DEX: An Expert System Shell for Decision Support[•]

Marko Bohanec[1] and **Vladislav Rajkovič**[2,1]

[1] Jožef Stefan" Institute, Ljubljana, Yugoslavia
[2] Faculty of Organizational Sciences, Kranj, Yugoslavia

ABSTRACT

An approach to decision making that integrates multi-attribute decision techniques with expert systems is described. The approach is based on the explicit articulation of qualitative decision knowledge which is represented by a tree of attributes and decision rules. The decision making process is supported by a specialized expert system shell for interactive construction of the knowledge base, evaluation of options and explanation/analysis of the results. Practical use of the shell is illustrated by an application in the field of performance evaluation of enterprises.

KEYWORDS:
Expert systems, knowledge representation, decision support, multi-attribute decision making.

1. INTRODUCTION

In this paper a software package named *Decision EXpert* (*DEX* for short) is presented. It is based on the methodology that combines multi-attribute decision making with expert systems (Bohanec and Rajkovič, 1987). DEX itself is designed as an interactive *expert system shell* that provides tools for building and verifying a knowledge base, evaluating options and explaining the results. The structure of the knowledge base and evaluation procedures closely correspond to the multi-attribute decision making paradigm. This makes the system specialized for decision support.

The following two sections give some basic explanations related to decision making and expert systems, respectively. The principles of integrating these two approaches in DEX, which are reflected primarily in the structure of the knowledge base, are presented in section 4. Section 5 describes DEX as a software system. Finally, a practical application of DEX for performance evaluation of public enterprises is illustrated in section 6.

[•] A reformatted version of the article published in *Sistemica* 1(1), 145–157, 1990.

2. DECISION MAKING

Decision making is a process of selecting a particular option from a set of possibles, so as to best satisfy the aims or goals of the decision maker (Efstathiou and Rajkovič, 1979). In practice, the *options* (also called *alternatives*) are objects or actions of (approximately) the same type, such as different computer systems, different people applying for a particular job, or different investment strategies.

Decision making problems occur daily in almost any field of human activity, ranging from everyday personal decisions to difficult problems in economy, management, medicine, etc. The difficulty of some problems is caused by their complexity which mainly originates in:

- a large number of parameters that influence the decision,
- incomplete, uncertain or conflicting goals and/or knowledge,
- numerous and/or loosely defined options,
- the presence of different decision making groups with different objectives, and
- time constraints imposed upon the decision making process.

Supporting humans in making complex decisions has long been a goal of many researchers and practitioners. A number of methods and computer-based systems have been developed (Humphreys and Wisudha, 1987). They are mainly studied in the framework of decision support systems (Keen and Scott Morton, 1978; Alter, 1980; Turban 1988), operations research and management sciences, decision theory (French, 1986) or decision analysis (Phillips, 1986).

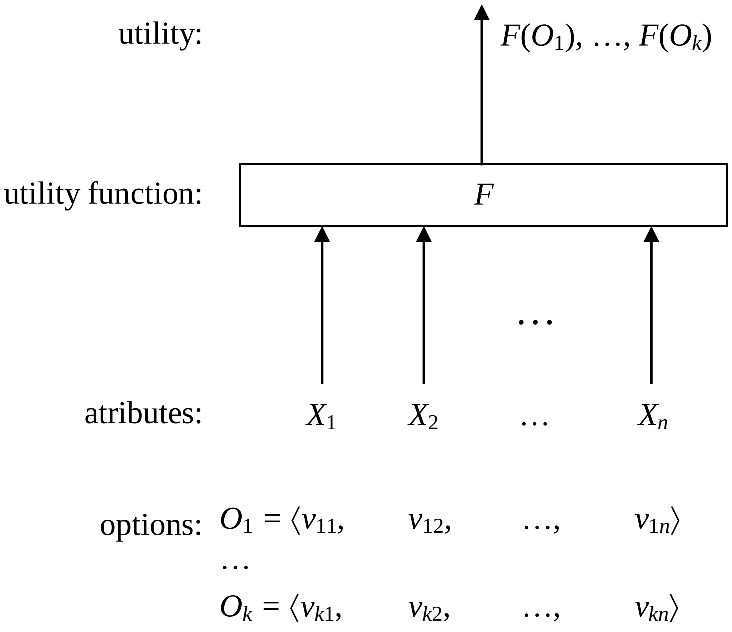


Fig. 1: General concept of multi-attribute decision making.

