AGGREGATION RULES IN COMMITTEE PROCEDURES

Javier Montero* and Vincenzo Catelio**
*Faculty of Mathematics
Complutense University
Madrid, Spain
**Department of Mathematics
University of Catania
Catania, Italy

Abstract

Very often, decision procedures in a committee compromise potential manipulations by taking into account the ordered profile of qualifications. It is therefore rejected the standard assumption of an underlying associative binary connective allowing the evaluation of arbitrary finite sequences of items by means of a one-by-one sequential process. In the paper we develop a mathematical approach for non-associative connectives allowing a sequential definition by means of binary fuzzy connectives. It will be then stressed that a connective rule should be understood as a consistent sequence of binary connective operators. Committees should precisely decide about which connective rule they will be considering, not just about a single operator.

Keywords: Fuzzy Connectives, Fuzzy Sets, Aggregation Operators.

1 Introduction.

Many real-life problems are solved by means of some information aggregation procedures. If we observe chains of partial information and we have to elaborate a global opinion we have an aggregation process which can be modeled in terms of aggregation rules. In principle, the only property which is required is just the ability to transform the data set into a simpler representation. For
example, if we expect data with a fixed and known dimension $n$, we do not need to define how $n+1$ or $n-1$ items of information have to be aggregated into one single index. Nevertheless, in practice we rarely know in advance the dimension of the real problem we are going to be faced to. Moreover, even if we do know the dimension of the input data, we have the semantic problem of using the same aggregation operator everywhere this is the case, on inputs of different dimensions. Therefore, any aggregation rule has to be able to operate with an arbitrary dimension of the data set, even when we assume that such a data set has nice mathematical properties like homogeneity and non-redundancy information.

When a committee has to decide about the qualification of candidates, the objective is just to aggregate the opinion of each committee member into a single index. Each committee member represents a piece of information. Quite often, the number of voting members may be not fixed (e.g., unavoidable last minute absences), or the number of voting members can not be a priori known (this is the case when we just have a qualitative definition of the small set of voters, i.e., the list of properties giving people the right to vote). Moreover, if the number of potential voters is too big, existence of an algorithm allowing some kind of sequential reckoning will help calculations.

A standard solution to such a basic sequential and operational problem is the assumption of Associativity. Under Associativity, we can aggregate information by pieces, item by item, by applying a unique binary connexion, no matter the dimension of our data set. The aggregated value can be obtained by aggregating information by means of a sequential one-by-one process. The aggregation process is fully characterized by a single binary aggregation rule, the one applied to the first couple of aggregated elements. Then the same composition is applied iteratively until the data set has been completely read. This is the case with classical fuzzy connective operators for conjunction and disjunction, t-norms and t-conorms (see e.g., [7]).

Associativity is a quite frequent assumption in group decision making models. Associative decision procedures are not affected by the order in which individuals express their opinion, and every aggregated opinion is considered just another individual opinion. However, important operators are not associative.

Some decision procedures in a committee sometimes do not take into account committee highest and lowest qualifications for each candidate, and then they evaluate the mean of the remaining middle values. This is a particular case of Ordered Weighted Averaging (OWA) operators, introduced by Yager [21] in order to fill the gap between min (which is the maximal t-norm) and max (which is the minimal t-conorm). OWA operators are not associative, and their application requires that the number of items to be aggregated has been previously fixed.

Classical Mean Rule, for example, takes into account social support of each alternative in such way that group opinions are no longer equivalent to individual opinions (see, e.g., [22]). Notice that a key OWA connective operator like the standard mean, defined as a mapping

$$M_a : [0, 1]^n \rightarrow [0, 1]$$

such that

$$M_a(a_1, \ldots, a_n) = \frac{1}{n} \sum_{i=1}^{n} a_i$$

is not associative. Each mean $M_n$ is just the mean of $n$ numbers (it has been defined for a fixed $n$). $M_a$ is just an operator, not a rule. When we refer to the Mean Rule we refer to the rule that evaluates the above mean for every $n$. The Mean Rule is not a single mapping $M_n$, but the complete sequence $\{M_n\}$ of all those mappings.

Moreover, it should be also pointed out that an operational calculus for the Mean Rule would not follow the above formula, but a left recursive calculus

$$M_n(a_1, \ldots, a_n) = \frac{(n-1)M_{n-1}(a_1, \ldots, a_{n-1}) + a_n}{n}$$

or alternatively, a right recursive calculus

$$M_n(a_1, \ldots, a_n) = \frac{a_1 + (n-1)M_{n-1}(a_2, \ldots, a_n)}{n}$$

where $M_1(a, b) = (a + b)/2$ (see [5, 13] for a discussion on some critical and computational issues).

