Translation from Transactions of IPSJ

A Knowledge Acquisition Method Based on a Multi-Attribute Utility Model

TOORU MITSUTA*, YASUHIRO KOBAYASHI* and HISANORI NONAKA*

In solving a design or planning problem, an expert sometimes selects a specific alternative on the basis of multiple criteria. It is difficult to elicit the expert's knowledge concerned with this kind of decision-making and represent it in the form of if-then rules in order to build an expert system. Here, a method for knowledge acquisition and representation is proposed that can overcome this difficulty. In this method, a multi-attribute utility function is employed as a knowledge representation model, which was originally proposed for decision-making with multiple objectives in multi-attribute utility theory.

An initial model is obtained by the conventional utility theory technique. This model is modified in order to eliminate conflicts between conclusions arrived at by the model and the expert. Modifications are performed by using an assumption selection module and a control module for focus on the modification process control. In order to improve the efficiency of modification, heuristics are used for assumption selection, backtracking planning, and so on.

The proposed method has been applied to knowledge acquisition problems in the construction scheduling domain, and its validity as a knowledge acquisition technique has been confirmed.

1. Introduction

Development of knowledge acquisition support systems is one of the key issues for practical building and use of expert systems. It is important to acquire knowledge from domain experts, since an expert system's performance depends strongly on the quality of its knowledge base. In practice, however, it is often difficult to derive sufficient knowledge through interviews with experts, because they are not necessarily aware of all the knowledge they use in everyday problem-solving processes. To compensate for this, several support methods have been proposed for efficient knowledge acquisition through interviews with domain experts by a knowledge engineer [1-6].

One such method is MORE, which employs an interview strategy that makes experts aware of their expertise in the domain of diagnosis. Another is SALT, which is based on an interview strategy for selecting or modifying parameter values in the domain of engineering design.

In some cases, however, it is difficult to obtain knowledge representation in the form of rules, not because experts are unaware of their expertise, but because they cannot describe their expertise easily. In selection of design alternatives, for example, a knowledge engineer and a knowledge supplier can identify dominant attributes in the design data such as cost, reliability, and efficiency, but the supplier cannot always clarify the detailed selection criteria for combination of these attribute values. The above-mentioned interview-based methods are not necessarily applicable to problems requiring selection of one of several alternatives on the basis of multiple performance indices, such as multi-criteria decision-making problems. The selection of a specific design out of many candidates is one such problem.

The Analytic Hierarchy Process (AHP) [7-8] and multi-attribute utility model [9-10] are methods for multi-criteria decision-making problems. These methods analyze the preference order of a decision-maker and choose a specific plan out of a set of alternative plans. They have been applied to planning the site of an industrial plant [11], planning the investigation of plant equipment [12], and other tasks. The AHP method is composed of two steps: assignment of the decision-maker's subjective judgment to alternative plans, and ordering of the alternative plans according to this judgment. It is advantageous when using judgments that are not directly quantifiable, and for simplying the ordering procedure, but it is disadvantageous when the validity of a given subjective judgment is ignored.

The multi-attribute model evaluates alternative plans indirectly. It selects quantifiable attributes, defines a formula for plan evaluation that specifies to what extent

This is a translation of the paper that appeared originally in Japanese in Transactions of IPSJ, Vol. 31, No. 6 (1990), pp. 763-771.

*Energy Research Laboratory, Hitachi, Ltd.

the criteria, represented in terms of attributes, are satisfied, and evaluates plans according to the formula. This method clearly describes the grounds and units for the adopted evaluation measure, thus elucidating the characteristics of the measure and taking account of the evaluation's validity. Judgmental criteria specifications can be regarded as a kind of knowledge acquisition, as can the extraction and quantification of an expert's preference. Expert systems in the domain of design and planning require judgmental criteria for selection of an appropriate one of several alternative plans, which are usually in the form of rules. Although rules are not directly extractable from experts, they may be indirectly obtained by the AHP or multi-attribute utility model methods, since these methods are able to extract criteria for plan selection according to judgmental results in cases of problem-solving by a decision-maker.

In the context of the relationship between decision-making and domain knowledge, recent papers have attempted to integrate multi-criteria decision-making and knowledge processing [13, 14]. The knowledge acqusition support system AQUINAS [15] evaluates performance values for alternative plans by means of an AHP model, for use as certainty measures of knowledge in rule form.

To do so, the system extracts the inter-attribute dependency required for classification, and describes the structure of rules. Although an AHP model is useful for pointing out dominant attributes, it requires direct comparison of attribute values. This makes the model less attractive in the domains of design and planning, where alternatives are not exhaustively enumerated and evaluated beforehand.

The multi-attribute utility model allows convenient treatment of the extended evaluation of candidate plans that are newly added, with minimum changes in the predefined model. This suggests that the model may be applicable to design and planning problems. Straightforward application of a knowledge acquisition method based on this model, however, would presumably cause some problems, and the original model should be extended so as to match the knowledge acquisition process. One requirement is that the knowledge should be consistent for use in the knowledge base. In the proposed method, knowledge is extracted from interviews with experts by a knowledge engineer, and refined with judgmental results in problem-solving cases by experts. Knowledge must be free from contradictions:

1. No item of knowledge should fail to match any other item of knowledge.
2. No decision made by the system with the knowledge base should fail to match a decision made by a human expert.

