Syntax and semantics of multi-adjoint normal logic programming

M. Eugenia Cornejo, David Lobo, Jesús Medina

Department of Mathematics, University of Cádiz. Spain
{mariaeugenia.cornejo,david.lobo,jesus.medina}@uca.es

Abstract

Multi-adjoint logic programming is a general framework with interesting features, which involves other positive logic programming frameworks such as monotonic and residuated logic programming, generalized annotated logic programs, fuzzy logic programming and possibilistic logic programming. One of the most interesting extensions of this framework is the possibility of considering a negation operator in the logic programs, which will improve its flexibility and the range of real applications.

This paper introduces multi-adjoint normal logic programming, which is an extension of multi-adjoint logic programming including a negation operator in the underlying lattice. Beside the introduction of the syntax and semantics of this paradigm, we will provide sufficient conditions for the existence of stable models defined on a convex compact set of an euclidean space. Finally, we will consider a particular algebraic structure in which sufficient conditions can be given in order to ensure the unicity of stable models of multi-adjoint normal logic programs.

Keywords: Multi-adjoint logic programs, negation operator, stable models.

1. Introduction

Syntax and semantics are two noticeably different parts in any theory of logic programming. On the one hand, the syntax describes the symbols and formulas chosen to represent formally statements that can be considered. On the other hand, the semantics gives meaning to the considered statements from its syntactic structure, and establishes an inference system to obtain which deductions and/or consequences are correct. In this paper, we will regard a specific type of structure called multi-adjoint logic programs. Multi-adjoint logic programming was introduced as a generalization of different non-classical logic programming frameworks in [33]. The main feature of this logical theory is based on the use of several implications in the rules of a same logic program, as well as general operators defined on complete lattices in the bodies of the rules. An interesting property of multi-adjoint logic programs is related to the existence of the least model. This fact allows us to check whether a statement is a consequence by using a simple evaluation. However, the existence of the least model cannot be guaranteed when we consider multi-adjoint logic programs enriched with a negation operator. Different semantics have been developed for logic programs with negation such as the well-founded semantics [46], the stable models semantics [16] and the answer sets semantics [18]. This work will focus on the study of the existence and the unicity of stable models for multi-adjoint normal logic programs. In some logical approaches, sufficient conditions have been stated in order to ensure the existence of stable models:

- In the 3-valued Kleene logic, every logic program with negation has stable models [39].
- In normal residuated logic programming, stable models exist for every normal logic program whose underlying residuated lattice has an appropriate bilattice structure [15, 25, 40, 41, 42].
- Another important survey on stable models in normal residuated logic programs on $[0, 1]$ was presented in [29], where it was proven that the continuity of the connectives appearing in the program guarantees the existence of stable models. In addition, the uniqueness of stable models is obtained when the product t-norm, its residuated implication, and the standard negation are considered.

The contribution of this paper consists in applying the philosophy of the multi-adjoint paradigm [8, 33] in order to develop a more general and flexible mathematical theory than the previous ones. Hence, we will present the syntax and semantics of the multi-adjoint normal logic programming framework and sufficient conditions to ensure:

1. the existence of stable models for multi-adjoint normal logic programs defined on any convex compact set of an euclidean space; and
2. the unicity of stable models for multi-adjoint normal logic programs defined on the set of subintervals $\mathcal{C}([0, 1])$.

When these logic programs correspond to some search problem, the stable models coincide with its possible solutions. Therefore, these goals are fundamental in order to know whether the program is related to a solvable problem and, in that case, whether only one solution exists. The characterization of programs with a unique solution and a deterministic procedure to obtain that solution is also important, since the solvability of this kind of programs will be at once.

These properties on the existence and uniqueness of stable models in multi-adjoint normal logic programming will be useful in other logic programming frameworks in which a negation operator is needed. Since monotonic and residuated logic programming [13, 12], fuzzy logic programming [47] and possibilistic logic programming [14] are particular cases of the multi-adjoint logic programming framework, we can straightforwardly apply the results given in this paper to their normal extensions (that is, when a negation operator is used).

Moreover, these results can be also applied to other frameworks with a different syntax, such as to generalized annotated logic programs [20]. This logic was related to the fuzzy logic programming introduced by Vojtás in [22], and so we can translate the results given in this paper to a new normal generalized annotated logic programming in which a negation operator is considered.

This paper is organized as follows: Section 2 includes a brief summary with concepts and results corresponding to the multi-adjoint logic programming framework and the algebraic topology. Section 3 presents the multi-adjoint normal logic programs as well as interesting properties of the immediate consequence operator and of the stable models. These properties allow to recognize which are the problems we have to solve in order to define the syntax and the semantics of multi-adjoint normal logic programs. A detailed study about the existence and the unicity of stable models in these programs is introduced in Section 4. Some conclusions and prospects for future work are included in Section 5.

2. Preliminaries

This section recalls some notions and results related to the propositional language used in multi-adjoint logic programming, which is composed of two important parts: the syntax and the semantics. Later, the definition of program in this general logic programming framework is included. Finally, some topological definitions and fix-point theorems will be introduced.

2.1. Syntax of the propositional language

The syntax of the propositional language of multi-adjoint logic programming is based on the concepts of alphabet and expressions of the language. These concepts require the use of some definitions of universal algebra as it is shown below.

Definition 1. 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 Ω_n will denote the set of elements with arity n in Ω.

Considering a graded set, the notions of algebraic structure and substructure of an algebraic structure are generalized by means of the following definitions.

Definition 2. Given a graded set Ω, an Ω-*algebra* is a pair $\mathfrak{A} = \langle A, I \rangle$ 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 $\omega_{\mathfrak{A}}$.
2. Each element $c \in \Omega_0$ (c is a constant) is interpreted as an element $I(c)$ in A, denoted by $c_{\mathfrak{A}}$.

Definition 3. Given an Ω-algebra $\mathfrak{A} = \langle A, I \rangle$, 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 $\omega \in \Omega_n$, then $J(\omega)\colon B^n \to B$ is the restriction of $I(\omega)\colon A^n \to A$.

Now, we introduce the notion of alphabet of a language, that is, the set of symbols from which expressions can be formed.

Definition 4. Let Ω be a graded set, Π a countable infinite set and L a set of truth-values. The *alphabet* $A_{\Omega,\Pi \uplus L}$ associated with Ω and $\Pi \uplus L$ is defined by the disjoint union $\Omega \uplus (\Pi \uplus L) \uplus S$, where S is the set of auxiliary symbols "(", ")" and ",".

From the set of operators Ω and the symbols of $\Pi \uplus L$, the algebra of expressions is defined as follows.

Definition 5. Given a graded set Ω and an alphabet $A_{\Omega,\Pi\uplus L}$. The Ω-algebra $\mathfrak{E} = \langle A^*_{\Omega,\Pi\uplus L}, I\rangle$ of *expressions* is defined as follows:

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

It is important to note that an expression does not need to be a well-formed formula, that is, an expression is only a string of letters of the alphabet. Indeed, the well-formed formulas is the subset of expressions defined as the next definition shows.

Definition 6. Let Ω be a graded set, Π a countable set of propositional symbols, L a set of truth-values and $\mathfrak{E}$ the algebra of expressions corresponding to the alphabet $A_{\Omega,\Pi\uplus L}$. The *well-formed formulas* (in short, formulas) generated by Ω over $\Pi \uplus L$ is the least subalgebra $\mathfrak{F}$ of the algebra of expressions $\mathfrak{E}$ containing $\Pi \uplus L$.