One of the approaches to decision support, which is widely used in practice, is *multi-attribute* decision making (Keeney and Raiffa, 1976; Chankong and Haimes, 1983). The basic principle is a decomposition of the decision problem into smaller, less complex subproblems (Fig. 1). Options are decomposed onto different dimensions *X*, usually called *attributes*, *parameters* or *criteria*. According to this decomposition, each option *O* is first described by a vector of values *v* of the corresponding attributes. The vectors are then evaluated by a *utility function F*. This function should be previously defined by the decision maker(s), representing his, her or their *goals*.

When applied upon a particular option *O*, the function *F* yields a *utility F*(*O*). According to this value, the options can be ranked and/or the best one chosen.

In the multi-attribute paradigm, the decision makers' *knowledge* about a particular decision problem is therefore *described* by *attributes X* and a *utility function F*. In addition, there is a data base of *options*, consisting of vectors *v*.

3. EXPERT SYSTEMS

Expert systems are intelligent information system that behave, in a certain sense, as a human expert in the application domain (Michie, 1979; Waterman, 1986; Goodall, 1985). A major new feature introduced by the methodology of expert systems is the ability to *explain* their decisions in user understandable terms.

Expert systems are typically composed of two modules: (1) a knowledge base and (2) an inference engine.

The *knowledge base* contains the knowledge about a particular problem domain. Knowledge can be represented in various forms. The most common ones are production (if-then) rules, semantic nets and frames (Goodall, 1985). In addition, these formalisms are usually capable of dealing with imprecision, uncertainty, and qualitative (non-numeric) nature of the expert knowledge.

The *inference engine* (1) solves problems stated by the user by applying a certain reasoning procedure upon the knowledge base and (2) generates user-oriented explanations of the solutions.

The modularity of expert systems allows a single inference engine to be connected with different knowledge bases, thus obtaining different expert systems that solve different (although more or less similar) problems. For this reason, *expert system shells* have emerged which are effectively "empty" expert systems. In this case, the user has an opportunity to build his or her own knowledge base. The shells are composed of an inference engine and, usually, different modules that support the building of the knowledge base by means of, for example, machine learning (Michie and Bratko, 1986) or question-answer dialogues (Bohanec and Rajkovič, 1988).

The concept of expert systems offers a number of potentials for decision support (Efstathou and Mamdani, 1986; Turban, 1988; Vari and Vecsenyi, 1988). With some adaptations to the field of decision making (Bohanec et al., 1988), this approach can improve the effectiveness of decision support in terms of:

- knowledge elicitation, verification and learning,
- explanation of the decisions,
- analysis of options, and
- handling qualitative ("soft") knowledge.

4. KNOWLEDGE REPRESENTATION IN DEX

The integration of expert systems and multi-attribute decision making in DEX is based on the explicit articulation of *knowledge* about a specific decision making problem. The structure of the knowledge base closely corresponds to the multi-attribute schema shown on Fig. 1. The main differences, influenced by expert systems, are:

1. Attributes are purely *qualitative* in DEX (as opposed to numerical attributes in the traditional decision making methods). They can take values from discrete and (optionally) ordered value domains. The values are usually *words* like “high” or “good”, or *intervals* of numerical values, for example “$100:$250”.
2. In DEX, utility functions are defined by sets of rules referred to as *elementary decision rules*. This is much different to the traditional methods where the functions are specified by a given analytical formula such as the weighted sum.The use of rules alleviates the application of expert system explanation techniques (Bohanec and Rajkovič, 1989).

In addition to Fig. 1, the attributes can be hierarchically structured into a *tree of attributes* in DEX. However, this is a common technique used in several traditional decision making methods (Humphreys and Wisudha, 1987). Such trees are also referred to as semantic trees, criteria trees or concept trees.

A particular *knowledge base* of DEX therefore consists of (1) tree of attributes and (2) utility functions (Fig. 2).

A *tree of attributes* represents the structure of a given decision problem. The attributes are structured according to their interdependence: a higher-level attribute depends on its descendants (sons) in the tree. Leaves of the tree, referred to as *basic attributes*, depend solely on the characteristics of options. Internal nodes of the tree are called *aggregate attributes*. Their values are determined on the basis of utility functions. The most important aggregate attribute is the root of the tree. Its purpose is to represent the overall *utility* of options.

Utility functions define the process of aggregation of lower-level attributes into the corresponding higher-level fathers. For each aggregate attribute X, a utility function F that maps values of sons of X into values of X, should be defined by the decision maker.

