KATHOLIEKE
UNIVERSITEIT
LEUVEN

DEPARTEMENT TOEGEPASTE ECONOMISCHE WETENSCHAPPEN

RESEARCH REPORT 9908

VERIFICATION AND VALIDATION OF KNOWLEDGE-BASED SYSTEMS WITH AN EXAMPLE FROM SITE SELECTION

by
G. WETS
F. WITLOX
J. VANTHIENEN

D/1999/2376/08

VERIFICATION AND VALIDATION OF KNOWLEDGE-BASED SYSTEMS WITH AN EXAMPLE FROM SITE SELECTION

GEERT WETS, FRANK WITLOX & JAN VANTHIENEN

Limburg University — LUC. Department of Business Economics, Universitaire Campus (building D), B-3590 Diepenbeek, Belgium, E-mail: geert.wets@luc.ac.be

University of Antwerp — UFSIA. Department of Applied Economic Sciences, Research Group "Transport & Spatial Analysis", Prinsstraat 13, B-2000 Antwerp, Belgium & Limburg University — LUC. Department of Business Economics, Universitaire Campus (building D), B-3590 Diepenbeek., Belgium, E-mail: frank.witlox@ufsia.ac.be

Katholieke Universiteit Leuven, Department of Applied Economic Sciences, Naamsestraat 69, B-3000 Leuven, Belgium, E-mail: jan.vanthienen@econ.kuleuven.ac.be

Keywords: V&V, site selection, decision tables, urban planning, KBS

Abstract

In this paper, the verification and validation of Knowledge-Based Systems (KBS) using decision tables (DTs) is one of the central issues. It is illustrated using real-market data taken from industrial site selection problems.

One of the main problems of KBS is that often there remain a lot of anomalies after the knowledge has been elicited. As a consequence, the quality of the KBS will degrade. This evaluation consists mainly of two parts: verification and validation (V&V). To make a distinction between verification and validation, the following phrase is regularly used: Verification deals with *"building the system right"*, while validation involves *"building the right system"*. In the context of DTs, it has been claimed from the early years of DT research onwards that DTs are very suited for V&V purposes. Therefore, it will be explained how V&V of the modelled knowledge can be performed. In this respect, use is made of stated response modelling designs techniques to select decision rules from a DT.

Our approach is illustrated using a case-study dealing with the locational problem of a (petro)chemical company in a port environment. The KBS developed has been named MATISSE, which is an acronym of Matching Algorithm, a Technique for Industrial Site Selection and Evaluation.

1 Introduction

One of the main problems of KBS is that there remain a lot of anomalies after the knowledge has been elicited. As a consequence, the quality of the KBS will degrade. This may result in users not wanting to use a KBS, or even worse, users who use the system, but who are not aware that the system is giving poor advice. This situation may result in financial loss or human damage. It is clear that this situation is undesirable and therefore a KBS should be evaluated. This evaluation consists mainly of two parts: verification and validation. To make a distinction between verification and validation, the following phrase is regularly used: verification is building the system right, validation is building the right system. These sentences paraphrase an adage used by Drucker (1974):

> "Efficiency is concerned with doing things right. Effectiveness is doing the right thing."

While verification is concerned with building a high-quality system, which contains no anomalies (such as incompleteness and inconsistency), the aim of validation is much more complex. In validation, we want to ascertain that the system which has been built meets the requirements of the user. This expression is rather vague, but it gives a good indication that validation is not a crisp concept. A system will never completely satisfy the user, nor it will completely dissatisfy him. Also the vagueness of the concept indicates that it will be very difficult to quantify validity. This is especially valid in the context of KBS, since they are typically useful in solving ill-structured problems. As a consequence, it is rather difficult to determine whether a system meets its requirements. Preece has tried to express those remarks by introducing the term "Pretty Good Validity" (Preece, 1995), meaning we can never be sure we have the optimal system, but we will try to make the best possible. Another conclusion we may draw is that validation subsumes verification. We cannot have a valid system if it contains anomalies.

The power of DTs to deal effectively with V&V issues has been recognized since the origin of DTs. Either this V&V can be performed immediately on the DTs because knowledge has been modelled using DTs, or the knowledge of KBS, which has been specified in another formalism, such as rules, can be transformed into a system of DTs for the purpose of V&V. Although the starting point is different, the problem which has to be resolved remains the same: "how can we V&V a system of DTs adequately?"

2 Decision tables

A DT is a tabular representation used to describe and analyze procedural decision situations, where the state of a number of conditions jointly determines the execution of a set of actions. Not just any representation, however, but one in which all distinct situations are shown as columns in a table, such that every possible case is included in one and only one column (completeness and exclusivity). The tabular representation of the decision situation is characterized by the separation between conditions and actions, on one hand, and between subjects and conditional expressions (states), on the other. Every table column (decision column) indicates which actions should (or should not) be executed for a specific combination of condition states. In this definition, the DT concept is deliberately restricted to the single-hit table, where columns are mutually exclusive. Only this type of table allows easy checking for consistency and completeness (Vanthienen and Dries, 1997).

A DT consists of four parts (Codasyl, 1982):

1. The condition subjects are the criteria that are relevant to the decision-making process. They represent the items about which information is needed to take the right decision. Condition subjects are found in the upper left part of the table.

2. The condition states are logical expressions determining the relevant sets of values for a given condition. Every condition has its set of condition states. Condition states are found at the upper right part of the table.

3. The action subjects describe the results of the decision-making process. They are found in the lower left part of the table.

4. The action values are the possible values a given action can take. They are found at the lower right part of the table.

A DT is a function from the Cartesian product of the condition states to the Cartesian product of the action values, by which every condition combination is mapped into one (completeness) and only one (exclusivity) action configuration. If each column only contains simple states (no contractions or irrelevant conditions), the table is called an expanded DT. An example is given in Figure 1.

1. Space (S)	S<20			20<=S<40			S>=40		
2. Costs (C)	C<2	2<=C<4	C>=4	C<2	2<=C<4	C>=4	C<2	2<=C<4	C>=4
1. Premium 1	-	-	x	-	x	x	-	x	x
2. Premium 2	x	x	x	x	-	x	-	-	x
	1	2	3	4	5	6	7	8	9

Figure 1: Example of an expanded DT

If necessary, columns in an expanded DT can be contracted. Contraction combines columns or groups of columns that only differ in the state value of one condition and that have equal action configurations into respectively one column. It is important to note that contraction does not change the knowledge contained in the DT. Only the format in which it is presented to the user is changed. Contraction is important in order to enhance the effectiveness of the decision-making or to provide a more compact formulation that can serve as a basis for discussion between the expert and the knowledge engineer. The contracted version of the expanded DT of Figure 1 is depicted in Figure 2. There are only five columns in the contracted DT instead of the nine columns in the expanded DT.

1. Costs (C)	C<2		2<=C<4		C>=4
2. Space (S)	S<20 or 20<=S<40	S>=40	S<20	20<=S<40 or S>=40	-
1. Premium 1	-	-	-	x	x
2. Premium 2	x	-	x	-	x
	1	2	3	4	5

Figure 2: Example of a contracted DT

3 Verification

Verification looks for potential inconsistencies in KBS. Considerable research in the V&V community has focused on determining a classification for these anomalies (e.g., Ginsberg, 1988; Nguyen, Perkins, Laffey and Pecora, 1987; Suwa, Scott and Shortliffe, 1982). A classification which is nowadays commonly used in the V&V community of KBS can be found in Preece and Shinghal (1994). It considers the following anomalies:

- Redundancy
  - Unfirable rule — Unsatisfiable condition
  - Subsumed rule — Duplicate rules
  - Unusable consequent
- Ambivalence — Contradictory rules
- Circularity
- Deficiency — Unused input

Figure 3: Preece's and Shinghal's anomaly classification

In Preece's classification, four anomaly types can be identified: redundancy, ambivalence, circularity and deficiency. These general types of anomalies are at the top level of the tree. Anomalies which occur at lower levels in the tree, are special cases of anomalies which are situated above them in the tree.

Redundancy occurs in a rule base, if there exists a rule *R* that for every possible input environment has no influence on the final result. Three types of redundancy may occur in a rule base, i.e. unfirable rules, subsumed rules and unusable consequent. A second anomaly type is ambivalence. Ambivalence occurs in a rule base if there a set of rules which infer contradictory knowledge. A third anomaly type which can occur is circularity. Circularity occurs in a KBS when it contains a set of rules, which can create a loop when the rules are fired. Finally, a KBS can be deficient. This means that for some combinations of input no conclusions can be derived.