2.2. Semantics of the propositional language

In this section, we will consider a graded set Ω, a set of propositional symbols Π, the corresponding Ω-algebra of well-formed formulas $\mathfrak{F}$ and an arbitrary Ω-algebra $\mathfrak{U}$ whose carrier is A. The notion of interpretation plays a fundamental role in the semantics of the propositional language of multi-adjoint logic programming.

Definition 7. A mapping $I\colon \Pi \to A$ which assigns to every propositional symbol appearing in Π an element of A is called *A-interpretation*. The set of all A-interpretations with respect to the Ω-algebra $\mathfrak{U}$ is denoted by $\mathcal{I}_{\mathfrak{U}}$.

If $(L, \preceq)$ is a complete lattice where L is the carrier of an Ω-algebra $\mathfrak{L}$ then the ordering $\preceq$ can be extended to the set of interpretations as follows:

$$I_1 \sqsubseteq I_2 \text{ if and only if } I_1(p) \preceq I_2(p), \text{ for all } p \in \Pi \text{ and } I_1, I_2 \in \mathcal{I}_{\mathfrak{L}}.$$

The new ordering $\sqsubseteq$ defined on the set of interpretations inherits some properties of the ordering $\preceq$ defined on the lattice, as the next proposition shows.

Proposition 8 ([33]). *If* $(L, \preceq)$ *is a complete lattice, then* $(\mathcal{I}_{\mathfrak{L}}, \sqsubseteq)$ *is a complete lattice where the least interpretation* Δ *applies every propositional symbol to the bottom element of* L*, and the greatest interpretation* ∇ *applies every propositional symbol to the top element of* L*.*

A similar result with respect to the convexity and the compactness of the set of interpretations will be proved in Section 4.1, which is focused on the study of the existence of stable models.

2.3. Multi-adjoint logic programs

The multi-adjoint framework arises as a generalization of several non-classical logic programming settings whose semantic structure is the multi-adjoint lattice [9, 10, 33]. In order to recall this definition, we need to introduce the concept of adjoint pair which was firstly presented in a logical context by Pavelka [37].

Definition 9. Let $(P, \leq)$ be a partially ordered set and $(\&, \leftarrow)$ be a pair of binary operations in P, such that

1. $\&$ is monotonic in both arguments.[1]
2. $\leftarrow$ is monotonic in the first argument (the consequent) and decreasing in the second argument (the antecedent).
3. For any $x, y, z \in P$, we have that $x \leq (y \leftarrow z)$ holds if and only if $(x \,\&\, z) \leq y$ holds.

Then we say that $(\&, \leftarrow)$ forms an *adjoint pair* in $(P, \leq)$.

Observe that, the monotonicity of the operators $\&$ and $\leftarrow$ is justified because they will be interpreted as generalized conjunctions and implications. It is important to highlight that $\&$ does not need to be either commutative or associative, and boundary conditions are not required. The last property in the previous definition corresponds to the categorical adjointness.

As well as the properties given in Definition 9, we will need to assume the existence of the bottom and top elements in the poset of truth-values, and the existence of joins for every directed subset, that is, we will assume a complete lattice.

The use of different implications and several modus ponens like inference rules to extend the theory developed in [13, 47] to a more general environment gave rise to consider various adjoint pairs in the lattice.

[1] A monotonic operator is also called order-preserving operator or increasing mapping.

Definition 10. The tuple $(L, \preceq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n)$ is a *multi-adjoint lattice* if the following properties are verified:

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

The algebraic structure shown in the next definition increases the expressive power of the multi-adjoint lattice by using extra operators.

Definition 11. 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 $\mathfrak{L} = (L, I)$ be an Ω-algebra whose carrier set L is a lattice under $\preceq$. We say that $\mathfrak{L}$ is a *multi-adjoint Ω-algebra* with respect to the pairs $(\&_i, \leftarrow_i)$, with $i \in \{1, \dots, n\}$, if $(L, \preceq, I(\leftarrow_1), I(\&_1), \dots, I(\leftarrow_n), I(\&_n))$ is a multi-adjoint lattice.

From this structure, a multi-adjoint logic program is defined as a set of rules and facts of a given language $\mathfrak{F}$.

Definition 12. Let $(L, \preceq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n)$ be a multi-adjoint lattice. A *multi-adjoint logic program* is a set of weighted rules $\langle (A \leftarrow_i \mathcal{B}); \vartheta \rangle$ 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 $\&_1, \dots, \&_n$ and $\wedge_1, \dots, \wedge_k$, disjunctors $\vee_1, \dots, \vee_l$, aggregators $@_1, \dots, @_m$ and elements of L.
5. *Facts* are rules with body $\top$.

Examples related to these preliminary notions can be found in [19, 31, 32, 33].

Note that, when the multi-adjoint lattice is enriched with a negation operator, we can define a particular type of multi-adjoint logic program called multi-adjoint normal logic program. Before presenting our study about the syntax and semantics of this special kind of non-monotonic logic program, we need to recall some topological notions and results.

2.4. Some notions of algebraic topology

This section includes different notions and results of algebraic topology, which will be used later. The definitions of compact set and convex set are listed below.

Definition 13. Let $(X, +, *, \mathbb{R})$ be an euclidean space. We say that $A \subseteq X$ is:

- a *compact set* if it is closed and bounded in X.
- a *convex set* if $t * x + (1 - t) * y \in A$, for all $x, y \in A$ and $t \in [0, 1]$.

Finally, we present two theorems related to the fix-point theory. The former is an extension of the Brouwer fix-point theorem, which is known as Schauder fix-point theorem [23].

Theorem 14 (Schauder fix-point theorem). *Let $(X, +, *, \mathbb{R})$ be an euclidean space and let $K \subseteq X$ be a non-empty convex compact set. Every continuous mapping $f\colon K \to K$ has a fix point.*

Before introducing the Banach fix-point theorem [4], it is necessary to show the definition of contractive mapping.

Definition 15. Let (X, d) be a complete metric space. We say that $f\colon X \to X$ is *a contractive mapping* if there exists a real value $0 < \lambda < 1$ such that:

$$d(f(x), f(y)) \leq \lambda\, d(x, y)$$

for each $x, y \in X$. Any real value λ satisfying the previous inequality is called *Lipstchiz constant.*

Theorem 16 (Banach fix-point theorem). *Let (X, d) be a complete metric space and let $f\colon X \to X$ be a contractive mapping in $A \subseteq X$. Then f has a unique fix-point in A.*

These previous concepts and results will play a crucial role in order to define the semantics of multi-adjoint normal logic programs.

3. On the syntax and semantics of multi-adjoint normal logic programs

As we mentioned above, we are interested in considering a multi-adjoint lattice $(L, \preceq, \leftarrow_1, \&_1, \ldots, \leftarrow_n, \&_n)$, with a maximum $\top$ and a minimum $\bot$ element, enriched with a negation operator. The considered negation will be a decreasing mapping $\neg\colon L \to L$ satisfying the equalities $\neg(\bot) = \top$ and $\neg(\top) = \bot$. The notion of default negation is modeled by the previous negation operator. The algebraic structure obtained from a multi-adjoint lattice and a negation operator will be called a *multi-adjoint normal lattice.*

The formal definition of a multi-adjoint normal logic program is given next.

Definition 17. A *multi-adjoint normal logic program (MANLP)* $\mathbb{P}$, defined on a multi-adjoint normal lattice $(L, \preceq, \leftarrow_1, \&_1, \ldots, \leftarrow_n, \&_n, \neg)$, is a finite set of weighted rules of the form:

$$\langle p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]; \vartheta\rangle$$

where $i \in \{1, \ldots, n\}$, $@$ is an aggregator operator, ϑ is an element of L and $p, p_1, \ldots, p_n$ are propositional symbols such that $p_j \neq p_k$, for all $j, k \in \{1, \ldots, n\}$, with $j \neq k$.