The original multi-attribute utility model does not support the handling of mismatches between an expert system decision and a human expert decision. Because of its bottom-up style of modelling preference, the utility model does not pay enough attention to consistency between the computational output of the model and the real-world solution, though this kind of consistency is crucial in the knowledge acquisition process. This problem derives from the intended use of the multi-attribute utility model, which is for developing applications in difficult situations when a decision-maker cannot directly determine the priority of alternatives. In the original application domain, no reference solution is given, and the quantification of a preference mismatch is not meaningful, unlike in the building and refinement of a knowledge base.

In order to apply this modelling method to the knowledge representation model, it is necessary to extend the method's treatment of output consistency. Very few attempts have been made to extend the multi-attribute utility model from the viewpoint of knowledge acquisition. The objective of this study is to propose a knowledge acquisition method that represents knowledge on the basis of the multi-attribute utility model, efficiently identifies knowledge in the form of utility functions which ensure that the inference result is consistent with an expert's judgment, and facilitates the utilization of the knowledge in a problem-solving system.

2. Knowledge Acquisition Method

2.1 Technical Issues

The multi-attribute utility model offers a basis for the identification of a decision-maker's preference model in the form of a utility function and for computerized decision-making according to the order of preference of alternatives. These processes are based on the quantitative value of the overall performance of each alternative, while analysis based on the decision-maker's judgments provides a utility function and weighting factor for each attribute. The weighting factor is hereafter called a trade-off coefficient.

$$U(x) = \Sigma W_i * U_i(x) \quad (1)$$

$$\Sigma W_i = 1 \quad (2)$$

$$0 < U_i < 1$$

x: alternative plan

U: overall performance of alternative plan

U_i: utility function (individual performance function) of attribute i

W_i: trade-off coefficient of attribute i

A utility function for each attribute is usually identified through reference experiments by the 50–50 lot method, and defines a plan with $U(x) = 0.5$. First, it extracts an indifferent value for an alternative plan x. The decision-maker considers the preference of the lot having this value with a 100% likelihood, which is equivalent to the preference of the lot having the best result with a probability of 50% and the worst result with a probability of 50%. Several sample data are ob-

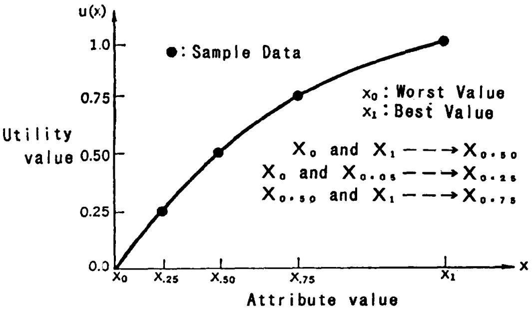


Fig. 1 Example of a utility function.

tained by repeating this kind of procedure. Secondly, the utility function is completed by fitting these sample results. An example of a utility function is shown in Fig. 1, where three representative and indifferent sample points $X_{0.50}$, $X_{0.25}$ and $X_{0.75}$ are obtained from the results of the decision-maker's judgment. Trade-off coefficients are similarly determined by the use of indifferent points between attributes.

If the model is to be usable for knowledge representation, all possible means should be provided of adjusting it so as to make its output of the order of preference of alternatives consistent with that of the decision-maker. These means will help the knowledge engineer to modify utility functions, if any mismatch is detected between the model output and the decision-maker's output and modification is required.

Assuming that the decision-maker's preference is correct, three items should be dealt with to modify the model for consistency:

1. Attributes;
2. A utility function for each attribute; and
3. A trade-off coefficient for each attribute.

Modification targets for items 2 and 3 are processed by trial-and-error, though item 1 can be selected by a method similar to the AQUINAS system. In a trial-and-error procedure for model adjustment, the following steps are repeated until the model output is consistent with that of the decision-maker:

1. Choice of target trade-off coefficients and utility functions to be modified;
2. Specification of the requirement, direction, and quantity of modification for chosen objects; and
3. Comparison of the order of preference of alternatives given by the modified model and that arrived at by the decision-maker's judgment.

It is difficult for a blind procedure to incorporate the decision-maker's preference into the model and, therefore, it is also difficult to produce multi-attribute utility functions efficiently and without mismatch.

One major technical issue in the application of the multi-attribute utility model to knowledge acquisition is how to realize an efficient modification process for utility functions and trade-off coefficients for attributes in such a way as to reproduce judgmental results arrived at by a decision-maker. To do this with a reasonable computational effort, it is necessary to limit the scope of the target objects to be modified to a tractable range in response to the results of comparison, and to control the change of focus in the scope, intelligently and at appropriate times, according to information fed back from the modification process.