In general, it is not so easy to talk about OWA rules. Although several interesting families of OWA operators have been introduced in the past, showing the great flexibility in the choice of types of OWA operators (see, e.g., [13, 14, 15]), not every family of OWA operators can be properly considered as a rule.

In this paper we generalize the arguments introduced by the authors in [2], where each OWA rule was represented in terms of a family of binary OWA operators (see also [3, 4, 5]). Any connective rule will be considered as a consistent family of connectives capable of solving arbitrary dimension problems. As a consequence, it is claimed that committees should always previously fix their coherent connective (aggregation) rule.

2 Recursive connective rules.

A connective rule should allow an aggregated value for any possible dimension of the list of items to be aggregated. That is, a connective rule should be a sequence of connective operators

$$\{\phi_n : [0, 1]^n \rightarrow [0, 1]\}_{n \geq 1}$$

to be used to aggregate any finite number of items. We shall focus our attention here on those connective rules which allow the aggregation of arbitrary lists in
a recursive manner. In particular, we shall consider some families of connective operators that can be defined by means of a left or a right recursive application of binary operators, once an appropriate rearrangement of the items to be aggregated has been previously realized. In this way, a connective rule should be understood as a family of connective operators which can be recursively evaluated.

Obviously, in order to be considered as a rule, some consistency assumption has to be imposed on the family of connectives. Not every family of connectives either defines a connective rule or can be considered to consist of, or allow a recursive definition.

DEFINITION 1 An ordering rule $\pi$ on a set of items $A, A \neq \emptyset$, is a family of permutations $\pi_\alpha : B \rightarrow D$

such that for every finite sequence of items

$B = (a_1, a_2, \ldots, a_n), \ a_i \in A \forall i,$

a new sequence is defined on $B$ in such a way that

$\pi_\alpha(a_i) < \pi_\alpha(b) \Rightarrow \pi(a_i) < \pi(b)$

whenever $a, b \in B \cap C$.

An ordering rule tells us the exact position each new element will placed in, any previously given ordered set of items.

An immediate example of ordering rule is the natural decreasing order of real numbers

$\sigma : R \rightarrow R$

which assigns to each list of numbers $(a_1, \ldots, a_n)$ its sorting permutation

$\sigma(a_1, \ldots, a_n) = (a_{\sigma(1)}, \ldots, a_{\sigma(n)})$

such that $a_{\sigma(i)} \geq a_{\sigma(j)}$ for all $i \leq j$.

DEFINITION 2 A left-recursive connective rule is a family of connective operators

$[\phi_n : [0, 1]^n \rightarrow [0, 1]]_{n \geq 1}$

such that there exists a sequence of binary operators

$[\theta_n : [0, 1]^2 \rightarrow [0, 1]]_{n \geq 1}$

verifying

$\phi_0(a_1, a_2) = \theta_1(a_1, \pi(a_2))$ and

$\phi_n(a_1, \ldots, a_n) = \theta_n(\phi_{n-1}(a_1, \ldots, \pi(a_{n-1})), \pi(a_n))$

for some ordering rule $\pi$.

Right-recursive can be analogously defined.

DEFINITION 3 A collection of connective operators

$[\psi_n : [0, 1]^n \rightarrow [0, 1]]_{n \geq 1}$

is said to be a right-recursive connective rule whenever

$\psi_0(a_1, a_2) = \rho_0(\pi(a_1), \pi(a_2))$

and

$\psi_n(a_1, \ldots, a_n) = \rho_n(\psi(a_1), \psi_{n-1}(a_2, \ldots, \pi(a_n)))$

hold for some family of binary operators

$[\rho_n : [0, 1]^2 \rightarrow [0, 1]]_{n \geq 1}$

and some ordering rule $\pi$.

We immediately have that an operator $\phi_n : [0, 1]^n \rightarrow [0, 1]$ can be right-recursive defined if and only if it can be left-recursive defined. This can be seen as follows. Let

$\rho_n(\pi(a_1), \pi(a_2), \ldots, \pi(a_n)) = \psi_n(a_1, \ldots, a_n)$

Define the permutation $\pi$ in such a way that for any $\pi(n)

$\pi(a) < \pi(b) \Rightarrow \pi(x(a)) < \pi(x(b))$

Moreover, let $\theta_n(a, b) = \rho_n(\pi(a), \pi(b))$ for all $a, b$. Then we have

(a) $\theta_n(\pi(a_1), \pi(a_2)) = \rho_n(\pi(a_1), \pi(a_2))$

(b) $\phi_n(a_1, a_2, a_3) = \rho_n(\pi(a_1), \pi(a_2), \pi(a_3))$

(c) In general, we can see by induction that

$\eta_n(\phi_1(a_1), \ldots, \phi_{n-1}(a_1, \ldots, \pi(a_{n-1})), \pi(a_n))$

The ordering rule $\pi$ is to be known as the dual ordering rule of $\pi$.