Henceforth, we will use the notation $\Pi_{\mathbb{P}}$ to denote the set of propositional symbols appearing in $\mathbb{P}$. In addition, the rules of a MANLP will be denoted as $\langle p \leftarrow_i \mathcal{B}; \vartheta\rangle$ where p is the *head* of the rule, $\mathcal{B}$ its *body* and ϑ its *weight.*

It is also convenient to mention that the whole set of rules that we can build from the well-formed formulas generated by Ω over $\Pi \uplus L$ will be denoted by $\mathfrak{R}_{\Pi\uplus L}$. The set $\mathfrak{R}^{+}_{\Pi\uplus K}$ will be formed by the rules of $\mathfrak{R}_{\Pi\uplus K}$ which do not contain the negation operator.

In order to avoid confusion, we will use a special notation to differentiate an operator symbol in Ω from its interpretation under $\mathfrak{L}$. Specifically, ω will denote an operator symbol in Ω and $\dot{\omega}$ will denote the interpretation of the previous operator symbol under $\mathfrak{L}$. In a similar way, the evaluation of a formula $\mathcal{A}$ under an interpretation I will be denoted as $\hat{I}(\mathcal{A})$, and it proceeds inductively as usual, until all propositional symbols in $\mathcal{A}$ are reached and evaluated under I. For instance, considering an interpretation $I \in \mathcal{I}_{\mathfrak{L}}$ and two formulas $\mathcal{A}, \mathcal{B} \in \mathfrak{F}$, the following equality holds:

$$\hat{I}(\mathcal{A} \,\&_i\, \mathcal{B}) = \hat{I}(\mathcal{A}) \,\dot{\&}_i\, \hat{I}(\mathcal{B})$$

Note that, every formula $\mathcal{A}$ can be written as $@[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]$, in which @ represents the composition of the monotonic operators in $\mathcal{A}$ (which is an aggregation operator) and $p_1, \ldots, p_n$ the propositional symbols appearing in $\mathcal{A}$. In this case, the equality $\hat{I}(@[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]) = \dot{@}[I(p_1), \ldots, I(p_m), \neg I(p_{m+1}), \ldots, \neg I(p_n)]$ is satisfied for any formula $\mathcal{A} = @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n] \in \mathfrak{F}$.

Now, after introducing the syntactic structure of MANLPs and some notational conventions, we will present the notions associated with the semantics of MANLPs. We will start with the definitions of satisfaction and model.

In a similar way to the semantics of multi-adjoint logic programs [33], we say that an interpretation satisfies a rule of a multi-adjoint normal logic program if the truth-value of the rule is greater or equal than the confidence factor associated with the rule.

Definition 18. Given an interpretation $I \in \mathcal{I}_{\mathfrak{L}}$, we say that:

(1) A weighted rule $\langle p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]; \vartheta\rangle$ is *satisfied* by I if and only if $\vartheta \preceq \hat{I}\,(p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n])$.

(2) An interpretation $I \in \mathcal{I}_{\mathfrak{L}}$ is a *model* of a MANLP $\mathbb{P}$ if and only if all weighted rules in $\mathbb{P}$ are satisfied by I.

Example 19. Consider the multi-adjoint normal lattice

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

where $\&_G$ and $\&_P$ are the Gödel and product conjunctors, respectively, $\leftarrow_G$ and $\leftarrow_P$ are their corresponding adjoint implications and $\neg(x) = 1 - x$, for each $x \in [0,1]$.

Let $\Pi_{\mathbb{P}} = \{p, q, r\}$ be the set of propositional symbols and let us define the following MANLP $\mathbb{P}$ valued in $[0,1]$ and consisting of two rules and one fact.

$$r_1 : \langle p \leftarrow_P q \;\; \&_G \neg r \; ; \; 0.7\rangle$$
$$r_2 : \langle r \leftarrow_G p \,\&_G\, q \; ; \; 0.2\rangle$$
$$r_3 : \langle q \leftarrow_P 1 \; ; \; 0.6\rangle$$

Let us prove that the interpretation $I \equiv \{(p, 0.5), (q, 0.7), (r, 0.4)\}$ satisfies the rules of $\mathbb{P}$.

For r_1 we obtain that

$$\begin{aligned}\hat{I}(p \leftarrow_P q \;\&_G \neg r) &= I(p) \dot{\leftarrow}_P \hat{I}\big(q \;\&_G \neg r\big) = I(p) \dot{\leftarrow}_P \big(I(q) \,\dot{\&}_G \,\dot{\neg}\, I(r)\big)\\ &= 0.5 \dot{\leftarrow}_P \big(0.7 \,\dot{\&}_G \,\dot{\neg}\, 0.4\big) = 0.5 \dot{\leftarrow}_P 0.6 = \frac{0.5}{0.6} = 0.8\widehat{3}\end{aligned}$$

Since the truth-value of r_1 is 0.7, we have that $0.7 \leq \hat{I}(p \leftarrow_P q \;\&_G \neg r)$, so I satisfies rule r_1.

Considering rule r_2 we obtain that

$$\begin{aligned}\hat{I}(r \leftarrow_G p \,\&_G\, q) &= I(r) \dot{\leftarrow}_G \hat{I}\big(p \,\&_G\, q\big) = I(r) \dot{\leftarrow}_G \big(I(p) \,\dot{\&}_G\, I(q)\big)\\ &= 0.4 \dot{\leftarrow}_G \big(0.5 \,\dot{\&}_G\, 0.7\big) = 0.4 \dot{\leftarrow}_G 0.5 = 0.4\end{aligned}$$

As the weight of rule r_2 is 0.2, we have that $0.2 \leq \hat{I}(r \leftarrow_G p \,\&_G\, q)$, hence I satisfies r_2.

Lastly, observe that rule r_3 is a fact, then I satisfies r_3 if and only if $I(q)$ is greater or equal than the weight of r_3, which holds, since $I(q) = 0.7$ and the weight of r_3 is 0.6. Therefore, I satisfies the three rules in $\mathbb{P}$ and we can conclude that it is a model of that program. □

The following section introduces the immediate consequence operator and the fix-point semantics of MANLPs.

3.1. Immediate consequence operator

The first definition generalizes the usual notion of immediate consequence operator for the flexible case of multi-adjoint normal logic programs.

Definition 20. Let $\mathbb{P}$ be a multi-adjoint normal logic program. The *immediate consequence operator* is the mapping $T_{\mathbb{P}}^{\mathfrak{L}} \colon \mathcal{I}_{\mathfrak{L}} \to \mathcal{I}_{\mathfrak{L}}$ defined for every L-interpretation I and $p \in \Pi_{\mathbb{P}}$ as

$$T_{\mathbb{P}}(I)(p) = \sup\{\vartheta \,\dot{\&}_i\, \hat{I}(\mathcal{B}) \mid \langle p \leftarrow_i \mathcal{B}; \vartheta\rangle \in \mathbb{P}\}$$

The following proposition ensures that, given a MANLP, we can obtain a partition of that program such that, for each propositional symbol p, there exists at most one rule in $\mathbb{P}$ whose head is p in each element of the partition. The interest of this result is that each part of the partition could be considered as an independent program. Note that, if for each propositional symbol p in a program $\mathbb{P}$, there exists at most one rule in $\mathbb{P}$ whose head is p, then the definition of $T_{\mathbb{P}}$ can be simplified, since the supremum operator can be removed from the definition.

