Logical Aggregation based on interpolative realization of Boolean algebra

Dragan G. Radojević
Mihajlo Pupin Institute, Volgina 15, 11000 Belgrade, Serbia
e-mail: dragan.radojevic@imp-automatika.bg.ac.yu

Abstract

In this paper, aggregation is treated as a logical and/or pseudo-logical operation what is important from many points of view such as adequacy and interpretations.

Keywords: Aggregation, Logical aggregation, Interpolative realization of Boolean algebra, Generalized Boolean polynomial, Pseudo Boolean polynomial, Generalized Choquet integral, OWA

1 Introduction

A very important problem in many fields of applications is the aggregation (fusion) of many partial aspects (attributes) into one global representative aspect. In the existing practice the weighted sum of partial aspects is used most often as an aggregation tool. This approach is additive and for all effects of interest which are not additive in their nature it is inadequate. For example: using a weighted sum as an aggregation tool even in the case of only two attributes (a, b), one can't to realize a simple and natural demand such as *a and b* is important. In multi-attribute decision making community this problem was recognized [2, 9] and as a solution they use theory of capacity [1] known in fuzzy community as fuzzy measure and fuzzy integrals [9]. In this approach additivity is relaxed by monotonicity, for which additivity is only a special case. As a consequence, the possible domain of application of this approach is much wider.

But from a logical point of view monotonicity is a superfluously strong constraint since many of logical functions are non monotone in their nature. A generalized discrete Choquet integral [8] is defined for a general measure – non monotone in a general case. This approach includes all logical and/or pseudo-logical functions but for only one arithmetic operator for interpolation intention, *min* function. *Interpolative realization of Boolean* algebra (IBA) [5] includes all logical functions and all interpolative operators – generalized product operators.

Logical aggregation as an adequate tool for aggregation in a general case is based on IBA. IBA is technically based on *generalized Boolean polynomials* (GBP) [5, 6].

GBP is described in Section 2. A representative example of logical aggregation is given in Section 4.

2 Generalized Boolean Polynomial

Primary attributes (properties) define a finite set $\Omega = \{a_1, \ldots, a_n\}$. No one of primary attributes can be calculated as a Boolean function of the remaining primary attributes from Ω. Set $BA(\Omega)$ of all the possible attributes generated by the set of primary attributes Ω by application of Boolean operators is a partially ordered set – Boolean algebra of attributes:

$$BA(\Omega) = \mathrm{P}(\mathrm{P}(\Omega)).$$

A partial order is based on the relation of inclusion and it is value irrelevant. The following structure with two binary and one unary operators is Boolean algebra:

$$\langle BA(\Omega), \cup, \cap, C \rangle.$$

Any element of Boolean algebra $\varphi \in BA(\mathbf{\Omega})$ is a corresponding attribute and it can be represented by the disjunctive normal form:

$$\varphi = \bigcup_{S \in \mathbf{P}(\mathbf{\Omega})} \left(\sigma_\varphi(S) \cap \alpha(S) \right), \qquad (1)$$

Atomic attributes $\alpha(S)$, $\left(S \in \mathbf{P}(\mathbf{\Omega})\right)$ are the simplest elements of $BA(\mathbf{\Omega})$ in the sense that they do not include in themselves anything except for a trivial Boolean constant **0**. The atomic attributes are described by the following expressions:

$$\alpha(S) = \bigcap_{a_i \in S} a_i \bigcap_{a_j \in \mathbf{\Omega} \setminus S} Ca_j, \quad S \in \mathbf{P}(\mathbf{\Omega}). \qquad (1.1)$$

Structural function $\sigma_\varphi : \mathbf{P}(\mathbf{\Omega}) \to \{0,1\}$ of analyzed attribute (element of Boolean algebra) $\varphi \in BA(\mathbf{\Omega})$:

$$\sigma_\varphi(S) = \begin{cases} \bar{1}, & \alpha(S) \subseteq \varphi; \\ \bar{0}, & \alpha(S) \not\subset \varphi; \end{cases}; \qquad (1.2)$$
$$\left(S \in \mathrm{P}(\Omega); \quad \bar{0}, \bar{1}, \varphi \in BA(\Omega)\right)$$

determines which atomic elements (attributes) are included in it ($\sigma_\varphi(S) = \bar{1} \Leftrightarrow \alpha(S) \subseteq \varphi$) and/or which are not included ($\sigma_\varphi(S) = \bar{0} \Leftrightarrow \alpha(S) \not\subset \varphi$), where:

$$\alpha(S) \subseteq \varphi \Leftrightarrow \left(\alpha(S) \cap \varphi = \alpha(S)\right)$$
$$\alpha(S) \not\subset \varphi \Leftrightarrow \left(\alpha(S) \cap \varphi = \bar{0}\right)$$
$$\left(S \in \mathrm{P}(\Omega); \quad \bar{0}, \bar{1}, \varphi \in BA(\Omega)\right)$$

