DEX (Decision EXpert): A Qualitative Hierarchical Multi-criteria Method

Marko Bohanec

Abstract DEX (Decision EXpert) is a hierarchical, qualitative, rule-based, multi-criteria decision modeling method. It combines multi criteria decision analysis with artificial intelligence and is particularly suited for sorting/classification decision problems. DEX puts special attention on the transparency, comprehensibility, consistency, and completeness of decision models, as well as on methods for the analysis, justification, and explanation of decisions. The approach relies on using software tools that actively support the decision maker in both the creation and utilization stages of the process. Since its inception in the 1980s, DEX has been successfully applied in hundreds of real-world decision projects in various areas, including economy, ecology, agronomy, medicine, and health care. In the last decade, there is an increasing trend of including DEX models in decision support systems. In this chapter, DEX is described from the theoretical and practical viewpoint and further explained in terms of motivation, history, software, applications, and method extensions. The presentation is supported by three examples: a didactic example of employee selection and two real-world industrial applications of choosing a raw-material location and assessing electric energy production technologies, respectively.

Keywords Decision Expert (DEX) · Multi-criteria modeling · Qualitative decision model · Decision rules · Decision support system · DEXi

List of Symbols

$\mathcal{A} = \{A_1, A_2, \ldots, A_q\}$ Alternatives

M. Bohanec (✉)
Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
e-mail: marko.bohanec@ijs.si

University of Nova Gorica, Nova Gorica, Slovenia

$A_i \in \mathcal{A}, A_i = \{a_{x,i} \in E_x, \forall x \in X\}$	An alternative
$a_{x,i} \in A_i$	Value of A_i assigned to attribute x
$B_x \subset D_x$	Subset of bad values of attribute x
$C_y = \prod_{x \in S(y)} D_x$	Domain of f_y
D_x	Value scale of attribute x
E_x	Range of values that can be assigned to attribute x
E_y	Range of f_y
$e = (\mathbf{x}, y) \in T_y$	Elementary decision rule, an entry in T_y
F	Set of aggregation functions of a DEX model
$\mathcal{F}_y$	Fuzzy distributions over D_y
$f_y : C_y \to E_y$	Aggregation function associated with attribute y
g_y	An approximation of f_y
$G_x \subset D_x$	Subset of good values of attribute x
$I_i \subset A_i$	Subset of values of A_i, assigned to input attributes
I_y	Set of intervals over D_y
$m_x = |D_x|$	Number of categories of scale D_x
$M = (X, D, S, F)$	A DEX model
$N_x \subset D_x$	Subset of neutral values of attribute x
$O_i \subset A_i$	Subset of values of A_i, assigned to output (aggregate) attributes
$\mathrm{ord}(v_{x,i}) = i$	Ordinal value of $v_{x,i}$
$P(x)$	Set of parents of attribute x
$\mathcal{P}_y$	Probability distributions over D_y
$r_y = |C_y|$	Size of C_y and the corresponding T_y
$S : X \to 2^x$	Descendant function
$S(x)$	Set of descendants of attribute x
$\mathcal{S}_y$	The power set of D_y
$T_y = \{(\mathbf{x}, y), \mathbf{x} \in C_y, y \in E_y\}$	Decision table associated with attribute y
$v_{x,i} \in D_x$	i-th qualitative value (category) of attribute x
$v_{x,i} \preccurlyeq v_{x,j}$	Weak preference relation
$w, w_i \in \mathcal{R}$	Relative weight (importance) of an attribute
$x, x_i, y \in X$	An attribute
$X = \{x_1, x_2, \ldots, x_n\}$	Set of attributes
$\omega \in [-0.5, +0.5]$	An offset to qualitative value v

Abbreviations

3D	Three-dimensional
AHP	Analytic Hierarchy Process, an MCDM method

AQ	Algorithm Quasi-optimal, a machine rule learning algorithm
CDPC	Consistency-Driven Pairwise Comparisons, an MCDM method
DECMAK	DECision MAKing, an early predecessor of DEX
DEX	Decision EXpert, a qualitative MCDM method
DEXi	Software implementing the DEX method
DRSA	Dominance-based Rough Set Analysis, an MCDM approach
DSS	Decision Support System
ELECTRE	ELimination Et Choix Traduisant la REalité (ELimination Et Choice Translating REality), a family of MCDM methods
HINT	Hierarchical INduction Tool, a machine-learning method for developing DEX models from data
MACBETH	Measuring Attractiveness by a Categorical Based Evaluation Technique, an MCDM method
MCDM	Multi-Criteria Decision Modeling
MCHP	Multi-Criteria Hierarchy Process, a hierarchical MCDM approach
QQ	Qualitative-Quantitative, an approach to ranking of alternatives using a DEX model

1 Introduction

DEX (Decision EXpert) [19] is a multi-criteria decision modeling (MCDM) method, conceived in the 1980s as a fusion of multi-criteria decision analysis and artificial intelligence. From MCDM, it adopted the ideas of modeling decision situations using multiple criteria, structuring and decomposing complex decision problems in smaller and less complex sub-problems, and solving problems through evaluation and analysis of decision alternatives. From artificial intelligence, it primarily adopted concepts used in expert systems: using qualitative (symbolic) variables, representing decision knowledge in terms of "if–then" rules, handling imprecision and uncertainty, emphasizing the transparency of decision models, and facilitating the explanation of results. DEX also includes some elements of machine learning, e.g., for constructing compact decision rules from decision tables.

According to the classification in [49], DEX belongs to the category of *full aggregation* or "American school" methods. This approach is characterized by using an explicit multi-criteria model, which is developed first, more or less independently from individual decision alternatives. These alternatives are then evaluated by the model, first by scoring them for each criterion and then aggregating these evaluations into a global score.

DEX is also characterized as follows [22]:

1. DEX is *hierarchical*: a DEX model consists of hierarchically structured *attributes* (in MCDM, also called *criteria* or *performance variables*). In this aspect, DEX is similar to other hierarchical MCDM methods [3, 45], such as AHP [85] and MCHP [34].
2. DEX is *qualitative*: all attributes in a DEX model are symbolic, taking values that are words rather than numbers, such as "bad", "medium", "excellent", "low", or "high". This relates DEX to verbal decision analysis [65], linguistic MCDM [31, 42], and MCDM methods that use words, such as MACBETH [2].
3. DEX is *rule-based*: hierarchical aggregation of values is defined with decision rules, acquired and represented in the form of decision tables. In this way, DEX is most similar to Dominance-Based Rough Set Analysis [43], which also uses decision tables and constructs decision rules from them.

Given its qualitative nature, DEX is particularly suitable for *sorting* [82] or *classification* [39, 53] decision tasks, which are aimed at assigning each decision alternative to the one category among a family of predefined categories. These categories can be preferentially ordered (sorting) or not (classification). There are also variations of DEX adapted for the ranking problem [8, 60].

In the remaining part of this chapter, the DEX method is presented in detail. After a brief historical overview, the concept of a DEX model is formally defined and illustrated using an employee selection example. This is followed by dynamic aspects of DEX, which are reflected in algorithms that support the creation and modification of decision tables and perform the evaluation and analysis of alternatives. Practical applications of DEX are reviewed and illustrated by two real-world industrial examples: choosing a clay-pit location and assessing electric energy production technologies. Final sections include notes on DEX extensions and a summary.

2 DEX Method and Software: A Brief History

The development of DEX can be traced back to Efstathiou and Rajkovič [40] who proposed using fuzzy sets [93, 94] and fuzzy inference rules to represent and evaluate decision alternatives. The authors also suggested representing decision knowledge in terms of a decision table together with fuzzy operators. The following development of DEX was mainly continued at the Jožef Stefan Institute, Ljubljana, Slovenia, where elements of expert systems [50, 73] and machine learning [30, 64] were gradually added to the basic concepts, leaving the fuzzy aspects somewhat aside. The method,

presented by presented by Rajkovič, et al. [77] and Bohanec and Rajkovič [6] under the name DECMAK, already had all the main ingredients: tree-structured qualitative attributes, decision tables and decision rules, and algorithms supporting knowledge acquisition and explanation, including a graphical representation of decision tables and a machine-learning algorithm for constructing aggregate rules. About 30 practical applications, mainly in Slovenia, were reported at that time [6].

The name DEX (Decision EXpert) was first used in [7], to denote both the method and the supporting software that was developed at that time. In 2000, the DEX software was replaced by next-generation software called DEXi; at that point, the development team decided to keep the name DEX only for the method and use other names for its implementations.

DEX has always been closely tied with the supporting software. Due to the combinatorial nature of DEX's decision tables (explained in the next section), the method is unsuitable for manual construction of models and becomes practical only when supported by appropriate user interfaces and algorithms for knowledge elicitation, representation, verification, and explanation. In many aspects, the definition of the DEX method followed the actual software implementations, which is a somewhat unusual practice in the MCDM area.

Three generations of DEX-related software have been developed so far:

1. *DECMAK* [6] was released in 1981 for mini and personal computers under operating systems RT-11, VAX/VMS, and MS-DOS.
2. *DEX* [7] was released in 1987 as an integrated interactive computer program for VAX/VMS and MS-DOS.
3. *DEXi* [27] was released in 2000 for Microsoft Windows.

Originally, DEXi was designed as educational software (the letter "i" in DEXi, pronounced "ee", actually comes from the Slovenian "izobraževanje", education). DEXi was—and still is—used in Slovenian secondary schools and universities in MCDM and decision-support courses. Since 2000, additional features were gradually added to DEXi, which eventually became a complete, stable, and de facto standard implementation of DEX. DEXi supports an interactive creation and editing of all components of DEX models (attributes, their hierarchy and scales, decision tables, and alternatives) and provides methods for the evaluation and analysis of alternatives (what-if analysis, "plus-minus-1" analysis, selective explanation, comparison of alternatives). DEXi is free software, available at http://kt.ijs.si/MarkoBohanec/dexi.html together with other DEX-related software, which includes the following:

- *DEXiEval*, *JDEXi,* and *DEXi.NET*: Implementations of DEX evaluation procedure in different environments: command-line, Java, and C#, respectively,
- *DEXi HTML Evaluator*: A software package for running DEXi models in Web browsers, and
- *DEXx*: A Java-based software library [90].

3 Formal Representation of a DEX Model

The DEX method is defined from two aspects: static and dynamic. The static aspect gives a formal description of components and concepts of a DEX model. The dynamic aspect addresses algorithms and tools necessary to develop and modify the model and to use it for the evaluation and analysis of alternatives. In this section, we begin with static aspects and continue with dynamic aspects in the next. The formal notation is adapted from Trdin and Bohanec [90].

A DEX *model M* is a four-tuple $M = (X, D, S, F)$, where X is the set of *attributes*, S is the descendant function that determines the hierarchical structure of attributes, D is the set of value scales of attributes in X, and F is the set of aggregation functions.

3.1 Attributes

The set X contains *n attributes*: $X = \{x_1, x_2, \ldots, x_n\}$. Attributes are variables that represent observable properties of the decision problem and decision alternatives. In DEX models, attributes are usually given unique and meaningful names, such as *Price* and *Productivity*. In such cases, the notation x_i is conveniently and conventionally replaced by the corresponding attribute name.
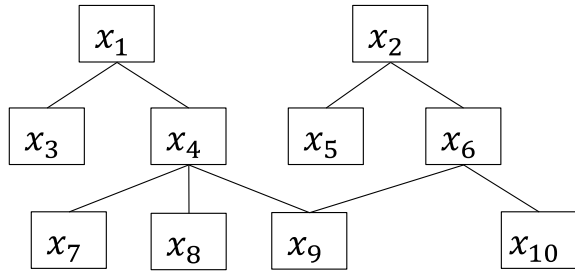
3.2 Model Structure: Hierarchy of Attributes

Attributes in a DEX model are structured *hierarchically*. The structure is defined by the function $S : X \to 2^x$, which associates each $x \in X$ with a set of its *descendants* $S(x)$ in the hierarchy. The relation between an attribute and its descendants represents both dependence and influence: an attribute x *depends* on attributes in $S(x)$ and attributes from $S(x)$ *influence* x.

Given S, the set of *parents* of each $x \in X$ is defined as $P(x) = \{p \in X : x \in S(p)\}$. Attributes without parents are called *roots* and represent main *outputs* of the model. Attributes without descendants, $S(x) = \emptyset$, are called *basic* attributes and represent model *inputs*. Attributes with $S(x) \neq \emptyset$ are referred to as *aggregate* attributes and are also considered (partial, lower-level) *outputs* of the model.

The function S is required to represent a *hierarchy*, i.e., a connected and directed (from attributes to their descendants) acyclic graph with one or more roots. Figure 1 shows an example of a hierarchy, composed of ten attributes x_1 to x_{10} so that $S(x_1) = \{x_3, x_4\}$, $S(x_2) = \{x_5, x_6\}$, $S(x_4) = \{x_7, x_8, x_9\}$, $S(x_6) = \{x_9, x_{10}\}$, and $S(x_i) = \emptyset, i \in \{3, 5, 7, 8, 9, 10\}$. This means that x_1 and x_2 are roots. There are six basic attributes: $x_3, x_5, x_7, x_8, x_9,$ and x_{10}. Among these, x_9 influences two parents, x_4 and

Fig. 1 Example of a hierarchy of attributes with two roots and 6 input attributes

x_6, while each of the remaining ones influences only one parent. Attributes x_1, x_2, x_4, and x_6 are aggregate and depend on their respective descendants.

