Iranian Journal of Fuzzy Systems

Volume 20, Number 5, (2023), pp. 71-88

Original paper

University of Sistan and Baluchestan

Immediate consequences operator through ordered weighted average operators

J. Medina[1] and J. A. Torné-Zambrano[2]

[1,2] *Department of Mathematics, University of Cádiz, Cádiz, Spain*

jesus.medina@uca.es,joseantonio.torne@uca.es

Abstract

The immediate consequences operator has been a widely studied and used operator for defining the semantics of a logic program. For instance, it has been considered in the fuzzy case for handling datasets with imperfect, imprecise or vague information. The natural generalization of this operator to the mentioned fuzzy framework is based on the supremum operator, which preserves the strict nature of the universal quantifier. As a consequence, errors in the data, which are usual in the uncertainty environment of the considered dataset, can cause loss of information. This is the main reason why this paper makes different generalizations of this operator by using weighted aggregation operators and introducing interesting results.

Keywords: Immediate consequences operator, ordered weighted operators.

1 Introduction

Multi-adjoint logic programming is based on the computational logic, one branch of the logic, which is the field of the mathematics that deals with modeling human language and has been considered in many fields [1, 3, 4, 11, 25]. From the efforts of diverse authors [7, 10, 26] in the field of computational logic, at the beginning of the 70s, logic programming was born, which was a novel conception considering that logic had only been used as a declarative language [2, 16]. In logic programming there are two parts well differentiated. On the one hand, the syntax, being the set of symbols and formulas that allow to represent the language in a natural way. On the other hand, semantics, which allows us to give meaning to formulas based on their syntactic structure, establishing an inference system that allows us to decide which deductions and conclusions are valid.

The syntax of a multi-set logic program is made up of logic programs, which consist of a series of equations $C \leftarrow \mathcal{B}_1, \dots, \mathcal{B}_n$, called *rules*, that models a given system. Based on these rules and experimental values obtained by observation, a process of diagnosis and prediction of the variables of the system can be carried out. Defining a system of inference to obtain which deductions or consequences are correct is one of the most common problems in logic programming. The multi-adjoint approach refers to a formal framework that has been recently introduced and which generalizes different approaches of fuzzy logic programming [6, 14, 30]. This is the main reason why this fuzzy framework has been considered in this paper. The algebraic structure of this framework is given by a complete lattice together with several conjunctors and implications, which form adjoint pairs. On the other hand, semantics of a logic program is defined as the post-fixed points of the immediate consequences operator $T_{\mathbb{P}}$ [13, 18, 19, 21, 27, 28, 29], which is defined as usual through the supremum of a set of values. Consequently, the existence of small alterations or variations in the data (due to an incorrect or deficient observation) can alter the final outcome. This relevant issue was recently considered in [22, 23] in which generalized quantifiers were considered to solve it. Another interesting alternative to reducing the impact of possible noise in the data, in the computation of the immediate consequences operator associated with a logic program, is the use of ordered weighted averaging (OWA) operators. Therefore, the

---

Corresponding Author: J. Medina

Received: October 2022; Revised: April 2023; Accepted: June 2023.

https://doi.org/10.22111/IJFS.2023.7696

fundamental objective of this paper will be to contribute with a new definition of the $T_{\mathbb{P}}$ based on OWA operators that can relieve the need to use the supremum in its definition.

The use of the OWA operators is justified. These operators allow us to aggregate the information, which consists of assembling all the values in a single one, being the representative of the rest of them. One of the main reasons for the popularity of the OWA operator, which has been target of study and research in the latest years [15, 17, 31, 32], is the great flexibility it provides for modeling a wide range of situations. Furthermore, it has some interesting properties such as monotonicity, and can also have other ones, such as continuity and idempotence.

The document is organized as follows: preliminary concepts of multi-adjoint structures, multi-adjoint logic programming and OWA operators are recalled in Section 2. In Section 3, we carry out a generalization of the $T_{\mathbb{P}}$ through OWA operators, first using a particular weights vector, and then generalizing the result with a generic weights vector. In Section 4 the continuity of this new operator is studied, necessary in order to obtain the least model of the program in a countable number of iterations. To finish, some conclusions and future work are presented in Section 5.

2 Preliminaries

We will begin recalling some relevant definitions that will be necessary in the next sections [19, 20].

2.1 Multi-adjoint algebra structure

The algebraic structure that we are going to consider in this paper is based on the notion of *adjoint pair* [19, 20].

Definition 2.1. [19] *Let $\langle P, \preceq\rangle$ be a poset and $(\leftarrow, \&)$ a pair of binary operators in P such that*

1. *The operator $\&$ is increasing in both arguments.*
2. *The operator $\leftarrow$ is increasing in the first argument (consequent) and decreasing in the second one (antecedent)*
3. *The adjoint property holds, $x \preceq (y \leftarrow z)$ verifies if and only if $(x\&z) \preceq y$, for all $x, y, z \in P$.*

In these circumstances, we say that $(\leftarrow, \&)$ is an adjoint pair *in $\langle P, \preceq\rangle$.*

Hence, the algebraic structure is composed of a lattice in which different adjoint pairs are defined.

Definition 2.2. [20] *Let $\langle L, \preceq\rangle$ be a complete lattice. A multi-adjoint lattice $\mathcal{L}$ is a tuple $(L, \preceq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n)$ satisfying*

1. *$\langle L, \preceq\rangle$ is bounded, i.e., it has a bottom $\bot$ and a top $\top$ elements.*
2. *$(\leftarrow_i, \&_i)$ is an adjoint pair in $\langle L, \preceq\rangle$, for all $i \in \{1, \dots, n\}$.*
3. *$\top \&_i \vartheta = \vartheta \&_i \top = \vartheta$, for all $\vartheta \in L$ and $i \in \{1, \dots, n\}$.*

2.2 Multi-adjoint logic programming

This section introduces some of the most important definitions in the framework of multi-adjoint logic programming, with the aim of making this article self-contained and facilitate the understanding of the following sections. For a more in-depth study, see [19, 20]. The first ones recall the usual definitions in logic programming.

Definition 2.3. [19] *A* graded set *is a set Ω with a function which assigns to each element $\omega \in \Omega$ a number $n \geq 0$, called the* arity *of ω. The set of operators from Ω with arity n will be denoted as Ω_n.*

Definition 2.4. [19] *Given a graded set Ω, an* Ω-algebra *$\mathfrak{U}$ is a pair $\langle A, I\rangle$ where A is a non-empty set called the* carrier, *and I is a function which assigns maps to the elements of Ω as follows:*

1. *Each element $\omega \in \Omega_n$, $n > 0$, is interpreted as a map $I(\omega)\colon A^n \to A$, denoted by $\omega_{\mathfrak{U}}$.*
2. *Each element $c \in \Omega_0$, called* constant, *is interpreted as an element $I(c)$ in A, denoted by $c_{\mathfrak{U}}$.*

It is also necessary the notion of substructure of an Ω-algebra.

Definition 2.5. [19] *Given an Ω-algebra $\mathfrak{U} = \langle A, I\rangle$, an* Ω-subalgebra *$\mathfrak{B}$ is a pair $\langle B, J\rangle$ such that $B \subseteq A$ and*

1. *$J(c) = I(c)$ for all $c \in \Omega_0$.*
2. *Given $\omega \in \Omega_n$, $J(\omega)\colon B^n \longrightarrow B$ is the restriction of $I(\omega)\colon A^n \longrightarrow A$ to the domain $B^n \subseteq A^n$.*

The following definition introduces the algebraic structure which enables to use extra operators.

Definition 2.6. [20] *Let Ω be a graded set containing operators $\leftarrow_i$ and $\&_i$, for $i \in \{1, \dots, n\}$, and possibly some extra operators, and let $\mathfrak{L} = \langle L, I\rangle$ be an Ω-algebra whose carrier set L is a complete lattice under $\preceq$. We say that $\mathfrak{L}$ is a* multi-adjoint Ω-algebra *with respect to the pairs $(\leftarrow_i, \&_i)$, for $i \in \{1, \dots, n\}$, if $(L, \preceq, I(\leftarrow_1), I(\&_1), \dots, I(\leftarrow_n), I(\&_n))$ is a multi-adjoint lattice.*

Example 2.7. *Given a graded set $\Omega = \{\leftarrow_P, \&_P, \leftarrow_G, \&_G, \wedge_{\text{Ł}}, @\}$ and an Ω-algebra $\mathfrak{L} = \langle [0,1], I\rangle$, where[1] $\dot{\&}_P$, $\dot{\&}_{\text{Ł}}$ and $\dot{\&}_G$ are the product, Gödel and Łukasiewicz t-norms respectively, together with their residuated implications, since the tuple $(L, \preceq, \dot{\leftarrow}_P, \dot{\&}_P, \dot{\leftarrow}_G, \dot{\&}_G)$ is a multi-adjoint lattice, we have that $\mathfrak{L}$ is a multi-adjoint Ω-algebra. Recall that the definitions of these operators, for all x, y, $z \in [0,1]$, are*

$$\&_G(x,y) = \min\{x, y\} \qquad z \leftarrow_G y = \begin{cases} 1 & \text{if } \; y \le z \\ z & \text{otherwise} \end{cases}$$

$$\&_{\text{Ł}}(x,y) = \max\{0, x + y - 1\} \qquad z \leftarrow_{\text{Ł}} y = \min\{1, 1 - y + z\}$$

$$\&_P(x,y) = x \cdot y \qquad z \leftarrow_P y = \begin{cases} 1 & \text{if } \; y \le z \\ \frac{z}{y} & \text{otherwise} \end{cases}$$

2.3 Syntax of propositional multi-adjoint logic programming

The syntax encompasses the set of symbols and formulae used to formally represent different statements.

Definition 2.8. [19] *Let Ω be a graded set and Π a countably infinite set. The* alphabet $A_{\Omega,\Pi}$ *associated with Ω and Π is defined to be the disjoint union $\Omega \uplus \Pi \uplus S$, where S is the set of auxiliary symbols "(", ")" and ",".*

Definition 2.9. [19] *Given a graded set Ω and alphabet $A_{\Omega,\Pi}$. The* Ω-algebra $\mathfrak{E} = \langle A^*_{\Omega,\Pi}, I\rangle$ of expressions *is defined as follows:*

1. *The carrier A^*_{Ω} is the set of strings over $A_{\Omega,\Pi}$.*
2. *The interpretation function I satisfies the following conditions for strings $a_1, \dots, a_n \in A^*_{\Omega,\Pi}$:*
   - *$c_{\mathfrak{E}} = c$, where c is a constant operation, that is, $c \in \Omega_0$.*
   - *$\omega_{\mathfrak{E}}(a_1) = \omega a_1$, where ω is a unary operation, that is, $\omega \in \Omega_1$.*
   - *$\omega_{\mathfrak{E}}(a_1, a_2) = (a_1 \omega a_2)$, where ω is a binary operation, that is, $\omega \in \Omega_2$.*
   - *$\omega_{\mathfrak{E}}(a_1, \dots, a_n) = \omega(a_1, \dots, a_n)$, where ω is an n-ary operation, that is, $\omega \in \Omega_n$ and $n > 2$.*

The idea of this definition is that the set $A_{\Omega,\Pi}$ provides syntactic structure to the set of symbols Ω. However, not all symbol combinations make sense. For this reason, the following definition is introduced.

