Building a Shared Sorting Function for a Group of Decision Makers

Olivier Cailloux Vincent Mousseau Patrick Meyer

ILLC - Universiteit van Amsterdam

11 April 2013

Outline

1. Context and motivation
2. Why not the obvious solution?
3. Our framework
4. Improving the procedure
5. Conclusions & Future work

Outline

1. Context and motivation
2. Why not the obvious solution?
3. Our framework
4. Improving the procedure
5. Conclusions & Future work

Our goal

- Obtain a way of evaluating objects (*alternatives*),
- by sorting them into preference-ordered *categories*, e.g. {Good, Medium, Bad},
- on the basis of several (objective) performance measures (*criteria*).

- Resulting sorting function must be consensual among multiple Decision Makers (DMs).

General framework

- Alternatives $\mathcal{A}$
- Criteria $\mathcal{J}$
- Performances $g_j : \mathcal{A} \rightarrow X_j, \forall j \in \mathcal{J}$
- Preference orders $\succeq_j, \forall j \in J$ (total orders)
- Preference ordered set of categories $\mathcal{C}$
  - C_1 worst category
- *Set* of decision makers $\mathcal{T}$

	g_1	g_2	g_3	g_4	
a_1	5	6	1	$++$	$\longrightarrow$?
a_2	4	7	8	$+$	$\longrightarrow$?
a_3	2	8	4	$-$	$\longrightarrow$?
a_4	4	6	9	$+$	$\longrightarrow$?
...					

C_3
C_2
C_1

Example: student evaluation

- Determine a way of evaluating students at the end of the year
  - Obtain a sorting function
- $\mathcal{A}$, space of all possible evaluations (may be infinite or large)
- We may reason on specific cases (possibly fictitious)
- Involves subjective appreciations
  - DMs may have different opinions

	math.	lang.	phys.	partic.	
St 1	B	D	A	$++$	$\longrightarrow$?
St 2	A	A	B	$+$	$\longrightarrow$?
St 3	A	C	C	$-$	$\longrightarrow$?
St 4	C	A	A	$+$	$\longrightarrow$?
...					

- C_4: highest d.
- C_3: distinction
- C_2: succeeds
- C_1: fails

Example: research projects

- Researchers submit projects to the board
- The board wants some way of evaluating these research projects
- Criteria: Redaction quality, scientific quality, experience of the team, publication score, ...

	redac.	sci.	exp.	publ.	
Pr 1	2	5	3	75.1	$\longrightarrow$?
Pr 2	5	5	4	32.2	$\longrightarrow$?
Pr 3	5	3	3	63.4	$\longrightarrow$?
Pr 4	3	5	4	61.7	$\longrightarrow$?
...					

C_2: fund project

C_1: reject

Example: green labelling

- Evaluate ecological quality of consumer products
- Criteria: amount of pollutants of different sorts, road distance, ...
- Use a representative set of products
- Obtain a transparent decision procedure

	pol. 1	pol. 2	rd dist.	
Pr 1	2	5	3	→ ?
Pr 2	5	5	4	→ ?
Pr 3	5	3	3	→ ?
Pr 4	3	5	4	→ ?
...				

- A+
- A
- B
- C
- D

Important features of the problem setting

Hypothesis

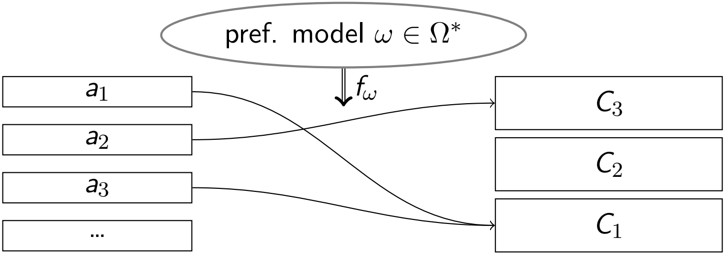
The situation is *not* about bargaining.

Sorting function depends on objective and subjective data.

- Objective, or consensual, data: most importantly, performances;
- subjective data: how the performances relate.

