

A framework for dynamic multiple-criteria decision making

Gianluca Campanella*, Rita A. Ribeiro

UNINOVA-CA3, Campus FCT-UNL, 2829-516 Caparica, Portugal

ARTICLE INFO

Article history:

Received 29 November 2010
Received in revised form 2 May 2011
Accepted 15 May 2011
Available online 18 May 2011

Keywords:

Dynamic decision making
Decision support systems
Multiple-criteria decision making

ABSTRACT

The classic multiple-criteria decision making (MCDM) model assumes that, when taking a decision, the decision maker has defined a fixed set of criteria and is presented with a clear picture of all available alternatives. The task then reduces to computing the score of each alternative, thus producing a ranking, and choosing the one that maximizes this value.

However, most real-world decisions take place in a dynamic environment, where the final decision is only taken at the end of some exploratory process. Exploration of the problem is often beneficial, in that it may unveil previously unconsidered alternatives or criteria, as well as render some of them unnecessary.

In this paper we introduce a flexible framework for dynamic MCDM, based on the classic model, that can be applied to any dynamic decision process and which is illustrated by means of a small helicopter landing example. In addition, we outline a number of possible applications in very diverse fields, to highlight its versatility.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Most real-world decision problems are dynamic, in the sense that the final decision is taken only at the end of some exploratory process, during which both alternatives and criteria may vary, as the examples of Section 6.1 testify.

However, the classic multiple-criteria decision making (MCDM) model is unable to capture this dynamicity, since it assumes that, before proceeding with the ranking, the decision maker must have identified fixed sets of criteria and alternatives. While, in principle, this model could be used in a dynamic setting by considering subsequent decisions to be completely independent one from the other, doing so would constitute a gross oversimplification of the way humans think about the fine interlinking that exists among decisions in a dynamic environment, in which earlier evaluations affect later ones.

The framework we propose in this paper aims to address this problem by extending the classic MCDM model in a flexible way that enables its use in very diverse fields requiring some form of dynamic decision making.

The rest of this paper is organized as follows. In Section 2, we briefly review the classic MCDM model and present the general theory our framework is set in. Subsequently, in Section 3, we delve into the crucial issue of choosing an appropriate aggregation function for this model, and present some well-known examples from the recent literature. We then give, in Section 4, a general overview of related work, before going into the details of our proposed framework in Section 5. To better illustrate our proposal, we make use of a numerical

example (Section 6) and present a number of possible applications (Section 6.1).

2. Classic MCDM model

The classic multiple-criteria decision making (MCDM) model [18,42] prescribes ways of evaluating, prioritizing and selecting the most favorable alternative from a set of available ones that are characterized by multiple, usually conflicting, levels of achievement for a set of attributes. The final decision is made by considering both inter-attribute and intra-attribute comparisons, possibly involving trade-off mechanisms.

Mathematically, a typical MCDM problem with m alternatives and n criteria is modeled by the matrix

$$\begin{matrix} & c_1 & c_2 & \dots & c_n \\ \begin{matrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{matrix} & \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix} & = & \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_m \end{bmatrix},
 \end{matrix}$$

where $x_{ij} \in [0, 1]$ represents the level of achievement of alternative a_i , $i = 1, \dots, m$ with respect to criterion c_j , $j = 1, \dots, n$, with 0 interpreted as “no satisfaction” and 1 corresponding to “complete satisfaction”. It is also common to introduce a weight vector $\mathbf{w} \in [0, 1]^n$, $\sum_{j=1}^n w_j = 1$ whose generic component w_j , $j = 1, \dots, n$ is the weight associated to criterion c_j representing its relative importance.

* Corresponding author.

E-mail address: gianluca@campanella.org (G. Campanella).

Evaluation of alternatives is performed by means of an aggregation function $f: [0, 1]^n \rightarrow [0, 1]$, which maps vectors of criteria values $x_i, i = 1, \dots, m$ to the $[0, 1]$ interval and satisfies, for all $\mathbf{x}, \mathbf{y} \in [0, 1]^n$,

$$\begin{cases} f(\underbrace{0, 0, \dots, 0}_{n \text{ times}}) = 0 \\ f(\underbrace{1, 1, \dots, 1}_{n \text{ times}}) = 1 \end{cases} \quad (\text{preservation of bounds}),$$

$$\mathbf{x} \leq \mathbf{y} \Rightarrow f(\mathbf{x}) \leq f(\mathbf{y}) \quad (\text{monotonicity}).$$

The resulting value is considered a score indicating how preferable the associated alternative is, with the common understanding that 0 corresponds to “no preference” and 1 to “strongest preference”. Given these scores, alternatives may then be ordered, thus producing a ranking, and the best one might be selected.

It is clear that the aggregation function chosen for distilling criteria values into a single score plays a crucial role in this model, which in turn means that its mathematical properties need to be better categorized and understood. For this reason, in the following section we will present some of the more commonly used aggregation functions, highlighting interesting properties and providing pointers to existing literature for the interested reader.

3. Aggregation functions

As we have seen in the previous section, the key component of the classic MCDM model is the aggregation function used to associate a single score to each alternative by distilling the different evaluations (one for each criterion). It is thus easy to understand that the mathematical properties of this function will have a direct impact on the produced values and, therefore, on the final ranking of alternatives.

In the rest of this section we will present well-known aggregation functions, highlighting interesting properties such as full or partial reinforcement [42], which might prove useful in the decision process.

For more information on the broad field of aggregation functions, as well as for identifying a set of general guidelines to help select one, the interested reader should refer to [40,2,10,44,7,33,3]; our exposition will broadly follow [3].

3.1. Averaging aggregation functions

Averaging aggregation functions are probably the most commonly used aggregation functions. An aggregation function f is averaging if, for every \mathbf{x} , $\min(\mathbf{x}) \leq f(\mathbf{x}) \leq \max(\mathbf{x})$.

A wide and well-known class of averaging aggregation functions is that of means, which includes the arithmetic, quasi-arithmetic, geometric, harmonic and power means, as well as their weighted counterparts. Another family of averaging aggregation functions, introduced by Yager [39] and especially popular in the fuzzy sets community, is that of Ordered Weighted Averaging functions (OWA), which associate weights to values rather than particular inputs.

When criteria cannot be considered preferentially independent, as is often the case, a natural choice for the aggregation function is the discrete Choquet integral [15–17], which is able to model the importance of single criteria as well as of subsets of criteria. Underlying the Choquet integral is a monotonic set function, called capacity [9], that plays a role similar to that of a weight vector in traditional weighted arithmetic means.