In practice, DEX models are most often structured as trees rather than general hierarchies. A tree is a special type of hierarchy in which all attributes, except a single root attribute, have exactly one parent.

Example. Hereafter, we illustrate DEX concepts using a simple didactic model called *Employ*, which is distributed with the DEXi software. The model is aimed at the assessment of applicants for a Project Manager position in a small company. An earlier version was published in [8].

Figure 2 shows the structure of *Employ*. It consists of 12 tree-structured attributes. The root attribute is also called *Employ* and represents the output evaluation of job applicants. Applicants are assessed according to three groups of attributes, represented by aggregate attributes *Educat*, *Years,* and *Personal*. All of them are structured further, leading to seven basic attributes: *Formal*, *For.lang*, *Exper*, *Age*, *Comm*, *Leader*, and *Test* (see Fig. 2 for descriptions). These represent the observed characteristics of applicants and have the role of input variables.

Fig. 2 Structure of the *Employ* model with descriptions of attributes

Attribute	Description
Employ	Employee selection demo: Project manager
├**Educat**	Education
│├Formal	Formal education (degree)
│└For.lang	Mastering of foreign language (English)
├**Years**	Age and experience
│├Exper	Professional experience in the field
│└Age	Age of the candidate
└**Personal**	Personal characterisrics
├**Abilit**	Abilities
│├Comm	Cummunicability
│└Leader	Leadership ability
└Test	Result of a psychological test

3.3 Scales

Each attribute $x \in X$ is associated with a value *scale* $D_x \in D$, which is defined as an ordered set of symbolic (qualitative) values: $D_x = \{v_{x,1}, v_{x,2}, \ldots, v_{x,m_x}\}$. Here, $m_x \geq 2$ denotes the number of discrete values that can be assigned to x. Usually, value scales are small and rarely consist of more than five values. Scale values are typically represented by words rather than numbers, for instance "low", "high", "unacceptable", and "good".

DEX scales can be either *ordered* or *unordered*.[1] Values of an ordered scale are assumed to be preferentially ordered so that $v_{x,1} \preccurlyeq v_{x,2} \preccurlyeq \cdots \preccurlyeq v_{x,m_x}$, where '$\preccurlyeq$' denotes a weak preference relation. Additionally, each scale D_x is partitioned in three subsets $B_x, N_x, G_x : B_x \cup N_x \cup G_x = D_x, B_x \cap N_x = B_x \cap G_x = N_x \cap G_x = \emptyset$. These subsets represent particularly bad, neutral, and particularly good values from D_x, respectively. They are convenient for displaying DEX values using different colors and fonts (usually red bold for **bad** and green bold italic for *good* values). By default, ordered scales are partitioned to $B_x = \{v_{x,1}\}$, $G_x = \{v_{x,m_x}\}$ and $N_x = D_x - (B_x \cup G_x)$, and unordered scales to $B_x = G_x = \emptyset$ and $N_x = D_x$.

According to the definition in [43], attributes that are associated with ordered scales are called *criteria*. In this way, a DEX model generally consists of attributes X, some of which are criteria. An attribute can be considered a criterion only after it has been associated with an ordered scale. For this reason, DEX is often referred to as a multi-attribute rather than multi-criteria method.

Example. Figure 3 shows the scales assigned to the attributes of *Employ*. The colors indicate that all scales, except D_{Age}, are ordered (increasingly) and partitioned using the default rule so that the worst and best attribute values appear at the beginning and end of the value list, respectively. Scale D_{Age} is unordered. Most of the values are represented by words: "unacc", "high", etc. Even though some values, for instance "1–5" and "21–25", are formulated as numeric intervals, they still represent single discrete symbols.

Fig. 3 Attributes of the *Employ* model associated with scales

Attribute	Scale
Employ	**unacc**; acc; good; *exc*
├**Educat**	**unacc**; acc; *good*
│├Formal	**prim-sec**; high; univ; MSc; *PhD*
│└For.lang	**no**; pas; *act*
├**Years**	**unacc**; acc; *good*
│├Exper	**no**; to1year; 1-5; 6-10; *more*
│└Age	18-20; 21-25; 26-40; 41-55; more
└**Personal**	**unacc**; acc; *good*
├**Abilit**	**unacc**; acc; *good*
│├Comm	**poor**; aver; good; *exc*
│└Leader	**less**; approp; *more*
└Test	**D**; C; B; *A*

---

[1] Actually, DEX implementations distinguish between *increasing*, *decreasing* and *unordered* scales. Here, we simplify the definition without loss of generality.

3.4 Aggregation Functions

The fourth and final component of the static DEX model definition is $F = \{f_x, x \in X\}$, a set of *aggregation functions* (also called *utility functions* in some software and older publications). An aggregation function serves for the evaluation of an aggregate attribute based on values of its immediate descendants in the model structure. Each aggregate attribute $y \in X$, $S(y) \neq \emptyset$ is thus associated with a total function.

$$f_y : D_{(1)} \times D_{(2)} \times \cdots \times D_{(k_y)} \to E_y,$$

where the Cartesian product refers to scales of $S(y) = \{x_{(1)}, x_{(2)}, \ldots, x_{(k_x)}\}$, where $x_{(1)}, x_{(2)}, \ldots, x_{(k_x)}$ are all descendants of y. In this way, $x_{(1)}, x_{(2)}, \ldots, x_{(k_x)}$ are arguments of f_y; in the context of f_y and corresponding decision tables, they are also referred to as *incoming* attributes. E_y denotes the range of f_y. Normally, the output range corresponds to the scale of y, that is, $E_y \equiv D_y$. However, for reasons that are explained later, E_y is often extended to:

- I_y, the set of intervals over D_y,
- $\mathcal{S}_y$, the power set of D_y,
- $\mathcal{P}_y$, probability distributions over D_y, or
- $\mathcal{F}_y$, fuzzy distributions over D_y.

In DEX, aggregation functions are represented by *decision tables*. Let us denote $C_y = D_{(1)} \times D_{(2)} \times \cdots \times D_{(k_x)}$ and $r_y = |C_y|$. Then, a decision table T_y consists of r_y entries $T_y = \{(\mathbf{x}_i, y_i), \mathbf{x}_i \in C_y, y_i \in E_y, i = 1, 2, \ldots, r_y\}$. Entries are often referred to as *elementary decision rules*: each rule defines the function value y_i for some combination of values of its arguments $\mathbf{x}_i$. Entries are required to be unique so that $\mathbf{x}_i \neq \mathbf{x}_j, i, j = 1, \ldots, r_y, i \neq j$. When completely defined, a decision table is normally expected to define output values for all possible $\mathbf{x} \in C_y$.

Example. Two completely defined decision tables are shown in Fig. 4. They define the functions that aggregate attributes *Abilit* and *Test* to *Personal* (left), and *Comm* and *Leader* to *Ability* (right). Each table contains 12 elementary decision rules, according

Fig. 4 Two decision tables, defining aggregation functions of *Personal* (left), and *Abilit* (right)

	Abilit	Test	Personal		Comm	Leader	Abilit
1	unacc	D	unacc	1	poor	less	unacc
2	unacc	C	unacc	2	poor	approp	unacc
3	unacc	B	unacc	3	poor	more	unacc
4	unacc	A	unacc	4	aver	less	unacc
5	acc	D	unacc	5	aver	approp	acc
6	acc	C	unacc	6	aver	more	acc
7	acc	B	acc	7	good	less	unacc
8	acc	A	good	8	good	approp	acc
9	good	D	unacc	9	good	more	good
10	good	C	acc	10	exc	less	unacc
11	good	B	good	11	exc	approp	good
12	good	A	good	12	exc	more	good

to the number of possible value combinations of the corresponding C_y. Each value combination appears only once in each table. Each row in the table can be easily interpreted as an elementary "if–then" rule; for instance, rule 4 in the *Personal* table can be read as.

if *Abilit* = "unacc" and *Test* = "A" then *Personal* = "unacc".

In addition to two functions shown in Fig. 4, the *Employ* model contains three other decision tables, associated with attributes *Employ*, *Educat,* and *Years*; these are not shown here.

3.5 Alternatives

Once developed, a DEX model serves for the evaluation and analysis of decision alternatives. Formally, *alternatives* $\mathcal{A} = \{A_1, A_2, \ldots, A_q\}$ are not part of a DEX model M, but are rather considered as external data objects processed by M. Each *alternative* $A_i, i = 1, 2, \ldots, q$, is represented by a set of values:

$$A_i = \{a_{x,i} \in E_x, \forall x \in X\},$$

where each $a_{x,i}$ represents the value of A_i that is assigned to attribute x. Similarly as with aggregation functions, E_x is normally identical to D_x. However, in order to represent incomplete and/or uncertain information about alternatives, E_x may be in some contexts extended to value intervals, subsets, or value distributions.

The sets A_i are naturally partitioned in subsets I_i and O_i so that $A_i = I_i \cup O_i, I_i \cap O_i = \emptyset$. The two subsets correspond to basic and aggregate attributes of X, respectively. The former, I_i, represents basic observable properties of each A_i, which are defined by the decision maker and provide input data for the evaluation. In contrast, the values aligned with aggregate attributes, O_i, are calculated using the model and are thus obtained as results (outputs) of the evaluation. The most important results are those assigned to one or more roots of the model.

Example. In the *Employ* use case, alternatives are job applicants. Table 1 shows input data (that is, the corresponding B_i) of four applicants, named A, B, C, and D. In this case, all alternatives are represented by single values taken from the scales of corresponding attributes.

Table 1 Four job applicants, described by the values of basic attributes

Basic attribute	Applicants			
	A	**B**	**C**	**D**
Formal	MSc	*PhD*	*PhD*	*PhD*
For.lang	pas	*act*	*act*	*act*
Exper	to1year	*more*	6-10	6-10
Age	21-25	26-40	26-40	26-40
Comm	good	aver	good	*exc*
Leader	*more*	**less**	**less**	*more*
Test	B	B	C	*A*

3.6 Evaluation of Alternatives

Evaluation of alternatives is a process aimed at calculating output values of all alternatives that have been previously described by values of input attributes. Given some model M, the evaluation is carried out as a bottom-up aggregation of model inputs toward its outputs, according to the hierarchical structure of attributes and associated aggregation functions. Algorithmically, considering that a DEX model generally consists of a hierarchy of attributes, all attributes in M are first topologically sorted with respect to S. This determines the order of aggregation function evaluations and ensures the availability of all incoming inputs for calculating the output values of each subsequent aggregation function. Given function arguments, the output of that function is determined by a simple lookup in the corresponding decision table.

Example. Figure 5 shows evaluation results of the four applicants, defined previously in Table 1. Each column in Fig. 5 represents a complete set of values A_i of the corresponding applicant. The main outputs are assigned to the attribute *Employ*, indicating that the applicant D was assessed as "exc", A as "good", and the remaining two applicants as "unacc". Other outputs include values assigned to the remaining

Fig. 5 Evaluation of job applicants

Attribute	A	B	C	D
Employ	good	**unacc**	**unacc**	*exc*
├**Educat**	acc	*good*	*good*	*good*
│├Formal	MSc	*PhD*	*PhD*	*PhD*
│└For.lang	pas	*act*	*act*	*act*
├**Years**	acc	*good*	*good*	*good*
│├Exper	to1year	*more*	6-10	6-10
│└Age	21-25	26-40	26-40	26-40
└**Personal**	*good*	**unacc**	**unacc**	*good*
├**Abilit**	*good*	**unacc**	**unacc**	*good*
│├Comm	good	aver	good	*exc*
│└Leader	*more*	**less**	**less**	*more*
└Test	B	B	C	*A*

aggregate attributes *Educat*, *Years*, *Personal,* and *Abilit*. These values provide additional information about the candidates and help explaining the main results. For instance, one can easily see that the "unacc" *Personal* values of B and C have likely caused their "unacc" overall assessments, despite excellent assessments achieved at *Educat* and *Years*.

4   Dynamic Aspects of DEX

Dynamic aspects of DEX modeling refer to procedures, algorithms, and tools that are primarily used in two distinct decision analysis stages:

1. *Creation*: Here, the task is to develop an operational DEX model, usually starting from the scratch and aiming to satisfy both the goals of the decision maker and formal requirements, presented in the previous section. The main challenges addressed in this stage are how to (1) define the model and its components, (2) modify, edit, and maintain the model, (3) verify the model and its components (e.g., for completeness and consistency), (4) deal with uncertainty of knowledge and modeled phenomena, and (5) ensure transparency and comprehensibility of the model.
2. *Usage*: In this stage, one or more DEX models are already available and we want to use them to effectively solve the decision problem. This is associated with questions of how to (1) obtain and represent data about alternatives, (2) handle incomplete or uncertain data about alternatives, (3) evaluate alternatives, and (4) analyze, explain, justify, and validate results.

Among these, the representation and evaluation of alternatives have already been covered in the previous section. The remaining aspects are addressed in this section. The presentation is restricted to—and illustrated by—solutions implemented in the DEXi software.

