AN IMPROVED RANKING ALGORITHM IN MULTI-CRITERIA DECISION MODELING METHOD DEX

Marko Bohanec
Jožef Stefan Institute, Department of Knowledge Technologies
Jamova cesta 39, SI-1000 Ljubljana, Slovenia
E-mail: marko.bohanec@ijs.si

Abstract: DEX (Decision EXpert) is a qualitative, hierarchical and rule-based decision-modeling method, particularly suitable for *sorting*: assigning decision alternatives into predefined categories. Under certain conditions and assumptions, DEX can be used also for *ranking*: ordering alternatives from best to worst. There already exists a DEX ranking method called QQ (Qualitative-Quantitative), specifically designed to evaluate alternatives both qualitatively — by assigning them to qualitative classes — and quantitatively — by ranking alternatives within each class. In this paper, we propose a novel method QQ2, aimed at improving some weaknesses of QQ: dependence on assessing criteria weights, and sub-optimal separation of alternatives. QQ2 achieves this by formulating and solving the ranking task as a quadratic optimization problem. An experimental comparison of QQ and QQ2 on 3322 real-life DEX decision tables demonstrated that QQ2 effectively overcomes the limitations of QQ.

Keywords: multi-criteria model, method DEX, decision rules, sorting, ranking, quadratic optimization

1 INTRODUCTION

Multi-criteria decision modeling (MCDM) is an approach used to make decisions when there are multiple, often conflicting, criteria or factors to consider [6]. Typically, MCDM proceeds [1] by defining a set of criteria that are relevant to the decision at hand. These criteria are then incorporated into a structured model that reflects the decision maker's preferences and priorities, and is used to systematically evaluate, compare and analyze different decision alternatives. The outcomes provide valuable information to making a well-informed decision.

According to Roy [8], there are three main types of decision problems: (1) *choosing*: selecting the best alternative from a set of available ones, (2) *ranking*: ordering alternatives from best to worst, and (3) *sorting*: assigning alternatives into predefined categories.

In this paper, we focus on a MCDM method DEX (Decision EXpert). DEX [4] is (1) *hierarchical*: a DEX model consists of hierarchically structured attributes; (2) *qualitative*: all attributes in a DEX model are symbolic, taking values such as "bad", "medium", "excellent"; (3) *rule-based*: decision alternatives are evaluated according to decision rules, acquired from the decision maker and represented in the form of decision tables.

DEX is primarily a sorting MCDM method: the evaluation process assigns each alternative to some distinct final evaluation class [7]. This is because all components of a DEX model are qualitative: attributes and their values, and decision rules that govern the evaluation process. Alternatives are described by qualitative input values and all evaluation results are qualitative.

In practice, however, it is often necessary to address other tasks than just sorting, i.e., choosing and ranking. For example, when there are several alternatives assigned to the same evaluation category, we may still want to distinguish between them and choose the best one. Therefore, it is tempting to think about a method that would rank alternatives using DEX models and would require very little additional effort from the decision maker. The method QQ [2] effectively demonstrates that this is indeed possible, provided that we accept some assumptions about preferential ordering of decision rules and their numerical interpretation. However, QQ exhibits two weaknesses: dependence on assessing criteria weights, and sub-optimal separation of alternatives. In what follows, we demonstrate that these weaknesses can be effectively addressed by employing the principle of dominance and formulating the ranking task in terms of a quadratic optimization problem. We propose a novel method called QQ2.

2 METHODS

2.1 Method QQ (Qualitative-Quantitative)

QQ is an extension of DEX that adds the ability to rank decision alternatives [2]. It is based on three main ideas:

1. *Combined qualitative-quantitative evaluation*: In addition to qualitative evaluation, which is normally carried out in DEX, it introduces a parallel numeric evaluation of alternatives. Numeric evaluations are used to rank alternatives within each qualitative class.
2. *Consistency of evaluations*: The two evaluations are kept consistent with each other. For each alternative assigned to a qualitative class C, the corresponding numerical evaluation must lie in the interval $[c - 0.5, c + 0.5]$, where c is the ordinal number of C. The values $+0.5$ and -0.5 are interpreted as “ideal” and “anti-ideal” evaluations within C, respectively.
3. *Automatic construction of the quantitative evaluation function,* which is developed (under some assumptions) from information already available in the DEX model. No additional information is requested from the decision maker.

QQ constructs the numeric evaluation function $Q: \mathbb{R}^m \to \mathbb{R}$, where m is the number of arguments (input attributes), in the way illustrated in Figure 1. The table on the left is an example of a DEX decision table that consists of rows, called elementary decision rules. In this case, the rules map all the combinations of values of attributes PRICE and TECH.CHAR. (technical characteristics) to the qualitative evaluation of a personal CAR.