To detect the anomalies, which we just have described, many tools have been developed. For an extensive survey see Murrell and Plant (1997). Currently, tools can adopt two strategies to analyse a knowledge base. Either they use meta-knowledge to check the system (domain dependent tools) or they transform the knowledge base in an intermediate representation, such as a table or a graph (domain independent tools). An example of a tool that uses meta-knowledge is the Expert system Validation Associate (EVA) system. This system was developed at Lockheed Corporation (Chang, Combs and Stachowitz, 1990). EVA is a set of tools built around a theorem prover and a database. These tools include anomaly checkers and validation tools. EVA works as a front end to several shells and transforms the syntax of those shells into EVA format. The meta-language of EVA allows the knowledge engineer to specify semantic constraints (e.g., impermissible sets).

A second category of tools are the domain independent tools. Early approaches in V&V made use of some form of tables. Examples of these tools include the Rule Checking Program (RCP) (Suwa, Scott and Shortliffe, 1982), PROLOGA (Vanthienen, 1986), Expert System Checker (ESC) (Cragun and Steudel, 1987), and Puuronen's approach (Puuronen, 1987). A disadvantage of these tools is that they merely check anomalies between pairs of rules, no checks over chains of rules are carried out. Second generation methods make it possible to detect anomalies across numerous rules. Examples of these tools are COVADIS (Rousset, 1988), KB-Reducer (Ginsberg, 1988), COVER (Preece and Shinghal, 1994) and PROLOGA95 (Vanthienen, Mues and Wets, 1997).

4 Validation

To decide whether a KBS is valid or not is a difficult task, mainly because a system is never completely valid or invalid. Furthermore, in most cases it is very difficult to quantify the validity of a system. In O'Keefe and O'Leary (1993) a framework has been proposed to validate KBSs.

4.1 O'Keefe's and O'Leary's validation framework

Their framework consists of four parts, i.e. criteria for validation, criterion vs. construct validity, maintaining objectivity and reliability.

4.1.1 Criteria for validation

The simplest criterion to decide whether the performance of a KBS is sufficient is to compare it with the performance of the expert. However, in most cases this criterion is not simply quantified. Furthermore, several experts (different from those who contributed to the system) have to be involved, so that the results of the tests are trustworthy. Another criterion, which might be used to validate the KB, is to specify a performance range. Hereby frequently a minimum level of competence is defined (e.g. the system should be 95% correct). A more formal way to determine a performance range is based upon the builder's user's risk technique Balci and Sargent (1981). This technique uses the in statistics well-known Type I and Type II errors. Type I errors occur in this context if a system is rejected as invalid when it is in fact valid. The probability of a type I error is called the builder's risk. On the other hand, a type II error occurs when an invalid system is accepted as valid. The probability of this event is called the user's risk.

4.1.2 Criterion vs. construct validity

The criteria to determine the validity of the KBS are variations of the so called criterion validity. This type of validity compares test scores with one or more external variables, Another type of validity, which the authors distinguish in their framework is construct validity. This kind of validity checks against the theory on which the system is based. In KBS, however, this kind of validity is not widely used, since most of the KBS are build in an empirical manner. The authors argue that in the future construct validity may play a more important role based upon 'first principles' derived from an understanding of the causality in the domain of discourse. However, currently it seems that this claim is not justified. Especially in practice, criterion validity is still the most important validation type.

4.1.3 Maintaining objectivity

An important issue in checking the validity of an expert is the objectivity of the human validator. For example, if will be clear that the developer of the system is not an ideal validator since he is not independent. In classical software engineering the last step in the development life cycle consists of an acceptance test of the system by the user. O'Keefe and O'Leary (1993) note that sometimes the user may have insufficient expertise to validate the system. Therefore, they argue that it can be interesting to contact third-party experts to validate the system.

4.1.4 Reliability

Finally, in their framework the authors mention the importance of reliability. To ensure a reliable KBS, the knowledge reported by the expert, and the actual knowledge of the expert should be the same. Subsequently, this knowledge should be translated in some computer interpretable formalism. During these subsequent steps, loss in reliability can cascade, and thus some initially small deviations may in the end lead to a major decline in the quality of the system under consideration.

4.2 Validation methods

Several methods to validate a KBS exist. Which method should be preferred is dependent on criteria such as available experts, time and money constraints, etc. An overview of methods to validate KBS is given in O'Keefe and O'Leary (1993). They distinguish three classes of methods, i.e. component validation, system validation and statistical methods.

4.2.1 Component validation

This type of validation focuses on components of the KBS individually. Three subtypes can be identified, i.e. rule validation, heuristics and meta-models.

Rule validation

A frequently used component validation method is rule validation. This should not be surprising, since the most important part of most KBS are rules. However, the direct examination of rules might be problematic in case of larger rule bases. In this case, a sample of the most important rules can be used to validate the rule base. A measure to determine the most important rules might be to select those rules which fire most frequently.

Heuristics

Since an expert system can be considered as a large heuristic, the expert system can be compared to the optimal solution in order to validate it. This method can be applied if the expert system models a problem where a mathematical optimization method can provide the optimal solution. The outcome of the expert system can then be compared to this optimal solution. Other examples of heuristic methods are scale up assumption. This method assumes that if the KB is valid for a small system, it will stay valid for larger systems. This may be the case, but is clear that this result is by no means certain. A last heuristic method we want to mention is called worst case analysis. This method tries to predict what the worst result is the system can generate.

Meta-models

Meta-models describe relationships between elements of the model. They are an higher level of the KBS. While in the early years of KB development they were used scarcely. If they were also used it was merely as a documentation technique and this only for large KBSs. Meta-models could then be used to check the validity of the KBS. Currently, meta-models are becoming increasingly popular. Their role is not limited anymore to documentation, but they are an essential building block of formal KBS development methods. The model which in this context is widespread is the KADS model (Fensel and van Harmelen, 1994).

4.2.2 System validation

System validation methods test the KBS as a whole. These methods checks how the system reacts given specific tests. Seven types of system methods will be described, i.e. test cases, Turing tests, simulation, control groups, sensitivity analysis, comparison against other methods and line of reasoning.

Test cases
The use of test cases is one of the most important methods to validate a KBS. Traditionally, the system has to solve cases, which were previously solved by an expert, and their respective results are compared. In O'Keefe and O'Leary (1993), four guidelines to select test cases are given:

- The boundaries of the inputs that the system will receive should be specified;
- A sufficient number of test cases is necessary. Especially, emphasizing the coverage of the test data and not the number of test cases that is used;
- The nature of the problems investigated should help to select the test cases. This means enough critical cases should be used;
- One should keep in mind that the expert's decision may have already had influence on the test case, and therefore the result of the system on the test case is biased by a previous decision of the expert.

Also there exist a number of approaches, which try to automatically generate test data, for example Bendou (1995). Furthermore, the use of testing techniques, which are well known in traditional software engineering, for testing a KBS is an active domain of research among them Kirani, Zualkerman and Tsai (1994) and Xantakis, Rabot and Richard (1995). Examples of such testing methods are black-box testing methods white box testing methods. Black-box testing methods do not take into account how the problem is solved, only the result counts. An example of a black-box testing method is input partition testing. In this method the input space is split into several partitions, and based on these partitions test cases are selected. White-box testing methods on the other hand, make use of the internal structure of the system to evaluate the test cases. An example of such a method is dynamic flow testing. This method generates test cases to exercise different paths of the execution of the program.

Turing tests
A well-known test in A.I. is the Turing test. In this test a person has decide, whether the output he receives is from a machine or a human. In the case of expert systems, an expert has to compare the results from the expert system with those from an expert and has to decide which result is from the expert system and which result is from the expert. To set up a Turing test, test cases should be selected, therefore, the remarks made in the prior sections, equally apply. For example, the well-known MYCIN expert system was validated using a Turing test. MYCIN (Buchanan and Shortliffe, 1984) originated of the Stanford Heuristic Project, and it is generally considered as the first expert system. MYCIN is an expert in diagnosing bacterial infections and describing treatment for them.