Another interesting approach that should be mentioned is that of mixture operators [26,24], which extend weighted averaging operators by considering weighting functions defined on the aggregation domain instead of constant weights. Depending on the type of weighting function used, one can for example penalize poorly satisfied attributes, and reward well-satisfied ones. Two kinds of

functions have been considered in this context, namely linear and quadratic weight generating functions [25,27].

Note, however, that these aggregation functions are in general not associative, and will thus not be the subject of further discussion in this review as they are not suited for the progressive aggregation process introduced later in this work.

3.2. Conjunctive aggregation functions

As their name implies, conjunctive aggregation functions are used to model conjunction, i.e. the logical *and*. They do not allow for compensation of low scores by other, higher scores, as it is the case, for example, of obtaining a driving license, for which one has to pass *both* the theory and the driving tests.

Therefore, their output is bound from above by the smallest input value, that is, for every \mathbf{x} , $f(\mathbf{x}) \leq \min(\mathbf{x})$.

3.2.1. Triangular norms

The prototypical example of a conjunctive aggregation function is the so-called triangular norm, or t-norm. It was first introduced by Menger [20] as an operation for the fusion of distribution functions on statistical metric spaces, and its current definition, due to Schweizer and Sklar [32], requires associativity, symmetry and neutral element 1.

Four basic examples of t-norms are the minimum, the product, the Łukasiewicz t-norm and the drastic product [3]. The weakest and the strongest t-norms are the drastic product and the minimum, respectively; for every \mathbf{x} and every t-norm T , it holds that $T_D(\mathbf{x}) \leq T(\mathbf{x}) \leq T_{\min}(\mathbf{x})$.

3.2.2. Parametric t-norms

Many families of related t-norms are defined by explicit formulas depending on some parameter. The main families of parametric t-norms are Hamacher's [44], Yager's [38] and Sugeno–Weber's [35], some of which include the basic t-norms as limiting cases.

3.3. Disjunctive aggregation functions

Disjunctive aggregation functions behave the opposite of conjunctive ones, in that satisfaction of *any* criteria is enough by itself, although positive inputs may reinforce one another. As their name implies, they are used to model disjunction, i.e. the logical *or*.

Therefore, their output is bound from below by the largest input value, that is, for every \mathbf{x} , $f(\mathbf{x}) \geq \max(\mathbf{x})$.

3.3.1. Triangular conorms

The dual aggregation function of a triangular norm is called a triangular conorm, or t-conorm. The current definition, again due to Schweizer and Sklar [32], requires associativity, symmetry and neutral element 0.

Four basic examples of t-conorms are the maximum, the probabilistic sum, the Łukasiewicz t-conorm and the drastic sum [3]. The weakest and the strongest t-conorms are the maximum and the drastic sum, respectively; for every \mathbf{x} and every t-conorm S , it holds that $S_{\max}(\mathbf{x}) \leq S(\mathbf{x}) \leq S_D(\mathbf{x})$.

3.3.2. Parametric t-conorms

As for t-norms, many families of related t-conorms are defined by explicit formulas depending on some parameter. The main families of parametric t-conorms are again Hamacher's [44], Yager's [38] and Sugeno–Weber's [35], some of which include the basic t-conorms as limiting cases.

3.4. Mixed aggregation functions

An aggregation function f is mixed if it does not belong to any of the classes hitherto presented, i.e. if it behaves differently on different parts of its domain. This kind of aggregation functions allow better modulation of the response; for example, in our context it might be especially useful to consider a positive (or upward) reinforcement behavior for high input values, a negative (or downward) reinforcement behavior for low ones, and perhaps an averaging behavior if some values are high and some are low.

Uninorms and nullnorms are two popular families of associative mixed aggregation functions built from triangular norms and conorms. A different approach, still close to t-norms and t-conorms, is adopted in the construction of compensatory T-S functions which, however, are not associative.

3.4.1. Uninorms

Uninorms [41,13] have conjunctive behavior when presented with low input values (below a given neutral element $e \in]0, 1[$), disjunctive behavior for high input values (above e) and averaging otherwise (Fig. 1a). The two extremal uninorms are shown in Fig. 2.

Any uninorm U with neutral element e is associated with a t-norm T_U and a t-conorm S_U (usually referred to as the underlying t-norm and t-conorm) such that

$$U(x,y) = \begin{cases} e T_U \left(\frac{x}{e}, \frac{y}{e} \right) & (x,y) \in [0, e]^2 \\ e + (1-e) S_U \left(\frac{x-e}{1-e}, \frac{y-e}{1-e} \right) & (x,y) \in [e, 1]^2 \end{cases}$$

Contrary to what happens in these two squares, the behavior of uninorms in the rest of the unit square is not bound to any specific class of (averaging) functions, which in turn led to the identification and characterization of several classes of uninorms.

3.4.2. FIMICA

FIMICA functions [42] are an interesting class of aggregation functions, derived from MICA functions [40], which exhibit full re-

inforcement and are thus well suited to the problem at hand. There exist two families of FIMICA functions, additive and multiplicative.

Given a fixed identity element $g \in [0, 1]$ and a monotonic mapping $f : \mathbb{R} \rightarrow [0, 1], x \geq y \Rightarrow f(x) \geq f(y)$, the family of additive FIMICA functions is defined as

$$M(\mathbf{x}) = f \left(\sum_{i=1}^n (x_i - g) \right)$$

Note that the additive family of FIMICA functions is, in general, not associative, though the argument of f is.

The family of multiplicative FIMICA functions is instead defined as

$$M(\mathbf{x}) = f \left(\prod_{i=1}^n \frac{x_i}{g} \right),$$

where $g > 0$ is the fixed identity element and f is as before. An example of a multiplicative FIMICA function is shown in Fig. 3.

It can be shown [42] that uninorms are FIMICA functions with two special properties:

1. idempotency: $M(\mathbf{x}) = \mathbf{x}$;
2. associativity.

However, FIMICA functions allow for a finer control of their response through the choice of an appropriate function f , making it possible, for example, to avoid undesired asymptotic behaviors (small changes in the argument resulting in huge differences in the aggregated value).

3.4.3. Nullnorms

Nullnorms are disjunctive for low input values (below a given absorbing element $a \in]0, 1[$), conjunctive for high input values (above a) and averaging otherwise (Fig. 1b). The two extremal nullnorms are shown in Fig. 4.