The existence of a right (left) recursion representation of a given operator does not imply in general the existence of an equivalent left (right) recursion representation by means of the same underlying ordering rule (see next section for an example). Of course, some rules will allow the one-side recursive definition (see also next section).

In some cases, such a recursive representation of a connective rule is fixed from the underlying ordering rule, as shown in the following result.
THEOREM 1: Let
\[ \{a_n \mid 1 \leq n \leq 10\} \]
be a left-recursive abductive rule with respect to the ordering rule \( L \), such that \( L_0(a, x) = \emptyset \) and \( L_1(x, x) = 1 \), with \( a \) continuous and strictly increasing in each coordinate, for all \( a \). Then \( \{a_n\}_{n=1} \) is unique in its range for each ordering rule \( L \) such that
\[ a_n(a_1, \ldots, a_n) = L_n(a, \{a_1, \ldots, a_n\}) . \]
(A similar version holds for right recursion)

Proof: First of all, notice that \( L_n(0, 0) = \emptyset \) for all \( n \). In fact, since \( L_2(0, 0) = 0 \), it follows that
\[ a_3(0, 0, 0) = L_3(0, 0, 0) = L_3(0, 0) \]
and so on.

Analogously, \( L_n(1, 1) = 1 \) for all \( n \). Moreover, every \( L_n \) is assumed to be strictly increasing. It is direct for \( n = 2 \), and if we assume it is true for \( n = 1 \), then we have

1. If \( \emptyset = \{a_1, \ldots, a_n\} \) increases and the other values remain constant, then \( \{a_1, \ldots, a_n\} \) increases, and therefore \( \{a_1, \ldots, a_n\} \) also increases.
2. If \( \emptyset = \{a_1, \ldots, a_n\} \) increases, due to continuity of \( a_{n+1} \) and the induction hypothesis it is assumed, the existence of a point \( \{a_1, \ldots, a_n\} \) such that \( \{a_1, \ldots, a_n\} \) is greater than \( \{a_1, \ldots, a_n\} \) for all \( n \), with some strict inequality, such that
\[ L_n(\ldots, L_3(\{a_1, \ldots, a_n\})) \]
\[ \{a_1, \ldots, a_n\} \]
such that \( \{a_1, \ldots, a_n\} \leq \{a_1, \ldots, a_n\} \) for all \( n \), with some strict inequality, such that
\[ L_n(\ldots, L_3(\{a_1, \ldots, a_n\})) \]
\[ \{a_1, \ldots, a_n\} \]
and therefore \( L_n \) has increased as well.

Hence, for each \( a \) there is at most one \( c \) such that \( L_n(c, a) = b \) (it is unique whenever the pair \( (a, b) \) belongs to the range of \( L_n \), and therefore \( c \leq c \) holds). That is, from
\[ a_n(a_1, \ldots, a_n) = L_n(a, \{a_1, \ldots, a_n\}) \]
we can evaluate
\[ c = L_n(\ldots, L_3(\{a_1, \ldots, a_n\})) \]
successively, from this equation we can evaluate
\[ L_n(\ldots, L_3(\{a_1, \ldots, a_n\})) \]
and so on, whenever we stay in the real range.

Thus, if a recursive abductive rule contains a continuous strictly increasing operator of dimension \( n \), then consistent operators of lower dimension can be obtained according to the above result. Obviously, consistent upper dimension operators cannot be freely chosen since for all \( n \)
\[ a_n(a_1, \ldots, a_n) = L_n(a, \{a_1, \ldots, a_n\}) \]
and, analogously,
\[ a_n(a_1, \ldots, a_n) = R_n(a, \{a_1, \ldots, a_n\}) \]

An interesting case to analyze is the one in which left and right recursions share the same underlying ordering rule. That is, when
\[ L_n(\ldots, L_3(\{a_1, \ldots, a_n\})) \]

hols for some ordering rule \( L_n \).

DEFINITION 2: If both left and right recursiveness hold for the same ordering rule then we have recursive rules.

In this way, recursion generalizes the concept of associativity, in the same sense that recursive rules are the ones that can be evaluated iteratively (both sides), after an appropriate pre-arrangement of data. The ability of being iteratively evaluated is in fact the deep reason for associativity in practice. An operational calculus algorithm usually implies an iterative reducing. But this iterative calculus does not necessarily require a unique binary operator. As shown above, the Mean Rule shows both left and right recursive definitions, although it is not associative.

The Mean Rule verifies an additional property: both left and right recursive definitions do not depend on the permutation, i.e., they are the same no matter the particular sequence of permutations being chosen. Left and right recursion hold for any possible ordering rule. If such a condition holds, we can talk about commutative recursive rules. Commutative recursive rules will be those connective rules which do not depend on any particular ordering rule.