Proposition 21. *Given a MANLP $\mathbb{P}$, there exists a partition $\{\mathbb{P}_\gamma\}_{\gamma\in\Gamma}$ of the program $\mathbb{P}$ such that:*

1. *$\mathbb{P}_\gamma$ does not contain two rules with the same head, for all $\gamma \in \Gamma$.*
2. *the equality $T_{\mathbb{P}}(I)(p) = \sup\{T_{\mathbb{P}_\gamma}(I)(p) \mid \gamma \in \Gamma\}$ holds.*

Proof. For each rule $r_\gamma \in \mathbb{P}$, let us consider the MANLP with only one rule $\mathbb{P}_\gamma = \{r_\gamma\}$. Then, the partition $\{\mathbb{P}_\gamma\}_{\gamma\in\Gamma}$ satisfies the first condition. Now, for each $\mathbb{P}_\gamma$ and interpretation I, the immediate consequence operator is:

$$T_{\mathbb{P}_\gamma}(I)(q) = \begin{cases} \vartheta \,\dot{\&}_i\, \hat{I}(\mathcal{B}) & \text{if } \ q = p \\ \bot & \text{otherwise} \end{cases}$$

where $\langle p \leftarrow_i \mathcal{B}; \vartheta\rangle$ is the unique rule in $\mathbb{P}_\gamma$. Thus,

$$T_{\mathbb{P}}(I)(p) = \sup\{\vartheta \,\dot{\&}_i\, \hat{I}(\mathcal{B}) \mid \langle p \leftarrow_i \mathcal{B}; \vartheta\rangle \in \mathbb{P}\} = \sup\{T_{\mathbb{P}_\gamma}(I)(p) \mid \gamma \in \Gamma\}$$

□

This proposition will play a crucial role in the proof of the unicity result of Section 4.

Another important property is that, if $\mathbb{P}$ is a *positive multi-adjoint logic program*, that is the rules in $\mathbb{P}$ do not contain any negation, we can ensure that its corresponding immediate consequence operator $T_{\mathbb{P}}$ is monotonic.

Proposition 22 ([33]). *If $\mathbb{P}$ is a positive multi-adjoint logic program, then $T_{\mathbb{P}}$ is monotonic.*

This fact allows to characterize the models of a positive multi-adjoint logic program by means of the postfix-points of $T_{\mathbb{P}}$.

Proposition 23 ([33]). *Let $\mathbb{P}$ be a positive multi-adjoint logic program and M be an L-interpretation. Then, M is a model of the program $\mathbb{P}$ if and only if $T_{\mathbb{P}}(M) \preceq M$.*

Note that, when $\mathbb{P}$ is positive, then the Knaster-Tarski fix-point theorem [44] ensures that $T_{\mathbb{P}}$ has a least fix-point. As a consequence, considering the monotonicity of $T_{\mathbb{P}}$ and the proposition above, we deduce that this least fix-point is the least model of $\mathbb{P}$ [33].

However, the immediate consequence operator is not necessarily monotonic in MANLPs. This fact implies that the existence of the least model cannot be ensured. In order to define the semantics for multi-adjoint normal logic programs, we will use the well-known notion of stable model of a program [16, 26].

3.2. Stable models

The notion of stable model of a normal program is related to the minimal models of a monotonic logic program obtained from the original one. Hence, first of all, we need to introduce a mechanism in order to obtain a positive multi-adjoint logic program from a MANLP.

Given a multi-adjoint normal logic program $\mathbb{P}$ and an L-interpretation I, we will build a positive multi-adjoint program $\mathbb{P}_I$ by substituting each rule in $\mathbb{P}$ such as

$$\langle p \leftarrow_i @[p_1, \dots, p_m, \neg p_{m+1}, \dots, \neg p_n]; \vartheta \rangle$$

by the rule

$$\langle p \leftarrow_i @_I[p_1, \dots, p_m]; \vartheta \rangle$$

where the operator $\dot{@}_I \colon L^m \to L$ is defined as

$$\dot{@}_I[\vartheta_1, \dots, \vartheta_m] = \dot{@}[\vartheta_1, \dots, \vartheta_m, \dot{\neg}\, I(p_{m+1}), \dots, \dot{\neg}\, I(p_n)]$$

for all $\vartheta_1, \dots, \vartheta_m \in L$. The program $\mathbb{P}_I$ will be called the *reduct* of $\mathbb{P}$ with respect to the interpretation I and the rules of the program $\mathbb{P}_I$ will be denoted as $\langle p \leftarrow_i \mathcal{B}_{\mathcal{I}}; \vartheta \rangle$.

Then, we say that I is a stable model of $\mathbb{P}$ if and only if I is a minimal model of the reduct $\mathbb{P}_I$.

Now, we can present the definition of stable model of a MANLP.

Definition 24. Given a MANLP $\mathbb{P}$ and an L-interpretation I, we say that I is a *stable model* of $\mathbb{P}$ if and only if I is a minimal model of $\mathbb{P}_I$.

Indeed, each stable model of a MANLP $\mathbb{P}$ is a minimal model of $\mathbb{P}$ as the next result shows.

Proposition 25. *Any stable model of a MANLP $\mathbb{P}$ is a minimal model of $\mathbb{P}$.*

Proof. Let I be a stable model of $\mathbb{P}$. By definition, I is a minimal model of the program $\mathbb{P}_I$. We will prove by reductio ad absurdum that I is a minimal model of $\mathbb{P}$.

Suppose that there exists an interpretation J such that it is a model of $\mathbb{P}$ and $J \sqsubset I$. That is, $J(p) \prec I(p)$ for each $p \in \Pi_{\mathbb{P}}$. If we prove that J is a model of $\mathbb{P}_I$, we will obtain a contradiction, since I is the minimal model of $\mathbb{P}_I$.

As J is a model of $\mathbb{P}$, for each rule in $\mathbb{P}$ of the form

$$\langle p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]; \vartheta\rangle$$

we obtain that

$$\vartheta \preceq \hat{J}(p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n])$$

That is,

$$\vartheta \preceq J(p) \dot{\leftarrow}_i \dot{@}[J(p_1), \ldots, J(p_m), \dot{\neg}\, J(p_{m+1}), \ldots, \dot{\neg}\, J(p_n)]$$

Because the operator $\neg$ is decreasing, we obtain that $\dot{\neg}\, I(p_k) \prec \dot{\neg}\, J(p_k)$ for all $k \in \{m+1, \ldots, n\}$. Hence, since $\dot{@}$ is monotonic, we can ensure that

$$\dot{@}[J(p_1), \ldots, J(p_m), \dot{\neg} I(p_{m+1}), \ldots, \dot{\neg} I(p_n)] \prec \dot{@}[J(p_1), \ldots, J(p_m), , \dot{\neg} J(p_{m+1}), \ldots, \dot{\neg} J(p_n)]$$

Finally, as the operator $\leftarrow_i$ is decreasing in the antecedent, we can conclude that

$$\begin{aligned}
\vartheta &\preceq J(p) \dot{\leftarrow}_i \dot{@}[J(p_1), \ldots, J(p_m), \dot{\neg}\, J(p_{m+1}), \ldots, \dot{\neg}\, J(p_n)] \\
&\preceq J(p) \dot{\leftarrow}_i \dot{@}[J(p_1), \ldots, J(p_m), \dot{\neg}\, I(p_{m+1}), \ldots, \dot{\neg}\, I(p_n)] \\
&= J(p) \dot{\leftarrow}_i \dot{@}_I[J(p_1), \ldots, J(p_m)]
\end{aligned}$$

so J is a model of $\mathbb{P}_I$, which contradicts the hypothesis. □

The following proposition introduces an important feature of stable models.

Proposition 26. *Any stable model of a MANLP $\mathbb{P}$ is a minimal fix-point of $T_{\mathbb{P}}$.*

Proof. We will prove that the immediate consequence operator of a MANLP $\mathbb{P}$ coincides with the immediate consequence operator of the positive multi-adjoint logic program $\mathbb{P}_I$, for any L-interpretation I.