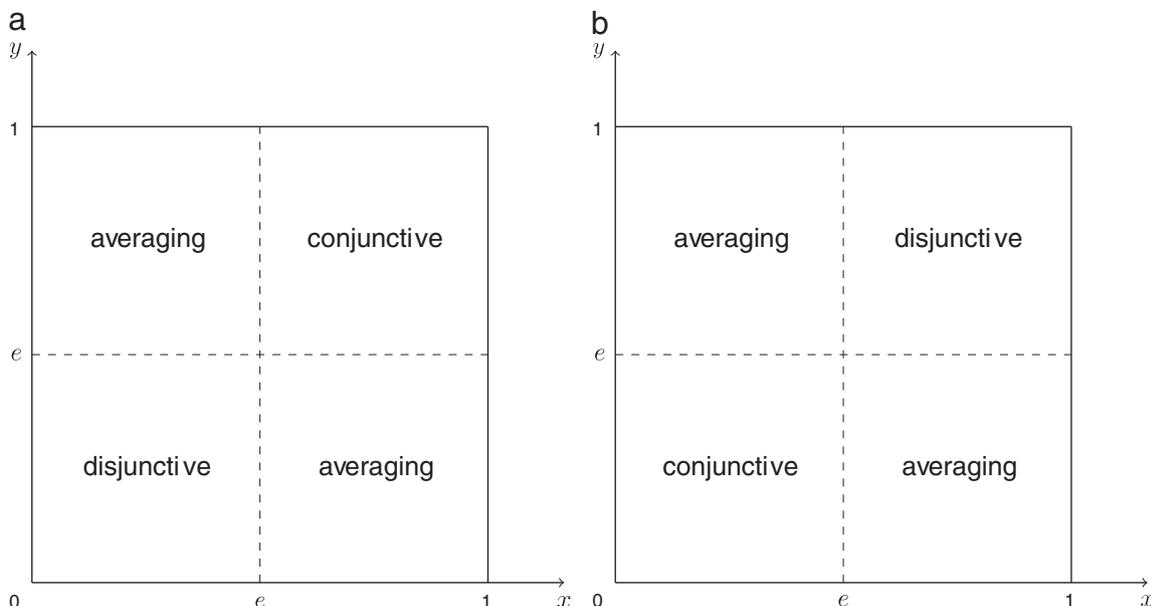
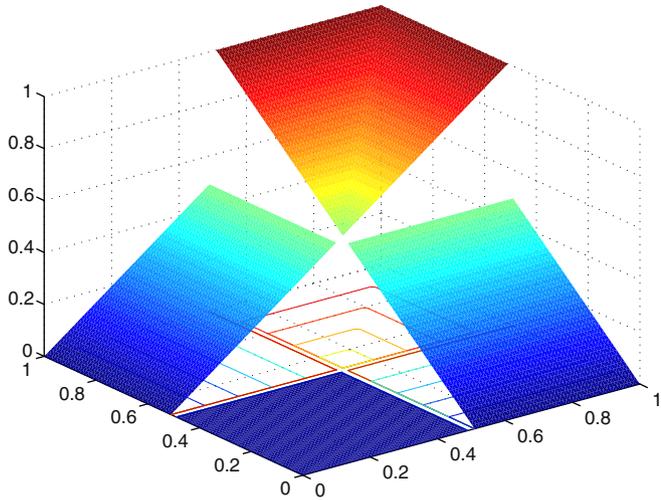


Fig. 1. Behavior of uninorms and nullnorms on the $[0, 1]$ interval: uninorms are conjunctive in the lower left rectangle and disjunctive in the upper right one; nullnorms behave in the opposite way.

a) Weakest uninorm



b) Strongest uninorm

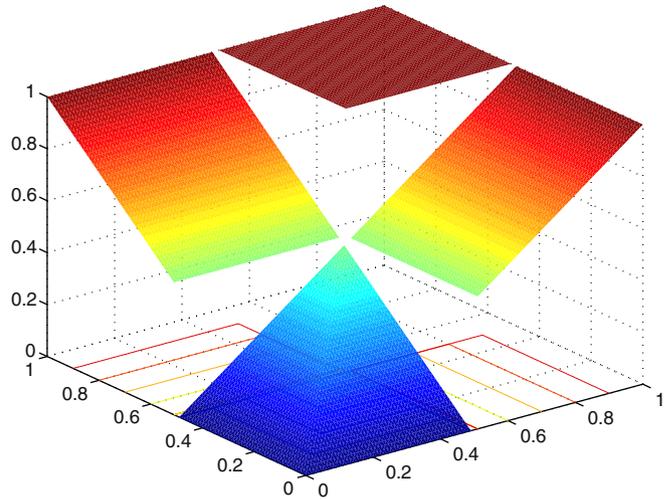


Fig. 2. Extremal uninorms: the weakest uninorm equals 0 in the lower left quadrant, *max* in the upper right one and *min* elsewhere; the strongest uninorm equals *min* in the lower left quadrant, 1 in the upper right one and *max* elsewhere.

Similarly to uninorms, nullnorms are averaging when dealing with mixed inputs, but their behavior in such cases is limited to a single value which coincides with the absorbing element *a*.

As for uninorms, any nullnorm *V* with absorbing element *a* is associated with a t-norm *T_V* and a t-conorm *S_V* such that

$$V(x,y) = \begin{cases} a S_V \left(\frac{x}{a}, \frac{y}{a} \right) & (x,y) \in [0, a]^2 \\ a + (1-a) T_V \left(\frac{x-a}{1-a}, \frac{y-a}{1-a} \right) & (x,y) \in [a, 1]^2 \\ a & \text{otherwise} \end{cases}$$

Therefore, similarly to uninorms, each nullnorm univocally defines a t-norm and a t-conorm. The converse (which is false in the case of uninorms), is true for nullnorms, that is, given an arbitrary t-norm, an

arbitrary t-conorm and an element $a \in]0, 1[$ there exists a unique associated nullnorm.

4. Overview of related work in dynamic MCDM

Having introduced the classic MCDM model as well as reviewed the most relevant aggregation functions, and before we move to our proposed extension, it is beneficial to give a brief overview of the current state of the art in the field of dynamic decision making, to allow a better understanding of where our framework finds its place.

The problem of making decisions in a dynamic environment has been the object of study in many different fields, but has received special consideration in psychology and cognitive science [4,34,12,5,14,36,37].