Definition 2.10. [19] *Let Ω be a graded set, Π a countable set of propositional symbols and $\mathfrak{E}$ the algebra of expressions corresponding to the alphabet $A_{\Omega,\Pi}$. The* well-formed formulas *(in short, formulas) generated by Ω over Π is the least subalgebra $\mathfrak{F}$ of the algebra of expressions $\mathfrak{E}$ containing Π. The set of formulas, that is the carrier of $\mathfrak{F}$, will be denoted by F_Ω.*

Finally, the unique homomorphic extension theorem is recalled.

Theorem 2.11. [9] *Let Ω be a graded set, Π a set of propositional symbols, $\mathfrak{F}$ the corresponding Ω-algebra of formulas. Let $\mathfrak{L}$ be an arbitrary Ω-algebra with carrier L. Then, for every function $J\colon \Pi \longrightarrow L$ there is a unique homomorphism $\hat{J}\colon F_\Omega \longrightarrow L$ such that:*

1. *For all $p \in \Pi$, $\hat{J}(p) = J(p)$.*
2. *For each constant $c \in \Omega_0$, $\hat{J}(c_{\mathfrak{F}}) = c_{\mathfrak{L}}$.*

---

[1] In order to simplify the notation, the interpretation of the operator symbols ($I(\&)$) will be denoted with a dot in the top ($\dot{\&}$).

3. *For every* $\omega \in \Omega_n$ *with* $n > 0$ *and for all* $F_i \in F_\Omega$ *with* $i \in \{1, \dots, n\}$

$$\hat{J}\left(\omega_{\mathfrak{F}}(F_1, \dots, F_n)\right) = \omega_{\mathfrak{L}}\left(\hat{J}(F_1), \dots, \hat{J}(F_n)\right).$$

From now on, a graded set Ω, a multi-adjoint Ω-algebra $\mathfrak{L}$, a set of propositional symbols Π, and the corresponding Ω-algebra of well-formed formulas freely generated from Π by the operators in Ω, with carrier L, denoted as $\mathfrak{F}$ will be fixed.

The syntax of multi-adjoint logic programming is focused on defining logic programs, which are built from a multi-adjoint Ω-algebra and the set of well-formed formulas $\mathfrak{F}$.

Definition 2.12. [19] *A* multi-adjoint logic program *is a set* $\mathbb{P}$ *of weighted rules of the form* $\langle (A \leftarrow_i \mathcal{B}), \vartheta \rangle$*, also denoted* $A \overset{\vartheta}{\leftarrow} \mathcal{B}$*, such that:*

1. *The rule* $(A \leftarrow_i \mathcal{B})$ *is a formula of* $\mathfrak{F}$*.*
2. *The confidence factor* ϑ *is an element (a truth-value) of* L*.*
3. *The head of the rule* A *is a propositional symbol of* Π*.*
4. *The body formula* $\mathcal{B}$ *is a formula of* $\mathfrak{F}$ *built from propositional symbols* $B_1, \dots, B_n$ $(n \geq 0)$ *by the use of conjunctors* $\&_1, \dots, \&_m$ *and* $\wedge_1, \dots, \wedge_k$*, disjunctors* $\vee_1, \dots, \vee_L$ *and aggregators* $@_1, \dots, @_r$*.*
5. *Facts are rules with body* $\top$*.*

Occasionally, a rule $A \leftarrow_i \mathcal{B}$ will be written as $A \leftarrow @[B_1, \dots, B_n]$, with $@$ being the aggregation of all operational symbols in $\mathcal{B}$, where $B_1, \dots, B_n$ are propositional symbols present in $\mathcal{B}$. Therefore, $\mathcal{B} = @[B_1, \dots, B_n]$.

Example 2.13. *Taking into account Example 2.7 and associating each operator symbol in* Ω *with their respective interpretions*[2]*, the following multi-adjoint* Ω*-algebra can be considered.*

$$\langle [0,1], \leq, \leftarrow_G, \&_G, \leftarrow_P, \&_P, \wedge_L \rangle,$$

where $\&_G$ *and* $\&_P$ *are the Gödel and the product conjunctors respectively, with their respective implications. Furthermore, we utilize the Łukasiewicz conjunctor* $\wedge_L$ *as an additional operator. We consider the set of propositional symbols* Π *as*

$$\Pi = \{\texttt{Spray water}, \texttt{Sunlight}, \texttt{Concentration of } CO_2, \texttt{Temperature}, \texttt{Humidity}\}.$$

The following multi-adjoint logic program $\mathbb{P}$ *can be considered under certain circumstances.*

$$\begin{aligned} &\texttt{Spray water} \overset{0.8}{\leftarrow}_G \texttt{Temperature} \wedge_L \texttt{Sunlight} \\ &\texttt{Spray water} \overset{0.9}{\leftarrow}_P \texttt{Concentration of } CO_2 \\ &\texttt{Spray water} \overset{0.2}{\leftarrow}_P \texttt{Humidity} \end{aligned}$$

For each individual different data/measures are obtained, these facts complement the set of rules in the program, which are formally represented as follows.

$$\begin{aligned} \texttt{Temperature} &\overset{0.6}{\leftarrow}_P 1 \\ \texttt{Sunlight} &\overset{0.8}{\leftarrow}_P 1 \\ \texttt{Concentration of } CO_2 &\overset{0.6}{\leftarrow}_P 1 \\ \texttt{Humidity} &\overset{0.7}{\leftarrow}_P 1 \end{aligned}$$

---

[2] We will avoid to put the dot on every symbol in order to simplify the notation.

2.4 Semantics of propositional multi-adjoint logic programming

Semantics of multi-adjoint logic programming will enable us to provide significance to the statements considered through their syntactic structure, and establishes an inference system to obtain which deductions or consequences are correct.

Definition 2.14. [19] *An* interpretation *is a mapping* $I\colon \Pi \longrightarrow L$ *which assigns values of the complete lattice* L *to the set of propositional symbols. The set of all interpretations of the formulas defined by the* Ω*-algebra* $\mathfrak{F}$ *in the* Ω*-algebra* $\mathfrak{L}$ *is denoted as* $\mathcal{I}_{\mathfrak{L}}$*.*

From the homomorphic extension theorem (Theorem 2.11), an extended interpretation $\hat{I}\colon F_\Omega \longrightarrow L$ can be defined on the set of formulas F_Ω from each interpretation $I\colon \Pi \longrightarrow L$. Moreover, the ordering $\preceq$ of the lattice $\langle L, \preceq\rangle$ is naturally extended to $\mathcal{I}_{\mathfrak{L}}$. The set of interpretations $\mathcal{I}_{\mathfrak{L}}$ together with the point-wise ordering forms a complete lattice [19], denoted by $\langle \mathcal{I}_{\mathfrak{L}}, \sqsubseteq\rangle$. The least interpretation $\vartriangle$ maps every propositional symbol to the least element $\perp$ of L.

Definition 2.15. [19] *Given an interpretation* $I \in \mathcal{I}_{\mathfrak{L}}$*, a weighted rule* $\langle A \leftarrow_i \mathcal{B}, \vartheta\rangle$ *is satisfied by* I *if and only if* $\vartheta \preceq \hat{I}\,(A \leftarrow_i \mathcal{B})$*. An interpretation* $I \in \mathcal{I}_{\mathfrak{L}}$ *is a model of a multi-adjoint logic program* $\mathbb{P}$ *if and only if all weighted rules in* $\mathbb{P}$ *are satisfied by* I*.*

Next, we present the definition of the immediate consequences operator [8] in the framework of multi-adjoint logic programming.

Definition 2.16. [19] *Let* $\mathbb{P}$ *be a multi-adjoint logic program. The immediate consequences operator* $T_{\mathbb{P}}^{\mathfrak{L}}\colon \mathcal{I}_{\mathfrak{L}} \longrightarrow \mathcal{I}_{\mathfrak{L}}$*, mapping interpretations to interpretations, is defined by considering*

$$T_{\mathbb{P}}^{\mathfrak{L}}(I)(A) = \sup\left\{\vartheta \dot{\&}_i \hat{I}(\mathcal{B}) \mid A \overset{\vartheta}{\longleftarrow}_i \mathcal{B} \in \mathbb{P}\right\}.$$

As the omission of $\mathfrak{L}$ is not misleading, we will use the notation $T_{\mathbb{P}}$ when we refer to the immediate consequences operator. Notice that the supremum of the definition exists because $\langle L, \preceq\rangle$ is a complete lattice. Next, we recall several important results that will allow us to define the semantics of multi-adjoint logic programming.

Theorem 2.17. [19] *An interpretation* $I \in \mathcal{I}_{\mathfrak{L}}$ *is a model of a multi-adjoint logic program* $\mathbb{P}$ *if and only if* $T_{\mathbb{P}}(I) \sqsubseteq I$*.*

Theorem 2.18. [19] *The operator* $T_{\mathbb{P}}$ *is monotonic.*

Taking into account that $T_{\mathbb{P}}$ is monotonic, together with the fact that $\mathcal{I}_{\mathfrak{L}}$ is a complete lattice, we obtain that $T_{\mathbb{P}}$ has a least fixed point.

Proposition 2.19. [28] *The operator* $T_{\mathbb{P}}\colon \mathcal{I}_{\mathfrak{L}} \longrightarrow \mathcal{I}_{\mathfrak{L}}$ *has a least fixed point* $\mathit{lfp}(T_{\mathbb{P}})$ *verifying that*

$$\mathit{lfp}(T_{\mathbb{P}}) \quad = \quad \inf\{I \in \mathcal{I}_{\mathfrak{L}} \mid T_{\mathbb{P}}(I) = I\} = \inf\{I \in \mathcal{I}_{\mathfrak{L}} \mid T_{\mathbb{P}}(I) \sqsubseteq I\}.$$

Theorem 2.17 allows us to relate the least model $M_{\mathbb{P}}$ with the least fixed point of the $T_{\mathbb{P}}$. Furthermore, the following result shows that it can be reached by iterating an ordinal number of times α the operator $T_{\mathbb{P}}$, starting from the least interpretation $\vartriangle$.

