparisons are connected together either directly or indirectly. Thus, the minimum set of complete pairwise comparisons may to some extent reflect the DM's initial overall preference structure about the relative importance of attributes. The minimum set of complete comparisons may be defined as follows.

DEFINITION 1. *Suppose there are n attributes. The minimum set of complete pairwise comparisons for the n attributes is composed of $(n - 1)$ pairwise comparisons, in which each of the n attributes must be compared with at least one of the other attributes and no single comparison or a subset of the comparisons is isolated from the other comparisons.*

Let y_i and y_j be two attributes and R_{ij} the preference relation for y_i and y_j. Then, Definition 1 may be interpreted with the help of a comparison diagram. Suppose a circle marked by y_i represents a node. If y_i and y_j are directly compared, then the node for y_i and that for y_j are linked together by a line segment marked by R_{ij}. Thus, a direct comparison for y_i and y_j may be depicted as in Figure 1. The set of pairwise comparisons for n attributes may then be represented using a network, composed of n nodes and many line segments linking these nodes.

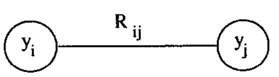

Figure 1. Direct comparison.

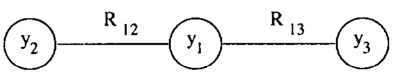

Figure 2. Basic comparison set.

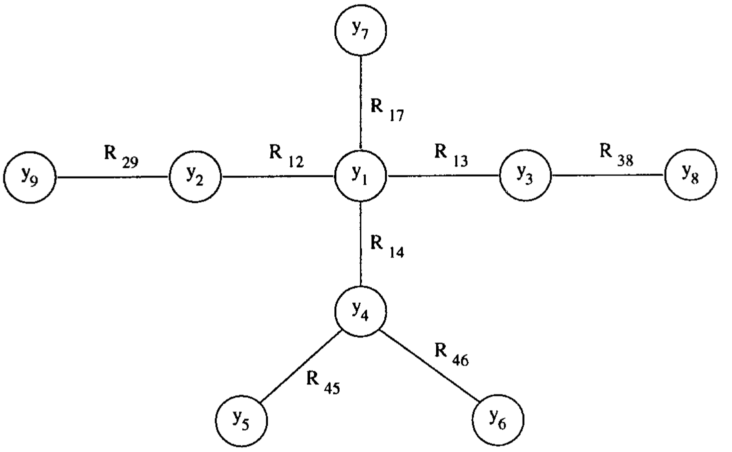

Figure 3. Complex comparison set.

The minimum set of complete pairwise comparisons for n attributes is composed of n nodes and $(n - 1)$ line segments. Each node is directly linked with at least another node by a line segment. From any node to each of the other nodes, there is one and only one uninterrupted chain connected by the line segments between the nodes. The two attributes at two nodes may not necessarily be compared explicitly. They can be compared with each other either directly if there is only one line segment between them, or indirectly if there is a chain of line segments between them and if all preference relations on the chain are transitive. For instance, Figure 2 shows a simple minimum set of complete comparisons for three attributes, y_1, y_2, y_3. Figure 3 displays a more complicated minimum set of complete comparisons for nine attributes, $y_i (i = 1, \ldots, 9)$.

In Figure 2, y_1 is directly compared with y_2 and y_3 respectively. y_2 and y_3 are linked together through y_1. They may be compared indirectly or may not be compared yet, dependent upon whether or not the preference relations R_{12} and R_{13} are transitive. If the DM states that "*y_2 is exactly three times as important as y_1*" and "*y_1 is at least as important as y_3*," for example, then

y_2 has actually been compared with y_3 indirectly, that is "*y_2 is at least three times as important as y_3.*" If the DM states that "*y_2 is at least three times as important as y_1*" and "*y_3 is also at least three times as important as y_1,*" it may not be appropriate to conclude that "*y_2 is as important as y_3.*" In fact, y_2 may be more or less important than y_3. In this case, more information is necessary if the exact preference relation between y_2 and y_3 has to be determined.

It may be noted that there is more than one way to construct the minimum set of complete pairwise comparisons if there are more than two attributes. A single-chain set and a star-shaped set may be two simplest minimum sets. In the former set, each of the attributes is compared with at least one but at most two other attributes; in the latter set, one attribute is used as the reference attribute, with which the other attributes are all compared. Other types of minimum sets may be spanned based on these two basic sets. For instance, the minimum set shown in Figure 2 may be regarded as a single-chain set or a star-shaped set with y_1 as the reference attribute. In Figure 3, if only $y_i (i = 1, \dots, 4)$ is considered, the minimum set for these four attributes is star-shaped with y_1 as the reference attribute. As a whole, the minimum set shown in Figure 3 is a complex set with y_1, y_2, y_3, and y_4 as the preference attributes.

3.3. Linear Programming Models for Weight Assignment

When the minimum set of complete pairwise comparisons for attributes is built, these comparisons can be transformed into linear equality or inequality constraints on the weights, which constitute the minimum constraint set $\Lambda_{\min}$. Generally, $\Lambda_{\min}$ may be written as follows

$$\Lambda_{\min} = \left\{ W \;\middle|\; \begin{array}{l} W = [\,w_1 \quad \dots \quad w_n\,]^\top, c_i w_i \Delta_r c_j w_j \text{ for } r = 1, \dots, n-1, \\ \textit{where } i, j \in \{1, \dots, n\}, i \neq j; \Delta_r \in \{<, >, =, \leq, \geq\}; \\ \text{and } c_i \text{ and } c_j \textit{ are real numbers.} \end{array} \right\} \tag{23}$$

According to Definition 1, $\Lambda_{\min}$ is not empty.