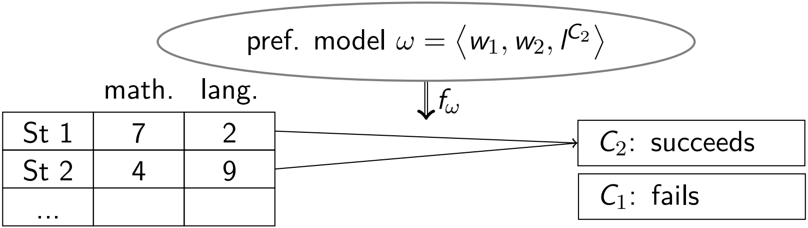
Modelling subjectivity

- Sorting function is parameterized
- Vector of parameters ω captures the subjective aspect of the sorting
- Set of possible parameter values is Ω^*
- Choosing Ω^* defines the set of possible sorting functions
- Can be done for one individual or for the group



Example: weighted sum with thresholds

- Criteria $\mathcal{J}$, $|\mathcal{J}| = n$, with functions $g_j : A \to \mathbb{R}$
- Preference model $\omega = \langle W, L \rangle$:
  - a vector of weights $W \in \mathbb{R}^n_+$,
  - a set of thresholds $L \in \mathbb{R}^{K-1}_+$ (K categories), l^C is the low threshold for category C.
- Class of models is $\Omega^* = \mathbb{R}^n_+ \times \mathbb{R}^{K-1}_+$
- Sorting function $f_\omega : A \to \mathcal{C}$ compares the score of a to the thresholds $l^{C_2}, l^{C_3}, \ldots$



Outline

1 Context and motivation
2 Why not the obvious solution?
3 Our framework
4 Improving the procedure
5 Conclusions & Future work

An obvious way

Two difficulties

1. Multiple criteria to aggregate
2. Multiple DMs whose points of view must be aggregated

A simple way to get around the second difficulty and come back to the mono-DM case:

- Come up with individual preference models $\omega_t, \forall t \in \mathcal{T}$ (typically using utility functions)
- The group preference model is some aggregation of the individual preference models;
- or the group sorting function is some aggregation of the individual sorting functions $\{f_{\omega_t}, t \in \mathcal{T}\}$.

Why not?

What we want

- Avoid unneccessary sacrifices
  - use of preference lability
- Achieve a better understanding of the points of consensus and disaggrements
- Explore non utility-based classes of preference models
- Ask questions in terms of the problem
  - effects on the sorting results
- Ask easy questions
  - assignment examples

(Not all of these points are specific to the approach presented here.)

Preference lability

- Preferences are labile: not determined precisely in one's head
- DMs may not know their own preferences
- Maybe several equally good ways of aggregating the criteria, for a given DM
- Documented (with a different perspective) in numerous experiments
[Kahneman and Tversky, 2000, Lichtenstein and Slovic, 2006]
    - "Preferences are constructed in context"

Outline

1. Context and motivation
2. Why not the obvious solution?
3. Our framework
4. Improving the procedure
5. Conclusions & Future work

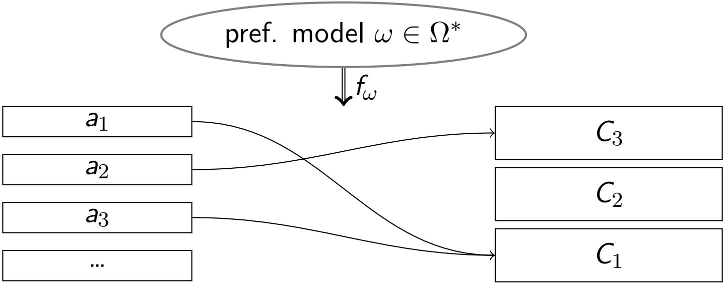
Sketch of the procedure

- We work with (a variant of) Electre Tri as the class of sorting functions
- Parameters to be elicited: $\omega = \langle L, W, \lambda \rangle$
- We ask for assignment examples (e.g. $a_1 \rightarrow C_2$)
- These constrain the set of candidate preference models
  - We must have $f_\omega(a_1) = C_2$
- If no preference model satisfy all examples:
  - Search which constraints should be removed to restore consistency

Defining the group sorting function

Explanation proceeds in two steps.

- Show how (the variant of) ELECTRE TRI works: how f_ω is defined, *assuming* the preference model ω is defined
- Then, explain how we find a suitable preference model for the group of DMs.