Simulation
Instead of generating test cases, it might be interesting to compare the outcome of the expert system to that of the simulation model. Each run of the simulation model can then be regarded as a test case. This type of validation, however, has to be handled with great care, since the simulation uses also a model. Clearly, we have to be sure that the simulation model itself is valid.

Control groups
Another method to validate KBS are control groups. This method can be interesting because in most KBS a lot of interaction between the user and the system is involved to solve problems. In the control groups method, problems are presented to two groups, one without the system, and one with the system. The performance of the two groups to solve the problems are then compared. The control group method is well-known in the field of

medicine. In this field, usually two groups of patients are formed. One group gets treatment with a new medicine, the other gets not. By comparing the results of the two groups the efficiency of the new treatment can be evaluated.

To be valuable, the experiment has to be setup so that the performance of the two groups without the system should be estimated equal. Furthermore, another problem with this method in the field of KBS is, that there may exist a learning curve to use the system. This should be kept in mind when evaluating the system.

Sensitivity analysis
A major problem to validate the expert system arises when there are only few test cases available. One could suggest to use credibility as a measure to validate the system. The developers could ask the question is the system credible to the expert, the users, etc. ? Another method, which might proof to be valuable in this context, is sensitivity analysis. This method starts from a single case where the results are satisfiable according to an expert. Subsequently, some inputs of the system are altered so, that the output should not change. Depending whether the output of the system changes or not, in this particular case, the quality of the expert system is assessed.

Given the fact, that there are only few real cases involved, this method may not cover sufficient parts of the input domain. Therefore, this method can in practice be combined with methods, which generate synthetic test cases to overcome this problem.

Comparison against other methods
To validate the system, it might be useful to compare the system with other models, which are developed to solve the problem under consideration. A typical example of this method occurs when the expert system will be used as an heuristic for a complex mathematical problem. In most cases, optimal solutions for such problems exist, but they are unusable in practice, because of all sorts of constraints (e.g. time constraints). Comparison against this optimal model will then give insight in the performance of the system.

Line of reasoning
To validate a KBS, it is important to see that the line of reasoning is correct. Because, typically an expert will only believe the system when it can explain, how it has reached a conclusion. A lot of commercial expert system shells have built in explanation facilities to enable this kind of validation.

4.2.3 Statistical methods

In many cases, besides using qualitative validation techniques some quantitative techniques using statistical models are useful. In O'Keefe, Balci and Smith (1987), the validation of a KBS is seen as the following hypothesis test:

H_O: the expert system valid for the acceptable performance ranges under the prescribed input domain;
H_A: the system is invalid

Many statistical techniques are available to the developer of the system to validate it. For example, techniques to compare statistically the conclusions of an expert with the outcome of the system. However, it is out of the scope of this paper to give an overview of these techniques.

5 The MATISSE KBS for industrial site selection

In order to be able to propose a new validation technique for KBS, the expert system has to be built first. An example is taken from urban land use planning. In particular, the problem of selecting a suitable location site for manufacturing (i.e. the chemical and petrochemical industry) in a dockland environment (i.e. the port of Antwerp) is analysed.

There are several reasons why the chemical and petrochemical industry can be considered an appropriate case study to illustrate the principal issues advanced in this study. First of all, the industry has, by comparison with others, received only scant attention in the economic and regional location theory, despite its importance to the national and regional economy. Second, the capital-intensive nature of the industry makes it a sector that is very locational sensitive and conscious. Consequently, the location decision of a (petro)chemical complex is taken with great care, which gives an indication of the importance the sector attaches to the selection of a satisfactory, suitable site. Moreover, once a commitment for a certain location site has been made, it is generally considered irreversible, and therefore every factor influencing the profitability of the plant should be evaluated carefully. Third and finally, the petrochemical industry, like the chemical industry in general, is essentially a supplier of intermediate products to other industries. This specific role has enabled it to become a vital element of economic growth, and also emphasizes the importance of functional linkages. The grounds to select the port of Antwerp as potential place of business are twofold. First, the present structure and development of the port is completely attributed to the gradual location of the (petro)chemical industry. Consequently, today, Antwerp is considered as one of the largest chemical and petrochemical complexes in the world. Furthermore, in Western Europe, it is one of the oldest (Molle and Wever, 1984, p. 128). Obviously, this fact highlights Antwerp's apparent magnetism to attract chemical and petrochemical companies to its port. Second, the current, phased development of the Left Bank makes the port of Antwerp unique in its ability to still offer to potential interested companies large areas of industrial sites. As a result, several new (petro)chemical companies have recently decided in favour of Antwerp for establishing their production operations.

The tool to be developed has been named MATISSE, which is an acronym of Matching Algorithm, a Technique for Industrial Site Selection and Evaluation (Witlox, 1998; Witlox and Timmermans, 1998; Witlox *et al.*, 1998). In order to develop the MATISSE-model, a series of in-depth interviews were conducted. At the time of the start of collecting the data (i.e. October 1996), the total number of chemical and petrochemical manufacturing companies located in the port of Antwerp was equal to 26 (Havenbedrijf Antwerpen, 1996). Although, this number may seem small, the industry's combined economic impact on the port of Antwerp is astonishing. To illustrate, in December 1996, these twenty-six chemical and petrochemical companies employed 12,371 people (representing almost half of the total industrial harbour employment) and occupied 1,162.7 ha on the Right Bank and 808.7 ha on the Left Bank (representing about 66% of the approximate 2,975 ha of allocated industrial sites). Moreover, accumulated over the years, the chemical and petrochemical industry has invested about BEF 325,120 million in their Antwerp port production installations, which represents almost 64% of all accrued industrial investments. The advantage of working with a rather small, but yet very important target population, is that no prior selection of a subgroup is necessary. However, it also implies that, in order to be representative, a high response rate is essential. In October 1996, a personalized letter was mailed to each of the 26 enterprises in the population with the request to grant an interview with the company's highest management executive on the subject of plant location decisions. In response to this letter, 14 companies answered spontaneously, while, as a result of a short (i.e. three week period) telephonic follow-up, an additional 9 companies could be persuaded to grant a short interview. This makes the total response rate equal to 23 companies (or 88%). Given the relatively small population size (26 firms), the resultant response rate is an exceptionally good result. However, viewed in terms of sampling, it is a well-known fact that the smaller the size of the population, the higher the imposed demands on the resulting response rate.

5.1 The in-depth interview technique

All the carried-out interviews proceeded in much the same way. First, the basic objective of the research was stated to the respondent, whereby the emphasis was placed on explaining the concept of a decision table (DT) as a formalism to represent data. The respondent was also shown an example of a DT, and asked (if possible) to think and express his or her information in terms of "if...then" decision rule structures. As pointed out by Vanthienen (1986), Merlevede and Vanthienen (1991), Santos-Gomez and Darnell (1992), Tanaka *et al.* (1993), and Arentze *et al.* (1995, 1996), this approach of data collection or elicitation, in which, from the start the respondent is confronted with the notion of DTs, is able to offer some significant model advantages, especially in the case when the decision information is complex. At least three reasons can be advanced in support of such an approach. First of all, the communicative properties offered by the DT technique makes it an ideal formalism for representing complex information and sets of decision rules in such a way that is intelligible and clear to lay people. Note that, in this respect, one of the main advantages of using a DT structure is its capacity of data representation. Second, DTs offer respondents the possibility to specify and verify the correctness of their supplied information represented in the rules of the DT. Consequently, the model's accuracy can be tested in a systematic way by checking each conditional statement (i.e. decision rule) of the DT separately. Third and finally, given that DTs allow for the use of subDTs, they support a hierarchical structuring of information that does not only provide a form of modularity, but also helps to keep an overview of the decision problem under investigation. As such, a so-called "top-down" decision-making approach is followed in which one starts with an abstract (head) DT that is then further worked out in a series of more concrete subDTs. This approach also seems to correspond with the way people tend to transfer their information to the interviewer (Arentze *et al.*, 1995, p. 240), and also concurs with the way in which a location decision is usually made.