In some way, we could say that a connective rule \( \{a_n\}_{n=1} \) is recursive if and only if a set of general associativity equations (in the sense of Max [9]) hold for each \( n \), once the items have been properly ordered. In fact, recursiveness holds whenever
\[ a_n(a_1, \ldots, a_n) = L_n(a, \{a_1, \ldots, a_n\}) = R_n(\{a_1, \ldots, a_n\}) \]
for all $n$ and some ordering rule $\sigma$. If each one of these binary connective $L_n, R_n$ is assumed to be defined in the cartesian product of two nontrivial compact intervals on the real line, being continuous strictly increasing in one coordinate, then it can be shown (see [9]) that they are commutative and basically additive, in such a way that

$$\phi_n(a_1, \ldots, a_n) = \phi_n^{-1}(\phi_1(a_1) + \cdots + \phi_n(a_n))$$

for some homomorphisms in the unit interval $[0, \phi_1 \cdots \phi_n]$. This result allows a particular representation of theorem 1. If we take, for example, the natural increasing order $\sigma$ as the underlying ordering rule, then each $L_n$ is defined on a simplex $S_n = S_2 = S_3 = R_n \forall n$ (that is, the whole recursive connective rule is characterized by a unique associative binary connective $F$, with no pre-arrangement of data).

Many connective rules $\phi_2, \ldots, \phi_n$ we can find in the literature are defined by means of a unique commutative and associative binary operator $\phi : [0, 1]^2 \to [0, 1]$ such that

$$\phi_n(a_1, \ldots, a_n) = \phi_n(\phi_2(a_1, b_2), \ldots, b_n) = \phi_n(b_1, \ldots, b_n - 1, \phi_2(b_n - 1, b_n))$$

for $(b_1, \ldots, b_n)$ any permutation of $(a_1, \ldots, a_n)$. When we refer to a $t$-norm or a $t$-conorm as a connective rule we really mean the family of connective operators in such a way univocally defined (only one binary connective not depending on the ordering rule). The whole family of connective operators is fully characterized by its first connective operator of dimension 2, and no pre-arrangement of data is needed.

3 OWA recursive rules.

As pointed out above, sometimes only one underlying ordering rule is allowed by the decision maker. Perhaps there is only one natural way of ranking our data, and data reach to us previously pre-arranged according to such an ordering rule. If this is the case, the above concepts should be modified in order to meet such a restriction. Either a recursive definition is consistent with such an ordering rule, or such a recursive definition can not be applied. Either both recursive definitions make use of such an ordering rule, or it can not be applied as a recursive rule. For example, it may be the case that data are assumed to be ranked in its natural increasing ordering, as happens with OWA rules.

Let us particularize the above ideas to the OWA case. First, we remind the reader some key concepts about OWA operators.

### 3.1 Basic on OWA operators.

**OWA operators** [12] are based upon the natural (decreasing) ordering. An OWA operator of dimension $n$ is a connective operator

$$\phi : [0, 1]^n \to [0, 1]$$

such that for any list $(a_1, \ldots, a_n)$ then

$$\phi(a_1, \ldots, a_n) = \sum_{i=1}^n \omega_i a_i$$

for some associated list of weights $W = (\omega_1, \ldots, \omega_n)$ such that

1. $\omega_i \in [0, 1]$ for all $1 \leq i \leq n$
2. $\sum_{i=1}^n \omega_i = 1$

OWA operators are therefore assuming the (decreasing) natural ordering on the real line as the underlying ordering rule $\sigma$.

OWA operators are obviously commutative, monotone and idempotent, but as pointed out above, not associative in general. In fact, a binary $\phi_2 (a = 2)$ OWA operator is associative if and only if it is the mean ($\omega_2 = 1$) operator or the max ($\omega_2 = 1$) operator. Therefore, given an OWA operator of dimension $n$, it can be only applied to aggregation problems of such a dimension $n$. If the dimension problem is modified, such OWA operator can not be applied.

Three short comments about OWA operators before going back to our operationality problem:

- A significant measure associated with OWA operators is the operator which estimates how close an OWA operator is to the mean operator. It is defined as

$$\text{closeness}(\phi) = \frac{1}{n-1} \sum_{i=1}^n (\phi_i - \phi)$$

Dual to the measure of cerness is the measure of andness defined as

$$\text{andness}(\phi) = 1 - \text{closeness}(\phi)$$

which therefore measures how close an OWA operator is to the min operator.

- Another important notion is duality. Given an OWA operator $\phi$ with weights $W = (\omega_1, \ldots, \omega_n)$, the dual $\phi$ of $\phi$ is the OWA operator whose weights are $[\omega_n, \ldots, \omega_1]$. It is not difficult to see that cerness($\phi$) = andness($\phi$).
A particular class of OWA operator is given by the frequency measure. They are OWA operators that verify the property \( w_i \geq w_j \) if \( i < j \). Any frequency measure \( f \) is such that \( f(\Phi) \geq \frac{1}{n} \).