2.2 Basic Configuration

In the problem of knowledge acquisition, additive independence is assumed in the formula for the preference of alternatives. This means that evaluation of an attribute is not influenced by that of other attributes. The identification of multi-attribute utility functions is identical with the specification of utility functions and trade-off coefficients. These two processes are performed independently in the conventional approach. The proposed approach is based on the modification process with information fed back from the comparison of preference results. The basic idea of this approach can be summarized as follows.

If utility functions are given to each attribute, the overall performance of each alternative plan can be expressed as a function of unknown variables that is, trade-off coefficients. If the order of preference is given for a pair of alternatives, this relation is expressed as an inequality as a function of trade-off coefficients.

As an example, consider a three-attribute utility function. When plan x is prefered to plan y, inequality $U(x) > U(y)$ holds true, and the following inequality is derived:

$$\{U_1(x)-U_1(y)\}\cdot W_1+\{U_2(x)-U_2(y)\}\cdot W_2 + \{U_3(x)-U_3(y)\}\cdot W_3>0.$$

This suggests that a set of inequalities is defined in terms of variables of trade-off coefficients according to a set of preference relations for a pair of alternatives, which are specified according to the decision-maker's judgment. Then, to identify a set of trade-off coefficients that reproduce the decision-maker's preference, the inequalities must be solved and values for the variables, satisfying all the inequalities, must be found. The latter requires a search of the sub-space satisfying all inequalities in the multi-dimensional space of the trade-off coefficients. In practice it is not easy to search this sub-space directly and to establish a satisfactory utility function. The proposed method identifies the multi-attribute utility model through iterative modification of the utility functions and refinement of trade-off coefficients.

The procedure for this knowledge acquisition is depicted in Fig. 2. Bounding the range of target objects is based on the results of comparing a simulated judgment by the model and a real judgment by a decision-maker, and is carried out at four levels of preference modelling:

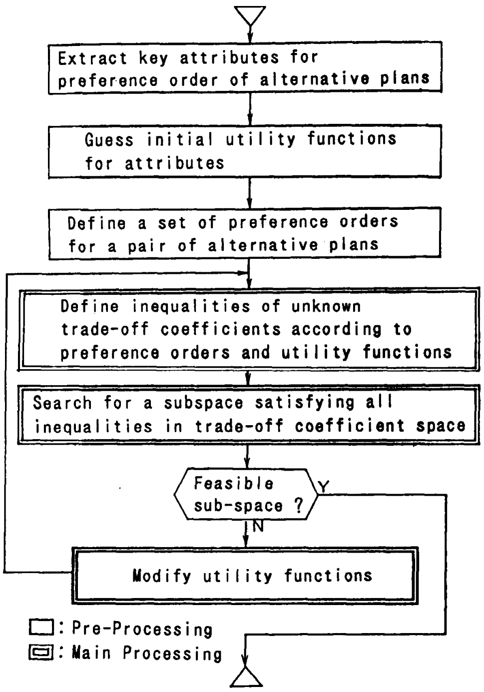


Fig. 2 Procedure for knowledge acquisition.

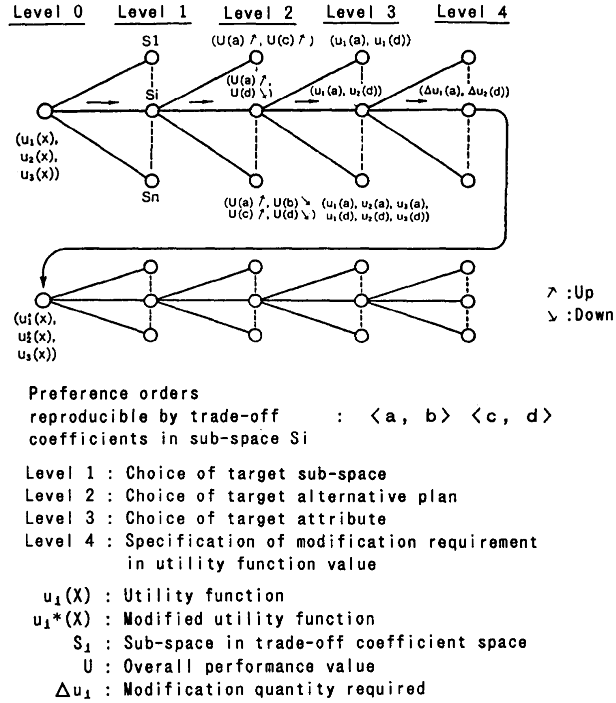


Fig. 3 Example of a utility function modification process.

1. Level 1: Choice of target sub-space.

Choose the subspace that is presumed most likely to be the true sub-space, and assume that the current trade-off coefficients are compatible with the chosen sub-space.

2. Level 2: Choice of target alternative plan.

Find a target pair of alternative plans for which the current trade-off coefficients cannot reproduce the decision-maker's preference, and choose one plan from the pair as a plan to be modified according to the overall performance measure.

