Software

The Rubis Decision-Deck software resources

Raymond Bisdorff, http://charles-sanders-peirce.uni.lu/bisdorff/

Rubis is a new best choice decision method in the tradition of the Electre IS method that is available in the Decision-Deck software package. A brief description of it is given hereafter followed by a short illustrative application.

A Decision-Deck software resource

The Decision-Deck (D2) project

The D2 project [2] provides an open source software, composed of various modular components, pertaining to the field of Multiple Criteria Decision Analysis (MCDA). It gives a user the possibility to add, modify or simply use existing plugged-in functionalities (plugins). These constituents can either be complete MCDA methods or elements common to a large range of procedures. The typical end-user of the Decision-Deck platform is an MCDA researcher, an MCDA consultant or a teacher in an academical institution.

The D2 project, started in early 2006, is at present actively supported by the MathRO laboratory of the Faculty of Engineering of Mons and the SMG of the Free University of Brussels (Belgium),the Lamsade laboratory of the University Paris-Dauphine and Karmic Software Research (France), the ILIAS laboratory of the University of Luxembourg, and the INESC (Coimbra, Portugal).

The D2 platform architecture

The Decision-Deck software is written in the Java programming language and is therefore platform independent. Its latest version can be downloaded from the collaborative software development management system Sourceforge [3]. Two kinds of implementation designs are available: on the one hand a rich Java client which implements locally the MCDA methods (D2), and on the other hand, a distributed web service and AJAX based architecture, serving the MCDA methods from a

distributed web server (D3). The Rubis choice method is actually implemented as such a web service on the *ernst-schroeder.uni.lu* RIA-server at the University of Luxembourg [1].

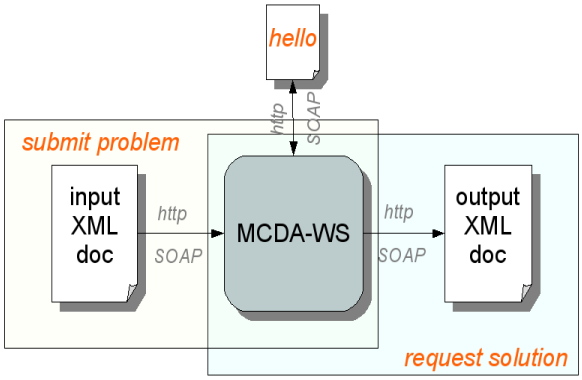

Figure 5: The Decision-Deck asynchronous MCDA web service layout

Examples of D2-plugins

The following MCDA methods are implemented in the current release of the D2 platform: - *sorting* of alternatives into ordered classes based on an outranking relation (IRIS), - *best choice* method based on an additive aggregation model accepting imprecise information on the scaling coefficients (VIP), - *ranking* of alternatives with a set of value functions (UTA-GMS/GRIP), and - *choosing a single best* alternative based on a bipolar-valued outranking relation (Rubis).

The principles of the Rubis MCDA method

The Rubis best choice method (Bisdorff, Meyer, Roubens 2007) [4] is a progressive multicriteria decision aid method in the tradition of the outranking methods. It is focused on the problem of selecting a single best alternative on the basis of the performances of all alternatives on a given consistent family of criteria. The Rubis solution consists mainly in a best choice recommendation (BCR) verifying the following principles:

1. Each non-recommended alternative is eliminated for well motivated reasons.

2. The number of alternatives retained in a BCR is as small as possible.

3. At each step of the progressive decision aiding a stable refinement of the previous BCR is delivered.

4. A BCR does not correspond simultaneously to a best as well as a worst choice recommendation.

5. The BCR is as credible as possible with respect to the preferential knowledge available in the current step of the decision aiding process.

Following recent formal results (Bisdorff, Pirlot, Roubens 2006) [5], it can be shown that such a BCR is given by the maximal credible and strict outranking kernels of the chordless odd circuits augmented bipolar-valued outranking digraph one may construct from a given performance tableau (see [4]).