To avoid using a nonlinear function as a standard for weight assignment such as the relative entropy defined by (22), a p-norm function is adopted for assigning the best compromise weight. The p-norm function is defined as follows

$$\|W^* - W\|_p = \left[\sum_{i=1}^{n} (w_i^* - w_i)^p \right]^{1/p}, \tag{24}$$

where p is positive and $W^* = [\,w_1^* \quad \dots \quad w_n^*\,]^\top$ is the ideal weight vector with w_i^* being the ideal weight for the attribute y_i. If the weight vector is normalized, that is $\sum_{i=1}^{n} w_i = 1$, then let $w_i^* = 1 (i = 1, \dots, n)$ as the maximum possible weight for each attribute is one. It is easy to show that the p-norm also possesses the aforementioned inferential properties which the relative entropy satisfies.

It is always desirable that the best compromise weight is assigned to be as close as possible to the ideal weight. The mathematical programming problem for initial weight assignment may thus be formulated as follows

$$\begin{cases} \min \|W^* - W\|_p, \\ \text{subject to } W \in \Lambda, \end{cases} \tag{25}$$

where

$$\Lambda = \left\{ W \mid W \in \Lambda_{\min}; \sum_{i=1}^{n} w_i = 1;\ w_i \geq 0;\ i = 1, \dots, n \right\}.$$

Λ represents the feasible region for weight assignment, in which there is at least one feasible solution. The optimal solution of (25) may be used as the best compromise weight vector which is nearest W^* in the sense of p-norm. Let $p = \infty$. The ∞-norm may then be used to search for the best compromise weight as the problem (25) with $p = \infty$ can be transformed into the following minimax problem, i.e.,

$$\begin{cases} \min \|W^* - W\|_\infty \\ \text{subject to } W \in \Lambda \end{cases} \Longleftrightarrow \min_{W\in\Lambda} \max_i \{(w_1^* - w_1), \ldots, (w_i^* - w_i) \ldots, (w_n^* - w_n)\}, \tag{26}$$

and (26) has the following equivalent

$$\begin{array}{l} \min \ \lambda \\ \text{subject to } w_i^* - w_i \le \lambda, \qquad i = 1, \ldots, n; \quad W \in \Lambda, \quad \lambda \ge 0, \end{array} \tag{27}$$

which is only a linear programming problem.

If there is exactly one feasible solution in Λ, which is the case when the relations Δ_r in $\Lambda_{\min}$ (see (23)) are all exact ones (i.e., Δ_r is "=" for all $r = 1, \ldots, n-1$), the best compromise weight vector is precisely determined by the DM's preference statements.

If there is more than one feasible solution in Λ, which is generally the case, the best compromise weight vector is under-determined and may be generated as the optimal solution of (27). However, other feasible solutions in Λ may also be selected as the best compromise weight vector by the DM if he is not satisfied with the optimal solution of (27) and if there exist other solutions in Λ, which are significantly different from and better than the current optimum. Hence, it may be useful to define a measure to check the determinacy of the DM's preference statements, so that the DM can clearly know how much room remains for weight assignment.

A determinacy index (simply, DI) is then defined as follows.

Suppose $\overline{W}^{(j)} = [\,\overline{w}_1^{(j)} \ \ \ldots \ \ \overline{w}_n^{(j)}\,]^\top$ is the optimal solution of the following problem

$$\max_{W\in\Lambda} w_j, \qquad j = 1, \ldots, n. \tag{28}$$

$\overline{W}^{(j)}$ is called an extreme weight vector and $\overline{w}_j^{(j)}$ is the maximal feasible weight value for the attribute y_j. The area of the feasible weight vectors on the normalization hyperplane (Λ) may be a measure to indicate the determinacy, although any other measure can be conveniently substituted. As this area is difficult to calculate, the area of the hyperpolygon enclosed by connecting the extreme weight vectors may be used to approximate the whole feasible area. As the feasible area is a convex set, the constructed hyperpolygon is always part of it.

Define $E(\overline{W})$ as the mean vector of the n extreme weight vectors, that is

$$E(\overline{W}) = \frac{1}{n} \sum_{j=1}^{n} \overline{W}_{(j)} = [\,E(\overline{w}_1) \ \ \ldots \ \ E(\overline{w}_n)\,]^\top, \tag{29}$$

$$E(\overline{w}_i) = \frac{1}{n} \sum_{j=1}^{n} \overline{w}_i^{(j)}, \qquad i = 1, \ldots, n. \tag{30}$$

Obviously, $E(\overline{W})$ is the geographical centre of the hyperpolygon. Then define a normalized Euclidian distance between the mean weight vector $E(\overline{W})$ and the j^{th} extreme weight vector as follows

$$D_j = \left[\frac{\sum_{i=1}^{n} \left(\overline{w}_i^{(j)} - E(\overline{w}_i) \right)^2}{n(n-1)} \right]^{1/2}, \qquad j = 1, \ldots, n, \tag{31}$$

where the denominator $n(n-1)$ is a scaling factor. The DI may then be defined by

$$DI = 1 - \sum_{j=1}^{n} D_j. \tag{32}$$

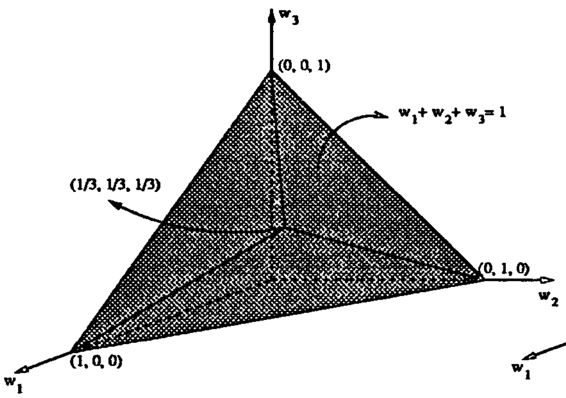

Figure 4. No preference information.