4.1   Developing Model Components and Structure

DEX models are typically developed by individual *decision makers* or *groups*, the so-called *decision-problem owners* who are responsible for making the decision at hand. In the case of complex decision problems, the team is often extended with *experts* and *decision analysts*. The former provides expertise about the problem domain and help formulate model components. The latter are responsible for an appropriate use of the methodology and supporting tools and usually guide or even lead the process.

In most cases, DEX models are developed through *expert modeling*, i.e., "hand-crafting" of model components and structure, following the approach of expert systems. In this process, DEX models do not only "grow" from the scratch, but are often changed in other ways: attributes are added or deleted, their scales and
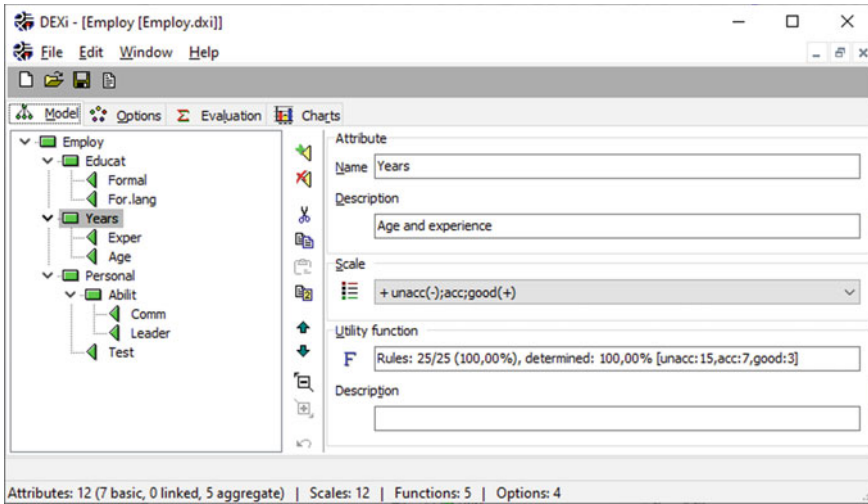
Fig. 6 DEXi model editor

aggregation functions are defined or changed, attribute hierarchies are restructured, etc. In practice, it is essential to support these needs by providing suitable software tools, such as DEXi.

For creating, editing and structuring attributes, DEXi provides an editor, shown in Fig. 6. All operations, mentioned above, are implemented, including model restructuring through drag-drop, duplicate, and copy-paste operations.

In addition to using software tools, many recommendations and "rules of thumb" of how to approach DEX modeling have been formulated from practical experience [27]. Regarding the selection of attributes, recommendations are the same as for any MCDM method: use attributes that are *relevant* for the problem and try not to overlook really *important* ones; avoid using *redundant* or closely correlated (*non-orthogonal*) attributes; assure that all input attributes are *operational* so that their values can be obtained for all alternatives in a sufficiently straightforward, well-defined, and accurate way.

With regard to developing model structure, DEX is similar to other hierarchical methods, such as AHP, but has some specific characteristics. In order to avoid too large decision tables (see the next section), it is recommended to make "narrow" hierarchies and limit the number of descendants of aggregate attributes to three or four at most. If an attribute requires, say, four descendants, consider structuring them further into sub-trees of $2 + 2$, $3 + 1$, or $2 + 1 + 1$ attributes.

In MCDM, two primary approaches are generally advocated for model structuring (see an overview in [58]): *top-down* (recursive decomposition of the root attribute to sub-trees) and *bottom-up* (defining input attributes first and gradually combining them toward the root). From practical experience, we can assure that none of them alone works really well; the most effective is the *middle-out* approach that combines

both. Usually, the process starts by making a preliminary and unstructured list of attributes. Related attributes from the list are then grouped together in a bottom-up way, and attributes that seem too complex, too general, or too difficult to measure are decomposed into simpler ones using the top-down approach. Often, new attributes are created in this process and old ones discarded, which normally requires several iterations of restructuring the model.

When combining attributes into a subtree, it is very important to group together attributes that are conceptually related and bear a common meaning. An excellent practical criterion is whether or not we can give a meaningful name to the newly created parent attribute. For example, considering basic attributes in Table 1, these were grouped together as shown in Fig. 2. For instance, *Formal* and *For.lang* were combined into *Educat*, and *Exper* and *Age* were combined into *Years*, which are both easy to interpret. As a didactic exercise, the reader is invited to combine the pairs {*Formal*, *Exper*}, {*Formal*, *Test*}, {*Exper*, *Leader*}, and {*For.lang*, *Test*} and try to find suitable names for the corresponding parent attributes.

With regard to designing scales, the following recommendations have been formulated [27]:

- For basic attributes: use the least number of values that is still sufficient to distinguish between importantly different characteristics of alternatives with respect to that attribute. Usually, two to four values are sufficient. For instance, there are only three qualitatively different levels relevant to assess mastering of formal language (*For.lang*) in the *Employ* model: "no", "passive", and "active".
- For aggregate attributes: The number of values should gradually increase from input attributes toward the root. For example, three four-valued attributes might be aggregated into a five-valued attribute. Five-valued scales are generally recommended for root attributes, as they are usually sufficient and work quite well.
- On scale ordering: Use preferentially ordered scales whenever possible; they really help in the definition of decision tables. If some attribute does not have a natural preferential order, try reformulating or converting it to an ordered one. Avoid using decreasing scales; they tend to be less comprehensible than increasing ones.

4.2 Acquiring Decision Tables and Decision Rules

The evaluation process in DEX is guided by decision tables. In general, a decision table consists of elementary decision rules that determine output values for each combination of input values. This adds a combinatorial aspect and makes DEX decision tables somewhat harder to define than the corresponding aggregation functions in other MCDM methods, including AHP. In practice, it turned out that it was really important to provide interactive software tools that aid the development of decision tables.

Figure 7 illustrates three typical stages of creating a decision table in DEXi. The leftmost screenshot shows that DEXi automatically generates all possible value
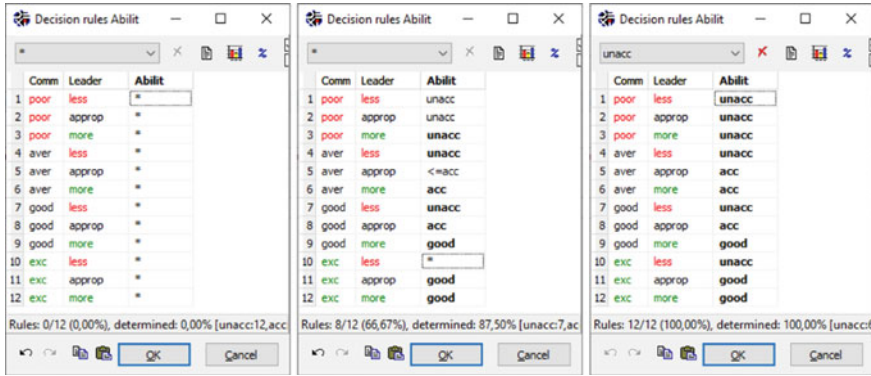
Fig. 7 Three stages of creating aggregation function for *Abilit* in DEXi

combination of descendant attributes (*Comm* and *Leader* in this example), releasing the decision maker from the burden of keeping track of all combinations. The right-most column initially contains asterisks '*', which indicate any possible value of the output attribute *Abilit*. It is important to understand that '*' represents the whole range of *Abilit*'s values, indicating that DEXi actually extends the notation E_y, introduced previously, to *intervals over D_y*. This extension is necessary for practical reasons and facilitates a smooth and user-friendly creation of decision tables from the scratch.

The second screenshot in Fig. 7 illustrates another important concept of DEX: considering the *principle of dominance* and trying to maintain the *consistency* of decision rules and *monotonicity* of aggregation function; for theoretical founda-tions, see [43, 44]. Let us assume that some decision table maps incoming attributes $x_1, x_2, \ldots, x_k$ to y, and all attributes are preferentially ordered. Suppose that a decision table already contains the entry

$$ e = (\mathbf{x}_e, y_e), \mathbf{x}_e = \left(a_{1,e}, a_{2,e}, \ldots, a_{k,e}\right), a_{i,e} \in D_i, i = 1, 2, \ldots, k, y_e \in D_y. $$

Then, the principle of dominance requires that for any other entry f, where $\mathbf{x}_f \succcurlyeq \mathbf{x}_e$, it should hold $y_f \succcurlyeq y_e$ (and analogously for '$\preccurlyeq$'). Here, $\mathbf{x}_f \succcurlyeq \mathbf{x}_e$ is defined to hold if $a_{i,f} \succcurlyeq a_{i,e}$ for all $i = 1, 2, \ldots, k$, and $a_{i,f} \succ a_{i,e}$ is true for at least one i. In this case, f is said to *dominate* e. If none of the $\mathbf{x}_f \succcurlyeq \mathbf{x}_e$ or $\mathbf{x}_f \preccurlyeq \mathbf{x}_e$ are true, the entries e and f are incomparable. A decision table in which all comparable pairs of entries comply with the principle of dominance is *consistent* and defines a *monotone* aggregation function.

Even though one can define decision table entries one by one in succession, this is rarely done in DEXi because of the substantial help provided by the dominance principle. The second screenshot in Fig. 7 shows the situation where the decision maker has already defined eight entries: 3, 4, 6, 7, 8, 9, 11, and 12 (the respective output values are shown in bold). Comparing the entries 3 and 2, one can easily see that they differ only in the value of *Leader*. Since "more" $\succcurlyeq$ "approp", rule 3

dominates 2. The output value of rule 3 is "unacc", and the value of rule 2 should be worse or equal than that; this leaves only one possibility for the value of rule 2: "unacc". In this way, the value of rule 2 has been fully determined only from the previously defined value of rule 3. In this case, rule 3 provided an *upper bound* for the value of rule 2.

Rule 5 in the second screenshot in Fig. 7 illustrates two additional facts: (1) rule values are indeed intervals (the display "<= acc" actually denotes the interval ["unacc", "acc"]), and (2) both lower and upper bounds of such intervals can be determined from already defined entries. Rule 5 dominates rules 1, 2, and 4, which are all "unacc", which sets the lower bound of rule 5. Rule 5 is dominated by rules 6, 8, 9, 11, and 12. The worst value of these rules is "acc", which is taken as the upper bound of rule 5.

In this way, one can effectively develop a decision table by first providing a few entries, and then gradually assigning single values to entries that still contain intervals.

The third screenshot in Fig. 7 shows a fully developed table. If not overridden by the user, DEXi checks the consistency at all times and issues a warning if it is violated. Strictly following this procedure assures that the resulting tables (and consequently the whole model) are consistent and *complete*, i.e., they explicitly define output values for all possible combinations of input attribute values.

As already mentioned, DEX decision tables are sensitive to the number of incoming attributes and the size of their value scales: for k incoming attributes $x_1, x_2, \ldots, x_k$, the total number of entries equals to $r = \prod_{i=1}^{k} |D_i|$. In practice, it turns out that decision tables with sizes of up to 25 are small and usually quite easy to define. The difficulty increases toward the size of about 100, which is already quite difficult. Everything above 100 is very difficult, and everything above 500 is extremely hard if not impossible to define. The number of incoming attributes also matters: the more the attributes, the more difficult the rules to define, even if the size of the tables is comparable. In all such cases, it is strongly recommended [27] to restructure the model into narrower subtrees.

4.3   Restructuring Decision Tables

In some circumstances, it might be necessary to restructure the space around some already defined decision table, for instance by adding or deleting an incoming attribute or changing the definition of bounding scales. In practice, it is important to *preserve* as much information already contained in the table as possible. DEXi automatically restructures tables whenever possible. For example, Fig. 8 shows what happens with the table when the value "acc" is deleted from the scale of *Abilit*: the rules with previously assigned values "unacc" and "good" are preserved, and only previous "acc" entries need to be redefined.
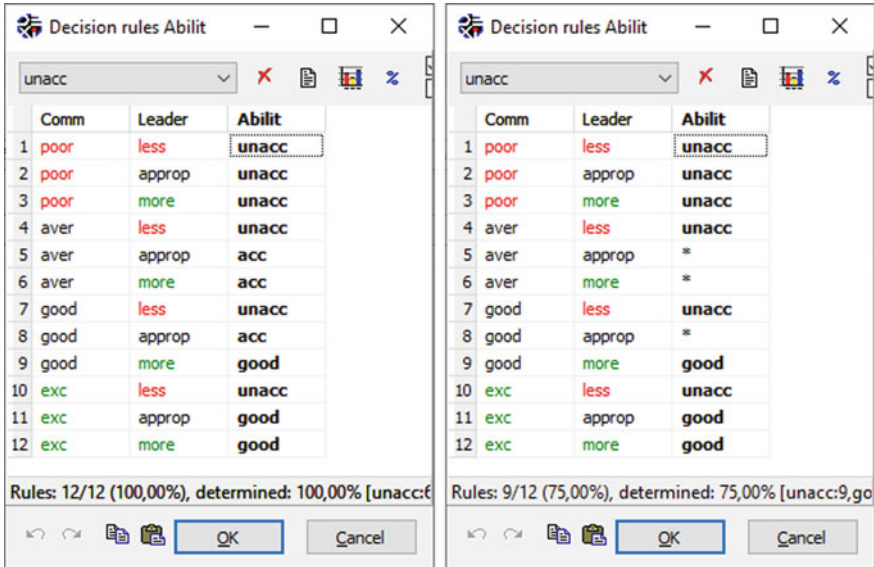
Fig. 8 Decision table *Abilit* before and after deleting "acc" from the output scale

## 4.4 Representation of Decision Tables: Complex Rules and 3D Graphics

Decision tables in DEXi are always acquired in terms of elementary decision rules (table rows). However, once completed, larger tables tend to become difficult to read and understand. To alleviate this problem, DEXi employs two methods: representation using complex rules and 3D graphic visualization.