It may be the case that the existing left and right recursive definitions do not make use of the same underlying ordering rule. For example, the following two OWA operators

\[
\phi(a_1, a_2) = \frac{1}{4} a_1 + \frac{1}{2} a_2
\]

and

\[
\phi(a_1, a_2, a_3) = \frac{1}{4} a_1 + \frac{1}{2} a_2 + \frac{1}{4} a_3
\]

allow a left recursive rule, since we can write

\[
\phi(a_1, a_2, a_3) = \frac{1}{2} \phi(a_1, a_2) + \frac{1}{2} a_3
\]

But once we have chosen such a decreasing natural ordering as our ordering rule, then there is no function \( f : [0, 1]^2 \rightarrow [0, 1] \) such that

\[
\phi(a_1, a_2) = f(a_{\Phi} - a_2) + \frac{1}{2} a_2
\]

Hence, they can not be together in the same left recursive rule.

Moreover, not every family of OWA operators allows some one-side recursive definition based upon the same natural ordering rule. For example, no left or right recursive rule can be defined if we take \( f_1 = f_2 \) as above, but

\[
\phi_2(a_1, a_2, a_3) = \frac{1}{4} a_1 + \frac{1}{2} a_2 + \frac{1}{4} a_3
\]

3.2 OWA recursive rules.

Although the standard associative procedure can not be considered when dealing with OWA operators, it may be the case that a recursive analysis can be applied to the decreasing ordered list \((a_1, \ldots, a_n)\). Thus, practical OWA aggregation problems where the number of values to be aggregated is not previously known, should be solved by choosing one of these consistent recursive families of OWA operators, by means of such a natural ordering rule. Each one of these families solves every aggregation problem for any arbitrary size of the input.

**Definition 5** A recursive OWA rule is a recursive convolution rule of OWA operators allowing left and right recursive definitions based upon the natural decreasing ordering rule, by means of binary OWA rules.

This recursive structure has the advantage that aggregation weights can be computed quickly by using a dynamic programming approach (see [1]). Such a recursive structure should not be confused with the sorted linkage property, considered in [6] in order to characterize OWA operators.

Anyway, we can check that once an OWA operator of dimension \( n \) has been fixed, all OWA operators of lower dimension belonging to its right and left OWA rules are almost uniquely defined. In fact, it will be shown that every OWA operator can be recursively defined, both left and right, since the values to be aggregated have been pre-aggregated according to the natural order in the real line. These two recursive representations will be basically unique.

**Theorem 2** Let us consider a fixed OWA operator \( \phi \) of dimension \( n \). Then there exist at least one family of \( n - 1 \) OWA operators of dimension 2

\[
L_2, \ldots, L_n
\]

and another family of \( n - 1 \) OWA operators all of them also of dimension 2

\[
R_2, \ldots, R_n
\]

allowing a left recursion and a right recursion, respectively, in such a way that

\[
\phi(a_1, \ldots, a_n)
\]

is equivalent to

\[
L_2(L_{n-1}(\ldots L_2(L_2(a_1, a_2, a_3), a_4), \ldots), a_n)
\]

and

\[
R_n(a_{1n}, R_{n-1}(a_{1n}, R_{n-2}(a_{1n}, \ldots, R_2(a_{1n}, a_2), a_3), \ldots), a_n)
\]

Moreover, each one of these binary OWA operators is either unique or it can be freely chosen.

**Proof:** Let us assume right recursion, for example. If

\[
R_n(a_1, a_2) = (1 - f(n)) a_1 + f(n) a_2
\]

Then,

\[
\phi(a_1, \ldots, a_n) = \sum_{i=1}^{n} w_i a_i = (1 - f(n)) a_1 + f(n) a_n
\]

where

\[
\Phi_2 = \Phi_{n-1}(a_2, \ldots, R_n(a_{n-1}, a_n), R_{n-2}(a_{n-2}, a_{n-1}), \ldots)
\]

in such a way that

\[
f(n) = 1 - \Phi_1
\]

Hence, if we now assume

\[
R_{n-1}(a_1, a_2) = (1 - f(n-1)) a_1 + f(n-1) a_2
\]

and another family of \( n - 1 \) OWA operators all of them also of dimension 2

\[
L_2, \ldots, L_n
\]
it must be

$$
\phi(x_1, \ldots, x_n) = (1 - f(n))a_{m1} + f(n)[1 - f(m-1)]a_{m2} + f(m-1)a_{m3}
$$

where

$$
b_m = R_{m-2}(a_{p1}, \ldots, R_2(a_{p1}, \ldots, R_2(a_{p1}, \ldots, R_2(a_{p1}, \ldots, a_{11}, a_{12}, \ldots, a_{1n})) \ldots )
$$

in such a way that

$$
f(n-1) = 1 - \frac{w_2}{1 - \sum_{j=1}^{n} w_j}
$$

whenever $w_i \neq 1$. In case $w_i = 1$, it is the max rule, and the remaining binary OWA operators are not relevant at all. The process continues till we reach the trivial case $R_2$ in particular, in each step we obtain

$$
f(n-1) = 1 - \frac{w_2}{1 - \sum_{j=1}^{n} w_j}
$$

whenever $\sum_{j=1}^{n} w_j < 1$ (otherwise, the remaining operators will be not relevant at all).