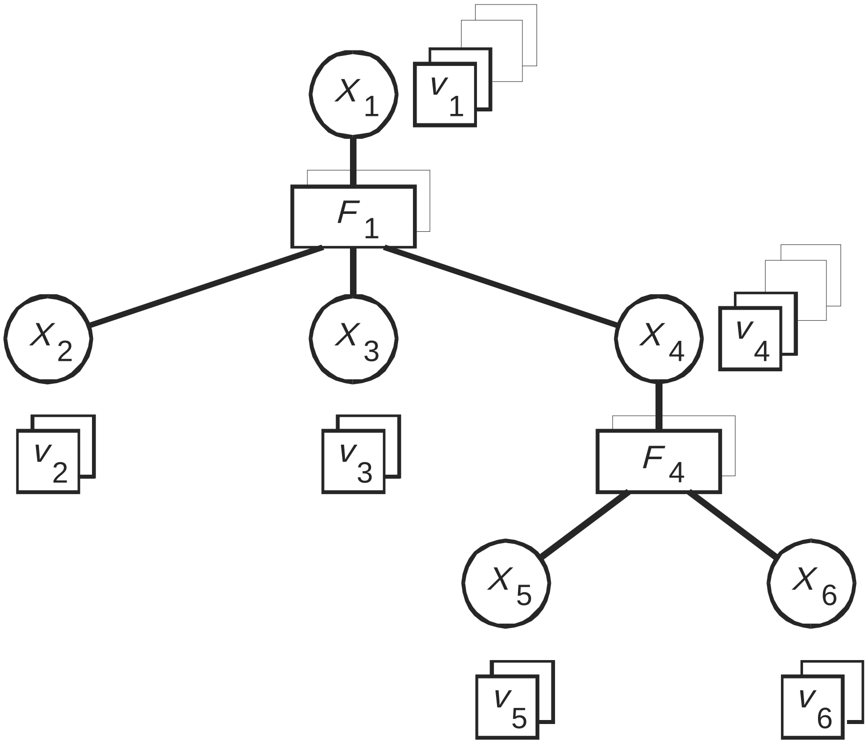


Fig. 2: Tree of attributes with utility functions and options.

Utility functions are represented by *elementary decision rules*. Let X_1, X_2, …, X_k be the sons of an aggregate attribute Y. Then, the function $Y = F(X_1,X_2,\ldots,X_k)$ is defined by a set of rules of the form

if $X_1 = x_1$ **and** … **and** $X_k = x_k$ **then** $Y = y_m : y_M$,

where x_i, y_m and y_M represent the values of the corresponding attributes. "$y_m : y_M$" stands for an interval of values between y_m and y_M, inclusive. Most commonly, y_m:y_M is a single-value interval. In this case, the rule is simplified to

if $X_1 = x_1$ **and** … **and** $X_k = x_k$ **then** $Y = y$.

Sets of elementary decision rules are grouped into tables.

In the case when more decision making *groups* with different objectives are involved in the decision process, each group can define their own set of utility functions. In Fig. 2, two such groups are assumed.

Options are represented by the values of basic attributes, i.e. by values v_1, v_2, …, that are assigned to the leaves of the tree. In Fig. 2, two options are assumed. Regardless of the number of groups, there can be only one value v_i assigned to a basic attribute for each option.

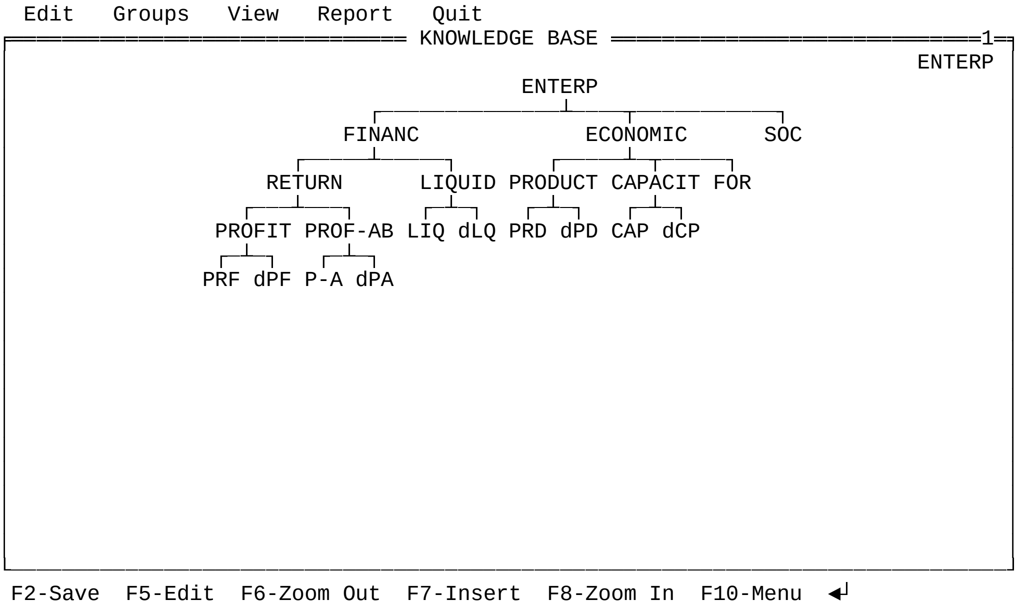
In the final stage of the decision making process, the above described components of the knowledge base are utilized in order to *evaluate options*, i.e. to determine the values of the root and the remaining aggregate attributes in the tree. Since there can be more than one group of utility functions, the evaluation process can result in several sets of aggregate evaluation results, as shown by separate sets of squares in Fig. 2.