Theorem 2.20. [19] *Let* L *be a complete lattice and* $\mathbb{P}$ *a multi-adjoint logic program. Then, for some ordinal* α *it is satisfied that* $M_{\mathbb{P}} = \mathit{lfp}\,(T_{\mathbb{P}}) = T_{\mathbb{P}}^{\alpha}(\vartriangle)$*.*

Example 2.21. *From Example 2.13 and Theorem 2.20, we will explain how the mapping* $T_{\mathbb{P}}$ *is computed and the least model of a program is obtained.*

$$\begin{aligned}
T_{\mathbb{P}}(\vartriangle)(\texttt{Spray Water}) &= \sup\left\{0.9\dot{\&}_P\widehat{\vartriangle}(CO_2), 0.2\dot{\&}_P\widehat{\vartriangle}(\texttt{Humidity}), 0.8\dot{\&}_G\widehat{\vartriangle}(\texttt{Temperature} \wedge_L \texttt{Sunlight})\right\}\\
&= 0.\\
T_{\mathbb{P}}^2(\vartriangle)(\texttt{Spray Water}) &= \sup\left\{0.9\dot{\&}_P\widehat{T_{\mathbb{P}}}(\vartriangle)(CO_2), 0.7\dot{\&}_P\widehat{T_{\mathbb{P}}}(\vartriangle)(\texttt{Humidity}), 0.5\dot{\&}_G\widehat{T_{\mathbb{P}}}(\vartriangle)(\texttt{Temperature} \wedge_L \texttt{Sunlight})\right\}\\
&= \sup\{0.54, 0.49, 0.4\}\\
&= 0.54.
\end{aligned}$$

The next table displays the iteration of the mapping of $T_{\mathbb{P}}$*. Notice that the least fixed point is reached in the second iteration. As a consequence, by Theorem 2.20, the least model of* $\mathbb{P}$ *is exactly* $T_{\mathbb{P}}^2(\vartriangle)$*.*

	$\triangle$	$T_{\mathbb{P}}(\triangle)$	$T_{\mathbb{P}}^2(\triangle)$	$T_{\mathbb{P}}^3(\triangle)$
Concentration of CO_2	0	0.6	0.6	0.6
Humidity	0	0.7	0.7	0.7
Sunlight	0	0.8	0.8	0.8
Temperature	0	0.6	0.6	0.6
Spray Water	0	0	0.54	0.54

Furthermore, this example shows that any interpretation greater than the least fixed point of $T_{\mathbb{P}}$ must not be a model of the program.

2.5 Continuity

In the multi-adjoint logic programming framework, the continuity is defined through the Scott-continuity, which is based on the notion of directed set. We recall some important definitions and results, which can be found in [18] and [19], for instance.

Definition 2.22. [18] *Let $\langle L, \preceq\rangle$ be a lattice and $X \subseteq L$. We say that X is directed if each finite subset of X has an upper bound in X.*

Definition 2.23. [18] *Let L be a complete lattice and $f\colon L \to L$ a mapping. We say that f is Scott-continuous if it preserves suprema of directed sets, that is, given a directed set X we have that*

$$f(\sup X) = \sup\{f(x) \mid x \in X\}.$$

A multivariable mapping $g\colon L^n \to L$ is said to be Scott-continuous if it is Scott-continuous in each argument separately.

The following result, typically ascribed to Kleene, provides the main reason why the continuity of the immediate consequences operator is really important.

Proposition 2.24. [18] *Let $\langle L, \preceq\rangle$ be a complete lattice and $T\colon L \to L$ a Scott-continuous mapping. Then, lfp(T) can be reached in numerable iterations ω of the operator T from the least element of the lattice.*

Next, we will consider the specific case of the multi-adjoint framework in order to recall a result we will use later in the continuity section (Section 4).

Definition 2.25. [19] *Let $\mathfrak{F}$ be a language interpreted on a multi-adjoint Ω-algebra $\mathfrak{L}$, and ω any operator symbol in the language. We say that ω is continuous if its interpretation under $\mathfrak{L}(\dot{\omega})$ is continuous in L.*

Lemma 2.26. [19] *Let $\mathbb{P}$ be a program interpreted on a multi-adjoint Ω-algebra $\mathfrak{L}$, and let $\mathcal{B}$ be any body formula in $\mathbb{P}$. Assume that all the operators @ in $\mathcal{B}$ are continuous, let X be a directed set of interpretations, and write $S = \sup X$; then*

$$\hat{S}(\mathcal{B}) = \sup\left\{\hat{J}(\mathcal{B}) \mid J \in X\right\}.$$

One important remark is that in this paper, due to the fact that we are considering OWA operators on the unit interval, the immediate consequences operator will be defined on this linear lattice. As a consequence, the Scott-continuity coincides with the usual left-continuity on real numbers. Therefore, it is not necessary either to study the other interesting notion of continuity on lattices: the lower semi-continuity (LSC) [19], since it also coincides with the left-continuity on real numbers. Thus, since these three notions of continuity coincides, we will only analyze the Scott-continuity of the new operator proposed in this paper.

2.6 Ordered weighted averaging aggregation operators

The generalization of the immediate consequences operator that will be studied in this paper is based on the concept of ordered weighted averaging aggregation operator, *OWA* from now on [15, 17, 31, 32].

Definition 2.27. [32] *An n-dimensional* OWA operator *is a mapping* $@\colon [0,1]^n \longrightarrow [0,1]$*, associated with a weights vector* $W = (w_1, \dots, w_n)$*, verifying that* $w_1 + \ldots + w_n = 1$ *and* $0 \le w_i \le 1$*, for all* $i \in \{1, \dots, n\}$*, such that*

$$@(a_1, \dots, a_n) = \sum_{i=1}^{n} w_i a_{\pi(i)}, \tag{1}$$

where π is a permutation of $\{a_1, \dots, a_n\}$ satisfying that $a_{\pi(n)} \le \ldots \le a_{\pi(1)}$.

In order to understand the great flexibility and versatility of this operator, in the following definition we present different kinds of it, which will be useful when generalizing the immediate consequences operator.

Example 2.28. *Different particular OWA operators are presented.*

1. *Consider the weights vector* $W^* = (1, 0, \ldots, 0)$. *It can easily be proved that* $@^* (a_1, \ldots, a_n) = \max \{a_i \mid 1 \le i \le n\}$. *In this case, we say that* $@^*$ *is the* maximum OWA.
2. *Taking into account the weights vector* $W_* = (0, \ldots, 0, 1)$, *we can obtain that* $@_* (a_1, \ldots, a_n) = \min \{a_i \mid 1 \le i \le n\}$. *In this case,* $@_*$ *is called the* minimum OWA. *OWA operators where the weights vector* w *verifies that* $w_j = 1$ *for some* j *and* $w_i = 0$ *for all* $i \neq j$ *are known as* step-OWAs.
3. *Given the weights vector* $W_{av} = (1/n, 1/n, \ldots, 1/n)$, *it is verified that* $@_{av} (a_1, \ldots, a_n) = \dfrac{1}{n}\sum_{i=1}^{n} a_i$. *In this case, the operator* $@_{av}$ *receives the name of* average *of* $\{a_1, a_2, \ldots, a_n\}$.
4. *A* window-OWA *is an OWA operator where the weights vector* $W = (w_1, \ldots, w_n)$ *verifies that there exist* k, $m \in \{1, \ldots, n\}$ *such that* $k + m \le n + 1$, $w_i = 0$ *for all* $i < k$ *and* $k + m \le i$, *and* $w_i = \frac{1}{m}$ *for all* $k \le i < k + m$.

Now, we recall the following result, which shows that the set of OWA operators is bounded by the maximum OWA (upper bound) and the minimum OWA (lower bound).

Proposition 2.29. [32] *For all OWA operator* $@$, *it is verified that*

$$@_* (a_1, \ldots, a_n) \le @ (a_1, \ldots, a_n) \le @^* (a_1, \ldots, a_n),$$

for all $(a_1, \ldots, a_n) \in [0, 1]^n$.

Finally, the following result shows that the OWA operator is monotonic.

Theorem 2.30. [32] *Let* $@_W$ *be an OWA operator associated with the weights vector* W. *Given two values vectors* $(a_1, \ldots, a_n)$ *and* $(b_1, \ldots, b_n)$, *ordered such that, for each* $j \in \{1, \ldots, n\}$, *it is satisfied that* $a_j \le b_j$, *then*

$$@_W (a_1, \ldots, a_n) \le @_W (b_1, \ldots, b_n).$$

2.6.1 Orness measure

The orness measure [15, 32] is an interesting notion for classifying different OWA operators depending on the "distance" from the maximum (which is also called "or" operator). This notion will also be considered in this paper.

Definition 2.31. [32] *(Orness). We define an orness measure as the mapping orness*: $[0, 1]^n \longrightarrow [0, 1]$ *defined as follows*

$$\mathit{orness}(W) = \frac{1}{n-1} \sum_{i=1}^{n} ((n-i) w_i).$$

Notice that the orness value will depend on the weights vector W considered. In the following example, we will see the orness value of the different types of OWA operator, given in Definition 2.28.

Example 2.32. *The following statements hold.*

1. *For* $W^* = (1, 0, \ldots, 0)$ *it is verified that* $\mathit{orness}(W^*) = 1$.
2. *For* $W_* = (0, \ldots, 0, 1)$ *it is satisfied that* $\mathit{orness}(W_*) = 0$.
3. *For* $W_{av} = (1/n, 1/n, \ldots, 1/n)$ *the orness measure is* $\mathit{orness}(W_{av}) = 1/2$.
4. *For* $W = (1/2, 1/2, 0, \ldots, 0)$ *it is verified that* $\mathit{orness}(W) = \frac{2n-3}{2n-2}$. *Clearly,* $\frac{1}{2} \le \mathit{orness}(W)$ *if* $2 \le n$ *and the orness of* W *converges to* 1, *when the value of* n *scales up.*

The following result shows a sufficient condition to obtain OWA operators with high (or low) orness.

Theorem 2.33. [32] *Let* $@$ *be an OWA operator. If the weights verify that* $w_{n-i+1} \le w_i$, *then* $\frac{1}{2} \le \mathit{orness}(@)$. *On the contrary, if the weights verify that* $w_i \le w_{n-i+1}$, *then* $\mathit{orness}(@) \le \frac{1}{2}$.

3 Immediate consequences operator generalized through OWA operators

The objective of this section is to give a new definition of the immediate consequences operator that allows reducing the impact of the possible presence of noise in the data. For that, we will use OWA operators. First of all, we will explain the difficulty that arises with the definition of the $T_{\mathbb{P}}$ through the supremum.