This step is illustrated with an example. Set A is a set of preferences of alternatives.

$$a_{ij} \in A \tag{3}$$

$$a_{ij} = \langle x_i, x_j \rangle:\ x_i \text{ is preferred to } x_j. \tag{4}$$

The statement that x_i is preferred to x_j is expressed as $U(x_i) > U(x_j)$. There exists a relation $U(x_i) < U(x_j)$, if trade-off coefficients based on the chosen sub-space fail to reproduce preference a_{ij}. This wrong inequality must be corrected to the target inequality $U(x_i) > U(x_j)$. Two choices for this correction process are to increase $U(x_i)$ or to decrease $U(x_j)$.

3. Level 3: Choice of target attribute.

Choose a target attribute from the attributes of the chosen alternative plan to modify the utility function. This step chooses one attribute from attributes $U_1(x_i)$, $U_2(x_i)$ and $U_3(x_i)$ to be increased at step (ii) in the example.

4. Level 4: Evaluation of the modification requirement in the utility function.

Determine the direction and quantity of modification in the chosen attribute to correct the utility function. This step for utility function modification is not straightforward and does not ensure a consistent set of trade-off coefficients satisfying all the inequalities. This is because a utility function is related to more than one alternative. Utility function modification for a focused inequality may have the side effect of violating other inequalities that were previously satisfied. This results in backtracking of modification trials at the same level. The procedure iteratively checks and maintains the validity of other preferences previously established after modification of a target preference. If the corrective action finally fails to find a proper set of trade-off coefficients at the present level, further backtracking is required at a higher level.

Knowledge acquisition is a kind of search problem that requires a procedure to be solved by backtracking. In this problem, the branches of the search tree expand drastically as utility functions are modified. Figure 3 shows the utility function modification process. The initially given utility functions are nodes placed at level 0, and subspaces of the trade-off coefficient space based on the utility function are nodes at level 1. Nodes at level 2 are sets of alternative plans, whose overall performance is corrected. From a chosen node of level 3, nodes of utility functions are corrected at level 4.

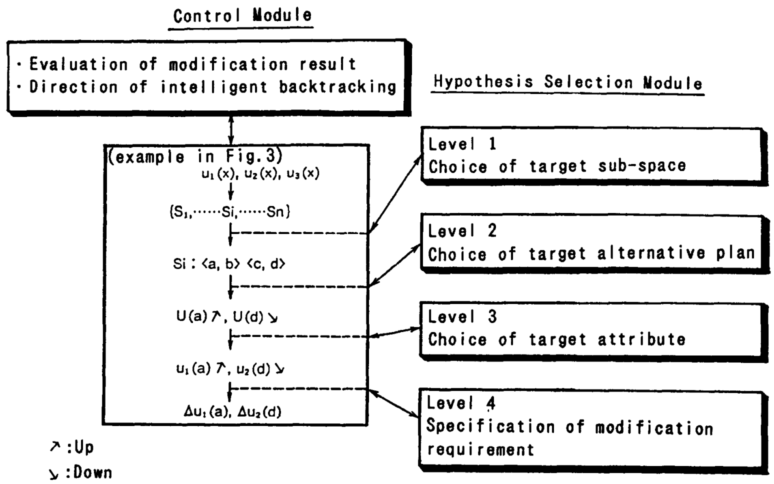
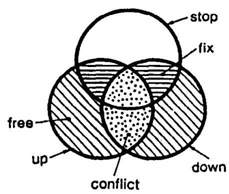


Fig. 4 Module configuration.

A : Set of alternative plan pairs
Aok : Set of alternative plan pairs reproducing target preference order
Ang : Set of alternative plan pairs not reproducing target preference order
$\langle x, y \rangle$: Preference order of object pair x and y; x is preferred to y



stop : Alternative plans included in Aok
up : Alternative plans that are the first pairs in Ang
down : Alternative plans that are the second pairs in Ang
free : Alternative plans that have a unique modification direction and quantity
fix : Alternative plans that have a unique modification direction
conflict : Alternative plans that have contradictory modification requirements

Fig. 5 Modification requirement for planning alternatives.

From the result of utility function modification at level 4, an updated utility function is generated that is a node at level 0. If the procedure fails to give trade-off coefficients reproducing the decision-maker's preference, the search tree expands below nodes at this level. If not, no further node expansion occurs in this tree.

The modification procedure for utility functions is implemented, and has the configuration shown in Fig. 4. It is supported by an intelligent backtracking mechanism. The procedure is composed of hypothesis selection modules for each level and a control module for them. Each hypothesis selection module selects a hypothesis at the corresponding level from candidates for modification that are screened at upper levels. The control module controls the flow of procedural steps. It judges whether a corrective action is successful or not. If an action is judged unsuccessful, the module backtracks to an appropriate level and selects another hypothesis on the basis of the predicted cause of the failure.