In the area of decision support and decision making, however, the problem is only starting to receive attention. A recently published work tries to integrate MCDM with the findings of several different fields, such as neural science, psychology and optimization theory [43], but does not propose a mathematical framework as we do in this work. Two other important developments are extensions to classic models [31,19], namely to the well-known Analytic Hierarchy Process (AHP) [29,30] and to the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) [18]. A similar tool, also based on AHP, is presented in [1], and incorporates user-specified probabilistic interactions between factors in the AHP hierarchy. A related problem, which was the object of the pioneering research of [21,22], is that of decision making with less than perfect information: in this case, it is often possible to acquire additional information at some cost, so that optimal information-gathering strategies, as well as optimal final decision strategies, are required. These issues continue to interest the decision making community, as attested by the recent paper by [8], in which the authors propose a framework that allows decision makers to provide incomplete preference values at multiple times.

A broad, interdisciplinary overview of tools that have been proposed in the past, such as expected and multi-attribute utility analyses, game theory, Bayesian approaches, decision trees and influence diagrams, stochastic optimal control theory, Markov decision processes, neural networks and rule-based cognitive architectures, can be found in [6].

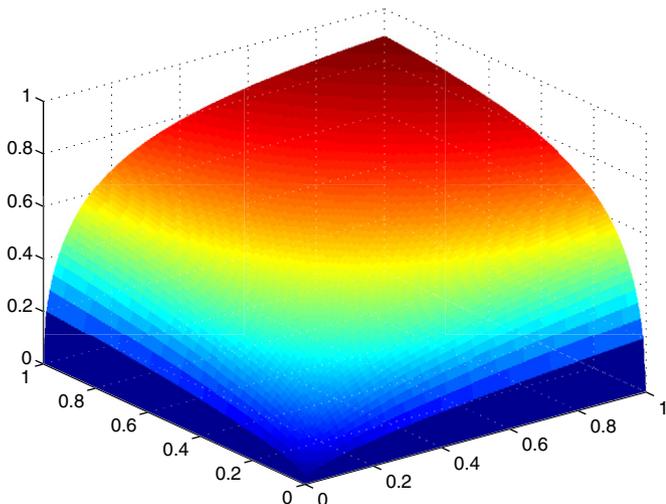
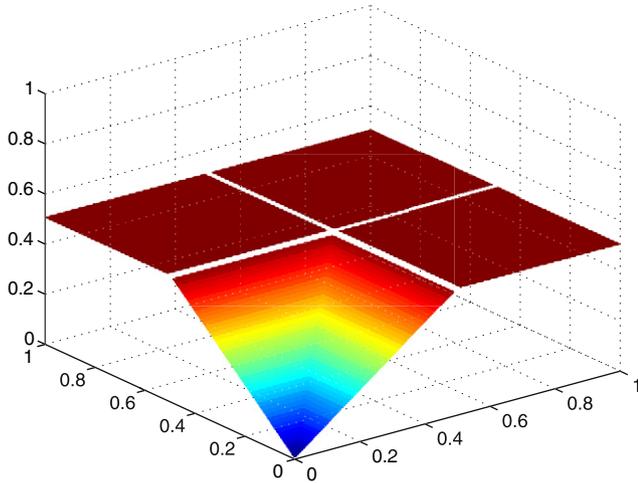


Fig. 3. Example of multiplicative FIMICA function using the function $f(x) = 1 - 1/(1+x)$ and the identity element $g = 0.25$.

a) Weakest nullnorm



b) Strongest nullnorm

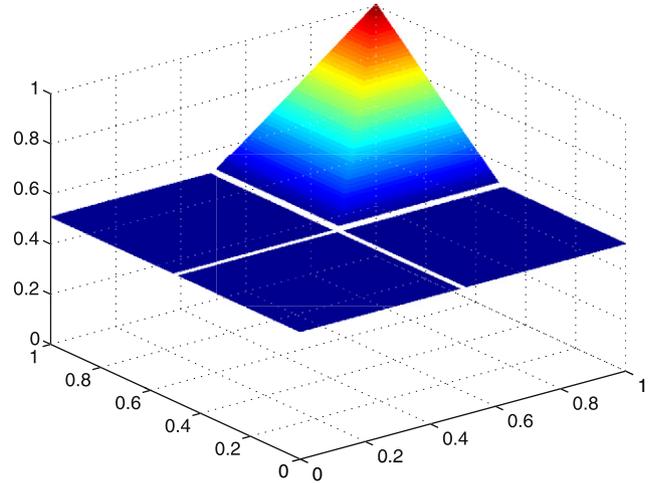


Fig. 4. Extremal nullnorms: the weakest nullnorm equals \max in the lower left and the neutral element a elsewhere; the strongest nullnorm equals \min in the upper right quadrant and a elsewhere.

Finally, it is crucial to understand that, although similar, our model is not aimed at solving consensus problems, since we consider a single decision maker taking multiple decisions over time, whereas consensus models deal with multiple decision makers reaching a common decision, nor is it related to optimal control systems, since our focus is not on process automation, but rather on supporting the decision making process.

5. Dynamic MCDM model

Given a (possibly infinite) set of positive time instants $\mathcal{T} = \{1, 2, \dots\}$, let \mathcal{A}_t denote the set of alternatives available at time $t \in \mathcal{T}$, $C_t: \mathcal{A}_t \rightarrow [0, 1]^n$ the function mapping each alternative to the corresponding vector of values for the n criteria over which alternatives are evaluated, and $\mathbf{w}_t \in [0, 1]^n$, $\sum_{w \in \mathbf{w}_t} w = 1, \forall t \in \mathcal{T}$ a weight vector for expressing criteria's relative importance. The set notation we have just introduced is used instead of the more common matrix notation presented in Section 2 because of the dynamicity of the problem, which makes it impossible to associate fixed positions to elements of a constantly changing set of alternatives.

The rating at each time $t \in \mathcal{T}$, denoted $R_t: \mathcal{A}_t \rightarrow [0, 1]$, is defined by the enclosed classic decision model based on C_t and (possibly) \mathbf{w}_t , and represents the (non-dynamic) aggregation of all criteria values, possibly taking into account their relative importance, for each available alternative.

The dynamic nature of the decision process is dealt with by means of a feedback mechanism, controlled by a second aggregation function ξ that makes use of a historical set of alternatives – its “memory”.

The historical set of alternatives is defined as

$$\mathcal{H}_0 = \emptyset \text{ and } \mathcal{H}_t \subseteq \bigcup_{\substack{t' \in \mathcal{T} \\ t' \leq t}} \mathcal{A}_{t'}, t \in \mathcal{T}. \quad (1)$$

Note that, in practical applications, it will be necessary to define the subset of alternatives to be included in the historical set and carried over to the next iteration by means of a selection rule, here called “retention policy”. This issue is discussed in more detail in Section 5.1.