Sorting method: a variant of ELECTRE TRI

Preference parameters

- *Category limits* $L = \langle l^C, C \in \mathcal{C} \setminus C_1 \rangle$: determine when the alternative is *good enough* on a criterion
- Weights $W = \langle w_j, j \in \mathcal{J} \rangle$, and a majority threshold λ: determine when the alternative is *globally good enough*

Alternatives $\mathcal{A}$

	g_1	g_2	g_3
a_1	3	1	1
a_2	5	3	3
a_3	0	5	1
a_4	2	0	2

Cat. limits L

	g_1	g_2	g_3
	C_3: Good		
l^{C_3}	4	4	3
	C_2: Average		
l^{C_2}	3	3	2
	C_1: Bad		

Weights W, λ

	g_1	g_2	g_3
W	0.2	0.6	0.2

$\lambda = 0.8$

Sorting method: a variant of Electre Tri

- a may reach at least C iff $\sum_{j \text{ in favor}} w_j \geq \lambda$ ($C \neq C_1$).
- j in favor of a reaching C iff $g_j(a) \geq l_j^C$.
- Thus, a sorted into the best category s.t.

$$\sum_{j | g_j(a) \geq l_j^C} w_j \geq \lambda.$$

Alternatives $\mathcal{A}$

	g_1	g_2	g_3
a_1	3	1	1
a_2	5	3	3
a_3	0	5	1
a_4	2	0	2

Cat. limits L

	g_1	g_2	g_3
	C_3: Good		
l^{C_3}	4	4	3
	C_2: Average		
l^{C_2}	3	3	2
	C_1: Bad		

Weights W, λ

	g_1	g_2	g_3
W	0.2	0.6	0.2

$\lambda = 0.8$

Strengths of Electre Tri

- A variant has been axiomatized [Bouyssou and Marchant, 2007a, Bouyssou and Marchant, 2007b].
- Justification of assignment easy to grasp (no complex computation needed).
- Might ease discussion among DMs.

Determining a group preference model

We want to determine a suitable $\omega = \langle L, W, \lambda \rangle$ for the group of DMs.

- We ask for assignment examples
- $A^* \subseteq \mathcal{A}$ the set of alternatives used as examples
- $\forall a \in A^*$, we know the category a should go into, according to one DM at least
- Assignment examples should be non contradictory

Mathematical program

Having assignment examples $E \subseteq A^* \times \mathcal{C}$,

- We want to find $\omega = \langle L, W, \lambda \rangle$ such that
  - $\forall (a, C) \in E : f_\omega(a) = C$.
- Idea (finding ω satisfying examples) existed already [Mousseau and Słowiński, 1998] but no efficient tools to solve it.
- We solve a Mixed Integer Program (MIP) [Cailloux et al., 2012].
- The MIP must represent the assignment examples as constraints on the decision variables L, W, λ.

Defining (some of) the constraints: idea

Consider example $a \rightarrow C_k$, with $C_k \neq C_1, C_{|\mathcal{C}|}$.

- Criterion j thinks that a deserves to reach C_k iff
  - $g_j(a) \geq l_j^{C_k}$.
- Introduce binary variable $b_j^{a,C_k} = 1$ iff a may reach C_k according to j.

Examples E

	g_1	g_2	g_3	C
a	3	1	5	C_k
…				

Cat. limits L

	g_1	g_2	g_3
	C_3: Good		
l^{C_3}	$l_1^{C_3}$	$l_2^{C_3}$	$l_3^{C_3}$
	C_2: Average		
l^{C_2}	$l_1^{C_2}$	$l_2^{C_2}$	$l_3^{C_2}$
	C_1: Bad		

Weights W, λ

g_1	g_2	g_3
w_1	w_2	w_3

Defining (some of) the constraints: idea

Consider example $a \to C_k$, with $C_k \neq C_1, C_{|\mathcal{C}|}$.

- a deserves to reach at least C_k iff

$$\sum_{j \mid b_j^{a,C_k}} w_j \geq \lambda.$$

Examples E

	g_1	g_2	g_3	C
a	3	1	5	C_k
...				