Given a rule $\langle p \leftarrow_i @[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n]; \vartheta\rangle$ in $\mathbb{P}$, for each L-interpretation I, we obtain the following chain of equalities:

$$\begin{aligned}
\vartheta \,\dot{\&}_i\, \hat{I}(@[p_1, \ldots, p_m, \neg p_{m+1}, \ldots, \neg p_n])\} &= \vartheta \,\dot{\&}_i\, \dot{@}[I(p_1), \ldots, I(p_m), \dot{\neg}\, I(p_{m+1}), \ldots, \dot{\neg}\, I(p_n)] \\
&= \vartheta \,\dot{\&}_i\, \dot{@}_I[I(p_1), \ldots, I(p_m)] \\
&= \vartheta \,\dot{\&}_i\, \hat{I}(@_I[p_1, \ldots, p_m])
\end{aligned}$$

where $\langle p \leftarrow_i @_I[p_1, \dots, p_m]; \vartheta\rangle$ is a rule in $\mathbb{P}_I$. Applying the supremum in both sides of the previous equality, we have that:

$$\begin{aligned} T_{\mathbb{P}}(I)(p) &= \sup\{\vartheta \mathbin{\dot{\&}_i} \hat{I}(\mathcal{B}) \mid \langle p \leftarrow_i \mathcal{B}; \vartheta\rangle \in \mathbb{P}\} \\ &= \sup\{\vartheta \mathbin{\dot{\&}_i} \hat{I}(\mathcal{B}_{\mathcal{I}}) \mid \langle p \leftarrow_i \mathcal{B}_{\mathcal{I}}; \vartheta\rangle \in \mathbb{P}_I\} = T_{\mathbb{P}_I}(I)(p) \end{aligned}$$

for all L-interpretation I.

Now, we will consider a stable model M of $\mathbb{P}$, which is a minimal model of the positive multi-adjoint program $\mathbb{P}_M$, by Definition 24. Taking into account Proposition 22 and Knaster-Tarski's fix-point theorem, we can assert that M is a fix-point of $T_{\mathbb{P}_M}$. As the equality $T_{\mathbb{P}} = T_{\mathbb{P}_M}$ holds, we can conclude that $M = T_{\mathbb{P}_M}(M) = T_{\mathbb{P}}(M)$ and therefore M is a fix-point of $T_{\mathbb{P}}$.

It only remains to demonstrate the minimality of M. Let us assume a fix-point N of $T_{\mathbb{P}}$ satisfying that $N \preceq M$. Then, by Proposition 23, we obtain that N is a model of $\mathbb{P}$. Moreover, by Proposition 25, we have that each stable model of $\mathbb{P}$ is a minimal model of $\mathbb{P}$. Therefore, we conclude that $N = M$. □

In general, the counterpart of Proposition 26 is not true because the $T_{\mathbb{P}}$ operator is not necessarily monotonic.

4. On the existence and unicity of stable models

Interesting results about the existence and unicity of stable models for normal residuated logic programs were presented in [7] and [29]. The aim of this section is to study which conditions are required in order to:

1. generalize the existence of stable models for MANLPs defined on any convex compact set of an euclidean space; and
2. ensure the uniqueness of a stable model for a MANLP defined on the set of subintervals of $[0, 1] \times [0, 1]$, which is denoted as $\mathcal{C}([0, 1]) = \{[x, y] \in [0, 1] \times [0, 1] \mid x \leq y\}$, together with the ordering relation $\leq$ defined as $[a, b] \leq [c, d]$ if and only if $a \leq c$ and $b \leq d$, for all $[a, b], [c, d] \in \mathcal{C}([0, 1])$.

4.1. Existence of stable models in convex compact sets

First of all, we will prove some properties of the set of interpretations. Given a finite multi-adjoint normal logic program $\mathbb{P}$ defined on K, the set of interpretations $\mathcal{I}_{\mathfrak{K}}$, together with the ordering relation defined in Section 2.2, verifies some properties of the underlying lattice. For example, by Proposition 8, we can ensure that $(\mathcal{I}_{\mathfrak{K}}, \sqsubseteq)$ is a complete lattice. Note that, each K-interpretation can be seen as an element of K^n, where n is the cardinal of $\Pi_{\mathbb{P}}$. As a consequence, the set of K-interpretations inherits the properties of K by means of the cartesian product.

In what follows, we will demonstrate that the whole set of interpretations of a MANLP defined on a lattice with convex (closed, respectively) carrier is a convex (compact, respectively) set.

Proposition 27. *Let $\mathbb{P}$ be a MANLP defined on a multi-adjoint normal lattice $(K, \preceq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n, \neg)$ where K is a convex (closed, resp.) set of an euclidean space X. Then the set of K-interpretations of $\mathbb{P}$ is a convex (compact, resp.) set in the set of mappings defined on X.*

PROOF. Given the euclidean space of functions from $\Pi_{\mathbb{P}}$ to K with the ordering relation $\sqsubseteq$ defined on the set of K-interpretations, and two K-interpretations $I, J \in I_{\mathfrak{K}}$, we will demonstrate that $tI + (1-t)J \in I_{\mathfrak{K}}$, for all $t \in [0,1]$. Since $I(p), J(p) \in K$, for all $p \in \Pi_{\mathbb{P}}$, we obtain that $tI(p) + (1-t)J(p) \in K$, and therefore $tI + (1-t)J \in I_{\mathfrak{K}}$, for all $t \in [0,1]$. As a consequence, we conclude that $I_{\mathfrak{K}}$ is a convex set.

In order to prove that $I_{\mathfrak{K}}$ is a compact set, we need to demonstrate that the set of K-interpretations of $\mathbb{P}$ is bounded and closed. Since $(K, \preceq)$ is a bounded lattice, with bottom and top elements $\bot$ and $\top$, respectively, we can define the constant bottom and top interpretation $I_\bot$ and $I_\top$. Taking into account the ordering relation defined on the set of interpretations, we obtain that $I_\bot \sqsubseteq I \sqsubseteq I_\top$, for all K-interpretation I. Therefore, $I_{\mathfrak{K}}$ is a bounded set. On the other hand, we can ensure that $I_{\mathfrak{K}}$ is also closed since each K-interpretation can be seen as an element of K^n, where n is the cardinal of $\Pi_{\mathbb{P}}$, and the cartesian product of closed sets is closed. □

Notice that, if K is closed then it is a compact set, since we are considering a multi-adjoint normal lattice. Hence, from now on, in order to not create confusion we will write compact instead of closed.

Now, we will show the considered mathematical reasoning to demonstrate the theorem related to the existence of stable models in MANLPs. Our purpose is to prove the continuity of the operator R defined by $R(I) = \mathrm{lfp}(T_{\mathbb{P}_I})$, where $\mathrm{lfp}(T_{\mathbb{P}_I})$ is the least fix-point of the operator $T_{\mathbb{P}_I}$, for a given fuzzy K-interpretation I where K is a convex compact set. This fact allows us to apply Theorem 14 and therefore, we can guarantee that an interpretation I exists such that it is the least fix-point of $T_{\mathbb{P}_I}$. Furthermore, this fix-point I is the least model of the positive multi-adjoint logic program $\mathbb{P}_I$. Therefore, we can ensure that I is a stable model of $\mathbb{P}$.

In order to reach this purpose, it will be fundamental to require the continuity of the conjunction connectives, the negation operator and the aggregator operators appearing in the body of the rules of MANLPs.