Finally, the evaluation function $E_t: \mathcal{A}_t \cup \mathcal{H}_{t-1} \rightarrow [0, 1]$, $t \in \mathcal{T}$ is defined as

$$E_t(a) = \begin{cases} R_t(a) & a \in \mathcal{A}_t \setminus \mathcal{H}_{t-1} \\ D_E(E_{t-1}(a), R_t(a)) & a \in \mathcal{A}_t \cap \mathcal{H}_{t-1}, \\ E_{t-1}(a) & a \in \mathcal{H}_{t-1} \setminus \mathcal{A}_t \end{cases}, \quad (2)$$

where D_E is some aggregation function.

For each alternative $a \in \mathcal{A}_t \cup \mathcal{H}_{t-1}$, either belonging to the current set of alternatives \mathcal{A}_t or carried over from the previous one by means of the historical set \mathcal{H}_{t-1} , we have that:

1. if the alternative a belongs only to the current set of alternatives, but not to the historical set \mathcal{H}_{t-1} , its evaluation $E_t(a)$ is simply equal to its rating $R_t(a)$ as computed by means of the enclosed MCDM model;
2. if the alternative a belongs to both the current and the historical set of alternatives, $a \in \mathcal{A}_t \cap \mathcal{H}_{t-1}$, its evaluation is the aggregation (performed by the aggregation function D_E) of its evaluation in the previous iteration with its rating in the current one, again computed using the enclosed MCDM model;
3. finally, if the alternative a does not belong to the current set of alternatives \mathcal{A}_t , but was carried over in the historical set \mathcal{H}_{t-1} , its evaluation is also carried over from the previous iteration, $E_t(a) = E_{t-1}(a)$.

Fig. 5 depicts a diagram outlining the important steps in the proposed dynamic multiple-criteria decision making process.

To ensure that repeated pairwise application of the aggregation function $D_E: [0, 1]^2 \rightarrow [0, 1]$ will yield, at time t , the same result as application over the whole set of past values $E_{t'}, t' \in \{1, \dots, t\}$, we require it to be associative, i.e.

$$D_E(D_E(x, y), z) = D_E(x, D_E(y, z)), \forall x, y, z \in [0, 1].$$

This condition also allows us to perform this computation incrementally, without the need to store all past values.

Apart from this requirement, the choice of the aggregation function used in the dynamic part of the framework is completely independent from that of the function used to score alternatives in the non-dynamic part. As a consequence, any suitable aggregation

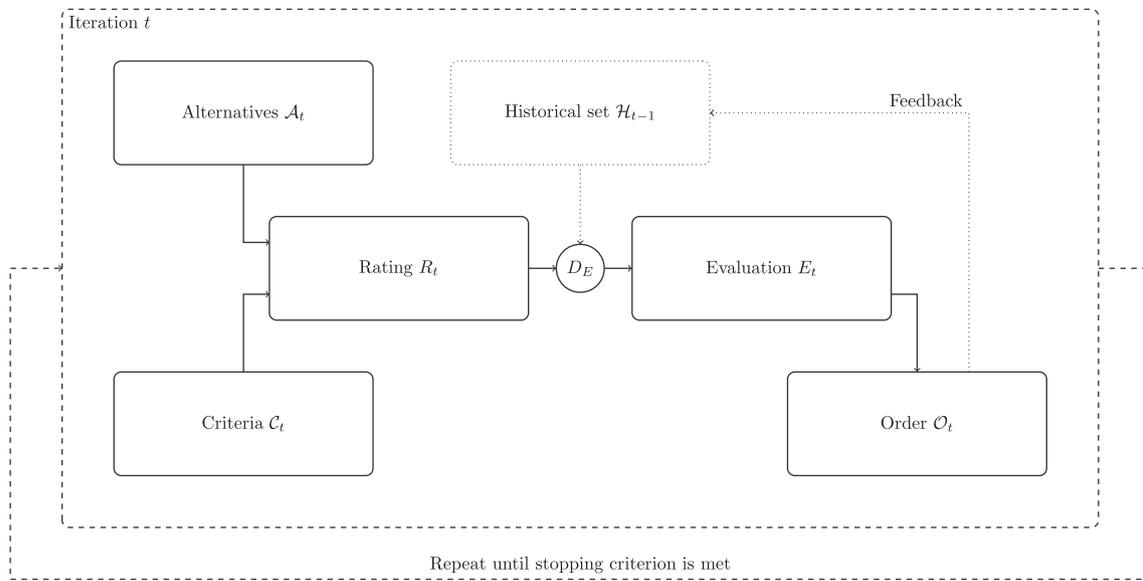


Fig. 5. Operations performed at each iteration t in the proposed dynamic framework: first, available alternatives and considered criteria are aggregated into utilities; then, by making use of the information stored in the historical set, a final ranking is produced; finally, the information in the historical set is updated and passed on to the next iteration. This process is repeated until the stopping criterion is met.

function, such as those presented in Section 3, can be used for D_E ; nonetheless, a property which we have found to be particularly relevant in this context is that of reinforcement.

5.1. Retention policy for the historical set

Another important constituent of our dynamic model is the historical set, whose function is to “remember” alternatives by carrying them over from iteration to iteration. As such, it is crucial to define a retention policy, i.e. a criterion for selecting a subset of past and current alternatives to be carried over.

We consider here three possibilities:

- accumulate alternatives, so that strict equality holds in (1);
- select the k alternatives which rank higher than all others;
- select all alternatives whose evaluation surpasses some threshold.

The first solution, while providing perfect “memory” of all past alternatives, is practically unimplementable in most situations, when we are faced with large (possibly infinite) sets of alternatives.

On the other hand, the second solution ensures that the cardinality of \mathcal{H}_t will never exceed k , but might potentially discard reasonable alternatives beyond the first k .

A good compromise is achieved by the third solution; note that the threshold might be fixed based on all current evaluations, for example by considering an appropriate quantile.

An additional strategy that might be used alongside is to drop alternatives that have not been available for a predefined number of iterations, effectively “forgetting” about them.

Finally, we observe that, if $\mathcal{H}_t \equiv \forall t \in T$, our framework reduces to applying the enclosed MCDM model independently at each instant t .

5.2. Stopping criterion

The dynamic decision process may or may not have an end, i.e. a final decision moment after which no further support is needed.

For example, in the case of the emergency department operation scheme described in Section 6.1, it is unclear if such an endpoint can

be marked as patients come and leave; on the other hand, the examples of medical diagnosis and planetary landing site selection do have such an endpoint: for the former, it would be determined by the doctor using the system, while for the latter it is imposed by the very nature of the decision being made.