The first method uses an algorithm that constructs a more compact table representation using *complex rules*. These are obtained by joining several elementary rules which have the same function value. The algorithm, whose presentation is beyond the scope of this chapter, belongs to the class of rule learning algorithms. Originally [6], it was adopted from the machine learning algorithm called AQ [59]. Recently, it has been enhanced for efficiency [51].

Using this algorithm, the *Abilit* decision table is presented in a more compact way with only 6 complex rules as shown in Fig. 9.

Fig. 9 Decision table *Abilit* represented with complex rules

	Comm	Leader	Abilit
1	**poor**	*	**unacc**
2	*	**less**	**unacc**
3	aver	>=approp	acc
4	aver:good	approp	acc
5	>=good	*more*	*good*
6	*exc*	>=approp	*good*
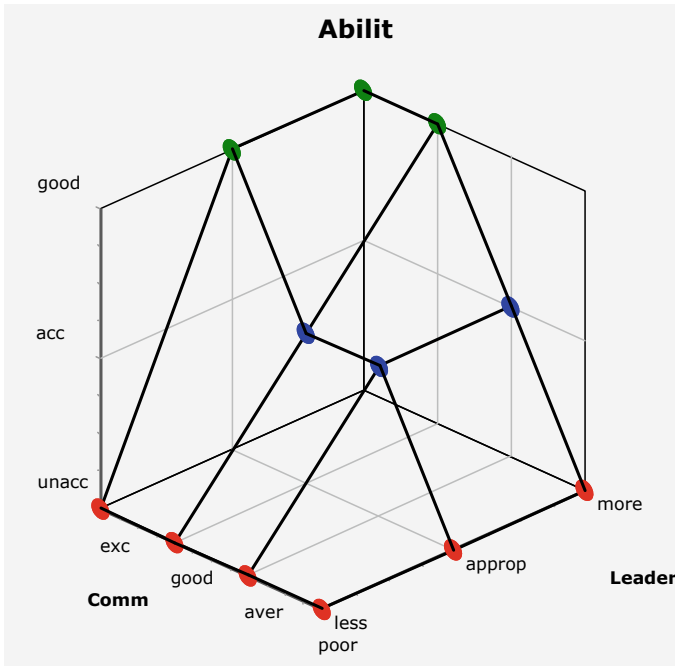
Fig. 10 Decision table *Abilit* represented with 3D graphic

The second method displays decision tables using 3D graphics (Fig. 10). There, table entries are interpreted as points in a multi-dimensional space. In the case of three or more incoming attributes, 3D intersections through the space are shown interactively. It is important to note that lines in Fig. 10 are there only to aid the 3D perception and are not part of the function definition, which remains discrete. It is also worth noticing that the function in Fig. 10 is somewhat typical for DEX; it resembles the minimum function and is not linear, in contrast with MCDM methods that use linear aggregation functions and weights.

4.5 Handling Incomplete Knowledge and Data

With this section, we turn attention to the usage stage, in which decision alternatives are represented and evaluated as described above in the formal section. In this stage, DEXi addresses two practically important issues: (1) handling incomplete data about alternatives and incompletely defined decision tables (this section) and (2) supporting analysis of the decision situation and individual alternatives (the next section).

As already indicated, DEX was inspired by ideas of expert systems. One of the most fundamental requirements for expert systems is that they must be able to process incomplete and uncertain knowledge. An expert system is expected to provide some answers, albeit incomplete or less accurate, even in the case of missing or uncertain input data, or "holes" in knowledge captured in the system.

The DEXi software implements a very simple version of this requirement using *value sets*: the notation E_x, introduced above, is extended to sets over D_x. In this way, values assigned to attributes by the evaluation algorithm are generally not single discrete values any more, but rather subsets of the corresponding scales. The evaluation algorithm iterates over all members of the input sets, and accumulates individual evaluations in corresponding output sets. Note that this approach handles both missing input data (which might be represented by '*', i.e., all values from the corresponding scale) and incompletely defined decision tables (by converting outgoing intervals to sets).

Figure 11 illustrates what happens in DEXi when some input data about job applicants is unknown. Candidate A has not been assessed with respect to his leadership abilities. Consequently, the model cannot really assess his *Personal* characteristics. The overall evaluation is represented by the set {"unacc", 'acc', "good"}, which does not say much, but indicates that A cannot reach the "exc" result. In contrast, candidates B and C are both assessed as "unacc", despite missing data of *Comm* and *For.lang*, respectively. Canididate D, whose *Test* results are currently unknown, achieved an extreme evaluation {"unacc", "exc"}. This indicates that she has the potential for becoming an excellent candidate, but subject to *Test* results, which may importantly determine the outcome.

Fig. 11 Evaluation of job applicants based on missing input data

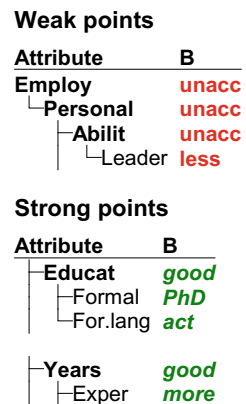| Attribute	A	B	C	D
Employ	unacc; acc; good	unacc	unacc	unacc; *exc*
├**Educat**	acc	*good*	*	*good*
│├Formal	MSc	*PhD*	*PhD*	*PhD*
│└For.lang	pas	*act*	*	*act*
├**Years**	acc	*good*	*good*	*good*
│├Exper	to1year	*more*	6-10	6-10
│└Age	21-25	26-40	26-40	26-40
└**Personal**	*	unacc	unacc	*
├**Abilit**	*	unacc	unacc	*good*
│├Comm	good	*	good	*exc*
│└Leader	*	*less*	*less*	*more*
└Test	B	B	C	*

4.6 Analysis of Alternatives

Analysis is one of the key concepts of MCDM and decision analysis in general. In contrast with evaluation, which merely calculates output results, analysis of alternatives is understood as an active involvement of participants who are trying to understand the decision situation, explain, and justify individual evaluations, explore the consequences of potential changes and search for better solutions. In DEXi, decision analysis is supported by three methods [27]: "what-if" analysis, selective explanation and "plus-minus-one" analysis.

What-if analysis is an exploration of consequences caused by changes of input data or aggregation functions. In DEXi, it is carried out through an iterative interactive process consisting of duplicating some alternative, changing data in one instance, and comparing both alternatives.

Selective explanation is aimed at the identification of particularly strong and weak characteristics of some alternative. Here, DEXi takes advantage of partitioning attribute scales into "good" and "bad" subsets. The method finds and displays all connected subtrees of attributes whose values are either all "good" or "bad". An example of such a display for job applicant B is shown in Fig. 12. It clearly highlights the candidate's main disadvantage, i.e., leadership abilities. On the other hand, the candidate does have advantages, reflected in *Educat* and *Exper*, so she might be considered for some other job position. Although based on a very simple idea, selective explanation has been found indispensable in practice for explaining and justifying decisions.

Plus-minus-one analysis investigates the effects of changing each basic attribute by one value down or up (if possible), independently of other attributes. Figure 13 shows results for candidate A. The column labeled A shows the current values, and the topmost value "good" is the current overall evaluation. The column "–1" shows the overall evaluation in the case that the corresponding attribute's value

Fig. 12 Job candidate B: Selective explanation

Weak points

Attribute	B
Employ	**unacc**
└**Personal**	**unacc**
├**Abilit**	**unacc**
│ └Leader	**less**

Strong points

Attribute	B
├**Educat**	*good*
│├Formal	*PhD*
│└For.lang	*act*
├**Years**	*good*
│├Exper	*more*

Fig. 13 Job candidate A: Plus-minus-one analysis

Attribute	-1	A	+1
Employ		good	
├─Formal		MSc	
└─For.lang	**unacc**	pas	*exc*
├─Exper	**unacc**	to1year	
└─Age	**unacc**	21-25	
├─Comm	acc	good	
└─Leader	acc	*more*	]
└─Test	acc	B	

drops by one. For instance, if *For.lang* were not "pas" but one step less (i.e., "no"), the candidate would have been evaluated as "unacc". In a similar way, the column " + 1" displays all possible improvements caused by one-step changes; it indicates that the candidate's evaluation may improve to "exc" if he improves his foreign language skills. Such displays require some practice to get used to, but effectively replace multiple "what-if" interactions.

5 Applications

The author of this chapter maintains a collection of DEX models that are available to him; they were developed mostly in the framework of various research and application projects, educational courses, or donated by other authors. In [22], he presented a study that included 582 models developed in 140 decision-making projects conducted in the period 1979–2015. Among these, 52 projects (38%) were documented in conference or journal publications, and further 20 (14%) projects were documented in internal reports. The collection is highly representative with respect to the addressed decision problems, decision makers involved, covered time period, and observed model characteristics.

The studied models addressed various decision problems from the following areas [22]:

- *Computer technology*: software, hardware, IT tools, programming languages, data base management systems, decision support systems;
- *Projects*: investments, research and R&D projects, tenders;
- *Organisations*: public enterprises, banks, business partners;
- *Schools*: quality of schools, programmes and teachers, school admission, choosing sports for schoolchildren;
- *Management*: production, portfolio management, trade, personnel (employees, jobs, teams), privatization, motorway;
- *Production*: location of facilities, technology, logistics, suppliers, office operations, construction, electric energy production, sustainability;
- *Ecology and Environment*: dumpsite/deposit assessment and remediation, emissions, ecological impacts, soil quality, ecosystem, sustainable development, protected areas;

- *Medicine and Health Care*: risk assessment (breast cancer, diabetes, ski injuries), nursing, technical analysis, knowledge management, healthcare network, therapy management for the Parkinson's disease and congestive heart failure;
- *Agriculture and Food Production*: economic and ecological effects of using genetically modified crops (GMOs), identification of (un)approved GMOs, coexistence of GMOs, crop protection, hop hybrids, garden quality;
- *Tourism*: nature trail, tourism farm facilities, mountain huts;
- *Services*: loans, housing loans, public portals, public services, leasing;
- *Other*: cars, hotels, electric motors, radars, game devices, awards, options, drug addiction, roof covering, coin design, data mining.

The study [22] also revealed some statistical properties of DEX models. An average model consists of roughly 28 attributes (16 of which are basic), 3.5 levels, and 2.5 descendants per aggregate attribute. The largest models may contain up to 400 attributes and 10 levels. An average scale contains 3.4 values and is preferentially ordered. An average decision table has 2.5 arguments, 3.7 output values, and 40 decision rules (with the median of 16). The overall completeness of decision tables is high (93%).

DEX applications generally belong to one of the following categories: (1) one-time decisions, (2) recurring decisions, and (3) decision support systems. These are reviewed next together with representative examples from the literature.

5.1 One-Time Decisions

Making *one-time decisions* is a classic MCDM task in which, given a set of decision alternatives, the goal is to choose the best alternative or to rank/sort them according to decision maker's preferences. Here, the main emphasis is on the quality of decision, i.e., trying to make the best possible decision in a given context. Consequently, the models tend to be very specific, they are often developed from the scratch or partly adapted from other sources, and they are quickly abandoned after the decision has been made.

First applications of DEX were mostly one-time and addressed decision problems related to computer technology, for instance choosing a data base management system [76] and purchasing a mainframe computer for a large factory [5]. The focus gradually shifted to other problem domains, such as employee selection [77]. Bohanec and Rajkovič [6] already report about 30 applications, including the selection of educational and production control software, microcomputers, as well as evaluation of trading partners, projects, and expert teams. Bohanec and Rajkovič [12] report on industrial applications, such as site suitability evaluation, product portfolio evaluation, and remediation of dumpsites. Similar problem types were approached ever since, for instance for evaluating public administration e-portals [57], project self-evaluation [99], mountain huts [88], and mountain lakes [79].

5.2 Recurring Decisions

Recurring decisions are essentially one-time decisions that occur periodically in similar circumstances, for instance, in approving loan applications or prescribing medical therapies. In this category, the emphasis shifts from the quality of individual decisions to the quality, generality, and usability of the model itself. The model has to "survive" multiple tries and be general enough to cope with changes from one case to another. The number of alternatives is initially unknown; sometimes, it may increase to hundreds or even thousands over time. This puts additional constraints on model design, which often proceeds by seeking the balance between including as many general attributes as possible (to facilitate considering cases that might emerge in the future) and reducing their number to only the most representative and easy to assess ones (to ease the burden of collecting input data for each considered alternative). Also, attributes and the whole decision-support procedure have to be clearly defined and meticulously documented, to prepare for multiple applications that may occur in long periods of time.