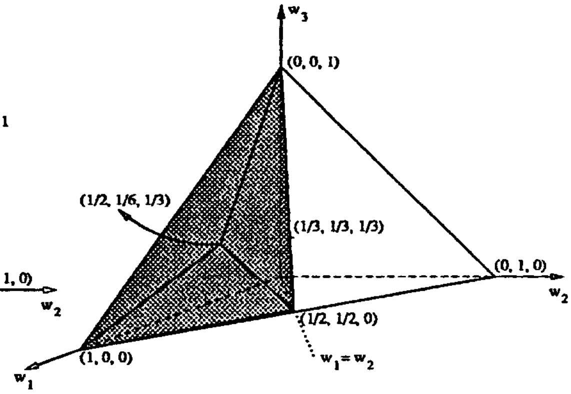

Figure 5. One comparison $w_1/w_2 \geq 1$.

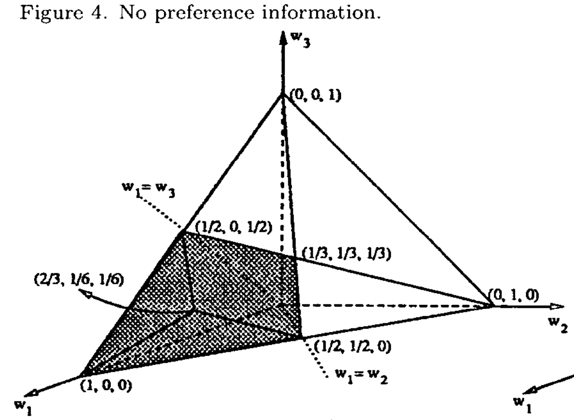

Figure 6. Two comparisons $w_1/w_2 \geq 1$ and $w_1/w_2 \geq 1$.

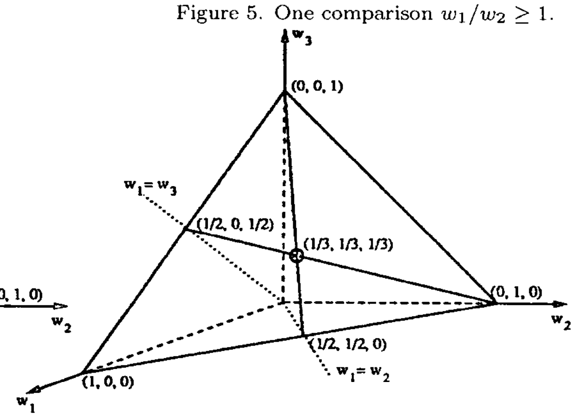

Figure 7. Two comparisons $w_1/w_2 = 1$ and $w_1/w_2 = 1$.

It is easy to prove that $DI \in [0 \quad 1]$ if $\overline{W}^{(j)}$ is generated using (28). When there is only one solution in Λ, $DI = 1$ as $\overline{W}^{(1)} = \cdots = \overline{W}^{(n)} = E(\overline{W}) = \hat{W}$ where $\hat{W}$ is the unique pre-determined weight vector in Λ. $DI = 0$ when no specific preference information is provided except for the normalizing constraint on weights, i.e., $\Lambda_{\min} = \varnothing$ in (23). In fact, it is easy to calculate from (28) that in this case $\overline{w}_i^{(j)} = 0 (i = 1, \ldots, n; i \neq j)$ and $\overline{w}_j^{(j)} = 1$ $(j = 1, \ldots, n)$ where $\overline{W}^{(j)} = [\overline{w}_1^{(j)} \quad \ldots \quad \overline{w}_n^{(j)}]^{\mathrm{T}}$. So, $E(\overline{W}) = [1/n \quad \ldots \quad 1/n]^{\mathrm{T}}$ and $D_j = 1/n, j = 1, \ldots n$.

Figure 4 to Figure 7 demonstrate a weight assignment problem for three attributes y_1, y_2, and y_3 with four sets of preference statements. It may be noted that the same best compromise weight vector $\hat{W} = [1/3 \quad 1/3 \quad 1/3]^{\mathrm{T}}$ can be obtained for the four sets of statements using (27). However, the determinacies of the four sets of statements are different. In Figure 4, no preference information is provided, so that any solution on the normalization plane $(w_1 + w_2 + w_3 = 1; w_1, w_2, w_3 \geq 0)$ might be selected to be the best weight vector. The three extreme weight vectors are $\overline{W}^{(1)} = [1 \quad 0 \quad 0]^{\mathrm{T}}$, $\overline{W}^{(2)} = [0 \quad 1 \quad 0]^{\mathrm{T}}$, $\overline{W}^{(3)} = [0 \quad 0 \quad 1]^{\mathrm{T}}$, and the corresponding mean weight vector $E(\overline{W}) = [1/3\ 1/3\ 1/3]^{\mathrm{T}}$. The value of DI^1 is then zero. In Figure 7, a minimum set of two exact complete comparisons is provided. Since $\hat{W} = \overline{W}^{(1)} = \overline{W}^{(2)} = \overline{W}^{(3)} = E(\overline{W}) = [1/3 \quad 1/3 \quad 1/3]^{\mathrm{T}}$, $DI^4 = 1$. In Figure 5, only one comparison is provided, which is incomplete, and $DI^2 = 0.21$. In Figure 6, a minimum set of two vague complete comparisons is provided and $DI^3 = 0.5$. In Figure 4, the area of the feasible weight vectors, the shaded area, is the same as the polygon (triangle) enclosed by connecting the three extreme weight vectors. This is also the case in Figures 5 and 7. In Figure 6, the latter is enclosed by the former.