QQ assumes that rules can be interpreted as points in a multi-dimensional space (Figure 1, right) and that they can be sufficiently well approximated (in the least squares sense) by a hyper-plane. Relative attribute weights are inferred from the slopes of this hyperplane along each argument direction (in this case, the weight ratio between PRICE and TECH.CHAR. is 60:40). Finally, QQ partitions the hyperplane to segments corresponding to each class value of the output attribute (CAR). In order to maintain the consistency between qualitative and quantitative evaluation, each segment is scaled so as to lie within the $c \pm 0.5$ interval for each class C.

	PRICE	TECH.CHAR.	CAR
1	high	bad	unacc
2	high	acc	unacc
3	high	good	unacc
4	high	*exc*	unacc
5	medium	bad	unacc
6	medium	acc	acc
7	medium	good	good
8	medium	*exc*	*exc*
9	*low*	bad	unacc
10	*low*	acc	good
11	*low*	good	*exc*
12	*low*	*exc*	*exc*

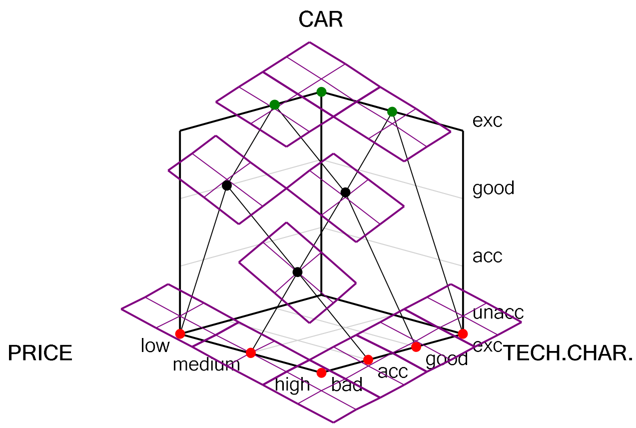

Figure 1: A decision table (left) and the numeric evaluation function Q constructed by QQ (right).

It is important to understand that Q is not used in isolation from other decision tables, but may depend on decision tables positioned at lower levels in a DEX model hierarchy. Since each Q outputs numeric values of the $c \pm 0.5$ type, it must also be able to accept inputs of the same type. This is the reason why Q is composed of ± 0.5 rectangles surrounding each elementary-rule point. Overall, Q is represented as a collection of such rectangles.

QQ has a major weakness: it depends on the notion of (numeric) weights. This seems “unnatural” for a fully qualitative method like DEX. Furthermore, weights make sense only if decision rules form a multi-dimensional pattern that is linear or close to linear. This is often not the case, and is particularly pronounced in cases involving arguments that are not preferentially ordered. Also, the assumption that the weight ratio is constant throughout the whole decision space restricts individual hyperplane partitions to adapt to the nearby contexts and better discriminate between the points.

2.2 Method QQ2

QQ2 is based on the same principles as QQ, but constructs the function Q differently. Instead of weights, it employs the concept of *preferential dominance* of decision rules. Consider decision table CAR in Figure 1. There, rules 1, 2, 3, 4, 5 and 9 all map to the same class CAR="unacc". However, by comparing conditional rule parts, we can see that rule 1 represents a worse situation than rule 2, because TECH.CHAR. are "bad" and "acc", respectively, and "bad" is worse than or equal to "acc". Therefore, we can say that $r_1 \preccurlyeq r_2$, where '$\preccurlyeq$' is a weak preference relation. In this way, we can establish two rankings for CAR="unacc": $r_1 \preccurlyeq r_2 \preccurlyeq r_3 \preccurlyeq r_4$ and $r_1 \preccurlyeq r_5 \preccurlyeq r_9$, and represent them with the lattice in Figure 2. Each rule has an associated value $q_i \in \mathbb{R}, i = 1, \dots, n$. Two elements are added, $q_L, q_U \in \mathbb{R}$: the lower and upper bound of Q, respectively. Similar lattices can be constructed for any other class.