Theorem 28. *Let $(K, \preceq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n, \neg)$ be a multi-adjoint normal lattice where K is a non-empty convex compact set in an euclidean space and $\mathbb{P}$ be a finite MANLP defined on this lattice. If $\&_1, \dots, \&_n$, $\neg$ and the aggregator operators in the body of the rules of $\mathbb{P}$ are continuous operators, then $\mathbb{P}$ has at least a stable model.*

Proof. Given a MANLP $\mathbb{P}$ and a K-interpretation I, the operator $R(I) = \operatorname{lfp}(T_{\mathbb{P}_I})$ can be expressed as a composition of the operators $\mathcal{F}_1(I) = \mathbb{P}_I$ and $\mathcal{F}_2(\mathbb{P}) = \operatorname{lfp}(T_{\mathbb{P}})$.

Note that $\mathcal{F}_1$ is a mapping from the set of K-interpretations to $(\mathfrak{R}^+_{\Pi \uplus K})^k = \mathfrak{R}^+_{\Pi \uplus K} \times \overset{k)}{\dots} \times \mathfrak{R}^+_{\Pi \uplus K}$, where k is the number of rules in $\mathbb{P}$. For each K-interpretation I, we obtain:

$$\mathcal{F}_1(I) = (p^1 \leftarrow_{j_1} @^1_I[p^1_1, \dots, p^1_m], \dots, p^k \leftarrow_{j_k} @^k_I[p^k_1, \dots, p^k_m])$$

where $j_i \in \{1, \dots, n\}$, with $i \in \{1, \dots, k\}$. Hence, $\mathcal{F}_1$ is a continuous mapping if and only if each component of $\mathcal{F}_1$ is continuous. But this is trivial since $@^i_I$ are continuous operators, by hypothesis.

On the other hand, $\mathcal{F}_2$ is a mapping from $(\mathfrak{R}^+_{\Pi \uplus K})^k$ to the set of K-interpretations. Since every operator used in the computation of $T_{\mathbb{P}}$ is continuous, we can ensure that the immediate consequence operator is continuous. In addition, taking into account Proposition 5.4 in [24], we can obtain the least fix-point of $T_{\mathbb{P}}$ by iterating ω times the inmediate consequence operator from the bottom interpretation. Hence, $\mathcal{F}_2$ is a continuos operator since it is a numerable composition of continuous operators.

Consequently, $R(I) = \operatorname{lfp}(T_{\mathbb{P}_I})$ is continuous because it is composition of two continuous operators. Applying Theorem 14 to the operator R, we conclude that R has a fix-point. Moreover, this fix-point coincides with the least model of $\mathbb{P}_I$ since it is a positive multi-adjoint logic program and we can apply Proposition 23. Thus, it is a stable model of $\mathbb{P}$. □

The following examples illustrate the result obtained in Theorem 28.

Example 29. Consider the euclidean space $(X, \oplus, \otimes, \mathbb{R})$ where X is the space of triangular functions defined by:

$$f_n(z) = \begin{cases} 10(z-n)+1 & \text{if} \quad n-0.1 \leq z \leq n \\ 10(n-z)+1 & \text{if} \quad n \leq z < n+0.1 \\ 0 & \text{otherwise} \end{cases}$$

with $n \in \mathbb{R}$. The operations $\oplus, \otimes \colon X \to X$ are defined as $f_n \oplus f_m = f_{n+m}$ and $k \otimes f_n = f_{k \cdot n}$, respectively, where $n, m, k \in \mathbb{R}$.

Now, we will consider the set of functions $K = \{f_x \mid x \in [0,1]\}$ together with the following ordering relation: $f_n \leq f_m$ if and only if $n \leq m$, for all $n, m \in \mathbb{R}$.

In order to see that K is a convex set, for all $f_x, f_y \in K$ and $t \in [0,1]$, we will prove that $t \otimes f_x \oplus (1-t) \otimes f_y \in K$, which is equivalent to demonstrate that $f_{t \cdot x+(1-t) \cdot y} \in K$. Clearly, $t \cdot x + (1-t) \cdot y \in [0,1]$ since $x, y, t \in [0,1]$. Therefore, we obtain that $f_{t \cdot x+(1-t) \cdot y} \in K$ and consequently K is convex.

From the ordering relation defined previously, we can assert that K is a bounded set because $f_x \leq f_1$, for all $f_x \in K$. Furthermore, K is a closed set since the boundary of K is contained by it, that is, $\{f_0, f_1\} \subseteq K$.

Hence, we can conclude that K is a convex compact set in X. Therefore, Theorem 28 ensures that every multi-adjoint normal logic program $\mathbb{P}$ defined on the multi-adjoint normal lattice $(K, \leq, \leftarrow_1, \&_1, \ldots, \leftarrow_n, \&_n, \neg)$, where the conjunctors, the negation and the aggregator operators in the body of the rules of $\mathbb{P}$ are continuous, has at least a stable model. □

Analogously, we can consider the functions f_n with any width different from 0.1. This kind of functions are interpreted as fuzzy numbers. A similar example can be obtained when we consider the set of functions f_x where x is an element of an arbitrary convex compact set.

The following example considers another algebraic structure with a more general family of triangular functions.

Example 30. Let $(X, \oplus, \otimes, \mathbb{R})$ be an euclidean space such that X is composed of the triangular functions f_{a_1,a_2,a_3} defined as follows:

$$f_{a_1,a_2,a_3}(z) = \begin{cases} \frac{z-a_1}{a_2-a_1} & \text{if} \quad a_1 \leq z \leq a_2 \\ \frac{a_3-z}{a_3-a_2} & \text{if} \quad a_2 \leq z \leq a_3 \\ 0 & \text{otherwise} \end{cases}$$

where $a_1, a_2, a_3 \in \mathbb{R}$. For all $a_i, b_i, k \in \mathbb{R}$ with $i \in \{1, 2, 3\}$, we define the operations $\oplus, \otimes\colon X \to X$ as:

$$\begin{aligned} f_{a_1,a_2,a_3} \oplus f_{b_1,b_2,b_3} &= f_{a_1+b_1,a_2+b_2,a_3+b_3} \\ k \otimes f_{a_1,a_2,a_3} &= f_{k\cdot a_1,k\cdot a_2,k\cdot a_3} \end{aligned}$$

which clearly are well defined. Following an analogous reasoning to the previous example and considering the ordering relation $f_{x_1,x_2,x_3} \leq f_{y_1,y_2,y_3}$ if and only if $x_1 \leq y_1$, $x_2 \leq y_2$ and $x_3 \leq y_3$, we can ensure that $K = \{f_{x_1,x_2,x_3} \mid x_1, x_2, x_3 \in [0, 1]\}$ is a convex compact set. Once again, we can assert that every multi-adjoint normal logic program defined on $(K, \leq, \leftarrow_1, \&_1, \dots, \leftarrow_n, \&_n, \neg)$, such that $\&_1, \dots, \&_n, \neg$ and the aggregator operators in the body of the rules of $\mathbb{P}$ are continuous operators, has at least a stable model. $\square$

As usual, the existence theorem does not ensure the uniqueness of stable models, as we will show next.