Notice that right recursion is unique for the min rule ($w_n = 1$), and left recursion will be unique for the max rule ($w_1 = 1$). In case $w_i \neq 0$ for all $i$, our OWA operator would be strictly increasing in each coordinate and theorem 1 would apply. The above result proves that in fact every OWA operator allows both left and right recursive definitions. OWA rules as considerned in this paper will consistently allow the recursive definition of each one of its operators. In other words, our OWA rules will be given by a sequence of OWA operators that can be explained in terms of a sequence of binary OWA operators allowing its right or left recursive representation. It is therefore natural to characterize each recursive OWA rule by means of the sequence of weights associated to its right or left recursive representation (see [2]).

**Definition 6** A basis function is any mapping $f$ that to any integer $n$ associates a number in the unit interval (that is, $f(n) \in [0,1]$ for all $n$) with $f(1) = 1$.

Each basis function $f$ will then allow the recursive definition of two families of OWA operators. For any $n \geq 2$, we can define $L_n$ and $R_n$, such that

$$
L_n(b_1, b_2) = (1 - f(n))a_{m1} + f(n)a_{m2}
$$

and

$$
R_n(b_1, b_2) = f(n)a_{m1} + (1 - f(n))a_{m2}
$$

Then any left recursive operation

$$
\phi(x_1, \ldots, x_n) = (1 - f(n))a_{m1} + f(n)[1 - f(m-1)]a_{m2} + f(m-1)a_{m3}
$$

is defined as

$$
\phi(x_1, \ldots, x_n) = L_n(\phi(x_{n-1}, \ldots, x_1), a_{m1}, a_{m2}, a_{m3})
$$

or

$$
\phi(x_1, \ldots, x_n) = R_n(\phi(x_{n-1}, \ldots, x_1), a_{m1}, a_{m2}, a_{m3})
$$

and any right recursive operation

$$
\phi(x_1, \ldots, x_n) = R_n(\phi(x_{n-1}, \ldots, x_1), a_{m1}, a_{m2}, a_{m3})
$$

will always lead to OWA operators, for every $n \geq 2$, as it will be shown below. Each one of these two families of OWA operators $\phi = \{\phi_\phi, \ldots, \phi_{\phi} \}$ if obtained via left-recursion call and $\phi = \{\phi_{\phi}, \ldots, \phi_{\phi} \}$ if obtained via right-recursion call, will be then associated to the basis function $f$. According to the last theorem, left recursive (LR) and right recursive (RR) families of OWA operators will be defined, in particular, as follows:

- $n$ is the dimension of the OWA operators $\phi_n$ and $\phi_n$.
- The weights of $\phi_n$ are denoted by $w_{n,1}, \ldots, w_{n,n}$ and $w_{n,1}, \ldots, w_{n,n}$ will denote the weights of $\phi_n$.
- For every $n \geq 2$ and every $i = 1, 2, \ldots, n$ we define

$$
w_{n,i} = \begin{cases} f(n) & \text{if } i = n \\ (1 - f(n))w_{n-1,i} & \text{if } i < n \end{cases}
$$

- For every $n \geq 2$ and every $i = 1, 2, \ldots, n$ we define

$$
w_{n,i} = \begin{cases} f(n) & \text{if } i = n \\ (1 - f(n))w_{n,i-1} & \text{if } i > 1 \end{cases}
$$

Therefore,

- $w_{n,i} = f(i)\prod_{j=1}^{i-1}(1 - f(j))$ for every $n \geq 2$ and every $i = 1, 2, \ldots, n$.
- $w_{n,i} = f(n-i+1)\prod_{j=i+1}^{n}(1 - f(j))$ for every $n \geq 2$ and every $i = 1, 2, \ldots, n$.

In view of the above equations it is immediate to check that $\phi_n$ and $\phi_n$ are in fact OWA operators, since for any $n \geq 2$ we have

$$
\sum_{i=1}^{n} w_{n,i} = \sum_{i=1}^{n} w_{n,i} = 1
$$

It is also easy to check now that not every family $\{\phi_1, \ldots, \phi_n \}$ of OWA operators can be recursively defined by means of binary OWA operators on the basis of the decreasing natural ordering. Recursive consistency can be easily characterized by means of the weights of the OWA operators. For example, if $w_{n,i} = 0$, in order to be able to provide a left recursive characterization it must also be $w_{n+1,i+1} = 0$. Analogously, $w_{n+1,i+1} = 0$ must hold for a right recursive definition, whenever $w_{n,i} = 0$.