Structural function of primary attribute $a_i \in \mathbf{\Omega}$ is given by the following expression

$$\sigma_{a_i}(S) = \begin{cases} \bar{1}, & a_i \in S \\ \bar{0}, & a_i \notin S \end{cases}; \quad \left(S \in \mathbf{P}(\mathbf{\Omega})\right)$$

Determination of structure of any attribute is based on the expression above and on the following rules:

$$\sigma_{\varphi \cap \psi}(S) = \sigma_\varphi(S) \cap \sigma_\psi(S),$$
$$\sigma_{\varphi \cup \psi}(S) = \sigma_\varphi(S) \cup \sigma_\psi(S),$$
$$\sigma_{C\varphi}(S) = C\sigma_\varphi(S).$$

where: $S \in \mathbf{\Omega}, \quad \varphi, \psi \in BA(\mathbf{\Omega})$.

Equation (1) can be described in the following form:

$$\varphi = \bigcup_{S \in \mathbf{P}(\mathbf{\Omega})} \left(\sigma_\varphi(S) \cap \left(\bigcap_{a_i \in S} a_i \bigcap_{a_j \in \mathbf{\Omega} \setminus S} Ca_j \right) \right).$$

Any attribute has its value realization on a valued level. A valued level is defined as a set of analyzed elements, objects, actions etc.

Any element of Boolean algebra of attributes can be represented by a *generalized Boolean polynomial*:

$$\varphi^{\otimes}\left(a_1^v, \ldots, a_n^v\right) = \sum_{S \in \mathbf{P}(\mathbf{\Omega})} \sigma_\varphi^v(S) \otimes \alpha^{\otimes}(S)\left(a_1^v, \ldots, a_n^v\right)$$
$$\varphi^{\otimes}\left(a_1^v, \ldots, a_n^v\right) = \sum_{S \in \mathbf{P}(\mathbf{\Omega})} \sigma_\varphi^v(S) \alpha^{\otimes}(S)\left(a_1^v, \ldots, a_n^v\right)$$
$$\left(a_i^v \in [0,1], \quad a_i \in \mathbf{\Omega}\right) \qquad (2)$$

Where: $\sigma_\varphi^v(S)$, $S \in \mathbf{P}(\mathbf{\Omega})$ is value realization of structural function $\sigma_\varphi(S)$, $S \in \mathbf{P}(\mathbf{\Omega})$, which is given by the following expression:

$$\sigma_\varphi^v(S) = \begin{cases} 1, & \sigma_\varphi(S) = \bar{1} \\ 0, & \sigma_\varphi(S) = \bar{0} \end{cases};$$
$$\left(S \in \mathbf{P}(\mathbf{\Omega}); \ \bar{0}, \bar{1}, \varphi \in BA(\mathbf{\Omega}); \ 0,1 \in N_0\right)$$

A generalized Boolean polynomial $\varphi^{\otimes}\left(a_1^v, \ldots, a_n^v\right)$ enables calculation of the value of a corresponding attribute $\varphi \in BA(\mathbf{\Omega})$ (element of Boolean algebra) for an analyzed object.