From the illustrative examples, it is obvious that a larger value of DI indicates that the quality of preference information is better. Preferences with better quality can more precisely determine the best weight vector but they are more difficult to provide. If DI is large enough (near one), it makes no sense to acquire more preference statements from the DM because in this case other feasible weight vectors in Λ are not significantly different from the optimum of (27). If the value of DI is not good enough, more information may be required so as either to revise the existing comparisons in the minimum set or to compare more pairs of attributes directly. In the latter case, the added direct comparisons may be inconsistent with the ones in the minimum set. It is therefore necessary to check the consistency of the added comparisons.

Suppose Λ_a is an additional sub-set, which is composed of the added direct comparisons except for those involved in the minimum set and is defined as follows

$$\Lambda_a = \left\{ W_a \left| \begin{array}{l} W_a = [\, w_1 \quad \dots \quad w_n \quad d_1^+ \quad d_1^- \quad \dots \quad d_T^+ \quad d_T^- \,]^{\mathrm{T}}, a_i w_i \Delta_t a_j w_j + d_t^+ - d_t^-, \\ \text{for } t = 1, \dots, T, \textit{ where } i, j \in \{1, \dots, n\}, i \neq j; \Delta^t \in \{<, >, =, \leq, \geq\}; \\ T \textit{ is the number of the additional comparisons}; a_i \text{ and } a_j \\ \textit{are real numbers} \text{ ; and } d_t^+ \text{ and } d_t^- \textit{ are deviation variables} \\ \textit{with } d_t^+, d_t^- \geq 0 \text{ and } d_t^+ \times d_t^- = 0 \text{ for } \textit{all } t = 1, \dots, T \end{array} \right. \right\} \tag{33}$$

where deviation variables d_t^+ and d_t^- measure the consistency of the added comparisons with those in $\Lambda_{\min}$. The best compromise weight vector is then assigned using the following linear goal programming

$$\begin{array}{l} \min \left\{ P_1 \sum_{t=1}^{T} \left(d_t^+ + d_t^- \right) + P_2 \| W^* - W \|_\infty \right\}, \\ \text{subject to } W_a \in \Lambda_a, \qquad W \in \Lambda, \end{array} \tag{34}$$

where $P_1 \gg P_2$.

As a whole, the MIPAC method assigns the best compromise weight vector using the following two main steps if the number of comparisons is larger than $(n-1)$. At first, the consistency is checked. If $\sum_{t=1}^{T}(d_t^+ + d_t^-) = 0$, then the additional pairwise comparisons are consistent with those already involved in the minimum set. Otherwise, inconsistency occurs, which indicates that the weights are over-determined. The inconsistent comparisons with d_t^+ or d_t^- being greater than zero can then be identified. The DM may either revise these comparisons or the relevant comparisons in the minimum set. Then, the best compromise weight vector is assigned to be the solution which is nearest to the ideal weight vector in the sense of ∞-norm.

The MIPAC method thus provides a flexible and systematic procedure to acquire preference information. It initially requires the DM to provide a minimum number of complete pairwise comparisons for attributes so as to generate the first weight assignment using (27). If the DM is not satisfied with the initially assigned weights and if he can provide more and perhaps useful preference information, it will ask the DM either to revise the existing comparisons in the minimum set or to take into account more direct comparisons so that better compromise weights can be assigned. The consistency and determinacy of the comparisons can be checked and numerically measured so that the DM clearly knows the quality of the preference information he has provided and hence the quality of the best compromise weights he has obtained.

4. SELECTION OF EFFICIENT DESIGNS FOR SEMI-SUBMERSIBLE

4.1. Problem Description and Efficient Design Generation

A mathematical model for preliminary design of a semi-submersible has been built as a multiple objective decision making (MODM) problem [6], which can be generalized as the following vector nonlinear programming problem

$$\begin{array}{lll} \max & \{ y_1(a) \quad y_2(a) \quad y_3(a) \quad y_4(a) \quad y_5(a) \} & \\ \min & \{ y_6(a) \} & \\ \text{subject to} & g_i(a) \leq 0 \qquad i = 1, \dots, 11 & \\ & h_i(a) \leq 0 \qquad j = 1, \dots, 9, & a = [\, x_1 \quad x_2 \quad \dots \quad x_9 \,]^{\mathrm{T}} \end{array} \tag{35}$$

where $y_i(a)\,(i = 1,\ldots,6)$ are nonlinear objective functions, $g_i(a)\,(i = 1,\ldots,11)$ are nonlinear constraint functions and $h_i(a)\,(i = 1,\ldots,9)$ are linear constraint functions. The objective functions and the design variables are described in Table 3. The purpose of design is to generate a best compromise design which can attain the best possible values for these six objectives.