Assume that in a specific situation we have that the set, from which the supremum is to be calculated, is composed of the values a_1, a_2 to a_n. These values have been obtained through observation. It is possible that a simple error in the calculation or computation of the data could lead to a value that is significantly higher than the rest of the values. Consequently, the value of the $T_{\mathbb{P}}$ would not be representative, since it is disturbed by the presence of noise in the data. This situation also serves to exemplify the great dependence of the $T_{\mathbb{P}}$ from only one observed data. Therefore, in certain cases, small variations in the number of observed data can lead to large variations in the value of the $T_{\mathbb{P}}$.

From now on, a finite and non-empty multi-adjoint logic program $\mathbb{P}$ over a set of propositional symbols Π, with cardinal $n \in \mathbb{N}^*$, and a multi-adjoint Ω-algebra $\langle [0,1], \leq, \&_1, \dots, \&_m \rangle$ will be fixed.

First of all, given a propositional symbol A and the interpretation I, we present the technical definition of the mapping C_I^A, in order to generalize the $T_{\mathbb{P}}$ using OWA operators, which also was used in other setting in [23].

Definition 3.1. [23] *(*Mapping C_I^A*). Let I be an interpretation on $\mathbb{P}$. We define the mapping $C_I^A \colon \mathbb{P} \longrightarrow [0,1]$, for each $\langle C \leftarrow_i \mathcal{B}, \vartheta \rangle \in \mathbb{P}$, as follows*

$$C_I^A\left(\langle C \leftarrow_i \mathcal{B}, \vartheta\rangle\right) = \begin{cases} \vartheta \dot{\&}_i \hat{I}(\mathcal{B}) & \textit{if } C = A \\ 0 & \textit{otherwise} \end{cases}$$

These mappings are monotonic on the set of interpretations $\mathcal{I}_{\mathfrak{L}}$ [23, Proposition 4]. In the following subsection, we are going to present the generalization of the immediate consequences operator, given by Definition 2.16, through OWA operators.

3.1 Extending the $T_{\mathbb{P}}$ through an arbitrary OWA operator

Taking into account Definition 2.27, we present the following generalization of the $T_{\mathbb{P}}$ using OWA operators.

Definition 3.2. *(*OWA immediate consequences operator*). Let $W \in [0,1]^n$ be a weights vector, with $\sum_{i=1}^{n} w_i = 1$. The* OWA immediate consequences operator *$T_{\mathbb{P}}^W \colon \mathcal{I}_{\mathfrak{L}} \longrightarrow \mathcal{I}_{\mathfrak{L}}$ is defined, for all $I \in \mathcal{I}_{\mathfrak{L}}$ and $A \in \Pi$, as*

$$T_{\mathbb{P}}^W(I)(A) \quad = \quad @_W\left(C_I^A(u_1), \dots, C_I^A(u_n)\right), \tag{2}$$

where the mapping C_I^A is given by Definition 3.1 and $\{u_1, \dots, u_n\}$ is the set of rules of $\mathbb{P}$.

The following result shows that $T_{\mathbb{P}}^W$ is really a generalization of the immediate consequences operator.

Proposition 3.3. *It is satisfied that*

$$T_{\mathbb{P}}^{W^*}(I)(A) = T_{\mathbb{P}}(I)(A),$$

where $W^* = (1, 0, \dots, 0)$ (Example 2.32) and C_I^A is the mapping given by Definition 3.1.

Proof. Given an interpretation I, a propositional symbol A, and the set of rules with head A, $\mathbb{P}_A$, we obtain that

$$\begin{aligned} T_{\mathbb{P}}^{W^*}(I)(A) &= @_{W^*}\left(C_I^A(u_1), \dots, C_I^A(u_n)\right) \\ &= \max\left\{\vartheta \dot{\&}_i \hat{I}(\mathcal{B}) \mid A \overset{\vartheta}{\leftarrow}_i \mathcal{B} \in \mathbb{P}\right\} \\ &\overset{(*)}{=} T_{\mathbb{P}}(I)(A), \end{aligned}$$

where $(*)$ holds because of the unit interval is totally ordered and $\mathbb{P}$ is finite. Notice that, if $\mathbb{P}_A = \varnothing$, then $C_I^A(u) = 0$, for all $u \in \mathbb{P}$ and trivially $T_{\mathbb{P}}^{W^*}(I)(A) = 0 = T_{\mathbb{P}}(I)(A)$ holds. ☐

Therefore, if the maximum OWA is considered, then the classical definition of the immediate consequences operator in a fuzzy framework is obtained (Definition 2.16). Next, we prove that $T_{\mathbb{P}}^W$ is monotonic.

Proposition 3.4. *The operator $T_{\mathbb{P}}^W$, given by Definition 2, is increasing in $\mathcal{I}_{\mathfrak{L}}$.*

Proof. Given two interpretations I_1, $I_2 \in \mathcal{I}_{\mathfrak{L}}$, such that it is verified that $I_1 \sqsubseteq I_2$, we have to prove that $T^W_{\mathbb{P}}(I_1) \sqsubseteq T^W_{\mathbb{P}}(I_2)$. Given $A \in \Pi$ and $\mathbb{P} = \{u_1, \dots, u_n\}$, we have that

$$\begin{aligned} T^W_{\mathbb{P}}(I_1)(A) &= @_W\left(C^A_{I_1}(u_1), \dots, C^A_{I_1}(u_n)\right) \\ &\overset{(*)}{\leq} @_W\left(C^A_{I_2}(u_1), \dots, C^A_{I_2}(u_n)\right) \\ &= T^W_{\mathbb{P}}(I_2)(A), \end{aligned}$$

where $(*)$ holds because of the mappings C^A_I are increasing on $\mathcal{I}_{\mathfrak{L}}$, by Theorem 2.30. Thus, $T^W_{\mathbb{P}}$ is increasing.

□

Next, we will study a certain example of the operator $T^W_{\mathbb{P}}$ for a better comprehension of its definition, with the aim of characterizing the weights vectors to generalize the immediate consequences operator for solving the problem of the possible presence of noise in the data. In this case, we are going to consider a weights vector in order to stay with the arithmetic mean of the two greatest values.

Example 3.5. *Assuming that the cardinal of* $\mathbb{P}$ *is greater or equal than* 2*, we can consider the following particular OWA immediate consequences operator*

$$T^{W_2}_{\mathbb{P}}(I)(A) = @_{W_2}\left(C^A_I(u_1), \dots, C^A_I(u_n)\right),$$

where the mapping C^A_I *is given by Definition 3.1, and* $W_2 = \left(\frac{1}{2}, \frac{1}{2}, 0, \overset{(n-2)}{\dots}, 0\right)$. *The OWA operator with the weights vector given by* W_2 *is a particular case of window-OWA.*

The usual next step is to analyze the semantics of the new immediate consequences operator $T^W_{\mathbb{P}}$. Before that, we will studied the specific case of $T^{W_2}_{\mathbb{P}}$, where the weights vector W_2 was introduced in Example 3.5, in order to facilitate the comprehension of the semantics in the general case.

3.2 The fixed point semantics associated with $T^{W_2}_{\mathbb{P}}$

Since the definition of $T_{\mathbb{P}}$ has been relaxed, the notion of model must be adapted to the new philosophy. In this section we will be focused on the particular weight W_2, and we will assume that $2 \leq |\mathbb{P}|$.

Definition 3.6. *(*W_2-model*). We say that an interpretation* $I \in \mathcal{I}_{\mathfrak{L}}$ *is a* W_2*-model of the program* $\mathbb{P}$ *if all the rules are satisfied by* I *except possibly one of them, in that case it is verified that the mean of the two highest values* $\vartheta \dot{\&} \hat{I}(\mathcal{B})$*, with* $\langle A \leftarrow \mathcal{B}, \vartheta \rangle \in \mathbb{P}$*, is less than the value of the interpretation in the propositional symbol* A*.*

The following theorem presents the characterization of W_2-models of a program $\mathbb{P}$ as the post-fixed points of $T^{W_2}_{\mathbb{P}}$.

Theorem 3.7. *An interpretation* I *is a* W_2*-model for* $\mathbb{P}$ *if and only if* $T^{W_2}_{\mathbb{P}}(I) \sqsubseteq I$*.*

Proof. Given the set of rules $\{u_1, \dots, u_n\}$ of $\mathbb{P}$, and the permutation π considered by the OWA operator (Definition 2.27). If all the rules are satisfied by I, then in particular it is verified that $C^A_I\left(u_{\pi(1)}\right)$, $C^A_I\left(u_{\pi(2)}\right) \leq I(A)$ and, therefore

$$T^{W_2}_{\mathbb{P}}(I)(A) = \frac{C^A_I\left(u_{\pi(1)}\right) + C^A_I\left(u_{\pi(2)}\right)}{2} \leq I(A).$$

On the contrary, if some rule is not satisfied by I, then it must necessarily be $u_{\pi(1)}$. But, as I is a W_2-model, from Definition 3.6, we obtain that

$$T^{W_2}_{\mathbb{P}}(I)(A) = \frac{C^A_I\left(u_{\pi(1)}\right) + C^A_I\left(u_{\pi(2)}\right)}{2} \leq I(A).$$

Now, in order to prove the reciprocal, suppose that the interpretation I satisfies that $T^{W_2}_{\mathbb{P}}(I) \sqsubseteq I$. Taking into account that $C^A_I\left(u_{\pi(2)}\right) \leq C^A_I\left(u_{\pi(1)}\right)$, it is verified that

$$C^A_I\left(u_{\pi(n)}\right) \leq \dots \leq C^A_I\left(u_{\pi(2)}\right) \leq \frac{C^A_I\left(u_{\pi(1)}\right) + C^A_I\left(u_{\pi(2)}\right)}{2} \leq I(A).$$

Consequently, all the rules of $\mathbb{P}$ are satisfied by I except possibly $u_{\pi(1)}$, in which case it is verified that

$$\frac{C^A_I\left(u_{\pi(1)}\right) + C^A_I\left(u_{\pi(2)}\right)}{2} \leq I(A).$$

□

We can characterize the least W_2-model as the least fixed point of the operator $T_{\mathbb{P}}^{W_2}$, using Theorem 3.7. First of all, the following proposition holds.

Proposition 3.8. *The interpretation* $M_{\mathbb{P}} \colon \Pi \longrightarrow [0,1]$ *defined as* $M_{\mathbb{P}} = \inf\{I \in \mathcal{I}_{\mathfrak{L}} \mid I \textit{ is a } W_2\textit{-model of}\mathbb{P}\}$ *is a* W_2*-model of the program* $\mathbb{P}$*, which is the least* W_2*-model of* $\mathbb{P}$ *and* $M_{\mathbb{P}} = \mathit{lfp}\left(T_{\mathbb{P}}^{W_2}\right)$.