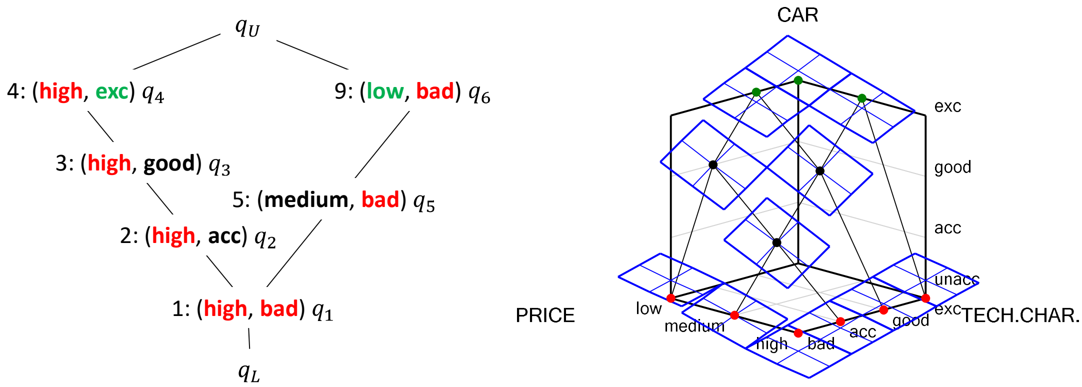

Figure 2: Lattice of CAR decision rules that map to C="unacc" (left) and Q constructed by QQ2 (right).

Having set the lattice, it is now clear what to do: place $q_1, \dots, q_n$ evenly between q_L and q_U, considering all lattice branches. We formulate this task in terms of a quadratic optimization problem, so as to minimize the sum of squared distances between adjacent points in the lattice:

Variables: $q_0 = q_L, q_1, q_2, \dots, q_n, q_{n+1} = q_U$
Minimize $\sum_{(i,j)\in L}(q_j - q_i)^2 = \sum_{i=0}^{n+1}(|N_i|{q_i}^2 - 2\sum_{j=0}^{n+1} q_i q_j)$
with respect to constraints: $q_0 = 0$, and $q_j - q_i \geq 1$ for $\forall(i,j) \in L$.

Here, L denotes the set of indices of adjacent rule pairs in the lattice, and N_i is the set of neighbors to rule r_i. The value $q_U = q_{n+1}$ is determined as part of the solution and need not be specified in advance. After solutions $q_1, q_2, \dots, q_n, q_{n+1}$ have been obtained, they have to be scaled so that the ± 0.5 rectangles around each rule stay within the $c \pm 0.5$ interval. Figure 2 (right) shows the result for the CAR decision table.

The results of QQ (Figure 1) and QQ2 (Figure 2) appear very similar. However, a closer look reveals that QQ's hyperplanes are generally narrower than QQ2's in the vertical direction. This is because QQ's are constrained with the 60:40 weight ratio, while QQ2's successfully adapt to the absence of adjacent rules that may have constrained their positions.

3 EXPERIMENTAL EVALUATION AND IMPLEMENTATION

We implemented QQ and QQ2 in R and experimentally evaluated them on a set of 3322 DEX decision tables, which were extracted from the database of real-life DEX models [3]. For each decision table, we observed: (1) *Separation*: average distance between dominated adjacent points, defined as $\frac{1}{|L|}\sum_{(i,j)\in L}|q_j - q_i|$; (2) *Time*: average execution time per decision table. Results (Table 1) indicate that QQ2 clearly outperforms QQ with respect to both measures; the difference is statistically significant. The difference in execution time is likely due to the fact that QQ was implemented fully in high-level R, while QQ2 used a machine-level optimization package “quadprog”. Anyway, both algorithms turned out fast and suitable for practice.

Table 1: Results of experimental evaluation of QQ and QQ2.

Measure	**QQ**	**QQ2**
Separation	0.1737442	0.2122702
Time [ms]	197.95	8.34

We also implemented QQ2 as part of the software DEXiWin [5]. The supported functionalities include: (1) calculating and presenting q-values for each decision table, (2) drawing Q for both QQ and QQ2, and (3) evaluating alternatives using QQ2.

4 CONCLUSIONS

The main contribution of this study is an improvement of the old, but still state-of-the-art DEX ranking algorithm QQ. Instead of using the concept of attribute weights, which is somewhat alien to the qualitative modeling approach of DEX, the new algorithm QQ2 employs the principle of dominance, which seems more “natural” and appropriate. QQ2 constructs the numeric aggregation function Q using quadratic optimization with linear constraints. Experimental evaluation confirmed that QQ2 statistically significantly outperforms QQ in terms of separation, i.e., coverage of the output dimension. Consequently, QQ2 is more sensitive and better separates decision alternatives being ranked. QQ2’s rectangles, drawn around each rule, are not constrained by QQ’s constant weight ratio and better adapt to the positions of adjacent decision rules.