$\alpha^{\otimes}(S)\left(a_1^v, \ldots, a_n^v\right)$, $S \in \mathbf{P}(\mathbf{\Omega})$, are Boolean polynomial of atomic elements defined by the following expression:

$$\alpha^{\otimes}(S)\left(a_1^v, \ldots, a_n^v\right) = \sum_{K \in \mathbf{P}(\mathbf{\Omega} \setminus S)} (-1)^{|K|} \bigotimes_{a_i \in K \cup S} a_i^v \qquad (3)$$
$$\left(S \in \mathbf{P}(\mathbf{\Omega}), \quad a_i \in \mathbf{\Omega}, \quad a_i^v \in [0,1], i = 1,..,n\right).$$

Example: *Atomic Boolean polynomials for the case when set of primary attributes is* $\Omega = \{a, b\}$*, are given in the following table:*

Table 1*: Example of Atomic Boolean polynomials*

S	$\alpha(S)$	$\alpha^{\otimes}(S)(a^v,b^v)$
$\varnothing$	$Ca \cap Cb$	$1-a^v-b^v+a^v\otimes b^v$
$\{a\}$	$a \cap Cb$	$a^v - a^v \otimes b^v$
$\{b\}$	$Ca \cap b$	$b^v - a^v \otimes b^v$
$\{a,b\}$	$a \cap b$	$a^v \otimes b^v$

In atomic Boolean polynomials the following operators +, - and $\otimes$ figure.

Operator $\otimes$ is *a generalized product*, defined in the same way as *T*-norms [4] with one additional axiom – *no-negativity* [5].

$$\otimes : [0,1]\times[0,1]\to[0,1]$$

1. $a_i^v \otimes a_j^v = a_j^v \otimes a_i^v$
2. $a_i^v \otimes \left(a_j^v \otimes a_k^v\right) = \left(a_i^v \otimes a_j^v\right) \otimes a_k^v$
3. $a_i^v \le a_j^v \quad \Rightarrow \quad a_i^v \otimes a_k^v \le a_j^v \otimes a_k^v$
4. $a_i^v \otimes 1 = a_i^v$
5. $\boxed{\sum_{K\in \mathbf{P}(\Omega\setminus S)} (-1)^{|K|} \bigotimes_{a_i \in S\cup K} a_i^v \ge 0,} \quad \forall S \in \mathbf{P}(\Omega)$

$$\left(\Omega = \{a_1,\ldots,a_n\},\quad a_i^v \in [0,1], i=1,\ldots,n\right)$$

Additional axiom "non-negativity" ensures that the values of atomic Boolean polynomials are non-negative: $\alpha^{\otimes}(S) \ge 0, \quad (S \in \mathbf{P}(\Omega))$. As a consequence all elements of Boolean algebra are non-negative: $\varphi^{\otimes}\left(a_1^v,\cdots a_1^v\right) \ge 0, \quad \varphi \in BA(\Omega)$.

Example: *In the case $\Omega = \{a,b\}$ generalized product, according to axioms of non-negativity can be in the following interval:*

$$\max(a+b-1,\, 0) \le a \otimes b \le \min(a,\, b)\ .$$

In spite of formal similarity between a *T*-norm and a generalized product, their roles are crucially different: while a *T*-norm in conventional fuzzy approaches has the role of a logical operator (which is impossible in a general case) a generalized product $\otimes$ is only an arithmetic operator on a value level.

A generalized Boolean polynomial given by the expression (2) can be represented in the following way:

$$\varphi^{\otimes}\left(a_1^v,\ldots,a_n^v\right) =$$
$$= \sum_{S\in \mathbf{P}(\Omega)} \sigma_\varphi^v(S) \otimes \sum_{K\in \mathbf{P}(\Omega\setminus S)} (-1)^{|K|} \bigotimes_{a_i\in K\cup S} a_i^v$$
$$= \sum_{S\in \mathbf{P}(\Omega)} \sigma_\varphi^v(S) \sum_{K\in \mathbf{P}(\Omega\setminus S)} (-1)^{|K|} \bigotimes_{a_i\in K\cup S} a_i^v,$$
$$\left(\varphi \in BA(\Omega),\quad a_i^v \in [0,1],\quad a_i \in \Omega\right).$$

A generalized Boolean polynomial can be represented as a scalar product of the following two vectors: (a) *structural vector* of analyzed Boolean algebra element – attribute

$$\vec{\sigma}_\varphi = \left[\sigma_\varphi^v(S) \middle| S \in \mathbf{P}(\Omega)\right] \tag{4}$$

where: $\Omega = \{a_1,\ldots,a_n\}, \quad \varphi \in BA(\Omega)$,

and (b) *vector of atomic Boolean polynomials*

$$\vec{\alpha}^{\otimes}\left(a_1^v,\ldots,a_n^v\right) = \left[\alpha^{\otimes}(S)\left(a_1^v,\ldots,a_n^v\right) \middle| S \in \mathbf{P}(\Omega)\right]^T \tag{5}$$
$$\left(a_i \in \Omega,\quad a_i^v \in [0,1], i=1,..,n\right).$$