From theorem 2 it is implied that $\phi_n$ of dimension $n$ being fixed, these all left-recursive and right recursive operators OWA operators with any dimension $\phi_1, \ldots, \phi_{n-1}$ are uniquely defined. More in general, we have the following result, which also gives a formal characterization of recursive consistency for OWA rules.
THEOREM 5 Let us consider a family of OWA operators \( \{ \phi_1, \ldots, \phi_n \} \).

Then it can be defined by LR (i.e., it is LR consistent) if and only if \( \phi_{i+1} = \phi_i \) for all \( i = 1, 2, \ldots, n \). Analogously, such a family of OWA operators can be defined by RR (i.e., it is RR consistent) if and only if \( \phi_i = \phi_{i+1} \) for all \( i = 1, 2, \ldots, n \).

Proof: Direct since every weight of the OWA operator of dimension \( k \) is multiplied by the same weight of the next binary OWA operator in order to allow the OWA operator of dimension \( k + 1 \). In case of right recursion, for example,

\[
\sum_{i=1}^{k+1} w_{i+k+1} = f(k+1) a_{k+1} + (1 - f(k+1)) \sum_{i=1}^{k} w_{i+k+1}
\]

Hence,

\[
\frac{w_{i+k+1}}{w_{i+k}} = 1 - f(k+1)
\]

for all \( i = 1, 2, \ldots, k \).

Therefore, once any \( \phi_n \) has been chosen, clear restrictions are implied in order to obtain families of OWA operators which are consistent with \( \phi_n \), both with respect to left recursion and right recursion. But if a left (right) recursion exists, the associated LR (RR) basis function is basically unique. Thus, each basis function is characterizing a LR (RR) consistent family of OWA operators. Moreover, it has been already pointed out above that right (left) recursive consistency for a given family \( \Phi = \{ \phi_1, \ldots, \phi_n \} \) of OWA operators does not imply left (right) recursive consistency.

4. Some examples.

We will now provide some interesting examples of recursive families of OWA operators.

4.1 Mean Rule.

The following result characterizes the Mean Rule as a recursive OWA rule.

THEOREM 6 Let \( f \) be a basis function with associated LR and RR families of OWA operators being identical (i.e., \( \Phi = \Phi' \), that is, \( \phi_n = \phi'_n \) for all \( n \)). Then it must be \( f(n+1) = 1/n \) for all \( n \), and in turn the weights of each \( \phi_n \) are \( w_{i,n+1} = 1/n \) for all \( i = 1, 2, \ldots, n \).

Proof: We shall prove it by induction, just for the RR case. The result is obvious for \( n = 2 \). Let us assume \( w_{i,n} = 1/n \) for all \( i = 1, 2, \ldots, n \). Then it must be

\[
w_{i,n+1} = f(n+1) = (1 - f(n+1)) w_{i,n} = (1 - f(n+1)) 1/n
\]

in such a way that \( f(n+1) = 1/n \). Therefore, since

\[f(n+1) = 1 - \frac{w_{i,n+1}}{w_{i,n}}\]

we obtain

\[w_{i,n+1} = (1 - f(n+1)) w_{i,n} = \frac{1}{n+1}\]

4.2 Constant basis function.

Together with the Mean Rule, another case a priori deserving our attention are those rule characterized by a constant basis function (i.e., there exists a value \( a \in [0, 1] \) such that \( f(n) = a \) for all \( n \geq 2 \)).

In case \( f(n) = a \) for all \( n \geq 2 \), each LR OWA operator \( \phi_n \) will have weights

\[w_{i,n} = (1-a)^{n-1}\]

for all \( i = 2, \ldots, n \). Analogously, weights for each RR OWA operator \( \phi'_n \) will be

\[w_{i,n} = (1-a)^{n-1}\]

for all \( i = 2, \ldots, n-1 \).

When \( n = 1 \) (i.e., 0) in left (right) recursion we obtain the min rule, and in right (left) recursion we obtain the max rule.
4.3 Harmonic OWA operators.