Therefore, the issue of identifying a suitable stopping criterion, if any, depends on the specific problem being solved: in some situations it will not be needed, in others it will be imposed exogenously (as is the case in planetary landing), or met only when the decision maker feels confident enough to declare the decision process over.

6. Numerical example: helicopter landing

To better illustrate our model, let us introduce a simple numerical example in which a decision maker has to pick a site for landing an helicopter from among a set of nine possible choices.

We shall consider the following criteria:

Effort the effort required to change route towards the site (for example, an estimate of the required amount of fuel).

Roughness the (estimated) roughness of the terrain.

Slope the (estimated) slope of the terrain.

Sunlight the (estimated) availability of sunlight at the site.

For the sake of simplicity, we will consider criteria values to be already normalized, and aggregate them using a weighted average with weight vector

$$\mathbf{w} = \begin{bmatrix} \text{Effort} & \text{Roughness} & \text{Slope} & \text{Sunlight} \\ 0.1 & 0.2 & 0.3 & 0.4 \end{bmatrix}.$$

Clearly, any other aggregation function may be used, as long as it is able to rank alternatives on a numeric scale.

As for the aggregation function of the dynamic part, we will use a simple probabilistic sum (hence, a t-conorm exhibiting upward

reinforcement); finally, the historical set will be used to carry over the best $k = 3$ alternatives.

6.1. First iteration

We consider an initial set containing only the first eight alternatives, $\mathcal{A}_1 = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8\}$, with the rationale that alternative a_9 was vetoed by some criteria. The criteria values and corresponding utilities are as follows:

	Effort	Roughness	Slope	Sunlight	$U_1 = E_1$
a_1	0.2	0.7	0.5	0.7	0.59
a_2	0.7	0.4	0.0	0.8	0.47
a_3	0.8	0.7	0.6	0.4	0.56
a_4	0.1	0.6	0.2	0.2	0.27
a_5	0.3	0.3	0.3	0.8	0.50
a_6	0.7	0.3	0.2	0.5	0.39
a_7	0.6	0.3	0.8	0.2	0.44
a_8	1.0	0.3	0.3	0.2	0.33

In the first iteration all evaluations and utilities correspond, since there is no historical information available; so, for example, the first aggregated value is computed as

$$R_1(a_1) = \mathbf{w}^T C_1(a_1) = 0.1 \times 0.2 + 0.2 \times 0.7 + 0.3 \times 0.5 + 0.4 \times 0.7 = 0.59.$$

Therefore, the initial order \mathcal{O}_1 is $a_1 < a_3 < a_5 < a_2 < a_7 < a_6 < a_8 < a_4$ and the alternatives carried over to the next iteration are the best $k = 3$, $\mathcal{H}_1 = \{a_1, a_3, a_5\}$.

6.2. Second iteration

In the second iteration we consider a set with seven alternatives $\mathcal{A}_2 = \{a_3, a_4, a_5, a_6, a_7, a_8, a_9\}$, obtained by:

- removing alternatives a_1 and a_2 from \mathcal{A}_1 (for example, because they were vetoed; note that the fact that an alternative was included in the historical set at the end of a previous iteration does not imply its availability in the next one);
- considering a modified alternative a_3 with different values for the last two criteria;
- taking into consideration alternative a_9 .

The criteria values for each alternative and the corresponding utilities are:

	Effort	Roughness	Slope	Sunlight	U_2
a_3	0.8	0.7	0.3	0.2	0.39
a_4	0.1	0.6	0.2	0.2	0.27
a_5	0.3	0.3	0.3	0.8	0.50
a_6	0.7	0.3	0.2	0.5	0.39
a_7	0.6	0.3	0.8	0.2	0.44
a_8	1.0	0.3	0.3	0.2	0.33
a_9	0.2	0.3	0.5	1.0	0.63

In the case of a_3 and a_5 , a further step is required to determine the corresponding evaluation, namely the aggregation with the historical value. As an example, for a_3 we have

$$E_2(a_3) = E_1(a_3) + R_2(a_3) - E_1(a_3) R_2(a_3) = 0.56 + 0.39 - 0.56 \times 0.39 = 0.73.$$

Therefore, at the end of the second iteration the evaluations are:

	a_3	a_4	a_5	a_6	a_7	a_8	a_9
E_2	0.73	0.27	0.75	0.39	0.44	0.33	0.63

and the associated ranking \mathcal{O}_2 is $a_5 < a_3 < a_9 < a_7 < a_6 < a_8 < a_4$ with $\mathcal{H}_2 = \{a_3, a_5, a_9\}$. Note that, while not considered for decision purposes, the value associated to a_1 , one of the alternatives that have gone missing from the previous iteration, is retained since it was present in the historical set.

6.3. Third iteration

In the third and final iteration we consider a set with five alternatives $\mathcal{A}_3 = \{a_2, a_6, a_7, a_8, a_9\}$, obtained by:

- removing alternative a_3, a_4 and a_5 from \mathcal{A}_2 ;
- considering a modified alternative a_9 with different values for the first two criteria;
- adding back alternative a_2 with different values for the second and third criteria.

The criteria values for each alternative and the corresponding utilities are:

	Effort	Roughness	Slope	Sunlight	U_3
a_2	0.7	0.6	0.5	0.8	0.66
a_6	0.7	0.3	0.2	0.5	0.39
a_7	0.6	0.3	0.8	0.2	0.44
a_8	1.0	0.3	0.3	0.2	0.33
a_9	0.0	0.8	0.5	1.0	0.71

Therefore, at the end of the third iteration the evaluations are:

	a_2	a_6	a_7	a_8	a_9
E_3	0.66	0.39	0.44	0.33	0.89

and the associated ranking \mathcal{O}_3 is $a_9 < a_2 < a_7 < a_6 < a_8$.

Note again that the values associated to a_3 and a_5 , two of the alternatives that have gone missing from the previous iteration, would be retained for use in the next iteration since they were present in the historical set.

6.4. Discussion of results

The best ranked alternative available in the last iteration is a_9 , and this represents the final decision since the decision process necessarily stops after landing. If at all possible, however, the decision maker might in fact opt to wait another iteration before committing to some decision.

At any rate, it is clear that our model is able to guide and support the decision maker as the decision process evolves over time, and thus offers added value over usage of the enclosed MCDM model in a non-dynamic fashion.

7. Potential applications

To illustrate the applicability of our model, we present in this section a number of very diverse contexts it could find use in. We are confident that, with due modifications, the model could be applied in even more situations, and are looking forward to further developments in this area.

7.1. Emergency department operation

As patients arrive at an emergency department, it is necessary to prioritize their cases based on clinical need and available resources, a process known as triage.