Proof. In the first place, we can consider the infimum of the set

$$\{I \in \mathcal{I}_{\mathfrak{L}} \mid I \text{ is a } W_2\text{-model of } \mathbb{P}\},$$

which exists because of $\langle \mathcal{I}_{\mathfrak{L}}, \sqsubseteq\rangle$ is a complete lattice. As $T_{\mathbb{P}}^{W_2}$ is a monotonic mapping, then by Proposition 2.19 and Theorem 3.7 it is verified that

$$\begin{aligned} \operatorname{lfp}\left(T_{\mathbb{P}}^{W_2}\right) &= \inf\left\{I \in \mathcal{I}_{\mathfrak{L}} \mid T_{\mathbb{P}}^{W_2}(I) \sqsubseteq I\right\} \\ &= \inf\{I \in \mathcal{I}_{\mathfrak{L}} \mid I \text{ is a } W_2\text{-model of } \mathbb{P}\} = M_{\mathbb{P}}. \end{aligned}$$

Therefore, by the definition of fixed point, it is verified that $T_{\mathbb{P}}^{W_2}\left(\operatorname{lfp}\left(T_{\mathbb{P}}^{W_2}\right)\right) = \operatorname{lfp}\left(T_{\mathbb{P}}^{W_2}\right)$, and so $\operatorname{lfp}\left(T_{\mathbb{P}}^{W_2}\right)$ is a W_2-model of $\mathbb{P}$ from Theorem 3.7. In conclusion, $M_{\mathbb{P}}$ is a W_2-model of $\mathbb{P}$, being the least W_2-model of $\mathbb{P}$. ☐

Lastly, the generalization of Theorem 2.20 is given.

Theorem 3.9. *For some ordinal* α*, it is satisfied that* $M_{\mathbb{P}} = \mathit{lfp}\left(T_{\mathbb{P}}^{W_2}\right) = T_{\mathbb{P}}^{\alpha}(\triangle)$.

Proof. This result can easily be proved from Proposition 3.8 and Theorem 2.20. ☐

3.3 Fixed-point semantics of the general case $T_{\mathbb{P}}^{W}$

In this section, we will introduce the fixed-point semantics of a multi-adjoint logic program considering an arbitrary OWA immediate consequences operator through a general weights vector W.

Owing to the intricateness of its syntax (Definition 3.2), it is not easy to define a notion of model for every vector W. As a consequence, considering the results presented in Section 3.1, the following general definition of model related to a weights vector W is presented.

Definition 3.10. *(*W-model*). We say that an interpretation* $I \in \mathcal{I}_{\mathfrak{L}}$ *is a* W-model *of a multi-adjoint logic program* $\mathbb{P}$ *if and only if it is a post-fixed point of* $T_{\mathbb{P}}^{W}$.

From the previous definition, it is clear that the least W-model of the program $\mathbb{P}$ is the least fixed point of $T_{\mathbb{P}}^{W}$. Furthermore, it is satisfied that the set of W-models with the point-wise order forms a complete lattice.

3.4 Generalization of $T_{\mathbb{P}}$ on a family of OWA operators

The sections above have conveniently generalized the immediate consequences operator through OWA operators, providing a more flexible fixed-point semantics for the obtention of information of data sets. Nevertheless, a problem related to the number of rules with head a propositional symbol can arise.

Assume that the program $\mathbb{P}$ is composed of a single rule $\langle A \leftarrow B, \vartheta\rangle$, i.e., $|\mathbb{P}| = 1$, and let $T_{\mathbb{P}}^{W_1}$ be the OWA immediate consequences operator associated with the vector $W_1 = (0, 1, 0, \ldots, 0)$. As we have only one rule, then the value of $T_{\mathbb{P}}^{W_1}(I)(A)$ would always be 0, for all $I \in \mathcal{I}_{\mathfrak{L}}$. Therefore, we have a problem with those propositional symbols heads of a single rule. One possible solution is to consider a weights vector given by

$$\tau(A) = \begin{cases} 1 & \text{if } \ |\mathbb{P}_A| = 1 \\ (\alpha_1, \alpha_2) & \text{if } \ |\mathbb{P}_A| = 2, \quad \text{with } \ \alpha_1 + \alpha_2 = 1 \\ \left(\alpha_1, \alpha_2, \alpha_3, \overset{(n)}{\ldots}, \alpha_n\right) & \text{if } \ |\mathbb{P}_A| = n \geq 3, \\ & \qquad \text{with } \ \alpha_1 + \ldots + \alpha_n = 1 \end{cases} \tag{3}$$

where $\mathbb{P}_A$ is the subset of rules with head A. For example, we can consider the following particular OWA immediate consequences operator

$$T_{\mathbb{P}}^{\tau}(I)(A) = @_{\tau(A)}\left(C_I^A(u_1), \ldots, C_I^A(u_n)\right),$$

where $\mathbb{P}_A = \{u_1, \dots, u_n\}$ and $\tau\colon \Pi \longrightarrow \left\{(1), \left(\frac{1}{2}, \frac{1}{2}\right), \dots, \left(\frac{1}{2}, \frac{1}{2}, 0, \overset{(n-2)}{\dots}, 0\right)\right\}$ is defined as

$$\tau(A) = \begin{cases} 1 & \text{if} \quad |\mathbb{P}_A| = 1 \\ \left(\dfrac{1}{2}, \dfrac{1}{2}\right) & \text{if} \quad |\mathbb{P}_A| = 2 \\ \left(\dfrac{1}{2}, \dfrac{1}{2}, 0, \overset{(m-2)}{\dots}, 0\right) & \text{if} \quad 3 \leq |\mathbb{P}_A| = m \end{cases} \tag{4}$$

Clearly this new definition is increasing in $\mathcal{I}_{\mathfrak{L}}$. The OWA operator with the weights vector given by τ is a particular case of window-OWA.

Observe that in this case we have a clear distinction between $|\mathbb{P}_A| = 1$ and $|\mathbb{P}_A| > 1$. This idea is also captured with the following definition, which provides with a more versatile solution based on considering a weights vector for each propositional symbol.

Definition 3.11. *(*$\mathcal{W}$-OWA immediate consequences operator*). Given a family of weights vectors*[3]

$$\mathcal{W} = \left\{ W_A \in [0,1]^{n_A} \mid \sum_{i=1}^{n_A} W_A[i] \leq 1, A \in \Pi, \mathbb{P}_A \neq \varnothing, |\mathbb{P}_A| = n_A \right\},$$

where $n_A \in \mathbb{N}^*$*. We define the* $\mathcal{W}$-OWA immediate consequences operator obtained from the family $\mathcal{W}$ *as the mapping* $T_{\mathbb{P}}^{\mathcal{W}}\colon \mathcal{I}_{\mathfrak{L}} \to \mathcal{I}_{\mathfrak{L}}$ *defined by*

$$T_{\mathbb{P}}^{\mathcal{W}}(I)(A) = \begin{cases} @_{W_A}\left(C_I^A(u_1), \dots, C_I^A(u_{n_A})\right) & \textit{if} \quad \mathbb{P}_A \neq \varnothing \\ 0 & \textit{if} \quad \mathbb{P}_A = \varnothing \end{cases}. \tag{5}$$

where $\mathbb{P}_A = \{u_1, \dots, u_{n_A}\}$*, for all* $A \in \Pi$ *and* $I \in \mathcal{I}_{\mathfrak{L}}$*.*

In the following result, we present the relation between the previous definition and Definition 2.

Proposition 3.12. *Let* $\mathbb{P}$ *be a multi-adjoint logic program and* W *a weights vector, then there exists a family of weights vectors*

$$\mathcal{W} \quad = \quad \left\{ W_A \in [0,1]^{n_A} \mid \sum_{i=1}^{n_A} W_A[i] \leq 1,\ A \in \Pi,\ \mathbb{P}_A \neq \varnothing, |\mathbb{P}_A| = n_A \right\},$$

such that $T_{\mathbb{P}}^{W} = T_{\mathbb{P}}^{\mathcal{W}}$*.*

Proof. Given a weights vector W, we can consider the family of weights vectors

$$\mathcal{W} = \left\{ W_A \in [0,1]^{n_A} \mid \sum_{i=1}^{n_A} W_A[i] \leq 1,\ A \in \Pi,\ \mathbb{P}_A \neq \varnothing, |\mathbb{P}_A| = n_A \right\},$$

where the weights vector W_A is defined as $W_A[i] = W[i]$, for all $1 \leq i \leq n_A$, $A \in \Pi$. Since $\displaystyle\sum_{i=1}^{n} W[i] = 1$, then clearly $\displaystyle\sum_{i=1}^{n_A} W_A[i] \leq 1$, for all $A \in \Pi$.

Given $A \in \Pi$, if $|\mathbb{P}_A| = 0$, then the equality straightforwardly holds, because of $T_{\mathbb{P}}^{\mathcal{W}}(I)(A) = 0$, by definition, and $T_{\mathbb{P}}^{W}(I)(A) = 0$, by the boundary condition of the aggregator operator with the bottom element.

Otherwise, the set $\mathbb{P}_A$ is not empty and, given π_A and π the corresponding permutations, the following equalities holds for every interpretation $I \in \mathcal{I}_{\mathfrak{L}}$.

$$\begin{aligned} T_{\mathbb{P}}^{\mathcal{W}}(I)(A) &= T_{\mathbb{P}}^{W_A}(I)(A) = \sum_{i=1}^{n_A} W_A[i] C_I^A(u_{\pi_A(i)}) = \sum_{i=1}^{n_A} W[i] C_I^A(u_{\pi_A(i)}) + \sum_{i=n_A}^{n} W[i] \cdot 0 \\ &= \sum_{i=1}^{n_A} W[i] C_I^A(u_{\pi(i)}) + \sum_{i=n_A}^{n} W[i] C_I^A(u_{\pi(i)}) = \sum_{i=1}^{n} W[i] C_I^A(u_{\pi(i)}) \\ &= T_{\mathbb{P}}^{W}(I)(A). \end{aligned}$$

---

[3] $W[i]$ denotes the i-th component of the vector W. Moreover, notice that the definition of OWA can trivially be adapted to consider weights vectors verifying $\displaystyle\sum_{i=1}^{n_A} W_A[i] \leq 1$.

□

This new operator $T_{\mathbb{P}}^{\mathcal{W}}$ is also increasing, as it is proved by the next proposition. From now on, a family of weights vectors $\mathcal{W} = \{W_A \in [0,1]^{n_A} \mid \sum_{i=1}^{n_A} W_A[i] \leq 1, A \in \Pi, \mathbb{P}_A \neq \varnothing, |\mathbb{P}_A| = n_A\}$ will be fixed.

Proposition 3.13. *The immediate consequences operator $T_{\mathbb{P}}^{\mathcal{W}}$ introduced in Definition 3.11 is increasing in $\mathcal{I}_{\mathfrak{L}}$.*

Proof. The proof is similar to the one given in Proposition 3.4. □

Consequently, we have presented a new definition of immediate consequences operator based on a family of weights vectors, which verifies interesting properties such as the monotonicity.