We recall that the $n$-th harmonic number is

$$H_n = \sum_{i=1}^{n} \frac{1}{i}.$$  

Harmonic OWA operators are obtained by taking

$$f(n) = \frac{1}{H_n}.$$  

Thus,

$$1 - f(n) = \frac{H_{n-1}}{H_n}.$$  

By using theorem 5 it is immediate to see that the family of $LR-H$-Harmonic OWA operators is a class of harmonic measures. For example, its first OWA operators will have the following weights:

$$w_{1,2} = \frac{2}{3}, w_{2,2} = \frac{1}{3},$$  

$$w_{1,3} = \frac{6}{11}, w_{2,3} = \frac{3}{11}, w_{3,3} = \frac{2}{11},$$  

$$w_{1,4} = \frac{12}{25}, w_{2,4} = \frac{6}{25}, w_{3,4} = \frac{4}{25}, w_{4,4} = \frac{3}{25}.$$  

4.4 A monotone fuzzy quantifier.

In [12, 13], it is shown how to obtain the evaluation of monotone fuzzy quantifiers by means of OWA operators. In particular, given a monotone non-decreasing fuzzy quantifier $Q$ such that $Q(0) = 0$ and $Q(1) = 1$, the weights $w_{i,n}$ for $i = 1, 2, ..., n$ of an OWA operator of dimension $n$ to evaluate $Q$ are defined as

$$w_{i,n} = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right).$$  

In case $Q(a) = a^r$ for some $r > 0$ we obtain that taking

$$f(n) = 1 - Q\left(\frac{n-1}{n}\right)$$

for all $n$, the associated left-recursive family of OWA operators does verify such a property. Hence, such a monotone fuzzy quantifier allows a left-recursive definition. But it cannot be right-recursive defined.

5 Final Comments.

OWA rules do play a main role in group decision making, since many aggregation procedures in practice are just particular cases. This paper generalizes previous results obtained just for OWA operators in [2]. A general approach to non-associative connective rules allowing an operational definition has been proposed. By operational we understand the ability of a recursive one by one evaluation, on the basis of a previous re-arrangement of the data set.

As a consequence, it has been stressed the fact that a connective rule, in order to be considered a rule, should be able to deal with any arbitrary number of items. An OWA operator is just an operator as the mean of $n$ numbers. None of them are connective rules, just single connectives. Considerably many real life decision problems require at different times to aggregate (possibly very large) lists of inputs of different dimensions. Connective rules have to be defined before knowing such a list. A connective rule is in general a rule allowing aggregation of any list, no matter its dimension.

Connective rules have been conceived here as consistent families of connective operators allowing a representation in terms of right or left recursion of binary connective operators. Associativity is just an easy way of meaning such an operational representation.

Obviously, there are families of OWA operators that represent rules in the sense that they allow the evaluation of any arbitrary number of items, not allowing the recursive approach as developed in this paper, but being consistent in some other alternative sense. This is the case, for example, of the Binarized OWA rule $\{\alpha, (a_1, ..., a_n)\}_{\alpha\leq 1}$ where each $\alpha$ is an OWA operator of dimension $n$ with weights

$$w_{i,n} = \left(\frac{n-i}{n}\right)^{\alpha-1}(1-n^{\alpha-1}) w_i = \frac{n-1}{n}$$

for some fixed $\alpha \in (0, 1)$. Each one of these operators can be recursively defined, but the family itself does not verify the recursive OWA rule condition given in definition 5, neither the more general recursiveness definition 4. An operative description of this family of OWA operators, still by means of a sequence of binary OWA operators and the natural decreasing ordering, can be based upon the ordered linkage property of OWA operators (see [6]).

Acknowledgement: This research has been partially supported by DGICYT of Spain (grant PB95-0471) and Complutense University (grant PB94/75-0145).
References


Table of Contents

Preface ix

1. BASIC ISSUES IN AGGREGATION
   Kolmogorov's theorem and its impact on soft computing
   H.T. Nguyen and V. Kreinovich 3
   Possibility and necessity in weighted aggregation
   C. Carlson, P. Ful hár and S. Ful hár 18
   OWA operators and an extension of the contrast model
   B. Bouchez-Maunoury and M. Rifqi 29
   Equivalence of changes in proportions at crossroads of
   mathematical theories
   J. Aczél, G. Rúst and J. Schweiger 36

2. FUNDAMENTAL ASPECTS OF OWA OPERATORS
   On the inclusion of importances in OWA aggregation
   R.R. Yager 41
   On the linguistic OWA operator and extensions
   F. Herrera and E. Herrera-Viedma 60
   Alternative representations of OWA operators
   M. Gómez-Chova 73

3. MATHEMATICAL ISSUES AND OWA OPERATORS
   Useful tools for aggregation procedures: some consequen-
   ces and applications of Širáň's measurable Hahn-Banach theorem
   H.J. Skala 80
THE ORDERED WEIGHTED AVERAGING OPERATORS

Theory and Applications

edited by

Ronald R. Yager
Iona College
New Rochelle, New York USA

and

Janusz Kacprzyk
Polish Academy of Sciences
Warsaw, Poland

Kluwer Academic Publishers
Boston/Dordrecht/London