So, a generalized Boolean polynomial is a scalar product of the above defined two vectors:

$$\varphi^{\otimes}\left(a_1^v,\ldots,a_n^v\right) = \vec{\sigma}_\varphi \vec{\alpha}^{\otimes}\left(a_1^v,\ldots,a_n^v\right) \tag{6}$$

where: $\varphi \in BA(\Omega), \quad a_i^v \in [0,1], \quad a_i \in \Omega$.

For structural vectors all Boolean axioms are valid: Associativity, Commutativity, Absorption, Distributivity, Excluded middle and Contradiction

$$\vec{\sigma}_{\varphi\cup(\psi\cup\phi)} = \vec{\sigma}_{(\varphi\cup\psi)\cup\phi} \qquad \vec{\sigma}_{\varphi\cap(\psi\cap\phi)} = \vec{\sigma}_{(\varphi\cap\psi)\cap\phi}$$
$$\vec{\sigma}_{\varphi\cup\psi} = \vec{\sigma}_{\psi\cup\varphi} \qquad \vec{\sigma}_{\varphi\cap\psi} = \vec{\sigma}_{\psi\cap\varphi}$$
$$\vec{\sigma}_{\varphi\cup(\varphi\cap\psi)} = \vec{\sigma}_{\varphi} \qquad \vec{\sigma}_{\varphi\cap(\varphi\cup\psi)} = \vec{\sigma}_{\varphi}$$
$$\vec{\sigma}_{\varphi\cup(\psi\cap\phi)} = \vec{\sigma}_{(\varphi\cup\psi)\cap(\varphi\cup\phi)} \qquad \vec{\sigma}_{\varphi\cap(\psi\cup\phi)} = \vec{\sigma}_{(\varphi\cap\psi)\cup(\varphi\cap\phi)}$$
$$\vec{\sigma}_{\varphi\cup C\varphi} = \vec{1} \qquad \vec{\sigma}_{\varphi\cap C\varphi} = \vec{0}$$

respectively; and all Boolean theorems: Idempotency, Boundedness, 0 and 1 are complements, De Morgan's laws and Involution:

$$\vec{\sigma}_{\varphi\cup\varphi} = \vec{\sigma}_{\varphi} \qquad \vec{\sigma}_{\varphi\cap\varphi} = \vec{\sigma}_{\varphi}$$
$$\vec{\sigma}_{\varphi\cup 0} = \vec{\sigma}_{\varphi} \qquad \vec{\sigma}_{\varphi\cap 1} = \vec{\sigma}_{\varphi}$$
$$\vec{\sigma}_{\varphi\cup 1} = \vec{1} \qquad \vec{\sigma}_{\varphi\cap 0} = \vec{0}$$
$$\vec{\sigma}_{C0} = \vec{1} \qquad \vec{\sigma}_{C1} = \vec{0}$$
$$\vec{\sigma}_{C(\varphi\cup\psi)} = \vec{\sigma}_{C\varphi\cap C\psi} \qquad \vec{\sigma}_{C(\varphi\cap\psi)} = \vec{\sigma}_{C\varphi\cup C\psi}$$
$$\vec{\sigma}_{CC\varphi} = \vec{\sigma}_{\varphi}$$

respectively; where: $\varphi,\psi,\phi \in BA(\Omega)$.

So, the structure of a Boolean algebra element preserves Boolean properties in a generalized case described by Boolean polynomials.

As a consequence for any two elements of Boolean algebra $\varphi,\psi \in BA(\Omega)$ the following equations are valid:

$$\begin{aligned}(\varphi\cap\psi)^{\otimes}\left(a_1^v,\dots,a_n^v\right) &= \vec{\sigma}_{\varphi\cap\psi}\vec{\alpha}^{\otimes}(x) \\ &= \left(\vec{\sigma}_{\varphi}\wedge\vec{\sigma}_{\psi}\right)\vec{\alpha}^{\otimes}\left(a_1^v,\dots,a_n^v\right) \\ (\varphi\cup\psi)^{\otimes}\left(a_1^v,\dots,a_n^v\right) &= \vec{\sigma}_{\varphi\cup\psi}\vec{\alpha}^{\otimes}\left(a_1^v,\dots,a_n^v\right) \\ &= \left(\vec{\sigma}_{\varphi}\vee\vec{\sigma}_{\psi}\right)\vec{\alpha}^{\otimes}\left(a_1^v,\dots,a_n^v\right) \\ (C\varphi)^{\otimes}\left(a_1^v,\dots,a_n^v\right) &= \vec{\sigma}_{C\varphi}\vec{\alpha}^{\otimes}\left(a_1^v,\dots,a_n^v\right) \\ &= \left(\vec{1}-\vec{\sigma}_{\varphi}\right)\vec{\alpha}^{\otimes}\left(a_1^v,\dots,a_n^v\right) \\ &= 1-(\varphi)^{\otimes}\left(a_1^v,\dots,a_n^v\right)\end{aligned}$$

Actually, Boolean polynomial maps a corresponding element of Boolean algebra into its value from the real unit interval [0, 1] on the value level so that a partial order on the value level is preserved. Since a partial order is based on Boolean laws, they are preserved on the value level in a general case too, contrary to other approaches.

3. Generalized Pseudo-Boolean Polynomial

To every element of IBA corresponds a generalized Boolean polynomial with the ability to process all values of primary variables from a real unit interval [0, 1]. A pseudo-Interpolative Boolean polynomial is a linear convex combination of analyzed elements of IBA – generalized Boolean polynomials:

$$\begin{aligned}&\pi\varphi^{\otimes}\left(a_1^v,\dots,a_n^v\right) = \sum_{i=1}^{m} w_i\varphi_i^{\otimes}\left(a_1^v,\dots,a_n^v\right), \\ &\sum_{i=1}^{m} w_i = 1, \quad \left(w_i \ge 0,\ i=1,\dots,m\right), \qquad (7) \\ &\left(a_i \in \Omega,\quad a_i^v \in [0,1], i=1,\dots,n\right).\end{aligned}$$

From the definition of generalized Boolean polynomials, an *interpolative pseudo-Boolean polynomial* is given by the following expression:

$$\begin{aligned}&\pi\varphi_{\mu}^{\otimes}\left(a_1^v,\dots,a_n^v\right) = \\ &= \sum_{i=1}^{m} w_i \sum_{S\in\mathbf{P}(\Omega)} \sigma_{\varphi_i}^v(S) \sum_{C\in\mathbf{P}(\Omega\setminus S)} (-1)^{|C|} \bigotimes_{a_i\in S\cup C} a_i^v \\ &= \sum_{S\in\mathbf{P}(\Omega)} \mu(S) \sum_{C\in\mathbf{P}(\Omega\setminus S)} (-1)^{|C|} \bigotimes_{a_i\in S\cup C} a_i^v . \quad (7.1)\end{aligned}$$

Structure function μ of interpolative pseudo-Boolean polynomial $\pi\varphi_{\mu}^{\otimes}$ is a set function

$$\mu : \mathbf{P}(\Omega) \to [0,1], \quad \Omega = \{a_1,\dots,a_n\}$$

defined by the following expression, [7]:

$$\begin{aligned}&\mu(S) = \sum_{i=1}^{m} w_i\sigma_{\varphi_i}^v(S), \\ &\left(S\in\mathbf{P}(\Omega),\quad \varphi_i \in BA(\Omega)\right) \qquad . \qquad (8) \\ &\sum_{i=1}^{m} w_i = 0, \quad w_i \ge 0, \quad i=1,\dots,m\end{aligned}$$

Where: $\sigma_{\varphi_i}^v$, $i=1,\dots,m$ are structure functions of the corresponding Boolean functions $\varphi_i \in BA(\Omega)$, $i=1,\dots,m$.

The characteristics of pseudo-Boolean polynomial depend on the generalized product, and its structure function. Structure functions can be classified into: (a) additive, (b) monotone and (c) generalized ($(a)\subset(b)\subset(c)$).