Analogously as in Definition 3.10, the next definition of model related to a family of weights vectors W can be established.

Definition 3.14. *($\mathcal{W}$-model). An interpretation $I \in \mathcal{I}_{\mathfrak{L}}$ is a $\mathcal{W}$-model of a multi-adjoint logic program $\mathbb{P}$ if and only if it is a post-fixed point of $T_{\mathbb{P}}^{\mathcal{W}}$.*

It can be directly shown that the set of $\mathcal{W}$-models, along with the point-wise ordering, forms a complete lattice on $\langle \mathcal{I}_{\mathfrak{L}}, \sqsubseteq \rangle$.

As in the classical case (Proposition 2.19), by the monotonicity of $T_{\mathbb{P}}^{\mathcal{W}}$ (Proposition 3.13), this operator has a least fixed point $\mathrm{lfp}(T_{\mathbb{P}}^{\mathcal{W}})$ verifying that

$$\mathrm{lfp}\left(T_{\mathbb{P}}^{\mathcal{W}}\right) = \inf\left\{I \in \mathcal{I}_{\mathfrak{L}} \mid T_{\mathbb{P}}^{\mathcal{W}}(I) \sqsubseteq I\right\}.$$

The following result shows the expected characterization of the least $\mathcal{W}$-model as the least fixed point of the operator $T_{\mathbb{P}}^{\mathcal{W}}$.

Proposition 3.15. *The interpretation $M_{\mathbb{P}} \colon \Pi \longrightarrow [0,1]$ given by*

$$M_{\mathbb{P}} = \inf\{I \in \mathcal{I}_{\mathfrak{L}} \mid I \text{ is a } \mathcal{W}\text{-model of } \mathbb{P}\},$$

is a $\mathcal{W}$-model of $\mathbb{P}$, which is the least $\mathcal{W}$-model of $\mathbb{P}$ and $M_{\mathbb{P}} = lfp\left(T_{\mathbb{P}}^{\mathcal{W}}\right)$.

Proof. This result can be proved analogously to the Proposition 3.8.

□

The next result is the generalization of Theorem 2.20 adapted to the new setting.

Theorem 3.16. *Let $\mathbb{P}$ be a multi-adjoint logic program. Then, for some ordinal α, it is satisfied that*

$$M_{\mathbb{P}} = lfp\left(T_{\mathbb{P}}^{\mathcal{W}}\right) = \left(T_{\mathbb{P}}^{W}\right)^{\alpha}(\Delta).$$

Proof. This result can be easily proved using Definition 3.9 and Proposition 2.20. □

To conclude, the immediate consequences operator has been generalized in a suitable wat using OWA operators, affording greater tractability in real problems, which is usual the presence of noise (imprecision, uncertainty, etc.).

3.5 Selecting accurate weights vectors to $T_{\mathbb{P}}^{\mathcal{W}}$

It is relevant to take into account that the weights vector is essential for calculate computationally the results and that not every weights vector W is appropriate to generalize the classical immediate consequences operator. For instance, the opposite natural definition of the $T_{\mathbb{P}}$ can be obtained if the minimum OWA, given at Example 2.28, is regarded. With respect to the choice of the weights vector and, consequently, the OWA operator, the most convenient way is to select those that can ensure values close to the supremum, with the aim of not obtaining a definition that considerably differed from the original $T_{\mathbb{P}}$.

Therefore, it is advisable in any new variant of $T_{\mathbb{P}}$ to obtain values close to the greatest one. In order to catalogue the different generalizations of the immediate consequences operator through OWA operators, we will consider the orness measure [15, 32], that can also be seen as the degree of approximation to the highest value given by $@^*$ (Example 2.28).

The first consequence of this measure is that we should consider vectors $W \in [0,1]^m$, with the greatest possible values for the components, that is, $\sum_{i=1}^{m} W[i] = 1$. Moreover, Theorem 2.33 provides a sufficient condition to ensure at least a medium orness value of the considered OWA.

Next, we adapt the orness measure to the general definition of OWA immediate consequences operator.

Definition 3.17. *(*$\mathcal{W}$-orness*). We define the* $\mathcal{W}$-orness measure associated with the family $\mathcal{W}$ *as the mapping*

$$\overline{orness}\colon \Sigma(\mathbb{P}_{A_1}) \times \ldots \times \Sigma(\mathbb{P}_{A_n}) \longrightarrow \ [0,1],$$

defined as follows

$$\overline{orness}(\mathcal{W}) = \inf\{orness(W_A) \mid W_A \in \mathcal{W}\},$$

where $\Sigma(\mathbb{P}_{A_i}) = \left\{W \in [0,1]^{|\mathbb{P}_{A_i}|} \mid \sum_{j=1}^{|\mathbb{P}_{A_i}|} W[j] \le 1\right\}$*, for all* $i \in \{1,\ldots,n\}$*, and the orness operator is given by Definition 2.31. Notice that if* $|\mathbb{P}_A| < 2$*, then the aggregator operator is the maximum with orness value of* 1*, which does not affect the computation of the infimum.*

Example 3.18. *Consider Example 2.13 with the extra rule*

$$\langle \texttt{Heat Up} \leftarrow_G \texttt{Sunlight} \wedge_L \texttt{Humidity}, 0.3\rangle\,.$$

As the propositional symbol `Spray Water` *is head of three rules, but* `Heat up` *is head of a single rule, then we should not consider the same weights vector for both rules, but a family of weights vectors, such as*

$$\mathcal{W} = \{W_{\texttt{Spray Water}}, W_{\texttt{Heat Up}}\} = \{W_2, W^*\},$$

where W^* *is the maximum OWA (Example 2.28) and* $W_2 = \left(\frac{1}{2}, \frac{1}{2}, 0\right)$*, which was defined in general in Example 3.5. Consequently, considering the interpretation* I *defined as*

$$I(\texttt{Temperature}) = 0.9, \qquad I(\texttt{Sunlight}) = 0.8,$$
$$I(\texttt{Concentration of } CO_2) = 0.6, \qquad I(\texttt{Humidity}) = 0.75,$$

we obtain the following values of the OWA immediate consequences operator.

$$\begin{aligned}
T_{\mathbb{P}}^{\mathcal{W}}(I)(\texttt{Spray Water}) &= T_{\mathbb{P}}^{W_{\texttt{Spray Water}}}(I)(\texttt{Spray Water})\\
&= T_{\mathbb{P}}^{W_2}(I)(\texttt{Spray Water})\\
&\overset{(1)}{=} \frac{1}{2}\cdot C_I^{\texttt{SW}}(u_{\pi(1)}) + \frac{1}{2}\cdot C_I^{\texttt{SW}}(u_{\pi(2)}) + 0\cdot C_I^{\texttt{SW}}(u_{\pi(3)})\\
&= \frac{1}{2}\cdot 0.7 + \frac{1}{2}\cdot 0.54 + 0\cdot 0.15\\
&= 0.62\\
T_{\mathbb{P}}^{\mathcal{W}}(I)(\texttt{Heat Up}) &= T_{\mathbb{P}}^{W_{\texttt{Heat Up}}}(I)(\texttt{Heat Up})\\
&= T_{\mathbb{P}}^{W^*}(I)(\texttt{Heat Up})\\
&= 1\cdot C_I^{\texttt{Heat Up}}(u_{\pi(1)})\\
&= 1\cdot 0.3\\
&= 0.3.
\end{aligned}$$

where in (1) we have shorten the propositional symbol `Spray Water` *to* `SW`*. From Definition 3.17, we obtain that*

$$\overline{orness}(\mathcal{W}) = \inf\{orness(W_{\texttt{Spray Water}}), orness(W_{\texttt{Heat Up}})\} = \inf\{0.75, 1\} = 0.75.$$

In addition, this example can be used to show Proposition 3.12 and how Definition 3.11 is more appropriate than the first proposal (Definition 3.2). Notice that, if we consider the weights vector $\left(\dfrac{1}{2}, \dfrac{1}{2}, 0, \overset{(n-2)}{\ldots}, 0\right)$*, where* n *is the number of rules in* $\mathbb{P}$*, then we obtain that* $T_{\mathbb{P}}^{\mathcal{T}}(I)(\texttt{Spray Water}) = 0.62$ *and* $T_{\mathbb{P}}^{\mathcal{T}}(I)(\texttt{Heat Up}) = 0.15$*, where the truth value of* `Heat Up` *is penalized because of only one rule in the program with this propositional symbol as head exists.*

Taking into account Proposition 3.12, there exists the following family of weights vectors

$$\mathcal{W}_1 = \left\{W_{\texttt{Spray Water}} = \left(\frac{1}{2}, \frac{1}{2}, 0\right) \in [0,1]^3, W_{\texttt{Heat Up}} = \frac{1}{2} \in [0,1]\right\},$$

so that $T_{\mathbb{P}}^{\mathcal{W}_1}(I)(\texttt{Spray Water}) = T_{\mathbb{P}}^{\mathcal{T}}(I)(\texttt{Spray Water})$ *and* $T_{\mathbb{P}}^{\mathcal{W}_1}(I)(\texttt{Heat Up}) = T_{\mathbb{P}}^{\mathcal{T}}(I)(\texttt{Heat Up})$*.*

4 On the continuity of the $T_{\mathbb{P}}^{\mathcal{W}}$ operator

In this section, we will study conditions to ensure the continuity of the new definition of the immediate consequences operator. As we have already mentioned, we will consider the Scott-continuity definition. This property is essential in order to obtain the least fixed point in numerable (ω) iterations. In the following, we are going to consider a multi-adjoint logic program $\mathbb{P}$, a multi-adjoint Ω-algebra $\mathfrak{L} = \langle L, \mathfrak{H}\rangle$ with respect to the pairs $(\leftarrow_l, \&_l)$, for $l \in \{1, \dots, m\}$, the complete lattice $\langle [0,1], \leq\rangle$ and the family of weights vectors

$$\mathcal{W} = \left\{ W_A \in [0,1]^{n_A} \mid \sum_{i=1}^{n_A} W_A[i] \leq 1, A \in \Pi, \mathbb{P}_A \neq \varnothing, |\mathbb{P}_A| = n_A \right\},$$

where $n_A \in \mathbb{N}^*$.

Our first result is a generalization of the theorem in the classical framework. This theorem shows that if all operators in $\mathcal{W}$ are Scott-continuous, then $T_{\mathbb{P}}^{\mathcal{W}}$ is Scott-continuous.