Cat. limits L

	g_1	g_2	g_3
	C_3: Good		
l^{C_3}	$l_1^{C_3}$	$l_2^{C_3}$	$l_3^{C_3}$
	C_2: Average		
l^{C_2}	$l_1^{C_2}$	$l_2^{C_2}$	$l_3^{C_2}$
	C_1: Bad		

Weights W, λ

g_1	g_2	g_3
w_1	w_2	w_3

Defining (some of) the constraints

Consider example $a \rightarrow C_k$, with $C_k \neq C_1, C_{|\mathcal{C}|}$.
We want that

- a reaches at least C_k;
- a does not reach C_{k+1}:

$$\sum_{j|b_j^{a,C_k}} w_j \geq \lambda \wedge \sum_{j|b_j^{a,C_{k+1}}} w_j < \lambda.$$

Define continuous variable [Meyer et al., 2008]:

$$v_j^{a,C_k} = \begin{cases} w_j & \text{if } b_j^{a,C_k} = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Therefore:

$$\sum_j v_j^{a,C_k} \geq \lambda \wedge \sum_j v_j^{a,C_{k+1}} < \lambda.$$

Recap of the procedure

- Obtain assignment examples
- Run the MIP
- Find a preference model $\omega = \langle L, W, \lambda \rangle$ such that f_ω satisfies the examples
- Present the results to the DMs by applying the function to a larger set of alternatives, or by explaining how it "reasons"
- They might want to correct or add examples

Also possible:

- No satisfying model exist
- Then some examples must be changed
- Existing procedures can find minimal sets of constraints to remove [Mousseau et al., 2006]

Outline

1. Context and motivation
2. Why not the obvious solution?
3. Our framework
4. Improving the procedure
5. Conclusions & Future work

Assignments stability

- Typically, DMs provide few examples compared to number of examples required to define a sorting function.
- Thus, great variability in possible assignments of other alternatives (from $\mathcal{A} \setminus A^*$).
- When one example changes (thus ω is changed to ω'), the other assignments may completely change (f_ω may be completely different than $f_{\omega'}$).
- This is called instability.
- As the procedure is used interactively, instability can occur at some point.
- Convergence may be slow and hard to see.

Improving the procedure

- Possible instability
- Possibly no consensus from the start
- What if starting examples are contradictory?
- We want the procedure to be more incremental
- First, agree on the category limits (L), then on the weights (W, λ).

Partially shared parameters

- We ask for examples E^t to each DM $t \in \mathcal{T}$.
- These can be contradictory, e.g. $\left(a_1 \xrightarrow{t_1} C_3\right), \left(a_1 \xrightarrow{t_2} C_2\right)$.
- For each DM $t \in \mathcal{T}$, we search for individual preference models $\omega^t = \langle L, W^t, \lambda^t \rangle$ satisfying examples E^t.
- Thus, category limits are shared but weights are chosen individually.
- This may exist even though there is no shared model satisfying all examples.
- This decomposes the problem into two simpler problems.
- Once shared category limits are found, better stability.
- Connects with existing procedures to find shared weights [Damart et al., 2007].

Implementation

- Search procedure may be implemented with a MIP.
- Has been implemented in a Java free (libre) package: `www.decision-deck.org/j-mcda/` [Cailloux, 2012].
- Available as a web service in the Decision Deck framework.
- Can be used through a client program (diviz): `http://www.decision-deck.org/diviz/`.

Performance test

Objective

Examine whether the MIP is able to find shared profiles within a reasonable time frame.

- Random generation : [3–10] criteria, [1–4] DMs, [2–5] categories, [1–700] examples per DM.
- Random performances $g_j(a)$.
- Shared profiles used to sort examples, then forgotten.

Performances

- Nb binaries $= |\mathcal{J}| \times |A^*| \times (|\mathcal{C}| - 1)$.
- 6 criteria, 3 categories, 3 DMs giving each 30 different examples: ≈ 1000 binaries

Results

- Solved within 90 minutes using less than 3 GB disk space:

Binary variables	Sample size	Problems solved
[0, 399]	477	100%
[400, 799]	441	87%
[800, 1199]	362	80%
[1200, 1599]	290	78%
[1600, 1999]	268	75%
[2000, 2199]	121	69%

- Mainly depend on the number of criteria.