4. Logical Aggregation

A starting point is a finite set of primary attributes $\Omega = \{a_1,\dots,a_n\}$. The task of logical aggregation (LA) [7] is the fusion of primary quality attribute values into one resulting globally representative value using logical tools. In a general case LA has two steps: (1) Normalization of primary attributes' values:

$$\cdot^{v} : \Omega \to [0, 1].$$

The result of normalization is a generalized logical and/or [0, 1] value of analyzed primary attribute, and

(2) Aggregation of normalized values of primary attributes into one resulting value by pseudo-logical function as a logical aggregation operator:

$$Aggr : [0,1]^{n} \to [0,1].$$

A Boolean logical function φ is transformed into a corresponding generalized Boolean polynomial (GBP), [5], $\varphi^{\otimes} : [0, 1]^{n} \to [0, 1]$. Actually, to any element of Boolean algebra of attributes $\varphi_i \in BA(\Omega)$ there corresponds uniquely GBP $\varphi_i^{\otimes}\left(a_1^{v},...,a_n^{v}\right)$. GBP is defined by expression (2) and/or (2.1).

Pseudo-logical function is a linear convex combination of generalized Boolean polynomials [5], defined by expression (7) and/or (7.1).

Operator of logical aggregation in a general case is a pseudo-logical function:

$$Agg_{\mu}^{\otimes}\left(a_1^{v},...,a_n^{v}\right) = \pi\varphi_{\mu}^{\otimes}\left(a_1^{v},...,a_n^{v}\right) \tag{9}$$

or

$$Agg_{\mu}^{\otimes}\left(a_1^{v},...,a_n^{v}\right) = \sum_{S \in \mathbf{P}(\Omega)} \mu(S) \sum_{C \in \mathbf{P}(\Omega \setminus S)} (-1)^{|C|} \bigotimes_{a_i \in S \cup C} a_i^{v}$$

Aggregation measure is a structural function of pseudo-logical function – logical aggregation operator (9). So, *Aggregation measure* is a set function $\mu : \mathrm{P}(\Omega) \to [0,1]$, which in a general case is not a monotone function (generalized capacity), defined by the following expression:

$$\mu(S) = \sum_{i=1}^{m} w_i \sigma_{\varphi_i}^{v}(S),$$
$$\left(S \in \mathrm{P}(\Omega), \quad \varphi_i \in BA(\Omega)\right) \tag{10}$$
$$\sum_{i=1}^{m} w_i = 0, \quad w_i \geq 0, \quad i = 1,...,m$$

As a consequence, a logical aggregation operator depends on the chosen: (a) measure of aggregation and (b) operator of generalized product. By a corresponding choice of the measure of aggregation μ and generalized product $\otimes$ the known aggregation operators can be obtained as special cases:

Weighted sum

For the aggregation measure and generalized product:

$$\mu_{add}(S) = \sum_{i=1}^{n} w_i \sigma_{a_i}^{v}(S), \quad S \in \mathrm{P}(\Omega); \quad \otimes := min.$$

Logical aggregation operator is a weighted sum:

$$Agg_{\mu_{add}}^{min}\left(a_1^{v},...,a_n^{v}\right) = \sum_{a_i \in \Omega} w_i a_i^{v}$$

Arithmetic mean

For the aggregation measure and generalized product:

$$w_i = \frac{1}{n}, \; \mu_{mean}(S) = \frac{|S|}{|\Omega|}; \quad \otimes := min$$

Logical aggregation operator is the arithmetic mean:

$$Agg_{\mu_{mean}}^{min}\left(a_1^{v},...,a_n^{v}\right) = \frac{1}{n} \sum_{a_i \in \Omega} a_i^{v}$$

K-th attribute only

For the aggregation measure and generalized product:

$$w_i = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases}; \quad \mu_k(S) = \begin{cases} 1 & a_k \in S \\ 0 & a_k \notin S \end{cases}; \quad \otimes := min$$

Logical aggregation operator is the k-th attribute only:

$$Agg_{\mu_k}^{\otimes}\left(a_1^{v},...,a_n^{v}\right) = a_k^{v}$$

Discrete Choquet integral

For any monotone aggregation measure μ_{mon} and generalized product:

$$\mu_{mon}, \qquad \otimes := min$$

Logical aggregation operator is a discrete Choquet integral:

$$Agg_{\mu_{mon}}^{\otimes}\left(a_1^{v},...,a_n^{v}\right) = C_{\mu_{mon}}\left(a_1^{v},...,a_n^{v}\right).$$