Table 3. Description of semi-submersible model.

design variables		objective functions		
symbol	description	symbol	units	description
x_1	corner column diameter	y_1	seconds	natural heave period
x_2	middle column diameter	y_2	tonnes	transit payload
x_3	length of the column	y_3	tonnes	operating payload
x_4	breadth of the pontoon	y_4	meters	permissible KG in transit
x_5	depth of the pontoon	y_5	meters	permissible KG in operation
x_6	length of the pontoon	y_6	pounds	cost of construction
x_7	height of the deck			
x_8	distance between pontoon centerlines			
x_9	diameter of transverse bracing			

Since there exists no single design which could optimize (maximize or minimize) the six objectives simultaneously, compromise among the objectives is necessary, based on the DM's preference information about the relative importance of the objectives. If preference information is acquired and represented by a utility function, the best compromise design may be obtained by optimizing the utility function. In this section, an alternative design synthesis strategy is presented. First, an interactive MODM method is used to generate a set of efficient (nondominated) designs. Then, the MIPAC method is used to assign weights for the attributes, these being the values of the various objectives. The best compromise design is then selected from the generated efficient designs using a MADM method.

The interactive step trade-off method (ISTM) is used to generate the efficient designs [8,14]. The interactive efficient design generation process is illustrated in [9]. Table 4 lists the values of the six objectives at the 13 generated efficient designs. The "$-$" symbol in the last column in Table 4 means that y_6 is for minimization. The first six designs ($a_1 \ldots a_6$) are the extreme designs (efficient ones) generated by optimizing each of the six objective functions separately. The values of the design variables for the extreme designs are shown in Table 5. The design a_{10} is referred to as the feasible ideal design which is closest to the imaginary ideal design taking the best feasible value of each objective. This feasible ideal design is generated assuming that all the objectives are of equal importance [8,9]. The other six efficient designs are generated near the feasible ideal design using an interactive decision making procedure [8,9]. The remaining problem is then to rank these 13 designs by taking the DM's preferences into account.

4.2. Preference Weight Assignment

Table 4 actually provides numerical values for multiple attribute evaluations of the efficient designs generated. If there were a design attaining the best values for all the six attributes, it would of course be selected as the best design. Unfortunately, such a design does not exist in the problem as some of the objectives are in conflict. Thus the ranking of the efficient designs depends not only on the multiple attribute evaluations but also on the preference information of the DM about the relative importance of the six attributes, which may be represented as weights.

The MIPAC method is used to assign the weights for the objectives. It is assumed that the DM initially provides the following pairwise comparisons for the objectives.

(1) *"COST OF CONSTRUCTION* (y_6)*" is at least twice as important as "NATURAL HEAVE PERIOD* y_1*"* (R_{16}).

Table 4. The decision matrix for the semi-submersible.

efficient designs	objective values					
	y_1	y_2	y_3	y_4	y_5	y_6
a_1	36.06	9594.95	18014.24	18.33	18.31	−7594448.00
a_2	32.64	13040.96	26194.62	21.33	21.34	−12404110.00
a_3	31.26	12284.91	30248.64	25.92	25.93	−15256452.00
a_4	28.21	6952.27	20811.53	32.29	32.32	−13551909.00
a_5	28.23	6748.00	20270.23	32.29	32.32	−13205347.00
a_6	21.87	2288.60	7608.83	20.36	20.35	−4805733.50
a_7	31.48	7892.82	16317.26	23.11	23.10	−7848295.50
a_8	31.48	7663.19	15785.09	22.11	22.10	−7430433.50
a_9	29.50	6663.20	14785.06	23.38	23.38	−7430433.00
a_{10}	31.48	8448.19	20186.60	26.11	26.09	−9450440.00
a_{11}	31.48	8312.49	19585.60	25.61	25.59	−9078700.00
a_{12}	31.61	8311.80	19584.40	25.21	25.19	−8946110.00
a_{13}	31.01	7985.11	17584.40	25.21	25.21	−8659400.00

Table 5. Variable values of the extreme designs.

design variables	extreme designs					
	a_1	a_2	a_3	a_4	a_5	a_6
x_1	12.59	15.00	15.00	14.99	15.00	9.86
x_2	7.00	10.00	12.00	7.50	7.00	8.22
x_3	26.80	26.80	30.00	40.00	40.00	26.80
x_4	25.00	25.00	25.00	19.99	20.00	13.15
x_5	8.93	8.93	10.00	10.00	10.00	8.93
x_6	117.53	150.00	150.00	132.18	128.98	90.83
x_7	4.00	7.07	10.00	10.00	10.00	4.00
x_8	75.53	90.00	90.00	89.96	90.00	59.17
x_9	2.69	1.00	1.00	1.00	1.00	1.00

(2) *"COST OF CONSTRUCTION* (y_6)*" is at least three times as important as "OPERATING PAYLOAD* (y_3)*"* (R_{36}).

(3) *"PERMISSIBLE KG IN OPERATION* (y_5)*" is at least twice as important as "OPERATING PAYLOAD* (y_3)*"* (R_{35}).

(4) *"OPERATING PAYLOAD* (y_3)*" is at least twice as important as "TRANSIT PAYLOAD* (y_2)*"* (R_{23}).

(5) *"PERMISSIBLE KG IN TRANSIT* (y_4)*" is as important as "PERMISSIBLE KG IN OPERATION* (y_5)*"* (R_{45}).

The comparisons R_{16}, R_{36}, R_{35} and R_{23} are vague ones and the last comparison R_{45} is an exact one. The above set of comparisons can be depicted as shown in Figure 8. Obviously, these five comparisons constitute a minimum set of complete pairwise comparisons for the six objectives.