Computing restrictions on weights

- We can now propose shared L to the DMs.
- For each $t \in \mathcal{T}$, accepting L imposes a restriction on possibility of choosing weights W^t satisfying the examples E^t.
- We define an LP to compute these restrictions, $\forall t \in \mathcal{T}, j_1, j_2 \in \mathcal{J}$:

$$j_1 \rhd^t j_2 \Leftrightarrow w^t_{j_1} > w^t_{j_2}, \forall \left\langle w^t_j, j \in \mathcal{J}, \lambda^t \right\rangle \text{satisfying } E^t.$$

- This may help the DMs in choosing shared category limits.

Variants and extensions

- Possible to set direct constraints on the model parameters.
- Possible to search for models including ELECTRE TRI vetoes.
- Possible to specify constraints on the (weighted) category size [Zheng et al., 2011].
  - Select research projects that fit the budget.
  - Mainly for consensual constraints.
- DMs may give imprecise assignments $\left(a \xrightarrow{t} [C_1, C_2]\right)$.
- Also implemented in software.

Possible use

This approach can be considered as a supplementary tool in the analyst's toolbox. Here is only one possible use.

- Ask for examples E^t.
- Resolve possible individual inconsistencies.
- Search for consensual model ω.
- If no such model, search for shared category limits.
- If still no model, allow for vetoes.
- Present resulting category limits and individual weights with restrictions on the weights.
- If not acceptable, DMs may provide supplementary examples or other constraints.

Outline

1. Context and motivation
2. Why not the obvious solution?
3. Our framework
4. Improving the procedure
5. Conclusions & Future work

Conclusion

- A path out of the "average of individual opinions" strategy.
- We search for consensus instead of compromise.
- We use a "divide and conquer" approach.
- Using a model possibly more intuitive than utility functions.
- Asks easy questions.
- Results are easily interpretable.
- Computation time: could be improved.

Future work

- Generalise the idea and separate the specifics to ELECTRE TRI.
- May also apply to other classes of models (such as utility functions)
- May apply to other problem types, e.g. ranking instead of sorting.
- Separate parameters in different manners?
- Formal description of the relation between "what we want" and "what we do".
- Validation of the model class.

Thank you for your attention!

Bibliography I

Bouyssou, D. and Marchant, T. (2007a).
An axiomatic approach to noncompensatory sorting methods in MCDM, I: The case of two categories.
European Journal of Operational Research, 178(1):217–245.

Bouyssou, D. and Marchant, T. (2007b).
An axiomatic approach to noncompensatory sorting methods in MCDM, II: more than two categories.
European Journal of Operational Research, 178(1):246–276.

Bibliography II

Cailloux, O. (2012).
Dealing with numerical imprecision in mathematical programs for electre tri models disaggregation.
Technical Report 2012-02, Laboratoire Génie Industriel, École Centrale Paris.
Cahiers de recherche 2012-02.

Cailloux, O., Meyer, P., and Mousseau, V. (2012).
Eliciting ELECTRE TRI category limits for a group of decision makers.
European Journal of Operational Research, 223(1):133–140.

Bibliography III

Damart, S., Dias, L., and Mousseau, V. (2007).
Supporting groups in sorting decisions: Methodology and use of a multi-criteria aggregation/disaggregation DSS.
Decision Support Systems, 43(4):1464–1475.

Kahneman, D. and Tversky, A., editors (2000).
Choices, Values, and Frames.
Cambridge University Press.

Lichtenstein, S. and Slovic, P., editors (2006).
The Construction of Preference.
Cambridge University Press, 1 edition.

Bibliography IV

Meyer, P., Marichal, J.-L., and Bisdorff, R. (2008).
Disaggregation of bipolar-valued outranking relations.
In Le Thi, H. A., Bouvry, P., and Pham Dinh, T., editors, *Proc. of MCO'08 conference*, pages 204–213, Metz, France. Springer.

Mousseau, V., Dias, L., and Figueira, J. (2006).
Dealing with inconsistent judgments in multiple criteria sorting models.
4OR, 4(3):145–158.

Mousseau, V. and Słowiński, R. (1998).
Inferring an ELECTRE TRI model from assignment examples.
Journal of Global Optimization, 12(2):157–174.