Example 31. We will consider the following MANLP $\mathbb{P}$, defined on the multi-adjoint normal lattice $\langle[0, 1], \leq, \leftarrow_G, \&_G, \leftarrow_P, \&_P, \neg\rangle$, with five rules and one fact.

$$\begin{array}{ll} r_1: \langle p \leftarrow_G \neg t \ ; \ 0.6\rangle & r_4: \langle t \leftarrow_P s \ ; \ 1\rangle \\ r_2: \langle q \leftarrow_P \neg s \ ; \ 0.8\rangle & r_5: \langle s \leftarrow_P 1 \ ; \ 0.5\rangle \\ r_3: \langle p \leftarrow_P q \ \&_P s \ ; \ 0.9\rangle & r_6: \langle t \leftarrow_G \neg q \ \&_G \neg p \ ; \ 0.7\rangle \end{array}$$

Notice that, the set of propositional symbols is given by $\Pi_{\mathbb{P}} = \{p, q, s, t\}$ and the operators included in the multi-adjoint normal lattice are the Gödel and product conjunctors, $\&_G$ and $\&_P$, together with their corresponding adjoint implications, $\leftarrow_G$ and $\leftarrow_P$. The considered negation operator is the standard negation defined as $\neg(x) = 1 - x$, for each $x \in [0, 1]$.

Clearly, $[0, 1]$ is a convex compact set and the operators in the body of the rules of $\mathbb{P}$ are continuous. Therefore, applying Theorem 28, we can guarantee that the multi-adjoint normal logic program $\mathbb{P}$ defined on the multi-adjoint normal lattice $\langle[0, 1], \leq, \leftarrow_G, \&_G, \leftarrow_P, \&_P, \neg\rangle$ has at least a stable model. In the following, we will compute two different stable models.

From the interpretation $M \equiv \{(p, 0.4), (q, 0.4), (s, 0.5), (t, 0.6)\}$, we can define the corresponding reduct $\mathbb{P}_M$ as follows:

$$\begin{array}{ll} r_1^M: \langle p \leftarrow_G 0.4 \ ; \ 0.6\rangle & r_4^M: \langle t \leftarrow_P s \ ; \ 1\rangle \\ r_2^M: \langle q \leftarrow_P 0.5 \ ; \ 0.8\rangle & r_5^M: \langle s \leftarrow_P 1 \ ; \ 0.5\rangle \\ r_3^M: \langle p \leftarrow_P q \ \&_P s \ ; \ 0.9\rangle & r_6^M: \langle t \leftarrow_G 0.6 \ ; \ 0.7\rangle \end{array}$$

First of all, we compute the least model of the program $\mathbb{P}_M$. For that, since $\mathbb{P}_M$ is a positive program, we iterate the $T_{\mathbb{P}_M}$ operator from the minimum interpretation $I_\bot$.

	p	q	s	t
$I_\bot$	0	0	0	0
$T_{\mathbb{P}_M}(I_\bot)$	0.4	0.4	0.5	0.6
$T^2_{\mathbb{P}_M}(I_\bot)$	0.4	0.4	0.5	0.6

Consequently, since $T_{\mathbb{P}_M}(I_\bot) = M$ and it is the least fix-point of $T_{\mathbb{P}_M}$, M is the least model of the reduct $\mathbb{P}_M$, which allows us to ensure that M is a stable model of the program $\mathbb{P}$.

Now, we will show that M is not the unique stable model of $\mathbb{P}$. Let M' be the interpretation given by $M' \equiv \{(p, 0.5), (q, 0.4), (s, 0.5), (t, 0.5)\}$ Then, the corresponding reduct $\mathbb{P}_{M'}$ is defined as:

$$\begin{array}{llll} r_1^{M'}: & \langle p \leftarrow_G 0.5\ ;\ 0.6\rangle & r_4^{M'}: & \langle t \leftarrow_P s\ ;\ 1\rangle \\ r_2^{M'}: & \langle q \leftarrow_P 0.5\ ;\ 0.8\rangle & r_5^{M'}: & \langle s \leftarrow_P 1\ ;\ 0.5\rangle \\ r_3^{M'}: & \langle p \leftarrow_P q\ \&_P s\ ;\ 0.9\rangle & r_6^{M'}: & \langle t \leftarrow_G 0.5\ ;\ 0.7\rangle \end{array}$$

By an analogous reasoning to that given for the least model of the reduct $\mathbb{P}_M$, it can be easily proved that M' is the least model of $\mathbb{P}_{M'}$. Therefore, M' is also a stable model of the program $\mathbb{P}$. □

Hence, ensuring the existence of a unique stable model is an important challenge. For example, when the logic program is associated with a search problem, if the stable model is unique, then the problem is solvable and it has a unique solution, which determines the optimal information we can obtain from the knowledge system. Therefore, studying sufficient conditions in order to ensure the uniqueness is an important goal, which will be developed in the next section.

4.2. Unicity of stable models in MANLPs defined on $\mathcal{C}([0,1])$

As we argued above and in the introduction section, the characterization of programs with a unique stable model is important. This section will consider a special algebraic structure and sufficient conditions from which we can ensure the unicity of stable models for multi-adjoint normal logic programs defined on the set of subintervals of $[0,1] \times [0,1]$, which is denoted by $\mathcal{C}([0,1])$.

In the following, we will present the particular operators which are considered in the programs.

Definition 32. Given $\alpha, \beta, \gamma, \delta \in \mathbb{N}$ such that $\beta \leq \alpha$ and $\delta \leq \gamma$, the operator $\&_{\beta\delta}^{\alpha\gamma} \colon \mathcal{C}([0,1])^2 \to \mathcal{C}([0,1])$ defined as:

$$\&_{\beta\delta}^{\alpha\gamma}([a,b],[c,d]) = [\, a^{\alpha} * c^{\gamma},\, b^{\beta} * d^{\delta} \,]$$

with $a, b, c, d \in \mathbb{R}$ and $*$ being the usual product among real numbers, will be called *exponential interval product with respect to* α, β, γ *and* δ (*ei-product*, in short).

Note that every ei-product with respect to four natural numbers α, β, γ and δ is well defined, since these values satisfy that $\beta \leq \alpha$ and $\delta \leq \gamma$.

In [30], different properties of these operators were introduced. In particular, the existence of the residuated implication $\leftarrow_{\beta\delta}^{\alpha\gamma}$ was proved (see [30, Theorem 1] for more details). Hence, we have that $(\&_{\beta\delta}^{\alpha\gamma}, \leftarrow_{\beta\delta}^{\alpha\gamma})$ forms an adjoint pair and so, they can be used in any multi-adjoint normal logic program.

An extension on $\mathcal{C}([0,1])$ of the standard negation will be the negation operator that we will consider in the programs. Specifically, this operator $\neg \colon \mathcal{C}([0,1]) \to \mathcal{C}([0,1])$ will be defined as $\neg[a,b] = [1-b, 1-a]$ for all $[a,b] \in \mathcal{C}([0,1])$, which is clearly decreasing and satisfies $\neg[0,0] = [1,1]$ and $\neg[1,1] = [0,0]$.

Before presenting the results associated with the unicity of stable models, we will take into consideration the following remarks. Due to the relation between $\mathcal{C}([0,1])$ and $[0,1] \times [0,1]$, we can introduce the inclusion mapping $\iota \colon \mathcal{C}([0,1]) \to [0,1] \times [0,1]$, defined as $\iota([a,b]) = (a,b)$. This mapping can easily be extended to tuples as follows, $\iota \colon \mathcal{C}([0,1])^n \to [0,1]^n \times [0,1]^n$, defined by $\iota([a_1,b_1],\ldots,[a_n,b_n]) = (a_1,\ldots,a_n,b_1,\ldots,b_n)$.

On the other hand, given the propositional symbols $p_1, \ldots, p_n$ appearing in $\mathbb{P}$, we can express each $\mathcal{C}([0,1])$-interpretation I as a tuple $(I(p_1),\ldots,I(p_n))$ which belongs to $(\mathcal{C}([0,1]))^n$. Therefore, the mapping ι can be defined on the set of $\mathcal{C}([0,1])$-interpretations and the image of each $\mathcal{C}([0,1])$-interpretation will be a n-tuple, which will be denoted with a bar, that is, given a $\mathcal{C}([0,1])$-interpretation I, we will write $\iota(I) = \bar{I}$.