Second, the actual interview technique, which was used to elicit the decision information from the respondents, can be termed a combination of an unstructured (informal) and a structured (guided) interview approach (Turban, 1995). In the unstructured, first part of the interview, the interviewee was asked to freely "think-aloud" on the subject of site selection (e.g. Which factors play a role in site selection?; Given a certain location factor, how is this factor assessed and what is its influence on locational decision-making?). The aim of this approach was to identify these location factors (i.e. conditions) that first came into the respondent's mind, and more important, to know how these factors should be interpreted and which evaluation criteria are used. As such, if the interviewee stated that a location factor like "site accessibility" is important in selecting a suitable site, it is essential to know what is meant by that particular condition. For example, it may refer to evaluating the site's accessibility in general, or as was often the case, refer to the site's accessibility in respect to the supply of raw materials, the transportation of finished goods, or the transfer of the work force to and from the industrial site? Obviously, interpretations and relative importance of factors differed across the sample. However, usually a close link with the organizational and production-related aspects of the economic activity could be observed. This fact again emphasizes the importance to take account of the so-called "context-dependent" nature of location factors. To illustrate, for an air-splitting company, using air (i.e. a ubiquity) as principal feedstock, other (if any) site accessibility requirements with respect to feedstock supply will be put forward than for an oil refinery which relies heavily on the overseas transport of crude oil and pipeline connections for the supply of its feedstock.

Another interesting point that could be deduced from the unstructured part of the interview, is that abstract location factors are usually not evaluated as such, but are first more concretely defined and then assessed. As a result, generic location factors such as "transport", "labour", "utilities" etc. are mentioned more than once in the decision-making process, but at different levels of importance in the site selection process and with varying interpretations. For instance, the location factor "transport" is first (on the highest level) interpreted and evaluated in terms of the availability of on-site "transport infrastructure". Conditional upon this evaluation, potential location sites may either be rejected simply because their existing transport infrastructure is totally inadequate, or further assessed in terms of, e.g., the level of additional "transport investments" needed. Again dependent upon this second evaluation, a further assessment may be required of the site's "transport costs" and "accessibility". The basic idea behind this thought process is that site selection makers are not interested in how good or bad a potential location site scores on, say, accessibility for personnel, if certain higher priority requirements concerning the general transport infrastructure are not met by the location site. As such, a hierarchical decision-making process can be distinguished in which: first, on the highest level, a number of elementary site conditions are being evaluated, then a number of investment considerations, and finally, a number of operating considerations.

Apart from stressing the activity-specific nature of location factors, the individual influences of location factors on the site selection problem (e.g. veto-dimension, trade-off dimension) and the existence of internal dependencies between factors (i.e. conditional relevance, conceptual interaction) should also be examined. In this respect, if a certain location factor is evaluated as (un)satisfactory, then how does this evaluation effect the overall site selection process, and also the consecutive evaluation of other location factors? In addition, it is also important to know which evaluation criteria the respondent uses to assess different location factors.

In the subsequent, structured part of the interview, the respondent was asked to react to a check list of different location factors, and was also more closely guided through a series of particular questions on the subject of site selection. The check list used was compiled through a review of the relevant existing literature on the subject of chemical and petrochemical plantsite selection (e.g. Winkelmans, 1973; Chapman, 1991; Gemeentelijk Havenbedrijf Rotterdam, 1993; Leuris, 1996). Among the additional questions posed, a number of so-called control questions were asked to eliminate certain basic inconsistencies. For instance, if the respondent stated that the general labour market conditions are evaluated as very good, but later in the interview, it is found that the recruitment of suitable workforce is very difficult, this contradiction will have to be rectified. Having identified the principal factors (and their associated evaluation states) that are important in industrial site selection — i.e. the so-called domain layer of model development (Arentze *et al.*, 1995) — the respondents were urged (there where possible) to make "if...then" like statements or decision rules (i.e. the inferential layer) expressing their decision-making process.

Given that the process of decision rule deduction and specification is an essential step in the construction of a DT, it worthwhile to discuss this specific aspect in somewhat more detail, and also point to some specific problems encountered. Although, nearly all the respondents showed a great affinity with the problem of site selection, some of them found it very difficult to express their decision-making process chiefly by means of a number of logical "if...then" rules. In particular, expressing compensatory ruling proved not to be an easy task, at least in comparison with making explicit certain non-compensatory rules. In respect to the latter, most respondents had little difficulty in using "if...then" statements to express non-compensatory relations. Frequently, respondents stated that *if* factor X is not satisfied or present, *then* under no circumstances a positive evaluation result (e.g. site suitability, sufficient labour market conditions, etc.) could be the outcome. Usually, these typical, non-compensatory factors related to primary site conditional aspects (e.g. geographical location, availability of basic utilities, etc.), but other non-compensatory statements were also used in the lower level decision-making process. Note that non-compensatory factors have a strong decision-making discriminating power — they are often also assessed on a strictly crisp basis (e.g. yes or no, present or absent, adequate or lacking) — and this is why they are (automatically) placed at the top of the DT. It also implies that in the process of decision support they are evaluated first by the respondent. In respect to handling compensatory rules by means of "if...then" constructions, it can be noted that in most cases respondents were able to indicate the major relations, but found it somehow more difficult to express complicated, more fine-drawn trade-off relations. Generally, compensatory decision rules of the following type were used: *if* factor X is evaluated as unsatisfactory but (i.e. *and*) factor Y (*and* or *or* factor Z, ...) is (are) being assessed as satisfactory, *then* a certain (evaluation) result X is obtained. Note that, in the present context, the interpretation of "factor" may relate to both locational factors as well as organizational aspects.

Translating the expressed non-compensatory and compensatory rules into a DT structure revealed a number of interesting points. First, a number of inconsistencies in the decision-making process could be observed. Some of these inconsistencies related to contradictions in certain decision rules, while others concerned the violation of the exclusiveness and completeness (i.e. domain coverage) properties. In a number of cases, the corrections to be implemented proved self-evident; in others, the respondents were either asked for the correct interpretation or admissible solutions were proposed to them for further consideration. The optimization of the DT in terms of minimum decision rules also revealed that redundant information in the decision-making process could be discarded. Second, as a result of the DT's capacity of generating all possible decision rules, a number of so-called "empty columns" (i.e. decision rules with undefined action states) were produced which needed to be completed. In some cases, this task was not considered to be an obviousness because certain (alternative) decision evaluations had to be made by the respondents which they were previously unaware of. Third, the use of subDTs offered an interesting means to structure the complex decision problem, and facilitates a "top-down" site selection process. Apart from the fact that abstract terms can be defined in more concrete and distinct factors, subDTs have the capacity to break-down a problem into a series of (less complicated) subproblems. The use of subDTs also avoids the common problem of a combinatorial explosion because each decision (sub)table only deals with a limited number of related conditions and actions, and DTs are being contracted as far as possible such that a minimum number of columns or decision rules is automatically adopted. Note further that, in view of a relational matching approach, the end tables of the nested, hierarchical DT structure should be formulated in such terms that they allow a direct matching with the observable object profiles (Arentze *et al.*, 1995, p. 239).

5.2 Some results

Following the results of our conducted in-depth interviews, (almost) an infinite number of location factors or attributes may potentially be relevant for defining a concept like "suitable location site". Combining these attributes, numerous definitions of site suitability are possible. Given that our focus is on the development and testing of a model to predict locational choice-making for a chemical or petrochemical industry, we concern ourselves with the modelling of the object-type *Site suitability of a harbour location for a (petro)chemical industry.* This object-type is depicted in Table 1.

On the basis of the conducted interviews, three abstract so-called INUS-conditions (i.e. a condition which is on its own insufficient, but within a conjunction indispensable) have been distinguished which play a crucial role (on the highest decision level) in determining the degree of locational site suitability: C_1: *^Site conditions*, C_2: *^Investment considerations* and, C_3: *^Operating considerations*. Each of these abstract conditions is further specified through a system of subDTs (denoted by "^"). Note also that each condition is defined using three condition states, which combined, results in 27 (3^3) different decision rules (i.e. an expanded table). However, as a result of the non-compensatory character of the third condition state of C_1 and the apparent conditional relevance in the condition states of C_3, the total number of decision rules in the contracted table is equal to 15. Of these 15 rules, several functionally equivalent rules (i.e. rules leading to identical action states, although having different condition state configurations) can be noted. Note further that Table 1 has only one action with five different action states. The action states express different degrees of site suitability, ranging from "excellent" (rule R_1) to "bad" (rules R_{14} and R_{15}), dependent on the outcome of the condition set.