These preference statements are then transformed into the constraints on the weights. Suppose w_i is the relative weight for $y_i, i = 1, \ldots, 6$. Then the initial minimum set $\Lambda_{\min}^{(0)}$ can be constructed

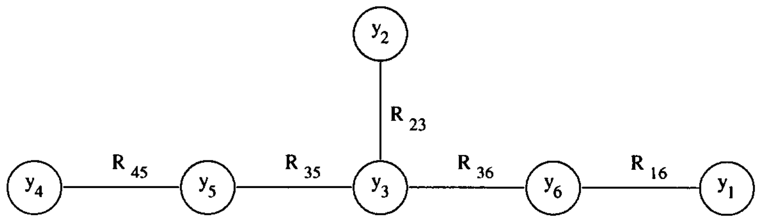

Figure 8. Minimum comparison set for semi-submersible.

as follows

$$\Lambda^{(0)}_{\min} = \left\{ W \,\middle|\, \begin{matrix} w_6 - 2w_1 \geq 0,\ w_6 - 3w_3 \geq 0,\ w_5 - 2w_3 \geq 0 \\ 2w_2 - w_3 \leq 0,\ w_4 - w_5 = 0,\ W = [w_1 \ \ \dots \ \ w_6]^{\mathsf{T}} \end{matrix} \right\}. \tag{36}$$

The initial linear programming for assigning the weights can be constructed as follows

$$\begin{aligned} &\min \|W^* - W\|_\infty, \\ &\text{subject to } W \in \Lambda^{(0)}, \end{aligned} \tag{37}$$

where

$$W^* = [1 \ \ \dots \ \ 1]^{\mathsf{T}} \tag{38}$$

$$\Lambda^{(0)} = \left\{ W \mid W \in \Lambda^{(0)}_{\min}, \sum_{i=1}^{6} w_i = 1, w_i \geq 0, i = 1, \dots, 6 \right\} \tag{39}$$

or equivalently

$$\begin{aligned} &\min \lambda, \\ &\text{subject to } 1 - w_1 \leq \lambda, \qquad w_6 - 2w_1 \geq 0, \\ &\qquad\qquad\ 1 - w_2 \leq \lambda, \qquad w_6 - 3w_3 \geq 0, \\ &\qquad\qquad\ 1 - w_3 \leq \lambda, \qquad w_5 - 2w_3 \geq 0, \\ &\qquad\qquad\ 1 - w_4 \leq \lambda, \qquad 2w_2 - w_3 \leq 0, \\ &\qquad\qquad\ 1 - w_5 \leq \lambda, \qquad w_4 - w_5 = 0, \\ &\qquad\qquad\ 1 - w_6 \leq \lambda, \qquad \sum_{i=1}^{6} w_i = 1, \\ &\qquad\qquad\ \lambda \geq 0, \quad w_i \geq 0, \quad i = 1, \dots, 6. \end{aligned} \tag{40}$$

The optimal solution of (40) is $\hat{W}^{(0)} = [0.0556 \ \ 0.0556 \ \ 0.1111 \ \ 0.2222 \ \ 0.2222 \ \ 0.3333]^{\mathsf{T}}$ and the value of the determinacy index is $DI^{(0)} = 0.5385$. $DI^{(0)}$ is rather small, which means that much room remains for improvement of the weight assignment. It could be considered, for example, that the DM is not satisfied with the initial weight assignment in that the first objective "*NATURAL HEAVE PERIOD*" (y_1)" is a very important performance index but it has been assigned the lowest weight. The DM therefore takes into consideration the following two additional comparisons.

(6) "*NATURAL HEAVE PERIOD* (y_1)" *is* 1.5 *times as important as* "*PERMISSIBLE KG IN OPERATION* (y_5)" (R_{15}).

(7) "*COST OF CONSTRUCTION* (y_6)" *is at most* 2.5 *times as important as* "*NATURAL HEAVE PERIOD* (y_1)" (R_{16}).

The two added comparisons can be transformed to the additional constraint subset $\Lambda^{(1)}_a$ on weights,

$$\Lambda^{(1)}_a = \left\{ W_a \,\middle|\, \begin{matrix} w_1 - 1.5w_5 + d_1^+ - d_1^- = 0,\ w_6 - 2.5w_1 - d_2^- \leq 0 \\ d_1^+ \times d_1^- = 0;\ d_1^+, d_1^-, d_2^- \geq 0,\ W_a = [w_1 \ \ \dots \ \ w_6 \ \ d_1^+ \ \ d_1^- \ \ d_2^-]^{\mathsf{T}} \end{matrix} \right\}. \tag{41}$$

Furthermore, the DM agrees that the "at least" in the comparisons R_{35} and R_{23} (statements (3) and (4)) can now be removed so that R_{35} and R_{23} become exact preference relations instead of the original vague ones. The initial minimum set $\Lambda^{(0)}_{min}$ is thus revised to be $\Lambda^{(1)}_{min}$,

$$\Lambda^{(1)}_{\min} = \left\{ W \,\middle|\, \begin{matrix} w_6 - 2w_1 \geq 0,\ w_6 - 3w_3 \geq 0,\ w_5 - 2w_3 = 0 \\ 2w_2 - w_3 = 0,\ w_4 - w_5 = 0,\ W = [w_1 \quad \ldots \quad w_6]^{\mathrm{T}} \end{matrix} \right\} \tag{42}$$

The linear goal programming for improving the initial weight assignment is then formulated by

$$\begin{aligned} &\min\ \left\{P_1\left[(d_1^+ + d_1^-) + d_2^-\right] + P_2\|W^* - W\|_\infty\right\}, \\ &\text{subject to } W_a \in \Lambda_a^{(1)}, \quad W \in \Lambda^{(1)}, \end{aligned} \tag{43}$$

where

$$\Lambda^{(1)} = \left\{ W \mid W \in \Lambda^{(1)}_{min}; \sum_{i=1}^{6} w_i = 1, w_i \geq 0, i = 1, \ldots, 6 \right\}. \tag{44}$$