5. THE DEX EXPERT SYSTEM SHELL

DEX is an interactive expert system shell implemented for the IBM PC/XT/AT/PS computers and true compatibles. It consists of two parts: (1) knowledge base building tools and (2) tools for the evaluation and analysis of options.

The expert system *building* phase is basically supported by two specialized editors of the two components of the knowledge base, the tree of attributes and utility functions.

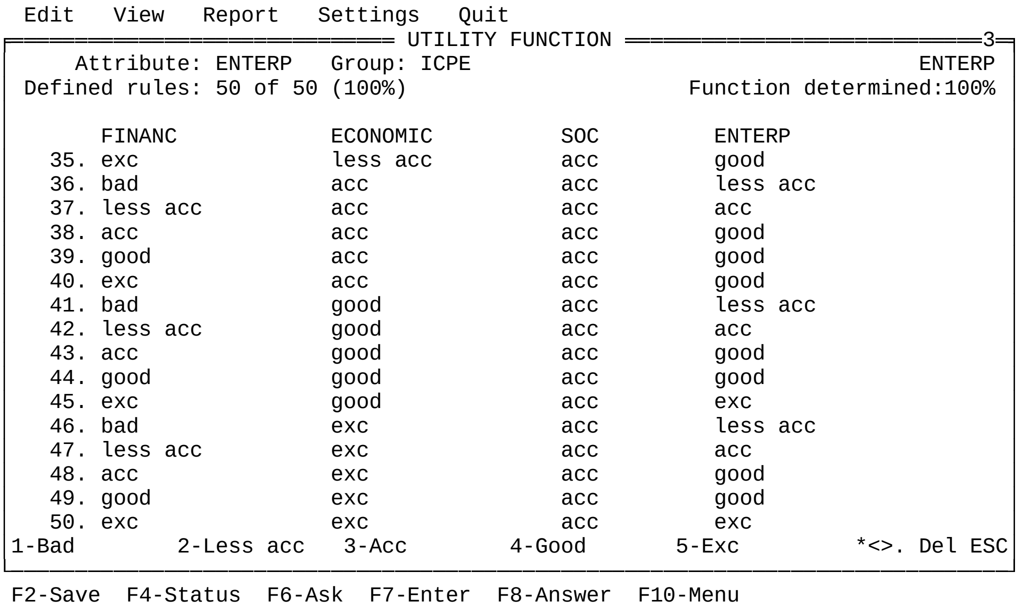
The tree editor (Fig. 3) allows the user to insert new attributes into the tree. Components of attributes, such as their names and values, can be edited and/or copied around the knowledge base. Subtrees of attributes can be copied, moved and deleted as well. If necessary, the underlying utility functionsare adjusted automatically.



```
  Edit   Groups   View   Report   Quit
╒═══════════════════════════════ KNOWLEDGE BASE ═════════════════════════════1═╕
│                                                                       ENTERP │
│                                        ENTERP                                │
│                            ┌──────────────┴────┬───────────┐                 │
│                          FINANC             ECONOMIC      SOC                │
│                      ┌─────┴─────┐       ┌─────┴─┬─────┐                     │
│                    RETURN      LIQUID PRODUCT CAPACIT FOR                    │
│                  ┌───┴───┐     ┌─┴─┐   ┌─┴─┐   ┌─┴─┐                         │
│                PROFIT PROF-AB LIQ dLQ PRD dPD CAP dCP                        │
│                ┌─┴─┐   ┌─┴─┐                                                 │
│               PRF dPF P-A dPA                                                │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
└──────────────────────────────────────────────────────────────────────────────┘
 F2-Save  F5-Edit  F6-Zoom Out  F7-Insert  F8-Zoom In  F10-Menu  ◄┘
```