Table 1: The object-type "Site suitability of a harbour location for a (petro)chemical industry" (contracted head table)

1. C1 ^Site conditions	superior						
2. C2 ^Investment considerations	good			about average		bad	
3. C3 ^Operating considerations	good	medium	bad	good or medium	bad	good	medium or bad
1. A1 Excellent	x	.	.	.	.	.	.
2. A2 Above average	.	x	.	x	.	.	.
3. A3 Average	.	.	x	.	x	x	.
4. A4 Below average	.	.	.	.	.	.	x
5. A5 Bad	.	.	.	.	.	.	.
	1	2	3	4	5	6	7

1. C1 ^Site conditions	moderate							inferior
2. C2 ^Investment considerations	good		about average		bad			-
3. C3 ^Operating considerations	good or medium	bad	good	medium or bad	good	medium	bad	-
1. A1 Excellent	.	.	.	.	.	.	.	.
2. A2 Above average	x	.	.	.	.	.	.	.
3. A3 Average	.	x	x	.	x	.	.	.
4. A4 Below average	.	.	.	x	.	x	.	.
5. A5 Bad	.	.	.	.	.	.	x	x
	8	9	10	11	12	13	14	15

Apparently, a so-called "top-down" approach is followed in which the respondents first evaluate a number of basic site conditions. These factors relate to the geographical location, the acquisition conditions, the available on-site transport infrastructure, and the available on-site utilities. In second place, the investment considerations are assessed. The idea is that if the site conditions are not fully satisfactory, perhaps this inadequacy could be compensated by making some additional investments. However, completely inferior evaluated site conditions cannot be compensated. In that case, the location site is rejected. The investment considerations refer to real estate considerations, the level of government intervention, transport investments and utility investments. If the site conditions and the investment considerations have both been evaluated, the costs of operating the site will also have to be assessed. It involves evaluating the site accessibility, agglomeration economies, labour market, and utility costs.

In total, the MATISSE-model consists of one head decision table and, linked to it, a hierarchy of 90 sub(sub...)tables. Obviously, not all these DTs will be mentioned here. The MATISSE-model was developed using the system shell PROLOGA, initially created in its crisp form by Vanthienen (1986) at the Catholic University of Louvain. The PROLOGA95 (PROcedural LOgic Analyzer) system, which runs under Windows95, is a PC-based interactive rule-based design tool for computer-supported construction and manipulation of DTs. This DT engineering workbench facilitates data acquisition and representation, offers adequate validation and verification support, and has a user friendly interface for consulting purposes (Vanthienen, 1991; Vanthienen and Dries, 1994).

6 The use of stated response designs as an alternative approach to validating decision rules

In contrast to the verification issue — where DTs have proven to be a very strong formalism — the validation aspect is less a decision table-related issue. Stated differently, given that the check for completeness and non-contradiction of information is automatically and systematically accomplished by the DT-workbench PROLOGA95, the requirement of correctness of information has to be validated explicitly. Usually, validation does not receive that much explicit attention. Decision rules tend to be generated by knowledge elicitation which typically involves experts who have to explicate their decisions; their knowledge is then represented by the DT formalism. As such, researchers seem to believe that this process of knowledge acquisition constitutes the actual validation. Hence, no attempt is made to examine whether this approach is indeed correct.

In order to validate the main tabular structures proposed in MATISSE, a number of experiments have to be conducted in which the respondents are being confronted with their given information to check whether the decision rules represented in the tables reflect a "correct" decision-making process. Moreover, it is also checked whether, based on these rules, new decision situations can be correctly predicted.

In general, a distinction can be made between checking the (i) *intra*-tabular and (ii) *inter*-tabular correctness of the decision rules. The intra-tabular check for correctness implies validating the correctness of a single table, while the inter-tabular check deals with the issue of the correctness of information in the interaction between (components of) different (sub)tables. In the present context, our validation check is limited to single tables. The process of checking the correctness of the decision rules represented in a single (expanded) DT may be developed and interpreted along two different lines.

In the first, more general approach, intra-tabular validation would signify that the correctness of the decision rules of the table has to be explicitly checked with the decision-maker(s). In principle, this approach implies validating each individual decision rule separately. In other words, a DT consisting of six conditions, each having three condition states, would result in a correctness check of $3^6 = 729$ individual decision rules. Given that this task goes well beyond even the most diligent respondent, this so-called *full factorial* approach, in which all rules are checked, can only be used when the DT consists of a relatively small number of conditions and condition states. Due to the fact that the respondents are often unable to evaluate more than a fairly small number of decision rules at a time, a selection in these rules can be made. Hence, when a full factorial design yields too many profiles, the number can be reduced by adopting a *fractional factorial* design. In that case, only a selection or fraction of all possible combinations of condition states is presented to the experts. The simplest fractional factorial design combines all main condition states without correlation. In the stated response modelling literature (e.g. Addelman, 1962; Steenkamp, 1985; Timmermans, 1986), several so-called "basic plans" for the construction of fractional factorial designs have been developed. These basic plans indicate the minimum number of trials needed to construct a valid design and also show how to combine different condition states in different profiles. The choice of a basic plan depends on the total number of conditions and total number of associated condition states in the DT. Addelman's basic plans are shown in the Appendix.

A second, though less commonly applied approach would be to use the expert's explicated decision rules to predict new decision situations. As such, the correctness of the decision rules is not validated by confronting the respondents *ex post* with their given answers, but by examining the capability of these given rules to predict the decision outcome of new situations. In this respect, the emphasis is on testing the external validity of the DT-model. The problem of which and how many decision rules should be selected for this external validation purpose is identical to the first approach. In other words, when the number of rules in the table is fairly large, a selected set of decision rules can be used. Therefore, instead of using all possible decision rules to predict future site selection behaviour, a fractional factorial design is applied to identify a reduced set of rules. In the present context, the second approach will be followed. Moreover, the intra-tabular correctness check of MATISSE's DT-structure will be limited to its head table.

7 KBS Validation: an example taken from MATISSE

Table 1 depicted MATISSE's head DT. In this head DT, three conditions, each having three condition states, were combined. The result was 27 (= 3^3) decision rules. It follows that a full factorial check would imply that the respondents have to evaluate all 27 rules. By contrast, an orthogonal fractional factorial design would involve evaluating only 9 decision rules. In this case, Addelman's (1962, p. 36) basic plan n° 2 can be used to construct these nine profiles (see Appendix). The process of encoding the condition states of the head DT is shown in Table 2.

Table 2: Encoding the condition states of the head table

Conditions	Condition states
C_1: Site conditions	0="superior" 1="moderate" 2="inferior"
C_2: Investment considerations	0="good" 1="about average" 2="bad"
C_3: Operating considerations	0="good" 1="medium" 2="bad"

In Table 2, it can be noted that a code (ranging from zero to two) is allotted to the various condition states of the three conditions. Using this code, different profiles of combined condition states can be constructed. This process of constructing a fractional factorial design is visualized in Table 3. By translating these encoded profiles in concrete combinations of condition states, nine different single decision rules are selected as a result of implementing Addelman's basic plan n° 2. Processing these nine profiles (i.e. combination of condition states) through MATISSE's head DT produces nine different decision rules (each leading to one of possibly five action states of A_1). By comparing the decision rules produced by the DT with the (ex post) answers given by the participants in the sample, a percentage of correctly predicted answers can be calculated. Depending upon the outcome of this prediction factor, the intra-tabular structure can be confirmed or suggestions made for alterations.

Table 3: Fractional factorial design for the head table

Profile N°	Condition and condition states		
	C_1	C_2	C_3
1	0	0	0
2	0	1	1
3	0	2	2
4	1	0	1
5	1	1	2
6	1	2	0
7	2	0	2
8	2	1	0
9	2	2	1

A point that may need some additional explanation is why the method of fractional factorial design used in stated response modelling is applied to make a selection of decision rules to be used to test the external validity of the model. Initially, experimental fractional factorial designs have been advocated in stated response modelling because they permit unbiased parameter estimations of all main effects of a factorial arrangement of attribute levels without correlation. In the present context of model validation, however, this attractive estimation property is not really relevant. In other words, given the fact that no parameters have to be estimated, the selection of decision rules could just as easily be made at random. The advantage of using experimental plans, however, is that the construction of these plans is based upon the principle of proportional frequencies of the factor levels. In other words, a correctly established and applied basic plan guarantees that the levels of one factor occur with each of the levels of the other factor with proportional frequencies. As such, a balanced or symmetrical selection of profiles of combined condition states is obtained.