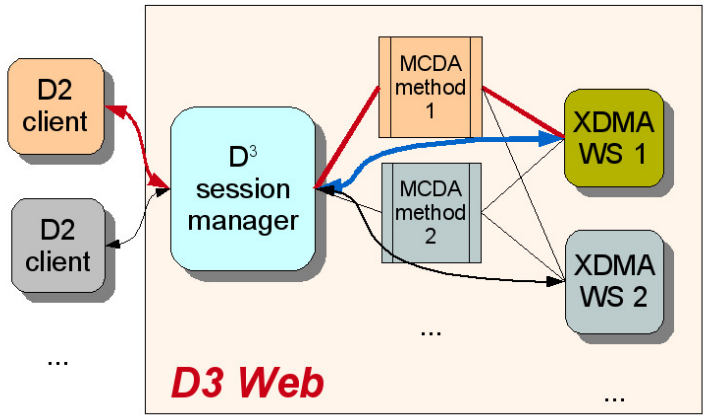

Figure 6: The D3 Web application

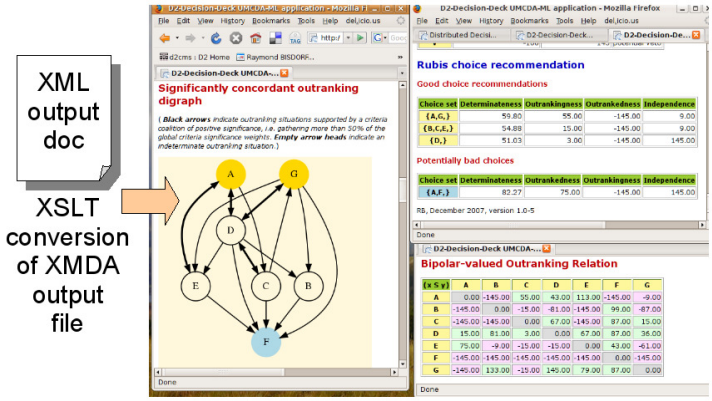

Figure 7: Browsing the Rubis solver's XML encoded response

The Python Rubis Solver

A Rubis best choice decision solver is actually implemented in the Python programming language via the digraphs Python module which can be downloaded from the following URL: http://ernst-chroeder.uni.lu/Digraph [6]. In order to distribute the solver in an operating system and programming language independent way, the Python Rubis solver offers also an asynchronous web service (WS) installed on the *ernst-schroeder.uni.lu* server at the University of Luxembourg [1].

The Rubis MCDA-web service

Accessing the Rubis solver may thus be done via an MCDA-WS which follows the general recommendations of the Decision-Deck project (see Figure 1). Three standard SOAP RPC literal ports over HTTP are indeed published:

1. A *hello* port for testing the connection with the Rubis service provider.

2. A *submitProblem* port for submitting an XML encoded problem description.

3. A *requestSolution* port for requesting the XML encoded solution of the Rubis best choice decision method.

#	Identifyer	Name	Comment
1	ant	Antequerra	An afternoon excursion to Antequerra and surroundings.
2	ard	Ardales	An afternoon excursion to Ardeles and El Chorro.
3	be	beach	Sun, fun and more
4	crd	Cordoba	A whole day visit by car to Cordoba.
5	dn	fa niente	Doing nothing
6	lw	long walk	A whole excursion with a picnic on the road.
7	mal	Malaga	A whole day visit by car to Malaga.
8	sev	Sevilla	A whole day visit by car to Sevilla.
9	sw	short walk	Less than a half day occupation.

Figure 8: Potential decision actions

Detailed description of the architecture and technical instructions for accessing the Rubis web service from local clients in any programming language may be found on the RIA-server *ernst-chroeder.uni.lu* of the University of Luxembourg [1]. At the same address may be found detailed and technical information concerning the XML encoding of Rubis specific performance tableaus to be submitted to the Rubis solver with the corresponding XML encoding of the Rubis Solver's response file.