2.3 Use of Heuristics

Heuristics are employed to enhance the efficiency of the search for trade-off coefficients. These search heuristics are not domain-specific knowledge but do-

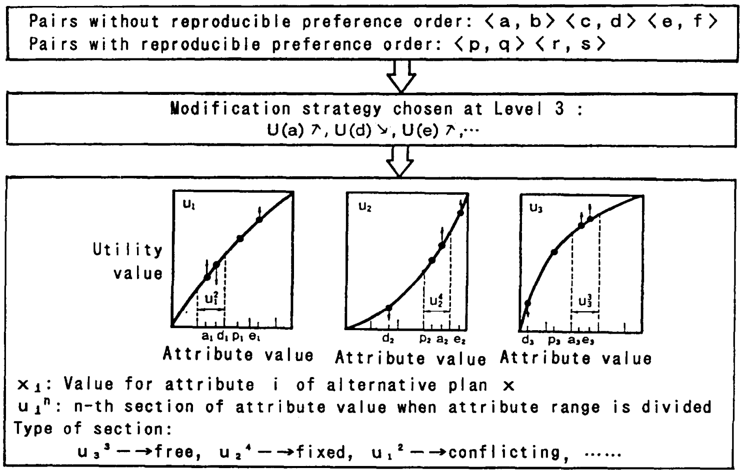


Fig. 6 Modification requirements for utility function values.

main-independent problem-solving knowledge, and help to guide the interactive flow for model identification. Two important steps guide a user in the knowledge acquisition process:

1. How to start the corrective action (support the process of initializing focus control for the modification process); and
2. What to do next if the corrective action fails (support the backtracking process caused by the modification failure).

A key to step 1 is the use of selection heuristics for bounding target objects to be modified, in order to structure the problem-solving process and suppress expansion of the search space. An efficient process for utility model identification is derived by the use of heuristics. Similarly, a key to step 2 is the use of control heuristics that realize an intelligent trial-and-error process in backtracking. This is particularily helpful in cases where modification failures are frequent. In the early phase of knowledge acquisition, naturally, step 1 is more important than step 2, as it guides the user in the interaction for knowledge acquisition.

2.3.1 Heuristics for Bounding Target Objects

For utility model identification, selection heuristics are available at each level. Examples are as follows:

1. Level 1: choice of a target sub-space.

A target sub-space is determined according to the majority decision principle, once the overall trade-off coefficient space has been divided into sub-spaces by the inequality planes. The primary and secondary rules at this level are as follows:

·Select the sub-space that satisfies the most inequalities.

·Select the sub-space with the most values available.

2. Level 2: choice of a target alternative plan.

Most alternative plans are assumed to have preference relations with more than one other plan. This suggests the possibility that modification of the overall performance of a plan is inconsistently requested by different plans. It is therefore necessary to select a target plan to be modified according to the importance of its preference relations with other plans.

Figure 5 shows the results of classified requests for modification of the overall performance *U* of each alternative plan. Here, alternative plans are classified into three types: free, fixed, and conflicting. The free type represents plans for which the direction and quantity of the modification are both uniquely determined. The fixed type includes plans for which the modification direction is uniquely fixed, while the conflicting type includes plans for which a contradictory modification direction is requested.

Furthermore, combinations of alternative plans that fail to reproduce the target preference order are classified into categories by using a pair of the types. For each category, selection rules are used to specify a target alternative plan to be modified in the course of knowledge acquisition. Six categories are important from the viewpoint of model modification:

·⟨free, free⟩
·⟨free, fixed⟩
·⟨free, conflict⟩
·⟨fixed, fixed⟩
·⟨fixed, conflict⟩
·⟨conflict, conflict⟩

Selection rules for each category are based on the heuristic that a plan with more degrees of freedom is preferred. The following are two examples of rules:

·Category ⟨free, fixed⟩: Select a free type of plan. This results in less influence on the other plans.

·Category ⟨fixed, fixed⟩: Select a plan with less possibility of disturbing the preference orders already established.

As an example, let us take the case in which ⟨x, y⟩ is not

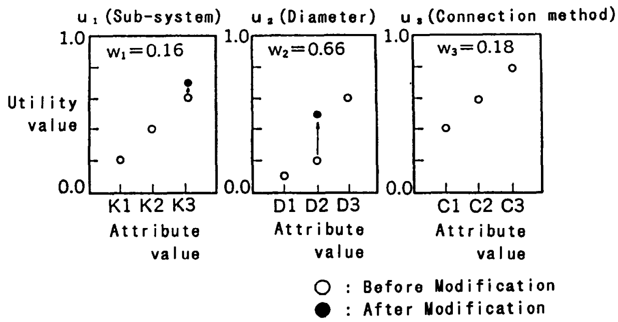


Fig. 7 Examples of utility function and trade-off coefficient.

reproducible and $\langle x, p\rangle$ and $\langle y, q\rangle$ are both reproducible. Increasing the overall performance of plan x is better than decreasing that of plan y, because the former has a higher possibility of preserving the preference order already reproduced.

3. Level 3: choice of target attribute

Level 3 of a modification tree handles the selection of a target attribute to be modified, using selection heuristics, which also concern the degree of freedom in the modification process. Once the modification direction has been fixed for the overall performance of an alternative plan, the same direction is assigned to a common policy for modifying the utility functions of the target attributes of the plan.