Bibliography V

Zheng, J., Cailloux, O., and Mousseau, V. (2011).
Constrained multicriteria sorting method applied to portfolio selection.
In Brafman, R. I., Roberts, F. S., and Tsoukiàs, A., editors, *Algorithmic Decision Theory*, volume 6992 of *Lecture Notes in Computer Science*, pages 331–343, Rutgers, United States. Springer Berlin / Heidelberg.

max s s.t.

$$\begin{cases} \sum_{j \in J} w_j = 1 \\ l_j^{C_h} \leq l_j^{C_{h+1}} \end{cases}$$

Variables

$$\begin{cases} l_j^{C_h}, \forall j \in J, C_h \in \mathcal{C} \\ w_j, \forall j \in J \\ \lambda \\ b_j^{a,C_h} \text{ (binaries)} \\ v_j^{a,C_h} \\ n(a, C_h) \text{ (binaries)} \end{cases}$$

$$\begin{cases} \frac{(g_j(a) - l_j^{C_h}) + \varepsilon}{M} \leq b_j^{a,h} \leq \frac{g_j(a) - l_j^{C_h}}{M} + 1 \\ v_j^{a,C_h} \leq w_j;\ b_j^{a,C_h} + w_j - 1 \leq v_j^{a,C_h} \leq b_j^{a,C_h} \\ \sum_{j \in J} v_j^{a,C_h} \geq \lambda + s \quad \forall a \to h, h \geq 2 \\ \sum_{j \in J} v_j^{a,C_{h+1}} + s \leq \lambda - \varepsilon \quad \forall a \to h, h < k \\ n(a, C_h) \leq 1 + \sum_{j \in J} v_j^{a,C_h} - \lambda \\ n(a, C_h) \leq 1 + \lambda - \sum_{j \in J} v_j^{a,C_{h+1}} - \varepsilon \\ \sum_{1 \leq h \leq k} n(a, C_h) = 1 \\ \underline{n_h} \leq \sum_{a \in A} n(a, C_h) P(a) \leq \overline{n_h} \quad \forall \langle C_h, P, \underline{n_h}, \overline{n_h} \rangle \end{cases}$$

All constraints (with qualifiers)

max s s.t.

$$\begin{cases} \frac{(g_j(a) - l_j^{C_h}) + \varepsilon}{M} \leq b_j^{a,h} \leq \frac{g_j(a) - l_j^{C_h}}{M} + 1 & \forall j \in J, a \in A, h \geq 2 \\ v_j^{a,C_h} \leq w_j;\ b_j^{a,C_h} + w_j - 1 \leq v_j^{a,C_h} \leq b_j^{a,C_h} & \forall j \in J, a \in A, h \geq 2 \end{cases}$$

$$\begin{cases} \sum_{j \in J} w_j = 1 \\ l_j^{C_h} \leq l_j^{C_{h+1}} & \forall j \in J, C_h \in \mathcal{C} \end{cases}$$

$$\begin{cases} \sum_{j \in J} v_j^{a,C_h} \geq \lambda + s & \forall a \to h, h \geq 2 \\ \sum_{j \in J} v_j^{a,C_{h+1}} + s \leq \lambda - \varepsilon & \forall a \to h, h < k \end{cases}$$

Variables

$$\begin{cases} l_j^{C_h}, \forall j \in J, C_h \in \mathcal{C} \\ w_j, \forall j \in J; \lambda \\ b_j^{a,C_h} \text{ (binaries)};\ v_j^{a,C_h}; \\ \quad n(a, C_h) \text{ (binaries)}, \\ \quad \forall j \in J, a \in A, C_h \in \mathcal{C} \end{cases}$$

$$\begin{cases} n(a, C_h) \leq 1 + \sum_{j \in J} v_j^{a,C_h} - \lambda & \forall a \in A, h \geq 2 \\ n(a, C_h) \leq 1 + \lambda - \sum_{j \in J} v_j^{a,C_{h+1}} - \varepsilon & \forall a \in A, h \leq k-1 \\ \sum_{1 \leq h \leq k} n(a, C_h) = 1 & \forall a \in A \\ \underline{n_h} \leq \sum_{a \in A} n(a, C_h) P(a) \leq \overline{n_h} & \forall \left\langle C_h, P, \underline{n_h}, \overline{n_h} \right\rangle \end{cases}$$