#	Identifyer	Name	Comment
1	dis	Distance	Minutes by car to attend the place of the activity.
2	phy	Physical investment	Contribution to physical health care.
3	rel	Relaxation	Anti-stress support.
4	rest	restauration	Quality of the expected restauration facilities.
5	sun	Sun, fun, and ... more !	No comment.
6	tour	Touristic interest	How many stars in the Michelin ?

Figure 10: The consistent family of criteria

Browsing the D3 Web Server offering the Rubis-WS

Following the previous design, the Rubis-WS requires a specific D3-Web session manager in order to asynchrounously submit a decision problem and subsequently request the corresponding solution in a coordinated and persistent way. Such a Decision-Deck D3 Web Application is at present installed at the following address: http://ernst-schroeder.uni.lu/d3/ (user:demo password: D3_demo).

The D3-Web application allows on-line submitting of XML encoded Rubis problem descriptions and visualization of the Rubis solver's response in a standad browser session (recent browser versions like IE 6+, Firefox 1.5+ etc are required due to the heavy use of javascripting).

criterion	ant	ard	be	crd	dn	lw	mal	sev	sw
dis	-120.00	-100.00	-30.00	-360.00	0.00	-90.00	-240.00	-240.00	0.00
phy	3.00	7.00	0.00	5.00	0.00	10.00	5.00	5.00	5.00
rel	1.00	-999.00	8.00	3.00	10.00	5.00	3.00	3.00	6.00
rest	8.00	10.00	4.00	8.00	10.00	1.00	8.00	10.00	1.00
sun	0.00	3.00	10.00	3.00	1.00	3.00	8.00	5.00	5.00
tour	5.00	7.00	3.00	10.00	0.00	8.00	10.00	10.00	5.00

Figure 9: The performance tableau

Using a D2 rich Java client

The D3-Web Application may also be accessed with the help of a classic D2 rich Java client when using the D2-Rubis plugin [3]. With this resource it is possible for an analyst or decision aid consultant to describe a set of alternatives and a family of criteria. External evaluators may then remotely assess the performances of the alternatives on each criterion. Eventually the decision-maker can tune the criteria family by choosing adequate significance weights and discrimination thresholds. The final problem description is then automatically transformed in an XML encoded problem description and submitted to a distant Rubis web service. A subsequent request for viewing the Rubis solver's outcome results is operated in a standard browser session (See Figure 3).

(x S y)	ant	ard	be	crd	dn	lw	mal	sev	sw
ant	0.00	-50.00	0.00	-33.33	0.00	-50.00	-33.33	-66.67	-33.33
ard	83.33	0.00	16.67	50.00	50.00	50.00	16.67	16.67	16.67
be	0.00	-16.67	0.00	0.00	16.67	33.33	0.00	0.00	16.67
crd	-100.00	-100.00	-100.00	0.00	-100.00	-100.00	33.33	0.00	-100.00
dn	33.33	-16.67	33.33	0.00	0.00	0.00	0.00	0.00	0.00
lw	66.67	50.00	-33.33	33.33	0.00	0.00	0.00	0.00	33.33
mal	66.67	-100.00	-100.00	100.00	-100.00	-100.00	0.00	66.67	-100.00
sev	66.67	-100.00	-100.00	100.00	-100.00	-100.00	66.67	0.00	-100.00
sw	66.67	-16.67	0.00	33.33	33.33	33.33	0.00	33.33	0.00

Figure 11: Bipolar outranking relation valued in the interval [-100;100]

Small Illustrative Example

The problem

A family, staying during their holidays in Ronda (Andalucia), is planning the next day's activity. The alternatives shown in Figure 4 are considered as potential actions. The family members agree to measure their preferences with respect to a set of six criteria such as the time to attend the place (*Distance* to be mimized), the required *physical investment*, the expected quality of the *restauration*, *touristic interest*, *relaxation*, *sun*, *fun*, and more ... (see Figure 5).