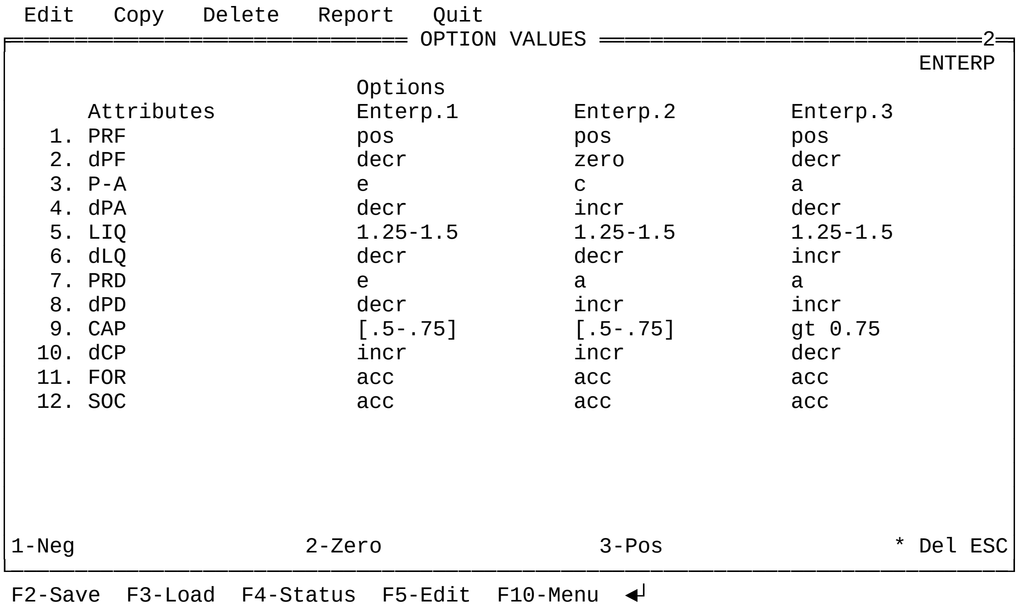
Fig. 3: The tree editor of DEX showing attributes for performance evaluation of enterprises.

For the definition, verification and justification of utility functions, a spreadsheet-like editor is provided (Fig. 4). DEX prepares a table of elementary decision rules with unknown values, which are then filled-in by the user (these values are placed in the rightmost column in Fig. 4).



```
  Edit   View   Report   Settings   Quit
╒═════════════════════════════ UTILITY FUNCTION ═════════════════════════════3═╕
│     Attribute: ENTERP   Group: ICPE                                    ENTERP │
│ Defined rules: 50 of 50 (100%)                      Function determined:100% │
│                                                                              │
│       FINANC           ECONOMIC          SOC         ENTERP                  │
│   35. exc              less acc          acc         good                    │
│   36. bad              acc               acc         less acc                │
│   37. less acc         acc               acc         acc                     │
│   38. acc              acc               acc         good                    │
│   39. good             acc               acc         good                    │
│   40. exc              acc               acc         good                    │
│   41. bad              good              acc         less acc                │
│   42. less acc         good              acc         acc                     │
│   43. acc              good              acc         good                    │
│   44. good             good              acc         good                    │
│   45. exc              good              acc         exc                     │
│   46. bad              exc               acc         less acc                │
│   47. less acc         exc               acc         acc                     │
│   48. acc              exc               acc         good                    │
│   49. good             exc               acc         good                    │
│   50. exc              exc               acc         exc                     │
│1-Bad        2-Less acc   3-Acc        4-Good       5-Exc         *<>. Del ESC│
└──────────────────────────────────────────────────────────────────────────────┘
 F2-Save  F4-Status  F6-Ask  F7-Enter  F8-Answer  F10-Menu
```

Fig. 4: Editor of decision rules; a part of the ENTERP table of decision rules is shown.