In order to illustrate that in Table 3 the proportional frequency condition is satisfied for the proposed fractional factorial design, consider, for example, the first (C_1) and second condition (C_2). For these two conditions (as for any other pair of conditions), the following requirement should be fulfilled (Addelman, 1962, p. 23):

$$n_{ij} = n_{i\bullet} \times n_{\bullet j} / N \qquad [1]$$

where N denotes the number of profiles in the plan, $n_{i\bullet}$ the number of times the i condition state of C_1 occurs in the plan, $n_{\bullet j}$ the number of times the j condition state of C_2 occurs in the plan, and n_{ij} the number of times the i condition state of C_1 occurs with the j condition state of C_2. In the present example, $N = 9$, $n_{0\bullet} = n_{1\bullet} = n_{2\bullet} = 3$ and $n_{\bullet 0} = n_{\bullet 1} = n_{\bullet 2} = 3$. To demonstrate that $n_{ij} = n_{i\bullet} \times n_{\bullet j}/N$ is satisfied, take for example n_{11} According to Table 3, n_{11} occurs once (i.e. profile 5); thus $n_{1\bullet}.n_{\bullet 1}/N$ should also be equal to 1, which is indeed the case: $3 \times 3/9 = 1$.

During the months of March and April of 1998, all 23 participating respondents, which were initially interviewed about a year and a half ago in order to be able to construct the MATISSE-model, were contacted again. Due to the fact that one (German) respondent had gone into retirement, and two others were no longer employed within the same company but took up a position abroad, the group of initial experts now equalled 20. In total, 19 respondents agreed to cooperate for a second time and were willing to grant a second interview. Only one respondent could not be persuaded to further cooperate.

In all cases, the interview proceeded much along the same way and lasted on average about forty minutes. First, the respondents were briefly given some feedback on the resulting MATISSE-model so that they had an idea of how their provided information and expert knowledge had been transformed in a tabular decision-making structure. Next, the two tasks (consider them two separate assignments), which the respondents were asked to perform, were explained and illustrated. The first task related to the issue of validation and will be discussed here; the second involved the issue of model fuzzification which will be dealt with in the next section. Given the difficulty (and also time-consuming nature) of both tasks, the respondents were not urged to answer immediately but were given sufficient time to complete the questionnaire outside the office hours. The respondents were nevertheless asked to write down their answers on pre-printed forms and to return them by mail as soon as possible. Although some respondents had to be reminded (several times) to return their questionnaire, they nearly all complied with our request. In total, 17 useful answers (74 %) were obtained for the purpose of validation.

Validation results

The head table of the model was validated using the nine "if...then" decision rules specified in Table 1. These decision rules were written on separate index cards and presented to the respondents ($N = 17$) in a random order. In each case, the respondent was asked to evaluate the hypothetical choice situation described on the index card in terms of the allowed action states used in the DT. These evaluations were then compared with what the DT claims to be the correct answer. As such, a percentage was computed of the number of correctly predicted decision rules. The main results for the head table are shown in Table 4.

Table 4: Validation results for the head table

Profile N°	Specified profile (and associated decision rule)	Expected AS according to DT	% and absolute distribution of given answers over the AS of the DT[(*)] ($N = 17$)									
			A_1 (excellent)		A_2 (above average)		A_3 (average)		A_4 (below average)		A_5 (bad)	
1	"000" (R_1)	A_1	**76.5**	13	23.5	4	—	—	—	—	—	—
2	"011" (R_4)	A_2	5.9	1	**58.8**	10	35.3	6	—	—	—	—
3	"022" (R_7)	A_4	—	—	—	—	6.2	1	**31.3**	5	62.5	10
4	"101" (R_8)	A_2	—	—	**17.6**	3	76.5	13	5.9	1	—	—
5	"112" (R_{11})	A_4	—	—	—	—	12.5	2	**62.5**	10	25.0	4
6	"120" (R_{12})	A_3	—	—	11.8	2	**23.5**	4	41.2	7	23.5	4
7	"202" (R_{15})	A_5	—	—	—	—	—	—	35.3	6	**64.7**	11
8	"210" (R_{15})	A_5	—	—	6.2	1	25.0	4	31.3	5	**37.5**	6
9	"221" (R_{15})	A_5	—	—	—	—	5.9	1	11.8	2	**82.3**	14

(*) % of correctly predicted AS are given in bold, highest % is underlined

Table 4 reflects in terms of percentage and in absolute terms the distribution of the given answers by the respondents for each specified profile (decision rule). To give some further interpretation to the percentages mentioned in Table 4, the figures in bold represent the percentages of what should be interpreted as "correctly predicted action states (AS)". This means that, in these cases, the answers of the respondents concurred with the answers produced by the DT. The other percentages (those not in bold), if mentioned, give an indication of the spread of all deviating answers. Finally, the figures which are underlined represent the highest percentage. Ideally, the percentages in bold should also be underlined because this means that a preponderance of respondents answered what the DT would also conclude to be the corresponding answer. Important to note is that, in the present context, it is difficult to reason in terms of the "correctness" of the answers. It is more an issue of what a majority of the respondents (experts) claim to be what they think is the most suitable answer, given a combination of location factor evaluations. If a strong discrepancy is noticed between the answers given by the respondents and the action states produced by the DT, then that particular rule in the DT cannot be validated and maybe the associated action state should be changed. In what follows, the validation results are first interpreted for the DT as a whole, and subsequently analyzed in more detail for each individual profile.

Analyzed over all cases (150 in total, being 17 experts evaluating 9 rules with 3 incomplete answers), the head DT is (only) able to predict the outcome of 51 % of the decision rules. Compared with the results of the other three subDTs which have also been put to the test (not included in this paper), this is the least satisfactory result. Two facts may be mentioned that could explain the rather "poor" result. First, the head DT, being the DT at the highest level, contains the most abstract location factors that needed to be evaluated. Hence, this fact may have confused the respondent in assessing the decision rules. Second, the head DT contains five possible action states which the respondents were allowed to use in evaluating the selected decision rules, while the other three subDTs all have only three possible action states. Clearly, the more action states, the more answer possibilities, the larger the probability of acquiring deviating answers. Nevertheless, in 6 out of the 9 rules the highest percentage corresponded with what was also assumed to be the expected action state (i.e. figures in bold and underlined).

If a closer look is taken at the results at individual decision rule level (i.e. per profile), a number of interesting elements can be distinguished and a more variegated interpretation can be given to the obtained validation result. The three best predictable profiles (rules) are 9 (R_{15}), 1 (R_1), and 5 (R_{11}). It is no coincidence that particularly the first two of these profiles are correctly assessed, given that they represent two extreme decision situations. However, this fact is not as self-evident as may seem. Almost equally satisfactory validation results have been obtained in the case when less extreme decision situations were to be evaluated (e.g. profiles 2 and 5). Also, the predictability of non-compensatory decision-making (i.e. profiles 7, 8 and 9 all referring to R_{15}) is very satisfactory. The two worst results were found for profiles 4 (R_8) and 6 (R_{12}). In respect to profile 4, only 17.6 % of the answers of respondents concurred with the assumed action state of the DT. Instead of evaluating this particular decision rule as "above average" (A_2), more than three-quarters of the respondents evaluated it as "average" (A_3). Given this unsatisfactory validation result, the action state of the particular rule in the head DT should best be changed. It would also increase the overall predictability of the head DT to 59 %. The problem with profile 6 is not as much the predictability of the rule but rather the spread in the given answers around the expected action state. Finally, viewed in terms of best and worst concordance by respondent, it can be noted three respondents scored a 7/9, while two others obtained only a 2/9 score, implying that for those two specific respondents the DT was only able to predict two out of the nine action states correctly.

8 Conclusions

In this paper, our attention focused on the verification and validation check of KBS based on DTs. Verification refers to whether the information depicted in the DTs is logically consistent and complete. By contrast, the process of validation relates to checking whether the information represented in the model is correct. While, as a formalism, DTs have very strong verification supporting properties, their potentiality with respect to validation issues is less straightforward (let alone existent). In respect to intra-tabular (within a single DT) validation, some notions of stated response design constructions have been advocated. By means of a series of fractional factorial experiments, the respondents were asked to evaluate *ex post* their given answers and decision heuristics represented in the tabular model structure. The model was tested used MATISSE's head table. For the purpose of validation, all participating respondents were contacted for a second time. In total, 19 respondents agreed to cooperate once more; 17 useful answers were obtained.