A discrete Choquet integral is defined by the following expression:

$$C_{\mu_{mon}}\left(a_1^v,...,a_n^v\right)=\sum_{k=1}^{n}\left(a_{(k)}^v-a_{(k-1)}^v\right)\mu_{mon}\left(A_{(k)}\right),$$

where:

$$a_{(1)}^v \le ... \le a_{(n)}^v; \quad A_{(k)}=\left\{a_{(k)}^v,...,a_{(n)}^v\right\}.$$

Minimal value of attributes

For the aggregation measure and generalized product:

$$\mu_{AND}(S)=\begin{cases}1, & S=\Omega\\ 0, & S\ne\Omega\end{cases}; \quad \otimes:=min.$$

Logical aggregation operator is the *min* function

$$Agg_{\mu_{AND}}^{min}\left(a_1^v,...,a_n^v\right)=min\left\{a_1^v,...,a_n^v\right\}.$$

Maximal value of attributes

For the aggregation measure and generalized product:

$$\mu_{OR}(S)=\begin{cases}1, & S\ne\varnothing\\ 0, & S=\varnothing\end{cases}; \quad \otimes;=min$$

Logical aggregation operator is the *max* function

$$Agg_{\mu_{OR}}^{min}\left(a_1^v,...,a_n^v\right)=max\left\{a_1^v,...,a_n^v\right\}.$$

OWA-ordered weight aggregation

For the aggregation measure and generalized product:

$$\mu_{OWA}(S)=\begin{cases}0, & S=\varnothing\\ \sum_{i=1}^{m}w_i, & |S|=m\end{cases}; \quad \otimes:=min$$

Logical aggregation operator is an OWA aggregation operator

$$Agg_{\mu_{OWA}}^{min}\left(a_1^v,...,a_n^v\right)=OWA\left(a_1^v,...,a_n^v\right).$$

OWA is defined by the following expression:

$$OWA\left(a_1^v,...,a_n^v\right)=\sum_{i=1}^{n}w_i a_{(i)}^v$$

$$a_{(1)}^v \le a_{(2)}^v \le ... \le a_{(n)}^v, \quad \sum_{i=1}^{n}w_i=1, \quad w_i\ge 0$$

.

k-th order statistics

For the aggregation measure and generalized product:

$$\mu_{k^{th}}(S)=\begin{cases}0, & |S|<k\\ 1, & |S|\ge k\end{cases}; \quad \otimes:=min$$

Logical aggregation operator is the k-th order statistics

$$Agg_{\mu_{k^{th}}}^{min}\left(a_1^v,...,a_n^v\right)=a_{(k)}^v,$$

where:

$$a_{(1)}^v \le a_{(2)}^v \le ... \le a_{(n)}^v.$$

5. Example of Logical Aggregation Application

A modified example from [3] is analyzed here.

Example: Objects A, B, C and D are described by quality attributes, whose values are from real unit interval [0, 1], given in the following table:

Table 2: Values of quality attributes

Object	*a*	*b*	*c*
A	.75	.9	.3
B	.75	.8	.4
C	.3	.65	.1
D	.3	.55	.2

An object should be compared on the base of a global quality. A global quality is actually aggregation of attributes so the following aspects should be incorporated: (a) the average value of quality attributes and (b) if the analyzed object is good by attribute *a* then attribute *c* is more important than *b* and if analyzed object is not good by attribute *a* then attribute *b* is more important than *c*.

A partial demand (a) is given by the following trivial expression:

$$\frac{a+b+c}{3}$$

A partial demand (b) is given by the following logical expression:

$$\varphi(a,b,c) = (a \cap c) \cup (Ca \cap b) \qquad (11)$$

A generalized Boolean polynomial of logical expression (11) is:

$$\varphi^{\otimes}(a,b,c) = \left((a \cap c) \cup (Ca \cap b)\right)^{\otimes}$$
$$= b + a \otimes c - a \otimes b$$

A possible logical aggregation operator is:

$$Aggr^{\otimes}(a,b,c) = \frac{1}{2}\frac{a+b+c}{3} + \frac{1}{2}\varphi^{\otimes}(a,b,c)$$
$$= \frac{1}{2}\frac{a+b+c}{3} + \frac{1}{2}(b + a \otimes c - a \otimes b)$$

A corresponding measure of aggregation is:

$$\mu = \frac{1}{6}(\sigma_a + \sigma_b + \sigma_c) + \frac{1}{2}(\sigma_a \wedge \sigma_c) \vee (C\sigma_a \wedge \sigma_b).$$

or given as a table:

Table 3: Measure of aggregation

S	$\mu(S)$
$\varnothing$	0
$\{a\}$	1/6
$\{b\}$	2/3
$\{c\}$	1/6
$\{a,b\}$	5/6
$\{a,c\}$	5/6
$\{b,c\}$	1/3
$\{a,b,c\}$	1

It is clear that the measure is non-monotone since $\mu(\{b\}) \geq \mu(\{b,c\})$, and as a consequence it is not possible to use a standard Choquet integral.