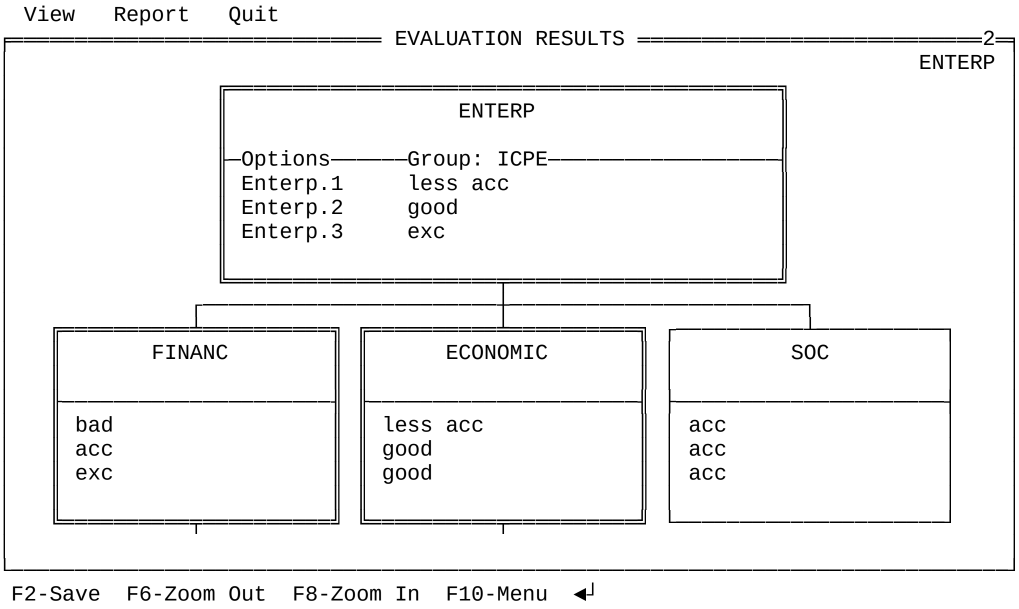


```
  Edit   Copy   Delete   Report   Quit
╒═════════════════════════════ OPTION VALUES ════════════════════════════════2═╕
│                                                                        ENTERP │
│                          Options                                             │
│      Attributes          Enterp.1         Enterp.2         Enterp.3          │
│   1. PRF                 pos              pos              pos               │
│   2. dPF                 decr             zero             decr              │
│   3. P-A                 e                c                a                 │
│   4. dPA                 decr             incr             decr              │
│   5. LIQ                 1.25-1.5         1.25-1.5         1.25-1.5          │
│   6. dLQ                 decr             decr             incr              │
│   7. PRD                 e                a                a                 │
│   8. dPD                 decr             incr             incr              │
│   9. CAP                 [.5-.75]         [.5-.75]         gt 0.75           │
│  10. dCP                 incr             incr             decr              │
│  11. FOR                 acc              acc              acc               │
│  12. SOC                 acc              acc              acc               │
│                                                                              │
│                                                                              │
│                                                                              │
│                                                                              │
│1-Neg                  2-Zero                 3-Pos                 * Del ESC│
└──────────────────────────────────────────────────────────────────────────────┘
 F2-Save  F3-Load  F4-Status  F5-Edit  F10-Menu  ◄┘
```

Fig. 5: Editor of descriptions of options; the columns represent basic values of three enterprises.

The editing process is continuously monitored by a subsystem for checking consistency and monotonicity of rules (Bohanec and Rajkovič, 1988). Its purpose is to warn the user in the case when a new rule (value) contradicts to the already defined rules.

In addition, a number of methods is implemented that translate the original representation of knowledge (elementary rules) into different other forms which show the same knowledge from different viewpoints using different levels of detail (Bohanec et al., 1988). Examples of such representations are graphics, decision trees and complex decision rules. The main role of these methods is in supporting a better understanding of the underlying knowledge and its justification (Rajkovič et al., 1988).



```
EVALUATION RESULTS                            2
                                          ENTERP

                     ENTERP

  Options        Group: ICPE
  Enterp.1       less acc
  Enterp.2       good
  Enterp.3       exc

       FINANC            ECONOMIC             SOC
  bad               less acc           acc
  acc               good               acc
  exc               good               acc
```

Fig. 6: Performance evaluation of three enterprises: the topmost two levels of attributes.

After the knowledge base has been built, the second main part of DEX, i.e. *evaluation and analysis* of options, can be applied. At the beginning, the user activates a specialized editor of options in order to describe the options by assigning the corresponding values to basic attributes (Fig. 5). After this, DEX automatically evaluates the options. The analysis of the results can follow which consists of one or more of the following activities (Bohanec and Rajkovič, 1989):

1. *Interactive inspection* of the results by "walking" around the tree and looking at the values that were assigned to aggregate attribute during the evaluation (Fig. 6).
2. *Explanation* of the evaluation: DEX can explain how each particular value has been obtained in terms of attributes' values involved in the process, triggered rules and descriptions of computations performed by DEX.
3. *What-if* analysis is performed interactively by changing values of basic attributes, reevaluating options and comparing the obtained results with the original ones.
4. *Selective explanation* of options: DEX finds and reports those subtrees that expressed the most advantageous or disadvantageous characteristics of a particular option (an example is shown in Fig. 8). The main point is in the explanation of options using only the most relevant information.

In the design of DEX, one of the most important goals was the transparency to the user. For this reason, the user can access a powerful *report generator* during all the stages of working with DEX. The generator is able to prepare complete or partial reports showing the components of the knowledge base, evaluation/analysis results and different kinds of explanation. The user can choose between different levels of detail and different forms of representation. The reports can be inspected interactively or printed out.