In 77 cases, the DT was able to predict the correct action state (51%) which may be considered a satisfactory result realizing that there will be heterogeneity in the expert opinions in the first place. Obviously, differences were found among the different decision rules tested. In those cases, where the respondents almost systematically gave another answer as the DT would, suggestions were made for action state alterations.

9 References

ADDELMAN, S. (1962) "Orthogonal main-effect plans for asymmetrical factorial experiments". *Technometrics*. Vol. **4** (1), pp. 21-46.

ARENTZE, T.A., A.W.J. BORGERS and H.J.P. TIMMERMANS (1995) "The integration of expert knowledge in decision support systems for facility location planning". *Computers, Environment and Urban Systems*. Vol. **19** (4), pp. 227-247.

ARENTZE, T.A., A.W.J. BORGERS and H.J.P. TIMMERMANS (1996a) "An efficient search strategy for site-selection decisions in an expert system". *Geographical Analysis*. Vol. **28** (2), pp. 126-146.

ARENTZE, T.A., A.W.J. BORGERS and H.J.P. TIMMERMANS (1996b) "Design of a view-based DSS for location planning". *International Journal of Geographical Information Systems*. Vol. **10** (2), pp. 219-236.

BALCI O. and SARGENT R. (1981), A methodology for cost risk analysis in the statistical validation of simulation models, *Comm. of the ACM 24*, pp. 190-197.

BENDOU, A. (1995), CT-DATAGEN, a constraint-based test data generator, *Proc. of the European Symposium on the Validation and Verification of Knowledge Based Systems*, pp. 19-29.

BUCHANAN, B. G. and SHORTLIFFE, E. H. (1984), *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Project*, Addison-Wesley, Reading (MA).

CHANG, C. L., COMBS, J. B. and STACHOWITZ, R. A. (1990), A report on the expert systems validation associate (EVA), *Expert Systems with Applications 1*, pp. 217-230.

CHAPMAN, K. (1991) *The International Petrochemical Industry: Evolution and Location*. Oxford and Cambridge (Mass.), Basil Blackwell.

CODASYL (1982), *A Modern Appraisal of Decision Tables*, Report of the decision table task group, ACM, New York (N. Y.).

CRAGUN, B. J. and STEUDEL, H. J. (1987), A decision-table-based processor for checking completeness and consistency in rule-based expert systems, *Man-Machine Studies 26*, pp. 633-648.

DRUCKER, P. F. (1974), *Management: Tasks, Responsibilities, Practices*, Heinemann, London.

FENSEL, D. and VAN HARMELEN, F. (1994), A comparison of languages which operationalize and formalize KADS models of expertise, *Knowledge Engineering Review 9,* pp. 105-146.

GEMEENTELIJK HAVENBEDRIJF ROTTERDAM (1993) *Bevindingen Onderzoek Rotterdam Chemiehaven*. Rotterdam, Gemeentelijk Havenbedrijf Rotterdam, Werkgroep Industriehaven-Chemie.

GINSBERG, A. (1988), Knowledge-base reduction: A new approach to checking knowledge bases for inconsistency and redundancy, *Proc. of the 7th National Conf. on Artificial Intelligence*, pp. 585-589.

HAVENBEDRIJF ANTWERPEN (1996) *The Establishment of Industries in the Antwerp Port Zone*. Antwerpen, Havenbedrijf Antwerpen.

KIRANI, S., ZUALKERMAN I. and TSAI W. (1994), Evaluation of expert system testing methods, *Comm. of the ACM 37*, pp. 71-81.

LEURIS, F. (1996) *Lokalisatieproblematiek in en rondom de Antwerpse Havenregio*. Antwerpen, UFSIA (unpublished).

MERLEVEDE, P. and J. VANTHIENEN (1991) "A structured approach to formalization and validation of knowledge". In: J. FEINSTEIN, A. AWAD, L. MEDSKER and E. TURBAN Eds. (1991) *Proceedings of the IEEE/ACM International Conference on Developing and Managing Expert Systems Programs*. Los Alamitos (CA), IEEE Computer Society Press, pp. 149-158.

MOLLE, W. and E. WEVER (1984) *Oil Refineries and Petrochemical Industries in Western Europe*. Aldershot, Gower Publishing Company.

MURRELL, S. and PLANT, R. (1997), A survey of tools for validation and verification, *Decision Support Systems Journal*, to appear.

NGUYEN, T. A., PERKINS, W. A., LAFFEY, T. J. and PECORA, D. (1987), Knowledge base verification, *AI Magazine 8*, pp. 69-75.

O'KEEFE, R. M., BALCI O. and SMITH E. (1987), Validation expert system performance, *IEEE Expert 2*, pp. 81-89.

O'KEEFE, R. M. and O'LEARY, D. E. (1993), Expert system verification and validation: a survey and tutorial, *AI Review 7*, pp. 3-42.

PUURONEN, S. (1987), A tabular rule-checking method, *Proc. 7th Intl. Workshop on Expert Systems and their Applications*, pp. 257-268.

PREECE, A. D. (1995), Building the right system right: Verification and validation in knowledge engineering, *Review Talk*, http://www.csd.abdn.ac.uk/ ~apreece/Research/KQpage.html.

PREECE, A. D. and SHINGHAL, R. (1994), Foundation and application of knowledge base verification, *Intelligent Systems 9*, pp. 683-701.

ROUSSET, M.-C. (1988), On the consistency of knowledge bases: The COVADIS system, *Computational Intelligence 4*, pp. 166-170.

SANTOS-GOMEZ, L. and M.J. DARNELL (1992) "Empirical evaluation of decision tables for constructing and comprehending expert system rules". *Knowledge Acquisition*. Vol. **4** (4), pp. 427-444.

STEENKAMP, J.E.B.M. (1985) "De constructie van profielensets voor het schatten van hoofdeffecten en interacties bij conjunct meten". In: A.E. BRONNER, P.S.H. LEEFLANG, C.P. MIDDENDORP, A.J. OLIVIER, W.F. VAN RAAIJ and B. WIERENGA Eds. (1985) *Marktonderzoek en consumentengedrag*. Jaarboek van de Nederlandse Vereniging van Marktonderzoekers 1985. Haarlem, Uitgeverij De Vrieseborch, pp. 125-154.

SUWA, M., SCOTT, A. and SHORTLIFFE, E. H. (1982), An approach to verifying completeness and consistency in a rule-based expert system, *AI Magazine 3*, pp. 16-21.

TANAKA, M., N. AOYAMA, A. SUGIURA and Y. KOSEKI (1993) "Integration of multiple knowledge representation for classification problems". In: *Proceedings of the 5th International Conference on Tools with Artificial Intelligence*. Boston (Mass.), November 8-11, 1993, pp. 448-449.

TIMMERMANS, H.J.P. (1986) "Locational choice behaviour of entrepreneurs: an experimental analysis". *Urban Studies*. Vol. **23** (3), pp. 231-240.

TURBAN, E. (1995) *Decision Support and Expert Systems*. Management Support Systems (4th Edition). Englewood Cliffs (New Jersey), Prentice-Hall.

VANTHIENEN, J. (1986), *Automatiseringsaspecten van de Specificatie, Constructie en Manipulatie van Beslissingstabellen,* PhD dissertation, Katholieke Universiteit Leuven, Dept. of Applied Economic Sciences.

VANTHIENEN, J. (1991) "Knowledge acquisition and validation using a decision table engineering workbench". In: *Proceedings of the World Congress on Expert Systems: December 16-19, 1991.* Orlando (Florida), Pergamon Press, pp. 1861-1868.

VANTHIENEN, J. and DRIES, E. (1997), Decision tables: Refining concepts and a proposed standard, *Comm. of the ACM,* to appear.

VANTHIENEN, J., MUES, C. and WETS, G. (1997), Inter-tabular verification in an interactive environment, *Proc. of the 4th European Symposium on the Validation and Verification of Knowledge Based Systems*, pp. 155-165.

WINKELMANS, W. (1973) *De Moderne Havenindustrialisatie.* Rijswijk, Nederlands Vervoerswetenschappelijk Instituut, Ph.D. thesis.