Since 1990s, with further development of supporting software, recurring decision problems became more and more accessible. Examples include supporting admission procedures in public schools [69], performance evaluation of enterprises [7], and evaluation of research and development projects [10]. Bohanec et al. [13] reported about recurring applications in health care in the assessment risks associated with breast cancer and diabetic foot. Probably the most important applications in the 1990s were *Talent*, a system for advising children in choosing sports [14], and a series of housing loan-allocation applications in collaboration with the Slovenian Housing Fund [11]. Both paved the way for decision support systems in the next period. More recent applications in recurring problems addressed, for instance, the evaluation of researchers [89], data mining workflows [100], detection of financial market manipulations [1], and water management investment projects [28].

5.3 Decision Support Systems

Many recurring decision problems look for the implementation of decision process in the form of a *decision support system* (DSS). DSSs are defined as interactive computer-based systems intended to help decision makers use communications technologies, data, documents, knowledge, and/or models to identify and solve problems, complete decision process tasks, and make decisions [72]. DEX models, developed for solving recurring problems, can be embedded in such DSSs in order to assess and analyze the given decision situations. DEX models usually provide just a fraction of the actual DSS functionality, which often adds a problem-specific user interface and includes additional support for user management, data acquisition, representation, search, and visualization, as well as other statistical, decision-analytic, and/or simulation methods.

Since 2005, many DSSs using DEX models were developed, most notably:

- *SMAC Advisor*: an advisory system on maize co-existence [16],
- *ESQI*: assessment of the impact of cropping systems on soil quality [17],
- a motorway traffic management system [70],
- *RIM*: assessment of bank reputational risk [20],
- *OVJE*: a DSS for the assessment of electric energy production technologies in Slovenia [23];
- *SIGMO*: assessment of GM presence in a food or feed products [24];
- *HeartMan*: a personal DSS for congestive heart management [25];
- *PD_manager*: a platform for Parkinson's disease management [91] with a DSS for the management of medication change [26, 63],
- *Soil Navigator*: assessment and management of soil functions [37],
- *IPSIM Chayote*: prediction and management of damage caused by fruit flies on the chayote in Reunion Island [38].

5.4 *Other Recent Applications*

Since 2005, DEX has been gaining more and more international reputation. It has been particularly well received in agronomy, agriculture, and related fields. Following a successful attempt of assessing economic and ecological impact of genetically modified crops [18, 98], a number of applications addressed the assessment of various cropping systems and their characteristics [4, 29, 33, 35, 36, 47, 54, 66, 71, 78, 80], production and marketing systems [32, 48, 55, 75, 83, 84], genetically modified crops [81, 92], farm management [67] and agri-food chains [61, 62].

Other recently conducted international applications of DEX addressed hydropower plant investments [87], assessment of offshore installation risks [41], employee redeployment [46], and development of ethno villages [74]. Ohunakin and Saracoglu [68] conducted a comparative study of methods MCDM, AHP, CDPC, DEX, ELECTRE III, and IV on the use case of very large concentrated solar power plants.

6 Two Real-World Examples

Among the above applications, we chose two for a more detailed showcase of the DEX approach and capabilities.

6.1 Example 1: Clay Pit Location

The first example came from the industry and was chosen because it represents a typical MCDM setting: a one-time decision problem aimed at choosing the best alternative from a given set. The problem was difficult and might have had critical consequences on the company and its long-term survival. Furthermore, initial alternatives were all unacceptable and better options had to be sought for in the process. The project was carried out in the 1990s; it is fully documented in the internal report [9] and partly in [12].

The company is called *Goriške opekarne* and is located near the Slovenian city of Nova Gorica. The company produces bricks and tiles. In 1993, they were faced with a difficult situation: the clay pit that had been providing raw material for their production became exhausted. The company had to find a replacement location, but this was difficult for a number of technological, logistic, financial, and environmental reasons, including a possible rejection of proposed solutions by local inhabitants. A group consisting of company managers, experts, and decision analysts was formed to define a DEX model and propose alternatives, while communicating with employees and inhabitants in a series of socio-psychological studies.

Eventually, a DEX model, whose complete structure is shown in Fig. 14, has been developed. A detailed description of individual attributes is beyond the scope of this chapter; however, one should note that the whole model is split in two main subtrees that address environmental and feasibility aspects of clay-pit locations, respectively. The model contains 30 basic and 19 aggregate attributes. Also, let us add that all scales in the model are preferentially ordered and the majority of them are either two-valued {"less-suit", "suit"} or three-valued {"unsuit", "less-suit", "suit"}. Scales of *ENVIRONMENT* and *ATTRACT* have four values, and the root attribute *SITE* has the scale {"unacc", "marg-acc", "less-acc", "acc", "good"}.

Decision rules from this model are illustrated here with just two examples shown in Fig. 15. The first example presents complex rules associated with attribute *TECH*, which aggregates three basic attributes: *TRANSPORT*, *CONSTRUCT,* and *LAND_ARCH*. TECH is located at the bottom of the tree; such attributes are often associated with specific decision rules and tables, which aim to resolve the decision problem at that level and provide useful evaluations/interpretations for higher levels of the model. The second example in Fig. 15 is located at the very root of the model and aggregates *ENVIRONMENT* and *FEASIBILITY* to the overall location evaluation (*SITE*). This is a typical representative of high-level aggregation functions, which tend to be symmetric or near-symmetric, and rule out all the cases that are evaluated poorly (i.e., "unacc") on lower levels of the hierarchy.

Three clay-pit locations were considered by this model: Okroglica, Marjetnica, and Bukovnik. Initially, all of them were assessed as "unacc". The team carried out a series of "what-if" scenarios, exploring possible improvements of the locations' characteristics, and anticipating an "optimistic" or "pessimistic" development of the investment project. Ultimately, eight variations were considered, which were

Attribute	Description
SITE	Site suitability
├─**ENVIRONMENT**	Environmental components
│ ├─**ATTRACT**	Site attractiveness
│ │ ├─DEVELOP	Development factor
│ │ ├─CHARACT	Site characteristics
│ │ └─**LAND**	Land
│ │ ├─ACCESS	Land accessibility
│ │ ├─QUALITY	Land quality (infrastructure)
│ │ └─TIME_AVAIL	Time availability (short/long term)
│ └─**VULNER**	Site vulnerability
│ ├─**POLLUTION**	Polution: Environmental impact
│ │ ├─**LIV_ENV**	Pollution impacts to living environment
│ │ │ ├─**HUMAN**	Pollution impacts to humans
│ │ │ │ ├─HEALTH	Health impacts
│ │ │ │ └─OTHER	Other impacts (e.g., noise)
│ │ │ ├─FAUNA	Pollution impacts on fauna
│ │ │ └─FLORA	Pollution impacts on flora
│ │ ├─SOC_ENV	Pollution impacts on social environment
│ │ └─**NON-LIV_ENV**	Pollution impacts on non-living environment
│ │ ├─SOIL	Impacts on soil
│ │ ├─WATER	Impacts on water
│ │ └─AIR	Impacts on air
│ └─**SITE_ORG**	Site organization
│ ├─**VALUATION**	Site valuation
│ │ ├─**ECOLOG**	Ecological valuation
│ │ │ ├─UNIQUE	Site uniqueness
│ │ │ └─DIVERS	Site diversity
│ │ └─PERCEP	Perceptional valuation
│ └─**USE**	Land use
│ ├─DEMOGR	Demography
│ ├─INFRAST	Infrastructure
│ └─**POTENTIAL**	Site potentials
│ ├─**PRIM_USE**	Primary use
│ │ ├─AGRICULT	Agriculture
│ │ ├─FOREST	Forestry
│ │ └─WATER	Water management
│ └─**OTH_POT**	Other potentials
│ ├─NAT_HER	Natural heritage
│ ├─CULT_HER	Cultural heritage
│ └─REC_TOUR	Recreation and tourism
└─**FEASIBILITY**	Feasibility of the project
├─SOC-PSYCH	Socio-psychological feasibility
├─**TECH**	Technical feasibility
│ ├─TRANSPORT	Transportation
│ ├─CONSTRUCT	Construction
│ └─LAND_ARCH	Landscape architecture design
└─**ECON**	Economic feasibility
├─DIRECT	Direct expenses
└─INDIRECT	Indirect expenses

Fig. 14 Structure of the *Clay Pit* DEX model

	TRANSPORT	CONSTRUCT	LAND_ARCH	TECH
1	*	*	unsuit	unsuit
2	less-suit	*	>=less-suit	less-suit
3	*	less-suit	>=less-suit	less-suit
4	*	*	less-suit	less-suit
5	*suit*	*suit*	*suit*	*suit*

	ENVIRONMENT	FEASIBILITY	SITE
1	unacc	*	unacc
2	*	unacc	unacc
3	less-acc	less-acc	marg-acc
4	less-acc	*acc*	less-acc
5	>=acc	less-acc	less-acc
6	acc	*acc*	acc
7	*good*	*acc*	*good*

Fig. 15 Two decision tables represented by complex rules: *TECH* and *SITE*
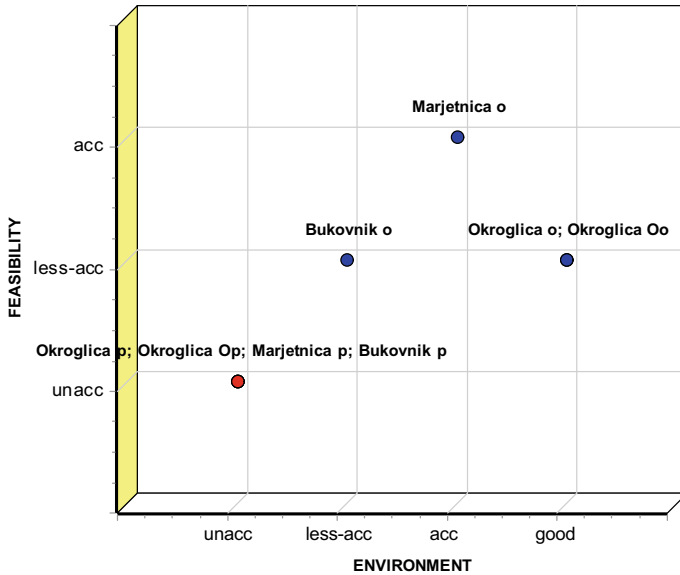
Fig. 16 Evaluation of Clay Pit locations along FEASIBILITY and ENVIRONMENT

evaluated as shown in the scatterplot in Fig. 16. Among these, "Marjetnica o" was considered the best and proposed for implementation.

6.2   *Example 2: Electric Energy Production Technologies*

The second example is taken from a more recent project aimed at the identification of reliable, rational, and environmentally sound production of electric energy in Slovenia by 2050 [23, 56]. Technology alternatives included both conventional and renewable energy sources: coal, gas, biomass, oil, nuclear, hydro, wind, and photovoltaic. This use case belongs to the category of complex and (potentially) recurring strategic decision problems, which occur and are relevant for any country. Without the ambition to go into any substantial detail, we wish to illustrate the capabilities of DEX to address really difficult real-world decision problems and handle models

consisting of several tens of attributes, which are eventually incorporated in a DSS (called OVJE in this case, see above).

The methodological approach consisted of three stages, in which two DEX and one simulation model were developed:

- *DEX Model T* for the evaluation of eight electric energy production technologies.
- *DEX Model M* for the evaluation of mixtures of technologies, considering the shares of individual technologies in the total installed capacity.
- *Simulation Model S* for the evaluation of possible implementations of technology mixtures in the period 2014–2050, taking into account various scenarios of shutting down the existing power plants and constructing new ones.

Here, we shall briefly sketch only the first one; for more information, the interested reader is referred to [23, 56]. Figure 17 shows the hierarchical structure of Model T. There are 35 input and 28 aggregate attributes. There are two attributes that influence more than one parent (*Licences* and *Contribution to development*); therefore, this is a true hierarchy rather than a tree. The model consists of three main subtrees:

- *Rationality*: assesses how much a particular technology contributes to the overall societal development, the economy, and the prudent use of land with low pollution.
- *Feasibility*: addresses the *Technical*, *Economic,* and *Spatial* feasibility aspects of the technology.
- *Uncertainties*: addresses common uncertainty themes associated with energy policy and comprises *Technological dependence*, *Possible changes in society and in the world*, and *Perception of risks* with respect to technical advancement of a technology and trust into safety management system.

Among the 28 decision tables that were formulated by an expert team, we show here only two in the form of complex rules. Both tables are complete, consistent, and monotone. The first one (Fig. 18) aggregates the assessments of *Rationality*, *Feasibility,* and *Uncertainties* into the root assessment of the suitability of *Technology*. This table is *evaluative* because it evaluates some criterion (in this case *Technology*) according to evaluations of the incoming criteria: the better the value of each incoming criterion, the better the overall evaluation. Evaluative aggregation functions are typical for most MCDM methods.