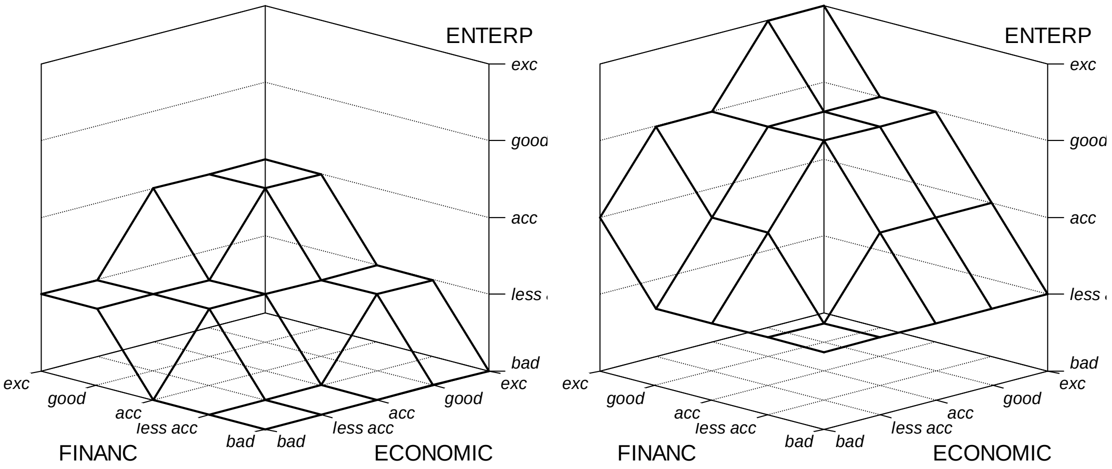


Fig. 7: Graphic representation of the ENTERP utility function;
SOC="unacc" on the left and SOC="acc" on the right.

6. DEX IN PERFORMANCE EVALUATION OF ENTERPRISES

In order to illustrate the above general statements about DEX, let us describe one of its applications for performance evaluation of public enterprises. Originally, this application was developed by Manuel Olave and Armando Barrera at the International Center for Public Enterprises (ICPE), Ljubljana. It was implemented using DECMAK, a research predecessor of DEX (Barrera and Bohanec, 1987). For the purpose of this section, this knowledge base has been reimplemented in DEX.

Actually, this application was mainly an experiment, testing the feasibility of this kind of approach for performance evaluation of enterprises. In the first stage, a knowledge base for performance evaluation was designed. This model was then compared to an existing performance evaluation model (called Signalling System), implemented in Pakistan. The comparison was done according to data about 54 public enterprises from Pakistan.

The tree of attributes, designed for this experiment, is shown on Fig. 3. The performance of enterprises is measured with three main groups of attributes: FINANCial, ECONOMIC and SOCial.

The FINANC attribute depends on RETURN and LIQUIDity. In turn, RETURN is decomposed into PROFIT and PROFit-ABility (representing the relation between profits and equity). PROFIT is measured according to the absolute value of profit in the evaluation period (PRO) and its trend relative to the previous period (dPR). The same approach of measuring the absolute value and its trend is used with the majority of the remaining attributes, i.e. PROF-AB, LIQUIDity and, in the ECONOMIC subtree, PRODUCTivity and CAPACITY utilization.

According to the fact that there were no data available for measuring SOCial and generation of FOReign exchange, these were not elaborated in detail. Rather, they were dealt with as basic attributes with fixed values for all the enterprises.

The above tree consists of 12 basic attributes (the leaves) and 9 aggregate ones (internal nodes). For each aggregate attribute, a set of decision rules was defined that map combinations of values of lower-level attributes into the values of that attribute. There are 9 sets of such rules, grouped into tables. A part of the table that determines the overall enterprise performance (ENTERP) according to FINANCial, ECONOMIC and SOCial, is given in Fig. 4. The same table is represented graphically in Fig. 7.

The basic data of three enterprises considered in the experiment are shown on Fig. 5. According to the rules in the knowledge base, the performance of these enterprises evaluates to “less acceptable"” “good” and “excellent”, respectively (Fig. 6). In Fig. 8, the “less acceptable” value of the first enterprise is selectively explained by highlighting its most advantageous and disadvantageous characteristic.

```
DISADVANTAGES                 ADVANTAGES

Attribute          Value      Attribute          Value

├─FINANC           bad        └─SOC              acc
│ ├─RETURN         bad
│ │ └─PROF-AB      bad        │ └─FOR            acc
│ │   ├─P-A        e
│ │   └─dPA        decr       │ │ └─dCP          incr

│ ├─PRODUCT        bad        │ │ │ ├─PRF        pos
│ │ ├─PRD          e
│ │ └─dPD          decr

│   └─dLQ          decr

│ │ │ └─dPF        decr
```

Fig. 8: Selective explanation of Enterprise 1.