An example is shown in Fig. 6. Here, several regions are defined according to the range of the attribute value and the fixed modification direction. Then, regions of the attribute value are classified into three types as in the level-2 description. The order of selection priority is free, fixed, and conflicting, and a target attribute to be modified is chosen for each alternative plan. In cases where more than one attribute belongs to the class with the highest priority, the following detailed criteria are applied:

· If there is more than one attribute of the free type, select the target attribute of the target plan to be modified according to the number of attribute values in the range of the focused attribute. This selection results in wider coverage for modification.

· If there is more than one attribute of the fixed type, select the target attribute according to the likelihood of maintaining the preference order already reproduced in the course of knowledge acquisition.

4. Level 4: choice of amount to be modified

The amount of the attribute value to be modified is specified on the basis of the current value of the attribute and the direction of modification. The preference $U(x) > U(y)$ is realized for a pair of alternative plans $\langle x, y\rangle$ that include target objects to be modified. Various different values may be requested for the modification quantity, since the value of an attribute is related to more than one alternative plan. If more than one value is requested, the modification request with the greatest absolute value is chosen. After the value of each attribute has been fixed, utility functions are re-defined over the necessary range of attributes by fitting.

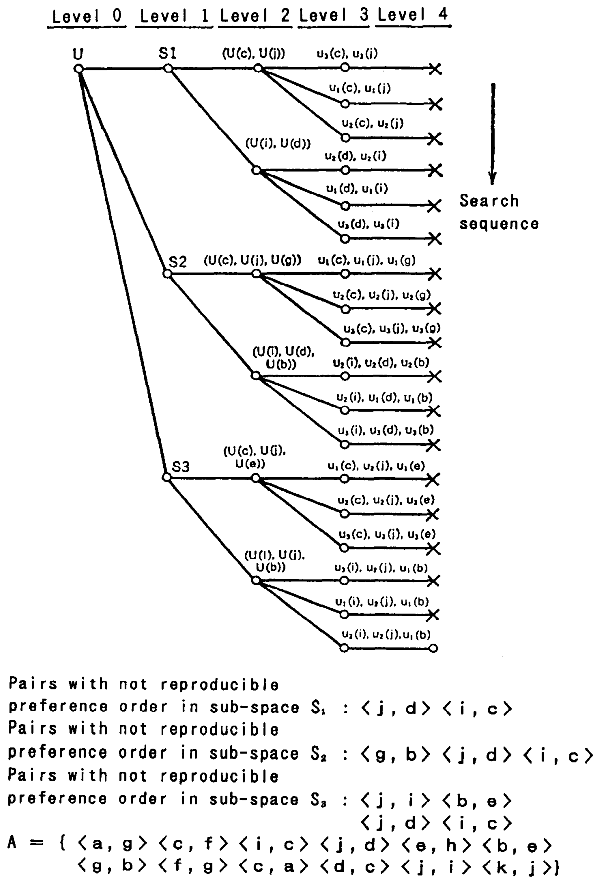


Fig. 8 Example of a modification process.

```
(IF   (and (No-of-pipes ¥section 0)              : cond1
           (type-of-task ¥task1 pipe-instal.)  : cond2
           (type-of-task ¥task2 pipe-instal.)  : cond3
           (eval-task-pref ¥task1 ¥task2))     : cond4

(THEN (pref-order ¥task1 ¥task2))
```

```
A meta-rule judges the preference order with an identified model
by comparing the overall performance of tasks.
        U(¥task1) > U(¥task1) ──→ cond4 satisfied
```

Fig. 9 Example of knowledge representation using a multi-attribute utility function.

2.3.2 Heuristics for Controlling Identification

A feasible region for all inequalities, which is the target sub-space, may not be found after modification of the utility functions and redefinition of inequalities. In this event it is necessary to backtrack and evaluate the effect of modification and reselection of the hypothesis.

The process of backtracking is guided by the use of control heuristics, which outline the step to which a process is backtracked and the hypothesis that is selected after the current selection failure. The following are examples:

1. Where to backtrack

(1) If the failure occurs just after the modification of a utility function at level 4, and no sub-space can be found that satisfies all inequalities, and the number of infeasible preference relations is reduced by the modification, then return to level 1 for a newly generated trade-off coefficient sub-space.

(2) If it is just after the modification of a utility function at level 4, and the gradient of the function has become opposite to that of the previous one, then return to level 3.

(3) If it is just after the selection of a target attribute at level 3, and all selected attributes are of conflicting type, then return to level 2.

2. What to select next after a failure

(4) If the process returns to step 3 after application of rule (2), and other attributes of free type are found there, then add them to the selected attributes.

(5) If step 2 has just been returned to after application of rule (3), and the other plan that makes a pair with the just-failed plan is found to be of free type, then select the plan as a target plan for the next trial.

3. Implementation and Evaluation

3.1 Application of Method

A prototype program has been implemented on the basis of the proposed method and experimentally applied to a knowledge acquisition problem extracted from actual knowledge base handling. The knowledge studied concerns the activity sequence of plant construction scheduling. Skeletal rules are extracted through interviews with plant construction engineers. One of them is as follows:

"If pipes are initially installed in a construction section, the sequence of installation activities simultaneously depends on the characteristics of the sub-system to which the pipe belongs, its diameter, and the method by which it is attached to components."