The second table (Fig. 19) combines possible societal and world changes into a common perception of *Possible changes*. Here, the values "neg", "no", and "pos" refer to the direction of changes. Despite that one can assign preferences to these categories, they are not really evaluative. The table actually specifies a multi-variate *logic* for combining some basic concepts into higher-level concepts. This shows that in DEX, using multi-valued qualitative variables, it is possible to express both evaluative and logical rules. The latter usually occur at lower model levels and define concepts that enter the evaluation process at higher levels of the hierarchy. Inference based on logic is rarely featured in MCDM methods.

Using Model T, the study [23] concluded that there were only three technologies of sufficient suitability for Slovenia: Hydro, Gas, and Nuclear. Among these, Hydro is
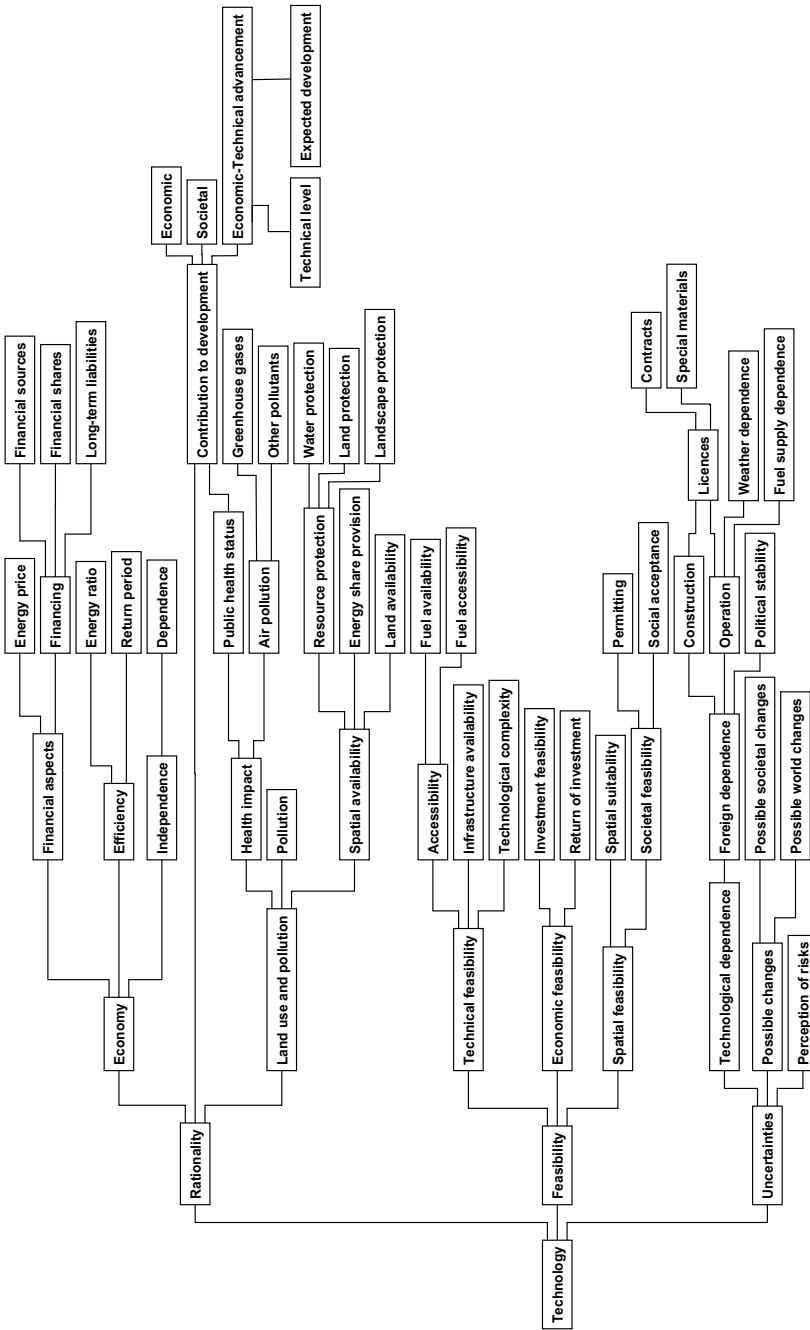
Fig. 17 Hierarchical structure of Model T

Fig. 18 Decision rules for
the assessment of *Technology*

	Rationality	Feasibility	Uncertainties	Technology
1	**inapprop**	*	*	**unsuit**
2	<=low	<=med	**v_high**	**unsuit**
3	<=*med*	**low**	**v_high**	**unsuit**
4	>=low	**low**	high:med	weak
5	>=low	*high*	**v_high**	weak
6	>=*med*	>=med	**v_high**	weak
7	*high*	**low**	<=med	weak
8	*high*	*	**v_high**	weak
9	low:*med*	**low**	>=low	suit
10	>=low	**low**	low	suit
11	>=low	>=med	high	suit
12	low	>=med	>=med	*good*
13	low:*med*	med	med:low	*good*
14	>=low	>=med	med	*good*
15	*high*	**low**	*none*	*good*
16	>=*med*	>=med	*none*	*exc*
17	>=*med*	*high*	>=low	*exc*
18	*high*	>=med	>=low	*exc*

Fig. 19 Decision rules for
determining the direction of
Possible changes

	Possible societal changes	Possible world changes	Possible changes
1	**neg**	*	**neg**
2	*	**neg**	**neg**
3	no	no	no
4	>=no	*pos*	*pos*
5	*pos*	>=no	*pos*

the best. Gas and Nuclear are similar, with Nuclear worse in terms of *Feasibility* and *Perception of risks*, but better in terms of *Economic feasibility* and *Possible changes*. Coal and Oil are unsuitable particularly because of inappropriate *Rationality* due to *Land use and pollution*. All the remaining "green" technologies are unsuitable for a number of reasons, including *Economy*, *Land use*, *Economic feasibility,* and *Technological dependence*.

7 DEX Extensions

A number of extensions to DEX have been proposed over the years, mostly motivated by the needs of complex real-world decision problems. The proposals were mainly coming from two directions:

1. *Bridging the gap between qualitative aspects of DEX and quantitative aspects of the "traditional" MCDM*. This includes introducing numeric variables and weights in DEX models and facilitating numeric evaluation to better support ranking tasks.
2. *Taking advantage of artificial intelligence approaches.* This includes extended uncertainty handling mechanisms and using machine learning algorithms to

develop DEX models (semi)automatically from examples of past decisions, whenever such data is available.

7.1 Numeric Attributes

In its basic form, DEX is strictly qualitative. Currently, for instance, this requires that all numeric input data is pre-processed and discretized externally; introducing numerical variables to DEX models would definitely alleviate such problems and advance the generality of the approach, making it suitable for a larger class of problems. In principle, adding numerical attributes per se to the static formal model is easy, one should only extend the types of scales D. However, this is not enough because any such change should also preserve the dynamic aspects of the method: supporting the creation and modification of aggregation functions, considering completeness, consistency, and monotonicity of aggregation functions, and performing in the case of missing or uncertain data or knowledge. This is much harder and explains why the progress is slow and hesitant. Trdin and Bohanec [90] proposed a number of methodological extensions of this type, which will guide future evolution of DEX.

7.2 Weights

Traditional MCDM methods heavily rely on weights to define the importance of attributes [45]. The formal DEX model does not define any weights to be associated with qualitative attributes and decision rules. However, to bridge the gap between MCDM and also for practical reasons, DEX actually was extended with the notion of weights. The principle is simple:

- given a decision table that defines the function $y = f(x_1, x_2, \ldots, x_k)$ and consists of entries $(\mathbf{x}_e, y_e)$, $e = 1, 2, \ldots, r$,
- interpret the entries as points in a multi-dimensional space, and
- construct g as an approximation of f in the form

$$g(x_1, x_2, \ldots, x_k) = w_0 + w_1 \text{ord}(x_1) + \cdots + w_k \text{ord}(x_k).$$

Here, $\text{ord}(x)$ denotes the ordinal number of value x, and $w_i \in \mathcal{R}$ are relative weights of the corresponding arguments for $i = 1, \ldots, k$. These coefficients are determined using the least squares measure.

This method is actually implemented in DEXi and is used for approximate bidirectional transformations between weights and decision tables: (1) estimating weights from defined rules using the above approximation and (2) determining the values of yet undefined decision rules on the basis of already defined rules and user-specified weights. For more information, the reader is referred to [15, 27] and supplementary material in [38].

7.3   Combining Qualitative and Quantitative Evaluation

As already indicated, the qualitative foundation of DEX makes it particularly suitable for sorting and classification problems. In practice, however, it is sometimes necessary to use an already developed model also for ranking. For instance, whenever there are several alternatives assigned to the same evaluation category, it is often still necessary to tell them apart in some way. In qualitative DEX, this is in principle possible by refining the model by adding new categories and/or modifying decision rules to improve the separation; however, this requires redefining at least some parts of the model. Or alternatively, one can proceed by comparing similar alternatives, using analytic techniques to understand their advantages and disadvantages, and ranking them on this basis. In any case, both approaches are time consuming, and a better out-of-the-box support for ranking might alleviate such issues.

In principle, it is not difficult to think of some kind of numerical evaluation based on a DEX model. For instance, why not just taking the weights from the previous section and use the function g to carry out the calculations? Unfortunately, this does not work well because f and g might give different rankings based on the same inputs. The real challenge is how to assure that both evaluation procedures are *consistent* with each other. We are actually looking for a method that would first assign alternatives to distinct classes and only then rank them within each class. If possible, the process should not involve any additional work and should rely only on information already available in the model.

So far, there were two attempts at this kind of approach [8, 60]. They both explored the idea of representing values of some ordered attribute $x \in X$ in the form $v + \omega$, where $v \in D_x$ is a qualitative value of x, and $\omega \in [-0.5, +0.5]$ is a numerical *offset* to that value. The offset -0.5 is interpreted as "particularly bad" in the context of v, and $+0.5$ is interpreted as "particularly good". For instance, a job candidate evaluated as $Employ =$ "good" $+ 0.33$ would have been considered better than another candidate with $Employ =$ "good"–0.12. In the evaluation algorithm, the qualitative evaluation of v remains exactly the same as before, and ω is assessed from the corresponding decision table using the principle of dominance and some additional assumptions. The approach of [8] uses a locally linear approximation of rules that map to some output category, whereas [60] uses copulas for the same purpose. The first approach is now called QQ (Qualitative-Quantitative). Unfortunately, these methods are not implemented in any currently available public software. We also think that the problem has not been solved in an entirely satisfactory way and remains a challenge for the future.

7.4 Handling Uncertainty Using Value Distributions

The idea of using fuzzy and probabilistic value distributions to cope with uncertain data and evaluations in DEX is actually quite old and originates from expert systems; it was first proposed in [5]. The idea is to allow using value distributions instead of single qualitative values in all places denoted E_x and E_y in the formal model. For instance, instead of assigning a single value to some input attribute, say *For.lang* = "pas", one can express their uncertainty about the real input using the probability distribution:

$$For.lang = \begin{pmatrix} ''no'' & ''pas'' & ''act'' \\ 0.1 & 0.7 & 0.2 \end{pmatrix}.$$

The same representation type can also be used for the outgoing values of decision rules.

This extension puts additional requirements on the evaluation procedure: the uncertainties, represented by probabilities or fuzzy possibilities, have to be propagated from input to output attributes in the hierarchy. Probabilistic inference employs product/sum operators, and fuzzy inference employs min/max or more general t-norm/t-conorm operators. For a more formal treatment of the subject, please see [90].

This evaluation procedure was actually implemented in the previous generation of DEX software and is still supported by software libraries *JDEXi, DEXi.NET,* and *DEXx.* It has been left out from DEXi for simplicity, but is destined to return in future software implementations.

7.5 Machine Learning of DEX Models

A large number of DEX application indicated that it is feasible for an individual decision maker or a group to develop a DEX model manually even for very difficult decision problems. On the other hand, it is also true that the task is demanding, particularly because the definition of decision rules generally requires more effort than definition of comparable aggregation functions in other MCDM methods. A natural question arising from DEX's artificial intelligence foundations is: could DEX models be constructed from data following the principles of machine learning? The answer is "yes, but it is hard"; none of the approaches attempted so far resulted in an entirely satisfactory solution for practice and no current general-purpose software implements any of the related methods.

The first and most ambitious attempt so far was made by Zupan et al. [95]. They proposed a method called HINT (Hierarchical Induction Tool) that is capable of transforming a large flat decision table into a hierarchical model, creating aggregate attribute and corresponding smaller decision tables along the way. This puts HINT in the category of concept learning methods [86]. Theoretically, the method did solve the task, but it also turned out very sensitive to noisy data (which is almost inevitable

in practice) and required a very good coverage of the decision space by input data (which is also difficult to assure in practice).

The second attempt by Žnidaršič et al. [96, 97] was somewhat more modest and explored the approach of *model revision*: given an already developed DEX model and some data, the task is to revise model's decision rules so as to better match the data. Eventually, the method worked satisfactory, but its implementation *proDEX* [96] has become obsolete and is currently unsupported.

In the third attempt, [21] took an intermediate approach: given the *structure* of attributes and data, construct all aggregation tables in the model, taking into account probability distributions of input attributes and enforcing the principle of dominance. The authors demonstrated the approach by developing a model for predicting injury risk in ski resorts. The approach seems promising and will be further investigated in the future.