Solving (43), the new optimum $\hat{W}^{(1)} = [0.2069 \quad 0.0345 \quad 0.069 \quad 0.1379 \quad 0.1379 \quad 0.4138]^{\mathrm{T}}$ can be obtained with $d_1^+ = d_1^- = d_2^- = 0$ and $DI^{(1)} = 0.9784$. Thus, the added comparisons R_{15} and R_{16} are consistent with those listed in the revised minimum set $\Lambda^{(1)}_{\min}$. $DI^(1)$ is now large enough and $\hat{W}^{(1)}$ may be used as the best compromise weight vector, that is,

$$W = \hat{W}^{(1)} = [0.2069 \quad 0.0345 \quad 0.069 \quad 0.1379 \quad 0.1379 \quad 0.4138]^{\mathrm{T}}. \tag{45}$$

If the DM is not satisfied with $\hat{W}^{(1)}$ either, he may further revise the minimum set and/or provide more direct comparisons. For instance, the DM may add that

(8) "*COST OF CONSTRUCTION* (y_6)" *is at most 7 times as important as* "*OPERATING PAYLOAD* (y_3)" (R_{36}).

Therefore, $\Lambda_a^{(1)}$ defined in (41) is changed into

$$\Lambda_a^{(2)} = \left\{ W_a \,\middle|\, \begin{matrix} w_1 - 1.5w_5 + d_1^+ - d_1^- = 0,\ w_6 - 2.5w_1 - d_2^- \leq 0,\ w_6 - 7w_3 - d_3^- \leq 0, \\ d_1^+ \times d_1^- = 0;\ d_1^+, d_1^-, d_2^-, d_3^- \geq 0,\ W_a = [w_1 \quad \ldots \quad w_6 \quad d_1^+ \quad d_1^- \quad d_2^- \quad d_3^-]^{\mathrm{T}} \end{matrix} \right\}. \tag{46}$$

Solving the following new linear goal programming problem

$$\begin{aligned} &\min\ \left\{P_1\left[(d_1^+ + d_1^-) + d_2^- + d_3^-\right] + P_2\|W^* - W\|_\infty\right\}, \\ &\text{subject to } W_a \in \Lambda_2^{(2)}, \quad W \in \Lambda^{(1)}. \end{aligned} \tag{47}$$

The optimal value of $\hat{W}^{(2)}$ is equal to $\hat{W}^{(1)}$ with $d_1^+ = d_1^- = d_2^- = d_3^- = 0$ and $DI^{(2)} = 0.985 > DI^{(1)}$. So the added direct comparison for y_3 and y_6 has improved the determinacy or quality of the preference information.

On the other hand, if the statement (8) is replaced by

(8.1) "*COST OF CONSTRUCTION* (y_6)" *is at most 5 times as important as* "*OPERATING PAYLOAD* (y_3)" (R'_{36}).

It can be shown that (R'_{36}) is inconsistent with the statements (1) to (7), that is $d_3^- > 0$. As a matter of fact, the statements (1) to (7) imply that $7.5 \geq w_6/w_3 \geq 6$.

4.3. Design Selection

Based on the decision matrix as shown in Table 4 and the assigned weights listed in (45), the 13 alternative efficient designs can be ranked using a MADM method. The CODASID method briefly outlined in Section 2 is chosen for this problem instead of the simple additive utility function method because the six objectives in the problem are not preferentially independent. For example, the trade-offs between "*OPERATING PAYLOAD*" and "*COST OF CONSTRUCTION*" may not be meaningful without considering the "*NATURAL HEAVE PERIOD*" concerning the safety of the structure.

It is assumed that the DM has a neutral attitude in ranking these designs, that is, he wishes to make a full use of all the information available without preferring to a specific type of information. In the CODASID method, this implies that $\rho_1 + \rho_2 = 0.5$ and $\rho_3 = 0.5$. It is also assumed that the multiple attribute evaluations (the decision matrix) is as reliable as the preference judgments (the weights) in the ranking of the efficient designs. This latter assumption means that $\rho_1 = \rho_2 = 0.25$ (see (19)).

In the CODASID method, the decision matrix and the weights are first aggregated into the following judgment-evaluation (J-E) matrix (Table 6) using formulae (4) to (12)

Table 6. The J-E matrix.

a	$pc(a)$	$ec(a)$	$d(a)$
a_1	1.794	−0.506	−0.505
a_2	−1.516	0.386	1.378
a_3	−2.552	1.242	1.802
a_4	−2.828	1.129	0.463
a_5	−1.794	1.110	0.311
a_6	−1.517	−3.932	0.944
a_7	0.690	−0.281	−0.595
a_8	1.586	−0.601	−0.523
a_9	1.310	−0.806	−0.393
a_{10}	1.655	0.797	−0.714
a_{11}	1.172	0.634	−0.748
a_{12}	1.862	0.558	−0.738
a_{13}	0.138	0.271	−0.682

Table 7. Ranking of the efficient designs.

a_r	$u(a_r)$	$rank(a_r)$
a_1	0.819761	6
a_2	0.394515	11
a_3	0.377091	12
a_4	0.556930	10
a_5	0.614344	9
a_6	0.266216	13
a_7	0.825511	5
a_8	0.812709	7
a_9	0.774379	8
a_{10}	0.952565	1
a_{11}	0.922411	3
a_{12}	0.934146	2
a_{13}	0.844813	4

Following formulae (13) to (20), the information contained in the J-E matrix is then synthesized into the relative closeness index $u(a_r)$ for each alternative design $a_r (r = 1, \dots, 13)$, based on which the ranking of the designs is then made. The values of the indices and the ranking of the designs are listed in Table 7.