This rule is not applicable to general situations for handling pipes of different sub-systems, diameters, and connection methods, since it does not mention the preference order of pipe installation activities for combination of these factors. The rule should be refined so as to cover the detailed relationships among factors. The knowledge acquisition problem is to refine the skeletal rule so as to provide sufficient coverage of the combination of attributes in the preference decision-making.

First, the hypothesis of additive independence was confirmed to be acceptable in this case, and the initial guess was extracted for a utility function for each attribute, as shown in Fig. 7. Initial utility values are defined as their roughly estimated values, since the attribute values are discrete, not continuous. The upper and lower bounds for attribute values are engineers' subjective data obtained through interviews. Twelve cases were collected from previous scheduling activities by experts and used as basic data for determining the order of preference of pipe installation activities. The knowledge representation model with a set of utility functions is identified by the proposed method according to the basic data.

Figure 8 shows an example of a modification process of a utility function. At level 4 in the figure, the modification trials finally fail because of violation of constraints on the upper and lower bounds. Later trials are conducted with the use of 32 heuristic rules. This suggests that a proper model is not identified in sub-spaces $S1$ and $S2$. Trade-off coefficients derived from sub-space $S3$ cannot reproduce preference orders for the activity pairs $\langle i, c\rangle$, $\langle j, d\rangle$, $\langle b, e\rangle$, and $\langle j, i\rangle$. A satisfactory model that satisfactorily reproduces experts' judgment is identified in sub-space $S3$ at the stage when i, j, and b are chosen for target activities to be modified, $U2$ is chosen for i and j, and $U1$ for b as attributes to be modified.

The prototype program is written in Common LISP. The final trade-off coefficients are also shown in Fig. 7. The computer use time for this case is about 30 minutes on a workstation with a speed of about 1 MIPS. This result demonstrates that model identification can be completed with a reasonably small computing effort. A model identified as trade-off coefficients by the above-mentioned procedure is transformed into knowledge in the rule form shown in Fig. 9. The predicate "eval-task-pref" in the fourth conditional clause is related to a meta-rule that judges the preference between items in arguments. Being invoked from the rule, the meta-rule calls the pre-defined function that compares the priority of activities specified, and determines their order of preference.

A set of a rule, meta-rule, and procedural function completes the general knowledge representation scheme available for various kinds of pipe installation activities in plant construction scheduling. Once this triplet has been given to a knowledge base, the system can be easily extended to accept tasks of a new type and to automate the ordering of activities in a schedule chart.

This result suggests that more refined judgmental knowledge is semi-automatically acquired from previous cases judged by an expert on the basis of the proposed method, if the assumption on the additive independence of attributes is valid. It is clear in the attributes related to activities of pipe installation that the assumption is acceptable, because no attribute value has any influence on other attribute values. The proposed method offers a general means for representation of judgmental knowledge to establish the preference order of objects from multi-attribute values.

If an attribute value depends on other attribute values, the assumption concerning attributes cannot be approved. Even then, however, the proposed method is applicable in principle, though choice of a target attribute is complicated at level 4 of the model adjustment. This modification step requires careful treatment in order to fix some attribute values as constant and to reduce the number of target attributes until they obey the condition mentioned in the assumption of additive independence.

One of the key issues in building better knowledge acquisition tools is visualization of the acquisition process. A more elaborate user interface should be provided to display the sub-space in the trade-off coefficient space, and to plot utility functions. The user interface makes it easier to grasp the whole process of model identification and to acquire new heuristics.

3.2 Construction Scheduling System with a Knowledge Acquisition Function

Two kinds of difficulties are anticipated in the process of putting the proposed method to practical use on a large scale:

1. Massive computer efforts due to the size of real world problems such as the number of attributes and the number of previous cases, and
2. Massive human efforts for experts and users to review cases and identify utility functions.

In the domain of construction scheduling, from which an example is taken, practical schedules are essentially planned on a trial-and-error basis. Unlike in other engineering scheduling domains, standardized guidelines and reference cases have not yet been established or computerized in this domain. Scheduling tasks including activity sequence choice are processed by a human expert with the support of highly interactive scheduling systems.

Unit sections for construction scheduling in industrial plants include hundreds of objects to be installed, such as components and pipes. Each object has several major attributes, though the number depends on the type of object. Scheduling experts seem to pay more attention to two or three attributes at a high abstract level than to the objects themselves when making decisions on the activity sequence. In view of the current style of problem-solving by human experts, it is suggested that expert systems for construction scheduling should be based on the man-machine interaction process.

Most domain knowledge to be extracted is relatively simple and is written as rules with a limited number of conditional clauses. In the course of putting the proposed method to practical use, the first of the above-mentioned difficulties is less serious than the second and, therefore, countermeasures to cope with the second are more necessary in light of this.