Theorem 4.1. *If all the operators occurring in the bodies of the rules of a program $\mathbb{P}$ are Scott-continuous, and the adjoint conjunctions are also Scott-continuous in the second argument, then $T_{\mathbb{P}}^{\mathcal{W}}$ is Scott-continuous.*

Proof. We have to prove, for each directed subset of interpretations X and each atomic formula A, that

$$T_{\mathbb{P}}^{\mathcal{W}} (\sup X) (A) = \sup \left\{ T_{\mathbb{P}}^{\mathcal{W}}(J)(A) \mid J \in X \right\}. \tag{6}$$

We will denote $S = \sup X$. If $\mathbb{P}_A = \varnothing$, then $T_{\mathbb{P}}^{\mathcal{W}}(\sup X)(A) = 0 = \sup \left\{ T_{\mathbb{P}}^{\mathcal{W}}(J)(A) \mid J \in X \right\}$. If $\mathbb{P}_A \neq \varnothing$, then we have the following chain of equalities:

$$\begin{array}{rcl}
T_{\mathbb{P}}^{\mathcal{W}} (\sup X) (A) & = & T_{\mathbb{P}}^{W_A} (\sup X) (A) \\
& = & \displaystyle\sum_{i=1}^{n_A} W_A[i]\, C_S^A \left(u_{\pi_A(i)}\right) \\
& = & \displaystyle\sum_{i=1}^{n_A} W_A[i] \, \left(\vartheta_i \dot{\&}_i \widehat{S}\,(\mathcal{B}_i)\right) \\
& \overset{(1)}{=} & \displaystyle\sum_{i=1}^{n_A} W_A[i] \, \left(\vartheta_i \dot{\&}_i \sup \left\{ \hat{J}(\mathcal{B}_i) \mid J \in X \right\}\right) \\
& \overset{(2)}{=} & \displaystyle\sum_{i=1}^{n_A} W_A[i] \, \sup \left\{ \vartheta_i \dot{\&}_i \hat{J}(\mathcal{B}_i) \mid J \in X \right\} \\
& \overset{(3)}{=} & \sup \left\{ \displaystyle\sum_{i=1}^{n_A} W_A[i] \, \left(\vartheta_i \dot{\&}_i \hat{J}(\mathcal{B}_i)\right) \mid J \in X \right\} \\
& = & \sup \left\{ \displaystyle\sum_{i=1}^{n_A} W_A[i]\, C_J^A \left(u_{\pi(i)}\right) \mid J \in X \right\} \\
& = & \sup \left\{ T_{\mathbb{P}}^{W_A}(J)(A) \mid J \in X \right\} \\
& = & \sup \left\{ T_{\mathbb{P}}^{\mathcal{W}}(J)(A) \mid J \in X \right\},
\end{array}$$

where equality (1) follows from Lemma 2.26, equality (2) follows from the Scott-continuity of the operators $\&_l$ on the second argument and equality (3) follows from the fact that the product and the sum are continuous mappings and so, Scott-continuous on the unit interval, in particular.

Therefore, we can conclude that the new definition of the immediate consequences operator, $T_{\mathbb{P}}^{\mathcal{W}}$, is Scott-continuous. □

In the following example, we will show that the consideration of directed sets cannot be dropped from the definition of continuity, in order to satisfy Equation 6.

Example 4.2. *Consider the multi-adjoint* Ω*-algebra* $\langle[0,1],\leq,\&_1,\&_2\rangle$ *and the program* $\mathbb{P}$ *composed of the rules* $\langle A\leftarrow_1 B,\vartheta_1\rangle$ *and* $\langle A\leftarrow_2 B\wedge C,\vartheta_2\rangle$*, with* $0\neq\vartheta_2<\vartheta_1$*. Let* X *be the set of interpretations* $\{I_1,I_2\}$*, where* I_1 *and* I_2 *are defined as follows:*

$$\begin{array}{lll} I_1(A)=0, & I_1(B)=1 & I_1(C)=0,\\ I_2(A)=0, & I_2(B)=0 & I_2(C)=1. \end{array}$$

Clearly X *is not directed, because* $\{I_1,I_2\}\subseteq X$ *has not an upper bound in* X*. Suppose that we are considering the weights vector* $W_1=(0,1,0,\dots,0)$*. Then, it is verified that*

$$\begin{aligned} \sup\left\{T_{\mathbb{P}}^{W_1}(J)(A)\mid J\in X\right\} &= \sup\left\{T_{\mathbb{P}}^{W_1}(I_1)(A),T_{\mathbb{P}}^{W_1}(I_2)(A)\right\}\\ &= \sup\{0,0\}\\ &= 0. \end{aligned}$$

On the other hand, we can write $\sup X=S$ *and it is defined as follows.*

$$\begin{array}{lll} S(A)=0, & S(B)=1, & S(C)=1. \end{array}$$

Then, we have the following chain of equalities:

$$\begin{aligned} T_{\mathbb{P}}^{W_1}\left(\sup X\right)(A) &= \sum_{i=1}^{2}W_iC_S^A(u_{\pi(i)})\\ &= \vartheta_2, \end{aligned}$$

because of $\vartheta_1\dot{\&}_1\hat{S}(B)=\vartheta_1\dot{\&}_1 1=\vartheta_1$*,* $\vartheta_2\dot{\&}_2\hat{S}\left(B\wedge C\right)=\vartheta_2\dot{\&}_2 1=\vartheta_2$*, and that* W_1 *consider the second greatest element in the set. Consequently, as* $0\neq\vartheta_2,\vartheta_1$*, Equality 6 does not hold for arbitrary subsets* $X\subseteq\mathcal{I}_{\mathfrak{L}}$*.*

We present the following result in which we prove that the Scott-continuous of the adjoint conjuntor is a necessary condition for the Scott-continuous of the operator $T_{\mathbb{P}}^{\mathcal{W}}$.

Theorem 4.3. *If the operator* $T_{\mathbb{P}}^{\mathcal{W}}$ *is Scott-continuous for all program* $\mathbb{P}$ *in* $\mathfrak{L}$*, then any operator in the body of the rules is Scott-continuous and the adjoint conjunctors* $\&_l$*, with* $l\in\{1,\dots,m\}$*, are Scott-continuous in the second argument.*

Proof. First of all, we start with the Scott-continuity of the adjoint operator $\&_l$ in the second argument, with $l\in\{1,\dots,m\}$. Given $l\in\{1,\dots,m\}$, $\vartheta\in L$ and the directed set $Y\subseteq L$, we are going to prove

$$\sup\{\vartheta\&_l y\mid y\in Y\}=\vartheta\&_l\sup Y.$$

Given a propositional symbol $B\in\Pi$ and the set of interpretations $X_Y=\{I_y\colon\Pi\to[0,1]\mid y\in Y\}$ such that

$$I_y(C)=\begin{cases} y & \text{if}\quad C=B\\ 0 & \text{otherwise}\end{cases}$$

for all $C\in\Pi$. We are going to prove that X_Y is directed. Given a subset $\{I_{y_1},\dots,I_{y_k}\}$ of X_Y, where $\{y_1,\dots,y_k\}\subseteq Y$, as Y is a directed set, then there exists $y_M\in Y$ such that it is an upper bound of $\{y_1,\dots,y_k\}$. Therefore, the interpretation I_{y_M} is an upper bound of $\{I_{y_1},\dots,I_{y_k}\}$. Hence, X is directed.

Consider the program $\mathbb{P}$ in $\mathfrak{L}$ composed of the rule: $\mathbb{P}=\{\langle A\leftarrow_l B,\vartheta\rangle\}$. Then, we have the following chain of equalities:

$$\begin{aligned} \sup\left\{\vartheta\dot{\&}_l y\mid y\in Y\right\} &= \sup\left\{\vartheta\dot{\&}_l I_y(B)\mid I_y\in X_Y\right\}\\ &\overset{(1)}{=} \sup\left\{\sum_{i=1}^{n_A}W_A[i]\left(\vartheta_i\dot{\&}_i I_y\left(B_i\right)\right)\mid I_y\in X_Y\right\}\\ &= \sup\left\{T_{\mathbb{P}}^{W_A}(I_y)(A)\mid I_y\in X_Y\right\}\\ &\overset{(2)}{=} T_{\mathbb{P}}^{W_A}\left(\sup X_Y\right)(A)\\ &= \sum_{i=1}^{n_A}W_A[i]\left(\vartheta_i\dot{\&}_i\widehat{\sup X_Y}(B_i)\right)\\ &\overset{(3)}{=} \vartheta\dot{\&}_l\sup Y, \end{aligned}$$

where (1) is verified taking into account that $I_y(C) = 0$, for all $C \in \Pi$, with $C \neq B$. Equality (2) follows from the Scott-continuity of $T_{\mathbb{P}}^{\mathcal{W}}$. Finally, (1) follows from the fact that $\sup X_Y(C) = \sup Y$, if $C = B$, and it is 0, otherwise. Therefore, $\&_l$ is continuous in the second argument, for all $l \in \{1, \dots, m\}$.

Now, we are going to prove the continuity of the operators in the body of the rules. In order to simplify the process we will use the following notation. If $\bar{y}$, $\bar{z} \in L^n$, where $\bar{y} = (y_1, \dots, y_n)$, $\bar{z} = (z_1, \dots, z_n)$, then $\bar{y} \leq \bar{z}$ if and only if $y_i \leq z_i$ for all $i \in \{1, \dots, n\}$.