From Table 7, the design a_{10} is ranked to be the best compromise design. As mentioned before, a_{10} is the feasible ideal design which has been generated to be closest to the imaginary ideal design with assuming that all the objectives are of equal importance. a_{10} is still evaluated to be the most favorable although the weights assigned to the objectives are different. This is because a_{10} has relatively good values on all the objectives. In particular, a_{10} has quite high values on the three important objectives, "natural heave period," "permissible KG in transit" and "permissible KG in operation" which are the measures of the safety of the structure. Thus, this design is quite safe with high "transit payload" and "operating payload" although its "cost of construction" is a little higher than some other designs.

5. CONCLUDING REMARKS

In any multiple attribute design selection situation, the preference information of a designer forms one of the bases for selection of the most favorable design. The MIPAC method explored in the paper provides a flexible and systematic way for preference acquisition and representation in terms of relative weights. It only requires minimal preference information for initial representation of the designer's overall preference structure. It also provides an iterative procedure for improvement of the representation if more preference information can be provided. In this process, the consistency and the determinacy of the preference judgments can be checked and measured explicitly.

The treatment of a multiple attribute design selection problem involves the definition of significant attributes, the generation of feasible candidate designs and the separate evaluations of each design with respect to every attribute, which requires a detailed understanding of the design problem. The illustrative design problem of a semi-submersible provides one possible way of problem formulation and solution. It also demonstrates how the MIPAC method can be implemented for weight assignment using minimal preference information.

REFERENCES

1. C.L. Hwang and K. Yoon, *Multiple Attribute Decision Making Methods and Applications: A State-of-the-Art Survey*, Springer-Verlag, New York, (1981).
2. T.L. Saaty, *The Analytic Hierarchy Process*, University of Pittsburgh, (1988).
3. G. Islei and A.G. Lockett, Judgmental modeling based on geometric least square, *European Journal of Operational Research* **36**, 27–35, (1988).
4. P. Gabbert and D.E. Brown, Knowledge-based computer-aided design of materials handling systems, *IEEE Transactions on Systems, Man, and Cybernetics* **19** (2), 188–196, (1989).
5. C.C. White III, A.P. Sage and S. Dozono, A model of multiattribute decisionmaking and trade-off weight determination under uncertainty, *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-14** (2), 223–229, (1984).
6. P.F. Meldrum, A semi-submersible model (Phase I), Research Report, EDCN/MCDM/RESC/1/1, Engineering Design Centre, University of Newcastle upon Tyne, U.K., (1991).
7. M.K. Starr and M. Zeleny, *Multiple Criteria Decision Making*, North-Holland, New York, (1977).
8. J.B. Yang, C. Chen and Z.J. Zhang, The interactive step trade-off method (ISTM) for multiobjective optimization, *IEEE Transactions on Systems, Man, and Cybernetics* **20** (3), 688–695, (1990).
9. J.B. Yang, An integrated multicriteria decision support system for engineering design, Research Report, Engineering Design Centre, University of Newcastle upon Tyne, U.K., (1992).
10. P. Levine, M.J. Pomerol and R. Saneh, Rules integrate data in a multicriteria decision support system, *IEEE Transactions on Systems, Man, and Cybernetics* **20** (3), 678–686, (1990).
11. J.B. Yang and M.G. Singh, An evidential reasoning approach for multiple attribute decision making with uncertainty, *IEEE Transactions on Systems, Man, and Cybernetics* **24** (1), 1–18, (1994).
12. J.B. Yang and P. Sen, A hierarchical evaluation process for multiple attribute design selection with uncertainty, In *Proceedings of 6^{th} International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Edinburgh, Scotland, 1993, (Edited by P.W.H. Chung, G. Lovegrove and M. Ali); IEA/AIE 93, pp. 484–493, Gordon and Breach Science Publishers, Switzerland, (1993).
13. J.B. Yang and P. Sen, A general multi-level evaluation process for hybrid MADM with uncertainty, *IEEE Transactions on Systems, Man, and Cybernetics* **24** (10), (1994).
14. J.B. Yang, C. Chen and Z.J. Zhang, The interactive decomposition method for multiobjective linear programming and its applications, *Information and Decision Technologies (Large Scale Systems)* **14** (3), 275–288, (1988).
15. P. Sen, Marine design: the multiple criteria approach, Transactions of the Royal Institute of Naval Architects, London, (1991).
16. J.B. Yang, P. Sen and P.F. Meldrum, Multiple attribute decision making through concordance and discordance analyses by similarity to ideal designs, *Journal of Multi-Criteria Decision Analysis*, (1993) (to appear).
17. I.L. Buxton and B.U. Akgul, The comparison of general cargo ship economic performance by simulation, *Maritime Policy & Management* **16** (1), 27–44, (1989).
18. J. Teghem Jr., C. Delhaye and P. L. Kunsch, An interactive decision support system (IDSS) for multicriteria decision aid, *Mathl. Comput. Modelling* **12** (10/11), 1311–1320, (1989).
19. D. Vanderpooten and P. Vincke, Description and analysis of some representative interactive multicriteria procedures, *Mathl. Comput. Modelling* **12** (10/11), 1221–1238, (1989).