In total, 54 enterprises were considered in the experiment. There were 10 enterprises evaluated differently compared to the reference model. They were additionally treated by the experts who considered that the knowledge-based model gave better results in 8 cases. In the remaining two cases, the reference model was superior.

The conclusion of this exercise was that it was feasible to use the expert system approach for performance evaluation of enterprises. In addition to better evaluation results, the following advantages over the traditional approach were identified:

- explicitness and transparency of the model,
- ability to handle qualitative concepts,
- ability to handle incomplete and uncertain information,
- powerful and user understandable explanation and analysis of enterprises.

7. CONCLUSION

The approach to decision making presented above integrates two technologies: multi-attribute decision making and expert systems. The expert system approach is reflected in

- qualitative domains of attributes,
- rule-based form of utility functions,
- the structure of the expert system shell which is composed of knowledge elicitation tools and an inference engine, with special emphasis on the explanation (transparency) of the knowledge and activities of the system.

On the other hand, the knowledge base is structured similarly to multi-attribute methods. It consists of tree-structured attributes and utility functions. The knowledge base serves as a model for the evaluation of options.

The DEX expert system shell and its research prototype, DECMAK, have been successfully applied in over 30 practical applications. The best performance has been observed in the following fields:

1. Evaluation of a kind of technology (there are 14 applications of evaluating computer hardware and software, radar systems, nuclear waste disposals and similar);
2. Personnel management (4 applications, such as expert team selection, performance evaluation of professionals and directing children to sports);
3. Management (evaluation of trading partners, tenders and enterprise performance).

REFERENCES

Alter, S.L.: *Decision Support Systems - Current Practice and Continuing Challenges*, Addison-Wesley, 1980.

Barrera, A., Bohanec, M.: *An Experiment with DECMAK in the Domain of Performance Evaluation of Public Enterprises*, Report DP-4732, Institut Jožef Stefan, Ljubljana, 1987.

Bohanec, M., Rajkovič, V.: *An Expert System Approach to Multi-Attribute Decision Making*, in Hamza, M.H. (ed.): IASTED International Conference “Expert Systems”, Geneva, Acta Press, 1987.

Bohanec, M., Rajkovič, V.: *Knowledge Acquisition and Explanation for Multi-Attribute Decision Making*, 8th International Workshop “Expert Systems and Their Applications”, Vol. 1, Avignon, 1988.

Bohanec, M., Rajkovič, V., Lavrac, N.: *Knowledge Explanation in Expert Systems: A Decision Support System and Machine Learning View*, IASTED International Conference "Expert Systems", Geneva, Acta Press, 1988.

Bohanec, M., Rajkovič, V.: *Knowledge-Based Explanation in Multi-Attribute Decision Making*, *Produktivnost* XXX, 2, 182–189, Belgrade, 1989.

Chankong, V., Haimes, Y.Y.: *Multiobjective Decision Making: Theory and Methodology*, North-Holland, 1983.

Efstathiou, J., Mamdani, E.H.: *Expert Systems and How They are Applied to Industrial Decision Making*, in Mitra, G. (ed.): *Computer Assisted Decision Making*, Elsevier Science Publishers B.V. (North-Holland), 1986.

French, S.: *Decision Theory*, Ellis Horwood Limited, 1986.

Goodall, A.: *The Guide to Expert Systems*, Learned Information Ltd., 1985.

Humphreys, C.P., Wisudha, D.A.: *Methods and Tools for Structuring and Analysing Decision Problems*, Decision Analysis Unit, The London School of Economics and Political Sciences, Tech. Rep. 87-1, London, 1987.

Keen, P.G.W., Scott Morton, M.S.: *Decision Support Systems – An Organizational Perspective*, Addison-Wesley, 1978.

Keeney, R.L., Raiffa, H.: *Decisions with Multiple Objectives*, John Wiley & Sons, New York, 1976.

Michie, D. (ed.): *Expert Systems in The Microelectronic Age*, Edinburgh University Press, 1979.

Michie, D., Bratko, I.: *Expert Systems: Automating Knowledge Acquisition*, Addison-Wesley, 1986.

Phillips, L.D.: *Decision Analysis and its Applications in Industry*, in Mitra, G. (ed.): *Computer Assisted Decision Making*, Elsevier Science Publishers B.V. (North-Holland), 1986.

Rajkovič, V., Bohanec, M., Batagelj, V.: *Knowledge Engineering Techniques for Utility Identification*, *Acta Psychologica* 68, 1988.

Turban, E.: *Decision Support and Expert Systems*, Macmillian, 1988.

Vari, A., Vecsenyi, J.: *Concepts and Tools of Artificial Intelligence for Human Decision Making*, *Acta Psychologica* 68,1988.

Waterman, D.A.: *A Guide to Expert Systems*, Addison-Wesley, 1986.