In the case $\otimes := min$ function $\varphi^{min}(a,b,c)$ is actually a generalized discrete Choquet integral and its values are given in the following table:

Table 4: Values for $\otimes := min$

Object	$\varphi^{min}(a,b,c)$
A	.45
B	.45
C	.45
D	.45

So, these results without discrimination are not adequate.

In the case when a generalized product is an ordinary product, $\otimes := *$, quitting conventional approaches, the corresponding values of function $\varphi^{*}(a,b,c)$ are given in the following table:

Table 5: Values for $\otimes := *$

Object	$\varphi^{*}(a,b,c)$
A	.450
B	.500
C	.485
D	.445

The values of aggregation function, for given aggregation measure, table 1, and for $\otimes := *$, are presented in the following table:

Table 6: Values of resulting aggregation function

Object	$Aggr^{*}(a,b,c)$
A	.5500
B	.5750
C	.4175
D	.3725

These results completely reflect all specified demands.

Comment: *All demands, defined in this example for aggregations of analyzed quality attributes, cannot be realized using the approaches which are conventional in the field of quality control.*

6. Conclusion

The aggregation of partial goals - attributes, into one representative global goal is a very important task. Conventional aggregation tools are very often inadequate. Partial demands for aggregation can be and usually are logical demand which can be adequately described only by logical expressions. In this paper logical aggregation as a tool for aggregation is analyzed. Logical aggregation has multiple advantages among others from the stand point of its possibility and interpretability. The new approach treats logical functions – partial aggregation demand, as a generalized Boolean polynomial which can process values from the whole real unit interval [0, 1]. Logical aggregation in a general case is a weighted sum of partial demands. Therefore, aggregation in a general case is a generalized pseudo-logic function. It is interesting that conventional aggregation operators are only a special case of logical aggregation operators and, as a consequence of using LA, one can do much more in an adequate direction than before.

References

[1] G. Choquet, Theory of capacities, *Annals de l'Institute Fourier,* 5:131-295, 1953.

[2] M. Grabisch , The application of fuzzy integrals in multicriteria decision making. *European Journal of Operational Research*, 89(3):445-456, 1996.

[3] M. Grabisch, Labreuche, Ch., Bi-capacities for decision making on bipolar scales. *EUROFUSE Workshop on Information Systems*, Varenna, Italy, 185-190, September 2002.

[4] E.P Klement., Mesiar R., Pap E., *Triangular norms*, Trends in Logic, Studia Logica Library, Vol 8, Kluwer Academic Publishers, 2000.

[5] D. Radojević, Interpolative realization of Boolean algebra as a consistent frame for gradation and/or fuzziness, *Forging New Frontiers: Fuzzy Pioneers II*, Ed. M. Nikravesh, J. Kacprzyk L. A. Zadeh, Studies in Fuzziness and Soft Computing, pp. 326-351, Springer, 2007.

[6] D. Radojević, Interpolative realization of Boolean algebra, NEUREL 2006, Eight Seminar on Neural Network Applications in Electrical Engineering, 201-206, 2006.

[7] D. Radojević, Logical aggregation based on Interpolative realization of Boolean algebra, submitted for publication

[8] D. Radojević, New [0,1]-valued logic: A natural generalization of Boolean logic, *Yugoslav Journal of Operational Research – YUJOR*, Belgrade, Vol 10, No 2, 185-216, 2000.

[9] M. Sugeno, *Theory of fuzzy integrals and its applications*. Doctoral Thesis, Tokyo Institute of Technology, Tokyo, Japan, 1974

[10] R. R Yager, and Kacprzyk, J., *The Ordered Weighted Averaging Operation: Theory, Methodology and Applications*, Kluwer: Norwell, MA, 1997.