The common evaluation of the performances of the nine alternatives on all the criteria results in the performance tableau shown in Figure 6. On the qualitative criteria all performances are marked on a same ordinal scale going from 0 (lowest) to 10 (highest). On the quantitative *Distance* criterion (to be minimized) the required travel time to go to and return from the activity is marked in negative minutes.

In order to model only effective preferences, an indifference threshold of 1 point and a preference threshold of 2 points is put on the qualitative performance measures. On the distance criterion, an indifference threshold of 20 min, and a preference threshold of 45 min. is considered. Furthermore, a difference of more than two hours to attend the activity's place is considered to raise a veto. Finally, all citeria are juged equi-significant for the action to be chosen.

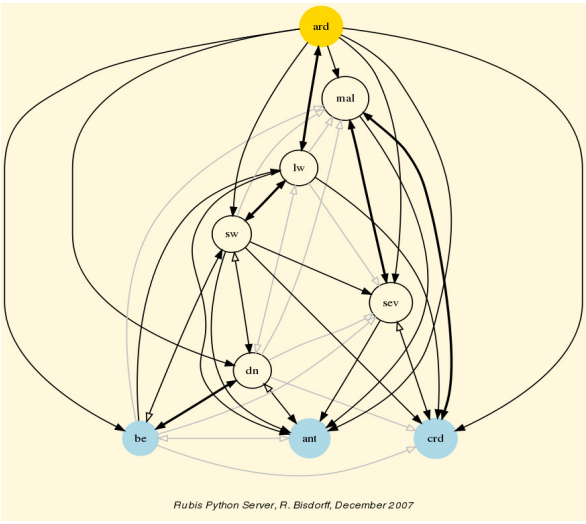

Figure 12: The resulting outranking digraph and the Rubis BCR

The solution

The resulting outranking relation, bipolar-valued in the credibility domain [-100.00, +100.00], is shown in Figure 7. The 0.00 values indicate indeterminate outranking situations as one may observe when comparing the very contradicting alternatives *doing nothing* and *long walk* for instance. The -100.00 values, observed for the large cities excursions, and especially for the *Cordoba* trip, results from the vetos that are raised due to the excessive travel time needed to go there and return.

In the corresponding outranking digraph (see Figure 8), the Rubis Solver marks the afternoon excursion to *Ardales* and *El Chorro* as the Rubis best choice recommendation (see Figure 8, empty arrow heads and grey lines indicate indeterminate outranking situations), whereas the *beach* and *Antequerra* or *Cordoba* excursions appear being the worst choices. It is worthwhile noticing that three coherent groups of more or less indifferent alternatives clearly emerge: - *Ardeles*, *long* and *short walks*; - the large cities excursions with *Sevilla*, *Malaga* and *Cordoba*; and the relaxing - *'fa niente'*, and *beach* alternatives. Our family members eventually appreciated very much the recommended *Ardales* excursion and all had a wonderful time the next day.

References

[1] The Rubis resources at the university of Luxembourg: http://ernst-schroeder.uni.lu/

[2] The web site of the Decision-Deck project: http://www.decision-deck.org/

[3] The Decision-Deck sourceforge repository: http://decision-deck.sourceforge.net/

[4] R. Bisdorff, P. Meyer and M. Roubens, "Rubis: a bipolar-valued outranking method for the choice problem". 4OR, *A Quarterly Journal of Operations Research*, Springer-Verlag. (Online)] Electronic version: DOI: 10.1007/s10288-007-0045-5, pp 1 - 27, in press.

[5] R. Bisdorff, M. Pirlot and M. Roubens (2006). "Choices and kernels from bipolar valued digraphs". *European Journal of Operational Research*, 175 (2006) 155-170. (Online) Electronic version: DOI:10.1016/j.ejor.2005.05.004.

[6] R. Bisdorff, *The Python digraphs module for Rubis: A user manual*. University of Luxembourg, 2008, http://ernst-schroeder.uni.lu/Digraph