WITLOX, F.J.A. (1998) *Modelling Site Selection: A relational matching approach based on fuzzy decision tables.* Eindhoven, Bouwstenen **49**, Ph.D. thesis.

WITLOX, F.J.A. and H.J.P. TIMMERMANS (1998) "MATISSE: A location model for assessing the site suitability of port-related companies using fuzzy decision tables". Working Paper (presented at the 8th World Conference on Transport Research). Antwerp, July 12-17, 1998, pp. 1-27.

WITLOX, F.J.A., H.J.P. TIMMERMANS and A.W.J. BORGERS (1998b) "MATISSE: Modelling site suitability of companies". Working Paper (presented at the 5th Recent Advances in Retailing & Services Science Conference). Baveno (Italy), August 25-28, 1998, pp. 1-26.

XANTHAKIS, S., RABOT W. and RICHARD P. (1995), Using conventional techniques for KBS: an overview, *Proc. of the European Symposium on the Validation and Verification of Knowledge Based Systems*, pp. 47-58.

Appendix : Addelman's basic plans

BASIC PLAN 1: 4; 3; 2^7; 8 trials

*	*	1234567
0	0	0000000
0	0	0001111
1	1	0110011
1	1	0111100
2	2	1010101
2	2	1011010
3	1	1100110
3	1	1101001
	*-1,2,3	

BASIC PLAN 2: 3^4; 2^4; 9 trials

1234	1234
0000	0000
0112	0110
0221	0001
1011	1011
1120	1100
1202	1000
2022	0000
2101	0101
2210	0010

BASIC PLAN 3: 4^5; 3^5; 2^{15}; 16 trials

12345 *****	12345 *****	00000 12345	00001 67890	11111 12345
00000	00000	00000	00000	00000
01123	01121	00001	10111	01110
02231	02211	00010	11011	10011
03312	01112	00011	01100	11101
10111	10111	01100	00110	11011
11032	11012	01101	10001	10101
12320	12120	01110	11101	01000
13203	11201	01111	01010	00110
20222	20222	10100	01011	01101
21301	21101	10101	11100	00011
22013	22011	10110	10000	11110
23130	21110	10111	00111	10000
30333	10111	11000	01101	10110
31210	11210	11001	11010	11000
32102	12102	11010	10110	00101
33021	11021	11011	00001	01011
1-000 *-123	2-000 *-456	3-000 *-789	4-111 *-012	5-111 *-345

BASIC PLAN 4: 7^3; 2^7; 18 trials

1234567	1234567
0000000	0000000
0112111	0110111
0221222	0001000
1011120	1011100
1120201	1100001
1202012	1000010
2022102	0000100
2101210	0101010
2210021	0010001
0021011	0001011
0100122	0100100
0212200	0010000
1002221	1000001
1111002	1111000
1220110	1000110
2010212	0010010
2122020	0100000
2201101	0001101

BASIC PLAN 5: 5^6; 4^6; 3^6; 2^6; 25 trials

123456	123456	123456	123456
000000	000000	000000	000000
011234	011230	011220	011110
022413	022013	022012	011011
033142	033102	022102	011101
044321	000321	000221	000111
101111	101111	101111	101111
112340	112300	112200	111100
123024	123020	122020	111010
134203	130203	120202	110101
140432	100032	100022	100011
202222	202222	202222	101111
213401	213001	212001	111001
224130	220130	220120	110110
230314	230310	220210	110110
241043	201003	201002	101001
303333	303333	202222	101111
314012	310012	210012	110011
320241	320201	220201	110101
331420	331020	221020	111010
342104	302100	202100	101100
404444	000000	000000	000000
410123	010123	010122	010111
421302	021302	021202	011101
432031	032031	022021	011011
443210	003210	002210	001110

BASIC PLAN 6: 9; 8; 7; 6; 5; 4; 3^{13}; 2^{13}; 27 trials

						00000	00001	111	00000	00001	111
*	*	*	*	*	*	12345	67890	123	12345	67890	123
0	0	0	0	0	0	00000	00000	000	00000	00000	000
0	0	0	0	0	0	00001	12121	212	00001	10101	010
0	0	0	0	0	0	00002	21212	121	00000	01010	101
1	1	1	1	1	1	01120	00111	122	01100	00111	100
1	1	1	1	1	1	01121	12202	001	01101	10000	001
1	1	1	1	1	1	01122	21020	210	01100	01000	010
2	2	2	2	2	2	02210	00222	211	00010	00000	011
2	2	2	2	2	2	02211	12010	120	00011	10010	100
2	2	2	2	2	2	02212	21101	002	00010	01101	000
3	3	3	3	1	1	10110	11001	111	10110	11001	111
3	3	3	3	1	1	10111	20122	020	10111	00100	000
3	3	3	3	1	1	10112	02210	202	10110	00010	000
4	4	4	4	3	3	11200	11112	200	11000	11110	000
4	4	4	4	3	3	11201	20200	112	11001	00000	110
4	4	4	4	3	3	11202	02021	021	11000	00001	001
5	5	5	4	3	3	12020	11220	022	10000	11000	000
5	5	5	4	3	3	12021	20011	201	10001	00011	001
5	5	5	4	3	3	12022	02102	110	10000	00100	110
6	6	6	5	4	2	20220	22002	222	00000	00000	000
6	6	6	5	4	2	20221	01120	101	00001	01100	101
6	6	6	5	4	2	20222	10211	010	00000	10011	010
7	7	6	5	4	2	21010	22110	011	01010	00110	011
7	7	6	5	4	2	21011	01201	220	01011	01001	000
7	7	6	5	4	2	21012	10022	102	01010	10000	100
8	0	0	0	0	0	22100	22221	100	00100	00001	100
8	0	0	0	0	0	22101	01012	012	00101	01010	010
8	0	0	0	0	0	22102	10100	221	00100	10100	001

*-1,2,3,4

BASIC PLAN 7: 4^9; 3^9; 2^{31}; 32 trials

123456789	123456789	00000	00001	11111	11112	22222	22	2233
*********	*********	12345	67890	12345	67890	12345	67	8901
000000000	000000000	00000	00000	00000	00000	00000	00	0000
011231111	011211111	00001	10111	01110	01101	10110	11	0000
022312222	022112222	00010	11011	10011	10110	11011	01	0000
033123333	011121111	00011	01100	11101	11011	01101	10	0000
101111032	101111012	01100	00110	11011	01100	01101	01	0011
110320123	110120121	01101	10001	10101	00001	11011	10	0011
123203210	121201210	01110	11101	01000	11010	10110	00	0011
132032301	112012101	01111	01010	00110	10111	00000	11	0011
202223102	202221102	10100	01011	01101	11001	10001	01	0101
213012013	211012011	10101	11100	00011	10100	00111	10	0101
220131320	220111120	10110	10000	11110	01111	01010	00	0101
231300231	211100211	10111	00111	10000	00010	11100	11	0101
303332130	101112110	11000	01101	10110	10101	11100	00	0110
312103021	112101021	11001	11010	11000	11000	01010	11	0110
321020312	121020112	11010	10110	00101	00011	00111	01	0110
330211203	110211201	11011	00001	01011	01110	10001	10	0110
002120213	002110211	00000	01010	11110	00010	10111	10	1111
013301302	011101102	00001	11101	10000	01111	00001	01	1111
020222031	020222011	00010	10001	01101	10100	01100	11	1111
031013120	011011120	00011	00110	00011	11001	11010	00	1111
103021221	101021221	01100	01100	00101	01110	11010	11	1100
112210330	112210110	01101	11011	01011	00011	01100	00	1100
121333003	121111001	01110	10111	10110	11000	00001	10	1100
130102112	110102112	01111	00000	11000	10101	10111	01	1100
200313311	200111111	10100	00001	10011	11011	00110	11	1010
211122200	211122200	10101	10110	11101	10110	1000	00	1010
222001133	222001111	10110	11010	00000	01101	11101	10	1010
233230022	211210022	10111	01101	01110	00000	01011	01	1010
301202323	101202121	11000	00111	01000	10111	01011	10	1001
310033232	110011212	11001	10000	00110	11010	11101	01	1001
323110101	121110101	11010	11100	11011	00001	10000	11	1001
332321010	112121010	11011	01011	10101	01100	00110	00	1001