8   Summary

DEX is a qualitative decision modeling method that combines hierarchical and rule-based MCDM with artificial intelligence, specifically expert modeling and machine learning. The basic concepts of DEX are very simple and only involve hierarchically structured attributes, discrete scales, and decision tables consisting of elementary decision rules.

Despite simplicity, DEX has been successfully used in hundreds of real-world applications. According to its qualitative design, it is best suited for supporting sorting and classification decision problems. Choosing and ranking problems can be addressed, too, but they generally require some additional effort (interactive exploration and analysis of alternatives) or methodological extensions (such as QQ). Although DEX is suitable for one-time decision problems, recent trends indicate a shift toward recurring decision problems and including DEX models in DSSs. This is probably related with the effort that is required to develop a DEX model, which is generally greater than with comparable MCDM methods. One-time decision problems rarely justify the effort, whereas recurring and DSS ones do.

Practical applicability of DEX depends on the availability of supporting software. This is particularly true for the acquisition of decision tables, which might be very difficult on paper but becomes feasible when supported by appropriate tools and user interfaces. In addition to merely representing a static formal DEX model, DEX software always attempted to actively support dynamic aspects of creating and using the model. For DEX, it is really important to:

- facilitate *editing* of the model and its components: attributes, their structure, scales, aggregation functions, and alternatives;
- support the *acquisition* of decision rules, which includes enforcing the principle of dominance and checking for consistency and completeness at all times;

- maintain the *transparency* of the model and provide comprehensible representations of its components, such as complex rules and 3D graphics;
- provide various methods for the *analysis* of alternatives and *explanation* of evaluations.

DEX models may suffer from the combinatorial explosion: the size of decision tables increases exponentially with the number of incoming attributes. When developing a DEX model, it is thus important to follow recommendations that aim to keep the size below about 100: make "narrow" hierarchies with only 2 or 3 descendants of an aggregate attribute, and use the least number of values per attribute that still distinguishes between qualitatively different states of that attribute. Another potential disadvantage is that DEX, in its original form, is alien to numbers. When alternatives are prevalently described by numeric properties, the options are either to discretize them externally or use another MCDM method.

In the future, the main evolution will go in the direction of *Extended DEX*, as proposed by [90]. The proposal includes introduction of numeric attributes in DEX models and explicitly addressing uncertainty using probabilistic and fuzzy distributions of values. Software that partly supports these extensions already exists (DEXx software library), and full support is under development. The plan is to gradually replace the existing software DEXi with a new generation of web-based [52] and desktop applications. There also two challenges still open for further research and eventual software implementation: combined qualitative-quantitative evaluation of alternatives and learning DEX models from data.

References

1. Alić, I., Siering, M., Bohanec, M.: Hot stock or not? A qualitative multi-attribute model to detect financial market manipulation. In: Wigland, D.L. (Ed.) eInnovation: Challenges and Impacts for Individuals, Organizations and Society, Proceedings of 26th Bled eConference Bled, Slovenia, Kranj: Moderna organizacija, 64–77 (2013)
2. Bana e Costa, C., De Corte, J.-M., Vansnick, J.-C.: MACBETH (Overview of MACBETH multicriteria decision analysis approach). Int. J. Inf. Technol. Decis. Mak. **11**, 359–387 (2003)
3. Belton, V., Stewart, T.J.: Multi Criteria Decision Analysis: An Integrated Approach. Springer, US (2002)
4. Bergez, J.-E.: Using a genetic algorithm to define worst-best and best-worst options of a DEXi-type model: Application to the MASC model of cropping-system sustainability. Comput. Electron. Agric. **90**, 93–98 (2013)
5. Bohanec, M., Bratko, I., Rajkovič, V. (1983): An expert system for decision making. In: Sol, H.G. (Ed.) Processes and Tools for Decision Support, pp. 235–248, North-Holland.
6. Bohanec, M., Rajkovič, V. (1988): Knowledge acquisition and explanation for multi-attribute decision making. In: Proceedings of the 8th International Workshop Expert Systems and Their Applications AVIGNON 88, vol. 1, pp. 59–78, Avignon, 1988 (also available from http://kt.ijs.si/MarkoBohanec/pub/Avignon88.pdf)

7. Bohanec, M., Rajkovič, V.: DEX: an expert system shell for decision support. Sistemica **1**(1), 145–157 (1990)
8. Bohanec, M., Urh, B., Rajkovič, V.: Evaluating options by combined qualitative and quantitative methods. Acta Physiol. (Oxf) **80**, 67–89 (1992)
9. Bohanec, M., Kontić, B., Kos, D., Marušič, J., Polič, S., Rakovec, J., Sedej, B., et al. (1993): Comparison of clay-pit locations Okroglica, Bukovnik, Marjetnica with respect to environmental protection (in Slovene language). Ljubljana: Jožef Stefan Institute, Report DP-6742.
10. Bohanec, M., Rajkovič, V., Semolič, B., Pogačnik, A.: Knowledge-based portfolio analysis for project evaluation. Inf. Manag. **28**, 293–302 (1995)
11. Bohanec, M., Cestnik, B., Rajkovič, V.: Evaluation models for housing loan allocation in the context of floats. In: Berkeley, D., Widmeyer, G.R., Brezillon, P., Rajkovič, V. (eds.) Context Sensitive Decision Support Systems, pp. 174–189. Chapman & Hall, London (1998)
12. Bohanec, M., Rajkovič, V.: Multi-attribute decision modeling: industrial applications of DEX. Informatica **23**, 487–491 (1999)
13. Bohanec, M., Zupan, B., Rajkovič, V.: Applications of qualitative multi-attribute decision models in health care. Int. J. Med. Inf. **58–59**, 191–205 (2000)
14. Bohanec, M., Rajkovič, V., Leskošek, B., Kapus, V.: Expert knowledge management for sports talent identification and advising process. In: Carlsson, S.A., Brezillon, P., Humphreys, P., Lundberg, B.G., McCosh, A.M., Rajkovič, V. (eds.) Decision Support through Knowledge Management, pp. 46–59 (2000)
15. Bohanec, M., Zupan, B.: (2004): A function-decomposition method for development of hierarchical multi-attribute decision models. Decis. Support Syst. **36**, 215–233 (2004)
16. Bohanec, M., Messéan, A., Angevin, F., Žnidaršič, M.: SMAC advisor: a decision-support tool on coexistence of genetically-modified and conventional maize. In: Proceedings of Information Society IS 2006, Ljubljana, 9–12 (2006)
17. Bohanec, M., Cortet, J., Griffiths, B., Žnidaršič, M., Debeljak, M., Caul, S., Thompson, J., Krogh, P.H.: A qualitative multi-attribute model for assessing the impact of cropping systems on soil quality. Pedobiologia **51**, 239–250 (2007)
18. Bohanec, M., Messéan, A., Scatasta, S., Angevin, F., Griffiths, B., Krogh, P.H.: A qualitative multi-attribute model for economic and ecological assessment of genetically modified crops. Ecol. Model. **215**, 247–261 (2008)
19. Bohanec, M., Rajkovič, V., Bratko, I., Zupan, B., Žnidaršič, M.: DEX methodology: three decades of qualitative multi-attribute modelling. Informatica **37**, 49–54 (2013)
20. Bohanec, M., Aprile, G., Costante, M., Foti, M., Trdin, N.: A hierarchical multi-attribute model for bank reputational risk assessment. In: Phillips-Wren, G., Carlsson, S., Respício, A., Brézillon, P. (eds.) DSS 2.0 -- Supporting Decision Making with New Technologies. Amsterdam: IOS Press. ISBN 978–1–61499–398–8, 92–103 (2014).
21. Bohanec, M., Delibašić, B.: Data-mining and expert models for predicting injury risk in ski resorts. In: Decision Support Systems V - Big Data Analytics for Decision Making. First International Conference ICDSST 2015, pp. 46–60. Springer (2015)
22. Bohanec, M.: Multi-criteria DEX models (2017): an overview and analysis. In: Zadnik Stirn, L., et al. (eds.) SOR-2017: 14th International Symposium on Operational Research in Slovenia, Bled, Slovenia, September 27–29, 2017, pp. 155–160. Slovenian Society Informatika, Section for Operational Research, Ljubljana
23. Bohanec, M., Trdin, N., Kontić, B.: A qualitative multi-criteria modelling approach to the assessment of electric energy production technologies in Slovenia. CEJOR **25**, 611–625 (2017)
24. Bohanec, M., Mileva Boshkoska, B., Prins, T.W., Kok, E.J.: SIGMO: a decision support system for identification of genetically modified food or feed products. Food Control **71**, 168–177 (2017)
25. Bohanec, M., Tartarisco, G., Marino, F., Pioggia, G., Puddu, P.E., Schiariti, M.S., Baert, A., Pardaens, S., Clays, E., Vodopija, A., Luštrek, M.: HeartMan DSS: A decision support system for self-management of congestive heart failure. Expert Syst. Appl. **186**, 115688 (2021). https://doi.org/10.1016/j.eswa.2021.115688

26. Bohanec, M., Miljković, D., Valmarska, A., Mileva Boshkoska, B., Gasparoli, E., Gentile, G., Koutsikos, K., Marcante, A., Antonini, A., Gatsios, D., Rigas, F., Fotiadis, D.I., Tsiouris, K.M., Konitsiotis, S.: A decision support system for Parkinson disease management: Expert models for suggesting medication change. J. Decis. Syst. **27**, 164–172 (2018)

27. Bohanec, M.: DEXi: program for multi-attribute decision making, user's manual, version 5.04. IJS Report DP-13100, Jožef Stefan Institute, Ljubljana (available from http://kt.ijs.si/MarkoBohanec/pub/DEXiManual504.pdf) (2020)

28. Brelih, M., Rajkovič, U., Ružič, T., Rodič, B., Kozelj, D.: Modelling decision knowledge for the evaluation of water management investment projects. CEJOR **27**, 759–781 (2019)

29. Carpani, M., Bergez, J.-E., Monod, H.: Sensitivity analysis of a hierarchical qualitative model for sustainability assessment of cropping systems. Environ. Model. Softw. **27–28**, 15–22 (2012)

30. Chandra, V.: Artifical intelligence and machine learning. PHI Learning (2014)

31. Chen, S., Liu, J., Wang, H., Xu, Y., Augusto, J.C.: A linguistic multi-criteria decision making approach based on logical reasoning. Inf. Sci. **258**, 266–276 (2014)

32. Chopin, P., Tiroilien, J., Blazy, J.-M.: Ex-ante sustainability assessment of cleaner banana production systems. J. Clean. Prod. **139**, 15–24 (2014)

33. Colomb, B., Carof, M., Aveline, A., Bergez, J.-E.: Stockless organic farming: strengths and weaknesses evidenced by a multicriteria sustainability assessment model. Agron. Sustain. Dev. **33**, 593–608 (2012)

34. Corrente, S., Greco, S., Słowiński, R.: Multiple criteria hierarchy process in robust ordinal regression. Decis. Support Syst. **53**(3), 660–674 (2012)

35. Craheix, D., Bergez, J.-E., Angevin, F., Bockstaller, C., Bohanec, M., Colomb, B., Doré, T., Fortino, G., Guichard, L., Pelzer, E., Méssean, A., Reau, R., Sadok, W.: Guidelines to design models assessing agricultural sustainability, based upon feedbacks from the DEXi decision support system. Agron. Sustain. Dev. **35**, 1431–1447 (2015)

36. Craheix, D., Angevin, F., Doré, T., de Tourdonnet, S.: Using a multicriteria assessment model to evaluate the sustainability of conservation agriculture at the cropping system level in France. Eur. J. Agron. **767**, 75–86 (2016)

37. Debeljak, M., Trajanov, A., Kuzmanovski, V., Schröder, J., Sandén, T., Spiegel, H., Wall, D.P., Van de Broek, M., Rutgers, M., Bampa, F., Creamer, R.E., Henriksen, C.B.: A field-scale decision support system for assessment and management of soil functions. Front. Environ. Sci. **7**(115), 1–14 (2019)

38. Deguine, J.-P., Robin, M.-H., Corrales, D.C., Vedy-Zecchini, M.-A., Doizy, A., Chiroleu, F., Quesnel, G., Païtard, I., Bohanec, M., Aubertot, J.-N.: Qualitative modeling of fruit fly injuries on chayote in Réunion : development and transfer to users. Crop Prot. **129** (2021). ISSN: 105367-1-105367–11

39. Doumpos, M., Zopounidis, C.: Multicriteria Decision Aid Classification Methods. Springer, US (2002)

40. Efstathiou, J., Rajkovič, V.: Multiattribute decisionmaking using a fuzzy heuristic approach. *IEEE Transactions on Systems, Man, and Cybernetics,* SMC-9, 326–333 (1979)

41. Erdogan, G., Refsdal, A., Nygård, B., Petter Rosland, O., Kvam Randeberg, B.: Risk-based decision support model for offshore installations. Bus. Syst. Res. **9**(2), 55–68 (2018)

42. García-Lapresta, J.L., del Pozo, R.G.: An ordinal multi-criteria decision-making procedure under imprecise linguistic assessments. Eur. J. Oper. Res. **279**(1), 159–167 (2019)