In this context, patients to be treated with higher priority would rank higher, with evaluations being computed from a set of criteria describing the severity of the case. Every time a new patient could be treated, the health professional would be able to quickly select the one who would benefit the most, removing it from the queue, while new patients would arrive, thus making the decision process dynamic. As we saw, our model is perfectly suited to deal with this kind of process, and is able to provide the needed ranking of patients at each iteration.

In a similar way, our model could also be used to effectively administer transplant waiting lists, or any other kind of prioritized queue, as it has been proven in a preliminary computer simulation.

A possible parametrization for this problem would be the following:

Alternatives all waiting patients.

Criteria body temperature, blood pressure, presence of bruises, burns, fractures, etc.

Retention policy keep all patients that have not been treated yet.

7.2. Medical diagnosis

A typical diagnostic process starts by gathering information directly from the patient and through physical examination in order to formulate a hypothesis of likely diseases, which will later be confirmed through further medical testing before providing treatment.

This process can be easily seen as a multiple-criteria decision making problem, in which possible diseases represent alternatives, and signs, symptoms and test results constitute the set of criteria.

Note that, in this case, both alternatives and criteria may change in time: the former because the diagnostician might consider diseases that look more likely given the available data, as well as discard others that fell below some likelihood threshold; the latter due to the fact that, as time passes, new test results will be available based on previous choices.

As such, we believe that our model is well suited to this area, and could be used both to guide the diagnosis, by indicating at each time which diseases are most probable, thus allowing the diagnostician to arrange appropriate tests to confirm or discredit the hypothesis, and to arrive at a solid final decision.

A possible parametrization for this problem would be the following:

Alternatives diseases under consideration.

Criteria clinical tests performed.

Retention policy keep all likely diseases, getting rid of those that are deemed too unlikely by the decision maker.

7.3. Planetary landing site selection

One of the most challenging research directions in space exploration is the development of autonomous hazard avoidance systems that would allow safe landing on dangerous or insufficiently characterized areas on distant planets, for which manual piloting is not an option due to communication delays.

A central component of these systems is clearly the selection of a suitable landing site, based on hazard mapping data that is continuously collected as the lander approaches the planet's surface.

Our model has already proven to be very effective in this area [11,23], making use of the continuous flow of data to provide, at each iteration, a ranking of possible landing sites. In particular [28], the aggregation functions considered were a hybrid function based on uninorms, using the Hamacher t-norm and t-conorm, and Yager's OWA function [39] in the averaging parts of its domain, as well as additive and multiplicative FIMICA functions [42] with appropriate underlying functions that ensured continuity in the $[0, 1]$ interval.

The parametrization chosen for this problem was:

Alternatives landing sites (represented by their coordinates on a hazard map).

Criteria estimated available sunlight, terrain slope and roughness, etc.

Retention policy keep a subset of the best k sites found so far.

7.4. Supplier selection

Many businesses rely on external suppliers for some of their operations, and often establish relationships with a number of possible suppliers.

In this scenario, every time the possibility to subcontract some work arises, it is necessary to understand whether it is convenient to do so and, in this case, which supplier should be selected.

Understandably, this selection would need to take into account previous choices made for similar decisions, thus allowing, for example, the formation of stronger collaborations with suppliers that score consistently higher than others, which is why our model is well suited to this type of problem.

A possible parametrization for this problem would be the following:

Alternatives suppliers under consideration.

Criteria estimated delivery time, previous reliability, price etc.

Retention policy keep all suppliers that have met quality standards and delivered on time in the past three months.

8. Conclusions

In this paper we introduced a versatile model for decision making in a dynamic environment, in which both sets of alternatives and criteria undergo modifications as the problem is further explored in a number of iterations. In this context, decisions may then be taken either frequently, or just at the end of the process.

The proposed framework can be flexibly adapted to different situations by choosing a retention policy for the historical set of alternatives, i.e. a rule for selecting alternatives to be "remembered", and an aggregation function, which is used to compute the final evaluations starting from scores and information in the historical set.

We shall continue our investigation on modeling dynamic decision processes, as well as understand to what extent our framework could be applied to the problem of consensus, and to what degree it could be integrated with optimal control systems.

Other interesting extensions that we aim at investigating include the application of aggregation functions not only to the final, aggregated rating, but also to values of some criteria that are known to be fixed, as well as finding ways to handle missing criteria values in some iterations.

References

- [1] A.B. Badiru, P.S. Pulat, M. Kang, DDM: decision support system for hierarchical dynamic decision making, *Decision Support Systems* 10 (1993) 1–18.
- [2] G. Beliakov, J. Warren, Appropriate choice of aggregation operators in fuzzy decision support systems, *IEEE Transactions on Fuzzy Systems* 9 (2001) 773–784.