This direction offers a promising approach. Thanks to advanced computer graphics technology, simulation systems have been computerized and utilized as practical design tools in the field of plant engineering [16]. Such systems are also promising as tools for visualizing the time-dependent process of plant construction, which contributes to quicker understanding of the state of construction in a target section and to a more detailed evaluation of the timing of transportation and installation of components, pipes, and so on.

As a framework for integrating this knowledge acquisition method with an overall construction scheduling system, simulation systems play a key role in the real-time collection of cases showing experts' preferences. It is not easy in this domain for a knowledge engineer to extract empirical knowledge with high availability. Expert systems with incomplete knowledge often draw a preference conclusion incompatible with an actual expert's preference. It is natural for this knowledge representation model to accommodate a variety of preference cases and to emulate human expert judgment in preference decision-making through structured modification of utility functions.

The knowledge acquisition and refinement process can be executed on a workstation as a background job to the main task of specifying the activity sequence by man-machine interaction with a simulation tool. The interactive scheduling process with a simulation tool leads to integration of the knowledge-based problem-solving and knowledge acquisition, which is one route to alleviating the knowledge acquisition bottleneck.

4. Concluding Remarks

A method for acquiring judgmental knowledge from experts on the order of preference of alternative plans has been developed on the basis of the knowledge representation model of the multi-attribute utility model, which is frequently applied to multi-criteria decision-making. In this method, knowledge is semi-automatically extracted from previous preference cases through model identification, if the attributes to be referred to are known but detailed and combined relationships between attributes are not available.

Model identification is completed in two steps: Comparison of the preferences of alternative plans calculated by a human expert and by the model, and modification of the model with feedback from the comparison result. In the latter step, the model is modified by two kinds of modules, namely, the hypothesis selection modules, to limit the target objects to four levels, and a control module for overall procedural control.

Heuristics for hypothesis selection are useful in realizing an efficient modification process. From the results of experimental application to an activity sequence problem in construction scheduling, the system successfully identified knowledge that reproduced the sequence determined by an expert planner. This suggests that the proposed method is applicable to the extraction of knowledge that determines the order of preference of objects by balancing several attribute values, though this kind of knowledge is not easily acquired in the framework of conventional rule refinement.

Interactive computation efforts for utility model identification were reduced by integrating the proposed method with a scheduling support system. The proposed method is a promising way of overcoming the knowledge acquisition bottleneck in building and using practical expert systems in engineering domains. Future studies should be directed to the handling of preference cases without additive independence and preference cases with multiple experts.

References

1. KAHN, G. et al. MORE: An Intelligent Knowledge Acquisition Tool, *Proc. of IJCAI-85* (1985), 581-584.
2. KAHN, G. et al. Strategies for Knowledge Acquisition, *IEEE Trans. on PAMI*, PAMI-7, 30 (1985), 511-522.
3. MARCUS, S. Knowledge Acquisition for Constructive Systems, *Proc. of IJCAI-85* (1985), 637-639.
4. MARCUS, S. Taking Backtracking with a Grain of SALT, *J. Man-machine Studies*, **26**, 4 (1987), 383-398
5. KAWAGUCHI, J. et al. Interactive Logic Design Support System for Data Base through Interviews with Users (in Japanese), *J. Inst. E.I.C.*, J70-D, 11 (1987), 2243-2249.
6. KAWAGUCHI, J. et al. SIS: A Shell for Interview Systems, *Proc. 1st Mng. J. Soc. A.I.* (1987), 6-21.
7. SATTY, T. L. The Analytic Hierarchy Process, McGraw-Hill, New York (1980).
8. BELTON, V. and GEAR, T. On a Shortcoming of Satty's Method of Analytic Hierarchies, *Omega* 11 (1983), 228-230.
9. KEENEY, R. L. and RAIFFA, H. Decisions with Multiple Objectives, Preferences and Value Tradeoffs, John-Wiley & Sons, New York (1976).
10. ICHIKAWA, A. Theory and Method for Multi-Criteria Decision-Making (in Japanese), SICE (1980).
11. TONE, K. Game-Styled Decision-Making (in Japanese), Nikka-Giren, Tokyo (1986).
12. NOMURA, T. et al. Application of Multi-Criteria Optimization Method to Design, Planning and Evaluation Problems in Industries (in Japanese), *Systems and Control*, **28**, 11 (1984), 651-659.
13. TAMURA, H. Multi-Criteria Evaluation and Knowledge Engineering for FMS Scheduling (in Japanese), *Preprint of 3rd Mtg. of the SIG for Knowledge Engineering at Osaka Univ.* (1986).
14. KOBAYASHI, S. Multi-Criteria Decision-Making—Knowledge Engineering Methodology (in Japanese), *Systems and Control*, **31**, 4 (1987), 275-285.
15. BOOSE, J. H. Use of Repertory Grid-Centered Knowledge Acquisition Tools for Knowledge-Based Systems, *Proc. of 2nd AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop* (1987), 2.0-2.19.
16. YOSINAGA, T. et al. A CAD System for Three-Dimensional Plant Layout Planning (in Japanese), *Hitachi-Hyoron*, **68** (1986), 325-330.