Given a directed set $Y \subseteq L^n$, we have to prove that

$$\dot{@}\,(\sup Y) = \sup \left\{ \dot{@}\,(y_1, \dots, y_n) \mid (y_1, \dots, y_n) \in Y \right\},$$

where @ is an operator (aggregator) in the body of one of the rules in the program. Clearly, if $(y_1, \dots, y_n) \in Y$, then $(y_1, \dots, y_n) \leq \sup Y$ and by the monotonocity of $\dot{@}$, $\dot{@}\,(y_1, \dots, y_n) \preceq \dot{@}\,(\sup Y)$. Therefore,

$$\sup \left\{ \dot{@}\,(y_1, \dots, y_n) \mid (y_1, \dots, y_n) \in Y \right\} \leq \dot{@}\,(\sup Y),$$

For the other inequality, given n propositional symbols $A_1, \dots, A_n \in \Pi$ and $\bar{y} = (y_1, \dots, y_n) \in Y$, we can consider the interpretation $I_{\bar{y}} \colon \Pi \to [0,1]$ defined as

$$I_{\bar{y}}(C) = \begin{cases} y_i & \text{if} \quad C = A_i \\ 0 & \text{otherwise} \end{cases}$$

for all $C \in \Pi$. On the one hand, we can consider the set $X_Y = \{I_{\bar{y}} \in \mathcal{I}_{\mathfrak{L}} \mid \bar{y} \in Y\}$ and its supremum $S_Y = \sup X_Y$. From the definition of $I_{\bar{y}}$ and as Y is directed, then X_Y is also directed. For all $\bar{y} \in Y$ it is verified that

$$(y_1, \dots, y_n) = (I_{\bar{y}}(A_1), \dots, I_{\bar{y}}(A_n)) \leq (S_Y(A_1), \dots, S_Y(A_n)).$$

Hence, it is satisfied that

$$\sup Y \leq (S_Y(A_1), \dots, S_Y(A_n)).$$

As a consequence, by the monotonocity of $\dot{@}$, we reach

$$\dot{@}\,(\sup Y) \leq \dot{@}\,(S_Y(A_1), \dots, S_Y(A_n)) = \widehat{S_Y}\,(@\,(A_1, \dots, A_n)).$$

On the other hand, consider the multi-adjoint logic program $\mathbb{P}$ composed of the rule: $u = \langle A \leftarrow_l @\,(A_1, \dots, A_n)\,, \top \rangle$. We have the following chain of equalities.

$$\begin{aligned}
\widehat{S_Y}\,(@\,(A_1, \dots, A_n)) &= \top \dot{\&}_l \widehat{S_Y}\,(@\,(A_1, \dots, A_n)) \\
&\overset{(1)}{=} \sum_{i=1}^{n_A} W_A[i] C^A_{S_Y}\left(u_{\pi(i)}\right) \\
&= T_{\mathbb{P}}^{W_A}\,(S_Y)\,(A) \\
&\overset{(2)}{=} \sup \left\{ T_{\mathbb{P}}^{W_A}\,(J_{\bar{y}})\,(A) \mid J_{\bar{y}} \in X_Y \right\} \\
&= \sup \left\{ \sum_{i=1}^{n_A} W_A[i] C^A_{J_{\bar{y}}}(u_{\pi(i)}) \mid J_{\bar{y}} \in X_Y \right\} \\
&= \sup \left\{ C^A_{J_{\bar{y}}}(u) \mid J_{\bar{y}} \in X_Y \right\} \\
&= \sup \left\{ \top \dot{\&}_l \widehat{J_{\bar{y}}}\,(@\,(A_1, \dots, A_n)) \mid J_{\bar{y}} \in X_Y \right\} \\
&\overset{(3)}{=} \sup \left\{ \dot{@}\,(y_1, \dots, y_n) \mid (y_1, \dots, y_n) \in Y \right\}.
\end{aligned}$$

where (1) and (3) follow from the definition of $I_{\bar{y}}$ and (2) follows from the Scott-continuity of $T_{\mathbb{P}}^{W_A}$. Consequently, it is verified that

$$\begin{aligned}
\sup \left\{ \dot{@}\,(y_1, \dots, y_n) \mid (y_1, \dots, y_n) \in Y \right\} &\leq \dot{@}\,(\sup Y) \\
&\leq \widehat{S_Y}\,(@\,(A_1, \dots, A_n)) \\
&= \sup \left\{ \dot{@}\,(y_1, \dots, y_n) \mid (y_1, \dots, y_n) \in Y \right\}.
\end{aligned}$$

□

Finally, from the previous results, we obtain the following version of Proposition 2.24 to the new more flexible immediate consequences operator.

Corollary 4.4. *If* $T_{\mathbb{P}}^{\mathcal{W}}$ *is continuous, then* $\left(T_{\mathbb{P}}^{\mathcal{W}}\right)^{\omega}(\Delta)$ *is the least model of* $\mathbb{P}$*, that is,* $\left(T_{\mathbb{P}}^{\mathcal{W}}\right)^{\omega}(\Delta) = M_{\mathbb{P}}$*.*

Proof. The proof follows from Proposition 2.24 and the previous results. $\square$

5 Conclusions and future work

Following the philosophy and motivation already introduced in [22, 23], this paper has presented a flexible definition of the immediate consequences operator based on the use of OWA operators, which has allowed us to effectively solve the problem of the possible loss of information caused by the use of the supremum. The semantics of a multi-adjoint logic program have also been conveniently adapted to this new framework through the use of a family of weights vectors. In addition, this new operator verifies very interesting properties such as the monotonicity and the continuity.

However, some open questions remain. The first one refers to the possible relationship that can be established between generalized quantifiers and OWA operators when the immediate consequences operator is to be generalized. Not all fuzzy measures (in the case of generalized quantifiers) nor all weights vectors (in the case of OWA operators) serve to generalize the $T_{\mathbb{P}}$. Consequently, it is interesting to study conditions that make it possible to ensure that a certain generalization of the $T_{\mathbb{P}}$ is adequate. Moreover, we will study the possibility of considering other kind of general operators [5, 12, 24].

On the other hand, all the results presented in this paper have been performed on the unit interval, which is a complete lattice that is totally ordered. New generalizations of the immediate consequences operator on a general lattice will be studied as future work.

Acknowledgement

Partially supported by the 2014-2020 ERDF Operational Programme in collaboration with the State Research Agency (AEI) in project PID2019-108991GB-I00, and with the Department of Economy, Knowledge, Business and University of the Regional Government of Andalusia in project FEDER-UCA18-108612, and by the European Cooperation in Science and Technology (COST) Action CA17124.

References

[1] E. Bas, U. Yolcu, E. Egrioglu, *Picture fuzzy regression functions approach for financial time series based on ridge regression and genetic algorithm*, Journal of Computational and Applied Mathematics, **370** (2020), 112656.

[2] A. Colmerauer, H. Kanoui, P. Roussel, R. Pasero, *Un systeme de communication homme-machine en Francais*, Technical report, Groupe de Recherche en Intelligence Artificielle, Universidad de Aix-Marseille, 1973.

[3] M. E. Cornejo, D. Lobo, J. Medina, *On the solvability of bipolar max-product fuzzy relation equations with the product negation*, Journal of Computational and Applied Mathematics, **354** (2019), 520-532.

[4] M. E. Cornejo, J. Medina, E. Ramrez-Poussa, *Algebraic structure and characterization of adjoint triples*, Fuzzy Sets and Systems, **425** (2021), 117-139.

[5] M. E. Cornejo, J. Medina, I. Stajner-Papuga, A. Tepavcevic, *Lattice valued aggregation operators applicable in digital forensics*, Studies in Computational Intelligence, (2023), 1-7.

[6] C. V. Damásio, L. M. Pereira, *Monotonic and residuated logic programs*, Symbolic and Quantitative Approaches to Reasoning with Uncertainty, ECSQARU'01, pages 748-759. Lecture Notes in Artificial Intelligence, 2143, 2001.

[7] M. Davis, H. Putnam, *A computing procedure for quantification theory*, Journal of the ACM, **7** (1960), 201-215.

[8] M. V. Emden, R. Kowalski, *The semantics of predicate logic as a programming language*, Journal of the ACM, **23**(4) (1976), 733-742.

[9] J. H. Gallier, *Logic for computer science: Foundations of automatic theorem proving*, Harper and Row Publishers, Inc., USA, 1985.

[10] P. Gilmore, *A proof method for quantification theory*, IBM Journal of Research and Development, **4** (1960), 28-35.

[11] N. V. Hung, V. M. Tam, N. H. Tuan, D. O'Regan, *Convergence analysis of solution sets for fuzzy optimization problems*, Journal of Computational and Applied Mathematics, **369** (2020), 112615.

[12] D. Jocic, I. Stajner-Papuga, *Distributivity between 2-uninorms and Mayor's aggregation operators*, Iranian Journal of Fuzzy Systems, **19**(6) (2022), 13-27.

[13] P. Julián-Iranzo, C. Rubio-Manzano, *A sound and complete semantics for a similarity-based logic programming language*, Fuzzy Sets and Systems, **317** (2017), 1-26.

[14] M. Kifer, V. S. Subrahmanian, *Theory of generalized annotated logic programming and its applications*, Journal of Logic Programming, **12** (1992), 335-367.

[15] A. Kishor, A. K. Singh, N. R. Pal, *Orness measure of OWA operators: A new approach*, IEEE Transactions on Fuzzy Systems, **22**(4) (2014), 1039-1045.

[16] R. Kowalski, *Predicate logic as a programming language*, Proceedings of IFIP'74, North Holland. Publishing Company, (1974), 569-574.

[17] I. Lizasoain, C. Moreno, *OWA operators defined on complete lattices*, Fuzzy Sets and Systems, **224** (2013), 36-52.

[18] J. Lloyd, *Foundations of logic programming*, Springer Verlag, 1987.

[19] J. Medina, M. Ojeda-Aciego, P. Vojtáš, *Multi-adjoint logic programming with continuous semantics*, Lecture Notes in Artificial Intelligence, **2173** (2001), 351-364.

[20] J. Medina, M. Ojeda-Aciego, P. Vojtáš, *Similarity-based unification: A multi-adjoint approach*, Proc. EUSFLAT Conference in Fuzzy Logic and Technology, (2001), 273-276.

[21] J. Medina, M. Ojeda-Aciego, P. Vojtáš, *Similarity-based unification: A multi-adjoint approach*, Fuzzy Sets and Systems, **146** (2004), 43-62.

[22] J. Medina, J. A. Torné-Zambrano, *Fuzzy logic programming with generalized quantifiers*, Springer International Publishing, **1040** (2023), 17-23.

[23] J. Medina, J. A. Torné-Zambrano, *Immediate consequences operator on generalized quantifiers*, Fuzzy Sets and Systems, **456** (2022), 72-91.

[24] R. Mesiar, A. Kolesarova, S. Borkotokey, *2-set-based extended functions*, Iranian Journal of Fuzzy Systems, **18**(3) (2021), 13-26.

[25] M. Ojeda-Hernndez, I. P. Cabrera, P. Cordero, *Quasi-closed elements in fuzzy posets*, Journal of Computational and Applied Mathematics, **404** (2022), 113390.

[26] D. Prawitz, *An improved proof procedure*, Theoria, **26** (1960), 102-139.

[27] U. Straccia, M. Ojeda-Aciego, C. V. Damásio, *On fixed-points of multivalued functions on complete lattices and their application to generalized logic programs*, SIAM Journal on Computing, **38**(5) (2009), 1881-1911.

[28] A. Tarski, *A lattice-theoretical fixpoint theorem and its applications*, Pacific Journal of Mathematics, **5**(2) (1955), 285-309.

[29] M. H. Van Emden, *Quantitative deduction and its fixpoint theory*, The Journal of Logic Programming, **1** (1986), 37-53.

[30] P. Vojtáš, *Fuzzy logic programming*, Fuzzy Sets and Systems, **124**(3) (2001), 361-370.

[31] R. R. Yager, *On ordered weighted averaging aggregation operators in multi-criteria decision making*, IEEE Transactions on Systems, Man, and Cybernetics, **18**(1) (1988), 183-190.

[32] R. R. Yager, *Families of OWA operators*, Fuzzy Sets and Systems, **59**(2) (1993), 125-148.