43. Greco, S., Matarazzo, B., Słowiński, R.: Rough sets methodology for sorting problems in presence of multiple attributes and criteria. Eur. J. Oper. Res. **138**(2), 247–259 (2002)

44. Greco, S., Matarazzo, B., Słowiński, R.: Dominance-based rough set approach to decision under uncertainty and time preference. Ann. Oper. Res. **176**(1), 41–75 (2010)

45. Greco, S., Ehrgott, M., Figueira, J.: Multi Criteria Decision Analysis: State of the art Surveys. Springer, New York (2016)

46. Hajnić, M., Mileva Boshkoska, B.: A decision support model for the operational management of employee redeployment in large governmental organisations. J. Decis. Syst. (2020). https://doi.org/10.1080/12460125.2020.1768681

47. Hawes, C., Young, M.W., Banks, G., Begg, G.S., Christie, A., Iannetta, P.P.M., Karley, A.J., Squire, G.R.: Whole-systems analysis of environmental and economic sustainability in arable cropping systems: a case study. Agronomy **9**, 438 (2019)

48. Iocola, I., Campanelli, G., Diacono, M., Leteo, F., Montemurro, F., Persiani, A., Canali, S.: Sustainability assessment of organic vegetable production using a qualitative multi-attribute model. *Sustainability* 10(10), 3820 (2018). https://doi.org/10.3390/su10103820

49. Ishizaka, A., Nemery, P.: Multi-criteria Decision Analysis: Methods and Software. Wiley, Chichester (2013)

50. Jackson, P.: Introduction to Expert Systems, 2nd edn. Addison-Wesley (1990)

51. Kikaj, A., Bohanec, M.: Complex decision rules in DEX methodology: jRule algorithm and performance analysis. In: Proceedings of the 21th International Conference Information Society IS 2018, Volume A, Ljubljana: Jožef Stefan Institute, 17–20 (2018)

52. Kikaj, A., Bohanec, M.: DEX2Web—A web-based software implementing the multiple-criteria decision-making method DEX. In: Jayawickrama, U., Delias, P., Escobar, M.T., Papathanasiou, J. (eds.) Decision Support Systems XI: Decision Support Systems, Analytics and Technologies in Response to Global Crisis Management, pp. 30–43 (2021). https://www.springerprofessional.de/en/dex2web-a-web-based-software-implementing-the-multiple-criteria-/19166702

53. Kotsiantis, S.B., Zaharakis, I.D., Pintelas, P.E.: Machine learning: a review of classification and combining techniques. Artif. Intell. Rev. **26**, 159–190 (2007)

54. Maksimović, A., Grgić, Z., Puška, A., Šakić Bobić, B., Ćejvanović, F.: Application of multi-criteria decision making for the selection of apple varieties for the Northwestern region of B&H. J. Central Eur. Agric. **19**(3), 740–759 (2018)

55. Montemurro, F., Persiani, A., Diacono, M.: Environmental sustainability assessment of horticultural systems: a multi-criteria evaluation approach applied in a case study in mediterranean conditions. Agronomy **8**, 98 (2018). https://doi.org/10.3390/agronomy8070098

56. Kontić, B., Bohanec, M., Kontić, D., Trdin, N., Matko, M.: Improving appraisal of sustainability of energy options - A view from Slovenia. Energy Policy **90**, 154–171 (2016)

57. Leben, A., Kunstelj, M., Bohanec, M., Vintar, M.: Evaluating public administration e-portals. Inf. Polity **11**(3/4), 207–225 (2006)

58. MacSweeney Brugha, C.: Structure of multi-criteria decision-making. J. Oper. Res. Soc. **55**(11), 1156–1168 (2004)

59. Michalski, R.S., Larson, J.: Incremental generation of VLL hypotheses: the underlying methodology and the description of program AQ11. Report ISG 83–5, University of Illinois at Urbana-Champaign (1983)

60. Mileva Boshkoska, B., Bohanec, M.: A method for ranking non-linear qualitative decision preferences using copulas. Int. J. Decis. Support Syst. Technol. **4**(2), 42–58 (2012)

61. Mileva Boshkoska, B., Liu, S., Chen, H.: Towards a knowledge management framework for crossing knowledge boundaries in agricultural value chain. J. Decis. Syst. **27**, 88–97 (2018)

62. Mileva Boshkoska, B., Liu, S., Zhao, G., Fernandez, A., Gamboa, S., del Pino, M., Zarate, P., Hernandez, J., Chen, H.: A decision support system for evaluation of the knowledge sharing crossing boundaries in agri-food value chains. Comput. Ind. **110**, 64–80 (2019)

63. Mileva Boshkoska, B., Miljković, D., Valmarska, A., Gatsios, D., Rigas, G., Konitsiotis, S., Tsiouris, K.M., Fotiadis, D., Bohanec, M.: Decision support for medication change of Parkinson's disease patients. In: Computer Methods and Programs in Biomedicine, vol. 196, November 2020, 105552 (2020)

64. Mitchell, T.M.: Machine Learning. McGraw-Hill (1997)

65. Moshkovich, H.M., Mechitov, A.I.: Verbal decision analysis: foundations and trends. Adv. Decis. Sci. **2013**, 1–9 (2013)

66. Mouron, P., Heijne, B., Naef, A., Strassemeyer, J., Hayer, F., Avilla, J., Alaphilippe, A., Höhn, H., Hernandez, J., Mack, G., Gaillard, G., Solé, J., Sauphanor, B., Patocchi, A., Samietz, J., Bravin, E., Lavigne, C., Bohanec, M., Golla, B., Scheer, C., Aubert, U., Bigler, F.: Sustainability assessment of crop protection systems: sustainos methodology and its application for apple orchards. Agric. Syst. **113**, 1–15 (2012)

67. Nikoloski, T., Udovč, A., Pavlovič, M., Rajkovič, U.: Multi-criteria assessment model for farm reorientation. J. Decis. Syst. **27**, 79–87 (2018)

68. Ohunakin, O.S., Saracoglu, B.O.: A comparative study of selected multi-criteria decision-making methodologies for location selection of very large concentrated solar power plants in Nigeria. Afr. J. Sci. Technol. Innov. Dev. **10**(5), 551–567 (2018)

69. Olave, M., Rajkovič, V., Bohanec, M.: An application for admission in public school systems. In: Snellen, I.Th.M., van de Donk, W.B.H.J., Baquiast, J.-P. (eds.) Expert systems in public administration, Elsevier, pp. 145–160 (1989)

70. Omerčević, D., Zupančič, M., Bohanec, M., Kastelic, T.: Intelligent response to highway traffic situations and road incidents. In: Proceedings of TRA 2008, 21–24 April 2008, Ljubljana, pp. 1–6 (2008)

71. Pelzer, E., Fortino, G., Bockstaller, C., Angevin, F., Lamine, C., Moonen, C., Vasileiadis, V., Guérin, D., Guichard, L., Reau, R., Messéan, A.: Assessing innovative cropping systems with DEXiPM, a qualitative multi-criteria assessment tool derived from DEXi. Ecol. Ind. **18**, 171–182 (2012)

72. Power, D.J.: Decision Support, Analytics, and Business Intelligence, 2nd edn. Business Expert Press, New York (2013)

73. Puppe, F.: Systematic Introduction to Expert Systems: Knowledge Representations and Problem-Solving Methods. Springer, Berlin (1993)

74. Prevolšek, B., Maksimović, A., Puška, A., Pažek, K., Žibert, M., Rozman, Č.: Sustainable development of ethno-villages in Bosnia and Herzegovina—a multi criteria assessment. Sustainability **12**, 1399 (2020). https://doi.org/10.3390/su12041399

75. Prišenk, J., Rozman, Č., Pažek, K., Turk, J., Bohak, Z., Borec, A.: A multi-criteria assessment of the production and marketing systems of local mountain food. Renew. Agric. Food Syst. 1–10 (2013)

76. Rajkovič, V., Bohanec, M. (1980): A cybernetic model of the computer aided decision making. In: Proceeding of 9th Internatioal Congress on Cybernetics, pp. 185–199, Namur

77. Rajkovič, V., Bohanec, M., Batagelj, V.: Knowledge engineering techniques for utility identification. Acta Physiol. (Oxf) **68**3(1–3), 271–286 (1988)

78. Ravier, C., Prost, L., Jeuffroy, M.-H., Wezel, A., Paravano, L., Reau, R.: Multi-criteria and multi-stakeholder assessment of cropping systems for a result-oriented water quality preservation action programme. Land Use Policy **42**, 131–140 (2015)

79. Ravnikar, T., Bohanec, M., Muri, G.: Monitoring and assessment of anthropogenic activities in mountain lakes: A case of the Fifth Triglav Lake in the Julian Alps. Environ. Monit. Assess. **188**(4), 1–17 (2016)

80. Rezaei, M.E., Barmaki, M., Veisi, H.: Sustainability assessment of potato fields using the DEXi decision support system in Hamadan Province, Iran. J. Integr. Agric. **17**(11), 2583–2595 (2018)

81. Ricci, B., Messéan, A., Lelièvre, A., Colénod, F.-C., Angevin, F.: Improving the management of coexistence between GM and non-GM maize with a spatially explicit model of cross-pollination. Eur. J. Agron. **77**, 90–100 (2016)

82. Roy, B.: Paradigms and challenges. In: Greco, S., Ehrgott, M., Figueira, J. (eds.) Multi Criteria Decision Analysis: State of the Art Surveys. Springer, New York (2016)

83. Rozman, Č, Grgić, Z., Maksimović, A., Ćejvanović, F., Puška, A., Šakič Bobić, B.: Multiple-criteria approach of evaluation of milk farm models in Bosnia and Herzegovina. Mljekarstvo **66**(3), 206–214 (2016)

84. Rozman, Č, Maksimović, A., Puška, A., Grgić, Z., Pažek, K., Prevolšek, B., Ćejvanović, F.: The use of multi criteria models for decision support system in fruit production. Erwerbs-obstbau **59**, 235–243 (2017)

85. Saaty, T.L., Vargas, L.G.: Models, Methods, Concepts & Applications of the Analytic Hierarchy Process. Springer, US, New York (2012)

86. Sammut, C.: Concept learning. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning. Springer, Boston, MA (2010). https://doi.org/10.1007/978-0-387-30164-8_154

87. Saracoglu, B.: A qualitative multi-attribute model for the selection of the private hydropower plant investments in Turkey. J. Ind. Eng. Manag. **9**(1), 152–178 (2016)
88. Stubelj Ars, M., Bohanec, M.: Towards the ecotourism: a decision support model for the assessment of sustainability of mountain huts in the Alps. J. Environ. Manag. **91**(12), 2554–2564 (2010)
89. Taškova, K., Stojanova, D., Bohanec, M., Džeroski, S.: A qualitative decision-support model for evaluating researchers. Informatica **31**(4), 479–486 (2007)
90. Trdin, N., Bohanec, M.: Extending the multi-criteria decision making method DEX with numeric attributes, value distributions and relational models. CEJOR **26**, 1–41 (2018)
91. Tsiouris, K.M., Gatsios, D., Rigas, G., Miljković, D., Koroušić-Seljak, B., Bohanec, M., Arredondo, M.T., Antonini, A., Konitsiotis, S., Koutsouris, D.D., Fotiadis, D.I.: PD_manager: an mHealth platform for Parkinson's disease patient management. Healthc. Technol. Lett. **4**(3), 102–108 (2017)
92. Wohlfender-Bühler, D., Feusthuber, E., Wäger, R., Mann, S., Aubry, S.J.: Genetically modified crops in Switzerland: implications for agrosystem sustainability evidenced by multi-criteria model. Agron. Sustain. Dev. **36**, 33 (2016)
93. Zadeh, L.A., Klir, G.J., Yuan, B. (eds.) Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A Zadeh. WSPC (1996)
94. Zimmermann, H.-J.: Fuzzy Sets, Decision Making, and Expert Systems, 4th edn.International Series in Management Science Operations Research, vol. 10. Springer, Netherlands (2001)
95. Zupan, B., Bohanec, M., Demšar, J., Bratko, I.: Learning by discovering concept hierarchies. Artif. Intell. **109**, 211–242 (1999)
96. Žnidaršič, M., Bohanec, M., Zupan, B.: proDEX–A DSS tool for environmental decision-making. Environ. Model. Softw. **21**(10), 1514–1516 (2006)
97. Žnidaršič, M., Bohanec, M.: Automatic revision of qualitative multi-attribute decision models. Found. Comput. Decis. Sci. **32**(4), 315–326 (2007)
98. Žnidaršič, M., Bohanec, M., Zupan, B.: Modelling impacts of cropping systems: demands and solutions for DEX methodology. Eur. J. Oper. Res. **189**, 594–608 (2008)
99. Žnidaršič, M., Bohanec, M., Lavrač, N., Cestnik, B.: Project self-evaluation methodology: the healthreats project case study. In: Proceedings of Information Society 2009, Ljubljana, pp. 85–88 (2009)
100. Žnidaršič, M., Bohanec, M., Trdin, N.: Qualitative assessment of data-mining workflows. In: Respício, A., Burstein, F. (eds.) Fusing Decision Support Systems into the Fabric of the Context, pp. 75–88. IOS Press, Amsterdam (2012)