Acknowledgement
The author acknowledges the financial support from the Slovenian Research and Innovation Agency under the programme Knowledge Technologies (No. P2-0103).

References

[1] Belton, V., Stewart, T. J. 2002. Multiple Criteria Decision Analysis: An Integrated Approach. Springer.

[2] Bohanec, M., Urh, B., Rajkovič, V. 1992. Evaluating options by combined qualitative and quantitative methods. Acta Psychologica 80, 67–89.

[3] Bohanec, M. 2017. Multi-criteria DEX models: An overview and analysis. SOR-2017: 14th International Symposium on Operational Research in Slovenia, Bled, Slovenia, 155–160.

[4] Bohanec, M., 2022. DEX (Decision EXpert). A qualitative hierarchical multi-criteria method. In: Multiple Criteria Decision Making (ed. Kulkarni, A.J.), Singapore: Springer, 39–78.

[5] Bohanec, M., 2025. DEXi Suite: DEXi decision modelling software. SoftwareX 31, 102240. https://doi.org/10.1016/j.softx.2025.102240

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

[7] López, L.M., Ishizaka, A., Qin, J., Carrillo, P.A.Á. 2023. Multi-Criteria Decision-Making Sorting Methods. London: Academic Press.

[8] Roy, B. 1996. Multicriteria Methodology for Decision Aiding. Springer.

Proceedings of the 18th International Symposium on Operational Research in Slovenia

SOR'25

Bled, SLOVENIA, September 24-26, 2025

Edited by:

S. Drobne, L. Zadnik Stirn, M. Kljajić Borštnar, J. Povh and J. Žerovnik

Slovenian Society INFORMATIKA (SDI)
Section for Operational Research (SOR)

Proceedings of the 18th International Symposium on Operational Research in Slovenia, SOR'25, Bled, Slovenia, September 24-26, 2025.

Proceedings of the scientific symposium SOR'25 (https://sor.fov.um.si/). All contributions were peer-reviewed in a blinded manner.

Editors: Assoc. Prof. Dr. Samo Drobne, UL FGG, Ljubljana, Slovenia
Prof. Dr. Lidija Zadnik Stirn, UL BF, Ljubljana, Slovenia
Prof. Dr. Mirjana Kljajić Borštnar, UM FOV, Kranj, Slovenia
Prof. Dr. Janez Povh, Rudolfovo Institute, Novo mesto, Slovenia
Prof. Ddr. Janez Žerovnik, Rudolfovo Institute, Novo mesto, Slovenia

Revievers are listed below.

Keywords: operations research, mathematical programming & optimization, computational optimization, multicriteria decision-making, graph theory, game theory, econometrics & statistics, data science & artificial intelligence, logistics & transportation, environment & human resources, production & management, agricultural economics & policy, finance & investments, inventory & location research, Slovenian Society Informatika (SSI), Section for operational research (SOR)

Available at: https://www.drustvo-informatika.si/sekcije-drustva?stran=publikacije-sor

Organiser of SOR'25: Slovenian Society INFORMATIKA – Section for Operational Research, SI-1000 Ljubljana, Litostrojska cesta 54, Slovenia (https://www.drustvo-informatika.si/sekcije-drustva)

Co-organiser of SOR'25: University of Maribor, Faculty of Organizational Sciences, SI-4000 Kranj, Kidričeva cesta 55a, Slovenia (http://www.fov.um.si/)

Co-organiser of SOR'25: Rudolfovo – Science and Technology Centre Novo mesto, SI-8000 Novo mesto, Podbreznik 15, Slovenia (https://www.rudolfovo.eu/)

Publisher: Slovenian Society INFORMATIKA – Section for Operational Research, SI-1000 Ljubljana, Litostrojska cesta 54, Slovenia (https://www.drustvo-informatika.si/sekcije-drustva)

Ljubljana, 2025

First electronic version.

Kataložni zapis o publikaciji (CIP) pripravili v Narodni in univerzitetni knjižnici v Ljubljani
COBISS.SI-ID 248708867

ISBN 978-961-6165-64-8 (PDF)

Proceedings of the 18th International Symposium on Operational Research in Slovenia (SOR'25) is cited in: ISI (Index to Scientific & Technical Proceedings on CD-ROM and ISI/ISTP&B online database), Current Mathematical Publications, Mathematical Review, MathSci, Zentralblatt für Mathematic / Mathematics Abstracts, MATH on STN International, CompactMath, INSPEC, Journal of Economic Literature

Technical editor: Samo Drobne

Designed by: Samo Drobne