- [3] G. Beliakov, A. Pradera, T. Calvo, *Aggregation Functions: A Guide for Practitioners*, volume 221 of *Studies in Fuzziness and Soft Computing*, Springer, 2008.
- [4] B. Brehmer, Dynamic decision making: Human control of complex systems, *Acta Psychologica* 81 (1992) 211–241.
- [5] J.R. Busemeyer, Dynamic decision making, in: N.J. Smelser, P.B. Baltes (Eds.), *International Encyclopedia of the Social & Behavioral Sciences*, Pergamon, 2001, pp. 3903–3908.
- [6] J.R. Busemeyer, T.J. Pleskac, Theoretical tools for understanding and aiding dynamic decision making, *Journal of Mathematical Psychology* 53 (2009) 126–138.
- [7] *Aggregation Operators: New Trends and Applications*, volume 97 of *Studies in Fuzziness and Soft Computing*, in: T. Calvo, G. Mayor, R. Mesiar (Eds.), *Physica-Verlag*, 2002.
- [8] Y.L. Chen, L.C. Cheng, An approach to group ranking decisions in a dynamic environment, *Decision Support Systems* 48 (2010) 622–634.
- [9] G. Choquet, Theory of capacities, *Annales de l'Institut Fourier* 5 (1954) 131–295.
- [10] M. Detyniecki, Numerical Aggregation Operators: State of the Art, in: *International Summer School on Aggregation Operators and their Applications*, Asturias, Spain, 2001.
- [11] Y. Devouassoux, S. Reynaud, G. Jonniaux, R.A. Ribeiro, T.C. Pais, Hazard avoidance developments for planetary exploration, 7th International ESA Conference on Guidance, Navigation & Control Systems, GNC, 2008.
- [12] A. Diederich, Dynamic stochastic models for decision making under time constraints, *Journal of Mathematical Psychology* 41 (1997) 260–274.
- [13] J.C. Fodor, R.R. Yager, A. Rybalov, Structure of uninorms, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 5 (1997) 411–427.
- [14] C. Gonzalez, Decision support for real-time, dynamic decision-making tasks, *Organizational Behavior and Human Decision Processes* 96 (2005) 142–154.
- [15] M. Grabisch, Fuzzy integral in multicriteria decision making, *Fuzzy Sets and Systems* 69 (1995) 279–298.
- [16] M. Grabisch, The application of fuzzy integrals in multicriteria decision making, *European Journal of Operational Research* 89 (1996) 445–456.
- [17] M. Grabisch, C. Labreuche, A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid, 4OR: A Quarterly Journal of Operations Research 6 (2008) 1–44.
- [18] Multiple Attribute Decision Making: An Introduction, Quantitative Applications in the Social Sciences, in: K.P.Y.C.L. Hwang (Ed.), Sage Publications, Inc, 1995.
- [19] Y.H. Lin, P.C. Lee, H.J. Ting, Dynamic multi-attribute decision making model with grey number evaluations, *Expert Systems with Applications* 35 (2008) 1638–1644.
- [20] K. Menger, Statistical metrics, *Proceedings of the National Academy of Sciences of the United States of America*, 28, 1942, pp. 535–537.
- [21] J.C. Moore, A.B. Whinston, A model of decision-making with sequential information-acquisition (Part 1), *Decision Support Systems* 2 (1986) 285–307.
- [22] J.C. Moore, A.B. Whinston, A model of decision-making with sequential information-acquisition (Part 2), *Decision Support Systems* 3 (1987) 47–72.
- [23] T.C. Pais, R.A. Ribeiro, Y. Devouassoux, S. Reynaud, Dynamic ranking algorithm for landing site selection, in: L. Magdalena, M. Ojeda-Aciego, J.L. Verdegay (Eds.), *Proceedings of the Twelfth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, pp. 608–613.
- [24] R.A.M. Pereira, The orness of mixture operators: the exponential case, in: U.P. de Madrid (Ed.), *Proceedings of the 8th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU 2000)*, Madrid, Spain, pp. 974–978.
- [25] R.A.M. Pereira, R.A. Ribeiro, Aggregation with generalized mixture operators using weighting functions, *Fuzzy Sets and Systems* 137 (2003) 43–58.
- [26] R.A.M. Pereira, G. Pasi, On non-monotonic aggregation: mixture operators, in: *Proceedings of the 4th Meeting of the EURO Working Group on Fuzzy Sets (EUROFUSE'99) and 2nd International Conference on Soft and Intelligent Computing*, Budapest, Hungary, 1999.
- [27] R.A. Ribeiro, R.A.M. Pereira, Generalized mixture operators using weighting functions: a comparative study with WA and OWA, *European Journal of Operational Research* 145 (2003) 329–342.
- [28] R.A. Ribeiro, T.C. Pais, L.F. Simões, Benefits of Full-Reinforcement Operators for Spacecraft Target Landing, volume 257 of *Studies in Fuzziness and Soft Computing*, Springer, 2010.
- [29] T.L. Saaty, *Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World*, volume 2 of *Analytic Hierarchy Process Series*, third edition RWS Publications, 1999.
- [30] T.L. Saaty, *Fundamentals of Decision Making and Priority Theory With the Analytic Hierarchy Process*, volume 6 of *Analytic Hierarchy Process Series*, first edition RWS Publications, 2000.
- [31] T.L. Saaty, Time dependent decision-making; dynamic priorities in the AHP/ANP: generalizing from points to functions and from real to complex variables, *Mathematical and Computer Modelling* 46 (2007) 860–891.
- [32] B. Schweizer, A. Sklar, *Probabilistic Metric Spaces*, Dover Publications, 2005.
- [33] V. Torra, Y. Narukawa, *Modeling decisions: information fusion and aggregation operators*, Cognitive Technologies, Springer, 2007.
- [34] J.T. Townsend, J. Busemeyer, *Dynamic representation of decision-making*, *Dynamic representation of decision-making*, The MIT Press, 1995, pp. 101–120.
- [35] S. Weber, A general concept of fuzzy connectives, negations and implications based on t-norms and t-conorms, *Fuzzy Sets and Systems* 11 (1983) 103–113.
- [36] M. Wittmann, M.P. Paulus, Decision making, impulsivity and time perception, *Trends in Cognitive Sciences* 12 (2008) 7–12.
- [37] M. Wittmann, M.P. Paulus, Temporal horizons in decision making, *Journal of Neuroscience, Psychology, and Economics* 2 (2009) 1–11.
- [38] R.R. Yager, On a general class of fuzzy connectives, *Fuzzy Sets and Systems* 4 (1980) 235–242.
- [39] R.R. Yager, On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Transactions on Systems, Man, and Cybernetics* 18 (1988) 183–190.
- [40] R.R. Yager, Aggregation operators and fuzzy systems modeling, *Fuzzy Sets and Systems* 67 (1994) 129–145.
- [41] R.R. Yager, A. Rybalov, Uninorm aggregation operators, *Fuzzy Sets and Systems* 80 (1996) 111–120.
- [42] R.R. Yager, A. Rybalov, Full reinforcement operators in aggregation techniques, *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics* 28 (1998) 757–769.
- [43] P.L. Yu, Y.C. Chen, Dynamic multiple criteria decision making in changeable spaces: from habitual domains to innovation dynamics, *Annals of Operations Research* (2010) 1–20.
- [44] H.J. Zimmermann, *Fuzzy Set Theory – And Its Applications*, fourth edition Springer, 2001.

Gianluca Campanella is currently pursuing his Master of Science degree in Applied Mathematics at the Universidade Nova de Lisboa, Portugal after graduating with maximum grades in Applied Computer Science at the Free University of Bolzano, Italy. He is currently working as a researcher at the CA3 research group of the research institution UNINOVA, under the supervision of Prof. Rita A. Ribeiro. His research interests span a variety of topics in operations research, fuzzy systems, decision support and complex networks. E-mail gianluca@campanella.org. Web page <http://www.campanella.org>.

Rita A. Ribeiro is Associate Professor in the Department of Electronic and Computer Engineering of the Universidade Nova de Lisboa, Portugal. She is also coordinator and senior researcher of the CA3 research group, part of the research institution UNINOVA. Her research interests include fuzzy multiple-criteria decision making, decision support systems, fuzzy inference systems, fuzzy optimization and their applications to real-world problems. E-mail rar@uninova.pt. Web page <http://www.ca3-uninova.org>.