For example, given $\Pi = \{p, q, s\}$ and the $\mathcal{C}([0,1])$-interpretation $I \colon \Pi \to \mathcal{C}([0,1])$, defined as $I(p) = [0.1, 0.4]$, $I(q) = [0,0]$, and $I(s) = [0.7, 0.9]$, if we consider the alphabetical ordering among the propositional symbols, I can be written as the tuple $\bar{I} = ([0.1, 0.4], [0,0], [0.7, 0.9])$.

Moreover, $T_{\mathbb{P}}$ can be considered as a real function from $\mathcal{C}([0,1])^n$ to $\mathcal{C}([0,1])^n$ since it assigns $\mathcal{C}([0,1])$-interpretations to $\mathcal{C}([0,1])$-interpretations.

Hence, we will write $T_{\mathbb{P}}(I)(p_i) = (T_{\mathbb{P}})_i(I)$ in order to express the value of $T_{\mathbb{P}}(I)$ for each propositional symbol p_i and we also have $(T_{\mathbb{P}})_i(I) = [(T_{\mathbb{P}})_i^1(I), (T_{\mathbb{P}})_i^2(I)]$ since $(T_{\mathbb{P}})_i(I) \in \mathcal{C}([0,1])$. Considering the mapping ι, we can write the n-tuple $([a_1, b_1], \dots, [a_n, b_n])$ in $\mathcal{C}([0,1])^n$ as $(a_1, \dots, a_n, b_1, \dots, b_n)$. This fact allows us to define the mapping $\overline{T_{\mathbb{P}}} \colon \iota(\mathcal{I}_{\mathfrak{L}}) \to [0,1]^n \times [0,1]^n$ as $\overline{T_{\mathbb{P}}}(\iota(I)) = \iota(T_{\mathbb{P}}(I))$, for each $I \in \mathcal{I}_{\mathfrak{L}}$. Notice that $\iota(\mathcal{I}_{\mathfrak{L}}) = \{\iota(I) \mid I \in \mathcal{C}([0,1])^n\}$.

Taking into account these previous considerations, we will introduce a lemma required to prove the uniqueness of the stable models for multi-adjoint normal programs defined on $\mathcal{C}([0,1])$ by using the extension of the standard negation defined above and a family of adjoint pairs formed by different ei-products together with their corresponding residuated implications.

Lemma 33. *Let $\mathbb{P}$ be a MANLP defined on a multi-adjoint normal lattice $(\mathcal{C}([0,1]), \leq, \leftarrow^{\alpha_1\gamma_1}_{\beta_1\delta_1}, \&^{\alpha_1\gamma_1}_{\beta_1\delta_1}, \dots, \leftarrow^{\alpha_m\gamma_m}_{\beta_m\delta_m}, \&^{\alpha_m\gamma_m}_{\beta_m\delta_m}, \neg)$ such that at most one rule with head p appears in $\mathbb{P}$ and the only possible operators in the body of the rules*[2] *are $\&^{\alpha\gamma}_{\beta\delta}$ with $\alpha = \beta = \gamma = \delta = 1$. If $I = [I^1, I^2]$ and $J = [J^1, J^2]$ are two $\mathcal{C}([0,1])$-interpretations, such that $J \sqsubseteq I$, then:*

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1), \dots, J^1(p_n), J^2(p_1), \dots, J^2(p_n))\right| \leq \alpha$$

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^2}{\partial p_j^l}(J^1(p_1), \dots, J^1(p_n), J^2(p_1), \dots, J^2(p_n))\right| \leq \beta$$

where

$$\begin{aligned}\alpha &= \sum_{j=1}^{h}(\vartheta^1)^{\alpha_w} \cdot \gamma_w \cdot I^1(q_j)^{\gamma_w - 1} \cdot (I^1(q_1) \cdots I^1(q_{j-1}) \cdot I^1(q_{j+1}) \cdots I^1(q_h))^{\gamma_w} \\ &\quad + (\vartheta^1)^{\alpha_w} \cdot \gamma_w \cdot (k-h)(I^1(q_1) \cdots I^1(q_h))^{\gamma_w} \\ \beta &= \sum_{j=1}^{h}(\vartheta^2)^{\beta_w} \cdot \delta_w \cdot I^2(q_j)^{\delta_w - 1} \cdot (I^2(q_1) \cdots I^2(q_{j-1}) \cdot I^2(q_{j+1}) \cdots I^2(q_h))^{\delta_w} \\ &\quad + (\vartheta^2)^{\beta_w} \cdot \delta_w \cdot (k-h)(I^2(q_1) \cdots I^2(q_h))^{\delta_w}\end{aligned}$$

[2]Notice that the implications in the rules can be the residuated implications of any general ei-product.

and $\langle p_i \leftarrow^{\alpha_w\gamma_w}_{\beta_w\delta_w} q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k;\ [\vartheta^1, \vartheta^2]\rangle$*, with* $w \in \{1, \ldots, m\}$*, is the unique rule in* $\mathbb{P}$ *with head* p_i*.*

PROOF. Let us assume that only one rule in $\mathbb{P}$ with head p_i exists, that is, $\langle p_i \leftarrow^{\alpha_w\gamma_w}_{\beta_w\delta_w} q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k;\ [\vartheta^1, \vartheta^2]\rangle$. Hence, we have that $(T_{\mathbb{P}})_i(I)$ is equal to:

$$[\vartheta^1, \vartheta^2] \dot{\&}^{\alpha_w\gamma_w}_{\beta_w\delta_w} \hat{I}(q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k)$$

Note that, by definition of the negation operator used here, we obtain that:

$$\hat{I}^1(q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k) = I^1(q_1) * \cdots * I^1(q_h) * (1 - I^2(q_{h+1})) * \cdots * (1 - I^2(q_k))$$

$$\hat{I}^2(q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k) = I^2(q_1) * \cdots * I^2(q_h) * (1 - I^1(q_{h+1})) * \cdots * (1 - I^1(q_k))$$

Therefore, considering each component of the immediate consequence operator and Definition 32, we have that:

$$(T_{\mathbb{P}})^1_i(I) = (\vartheta^1)^{\alpha_w} * \big(I^1(q_1) * \cdots * I^1(q_h) * (1 - I^2(q_{h+1})) * \cdots * (1 - I^2(q_k))\big)^{\gamma_w}$$

$$(T_{\mathbb{P}})^2_i(I) = (\vartheta^2)^{\beta_w} * \big(I^2(q_1) * \cdots * I^2(q_h) * (1 - I^1(q_{h+1})) * \cdots * (1 - I^1(q_k))\big)^{\delta_w}$$

Now, by using the mapping ι, the first component of the immediate consequence operator can be written as[3]:

$$(\overline{T_{\mathbb{P}}})^1_i(p^1_1, \ldots, p^1_n, p^2_1, \ldots, p^2_n) = (\vartheta^1)^{\alpha_w} \cdot \big(q_1 * \cdots * q_k * (1 - q_{k+1}) * \cdots * (1 - q_m)\big)^{\gamma_w}$$

where each q_j with $j \leq k$ is actually some $p^1_1, \ldots, p^1_n$ and each q_j with $j > k$ is some $p^2_1, \ldots, p^2_n$.

Analogously, the second component of the immediate consequence operator can be expressed as:

$$(\overline{T_{\mathbb{P}}})^2_i(p^1_1, \ldots, p^1_n, p^2_1, \ldots, p^2_n) = (\vartheta^2)^{\beta_w} \cdot \big(q_1 * \cdots * q_k * (1 - q_{k+1}) * \cdots * (1 - q_m)\big)^{\delta_w}$$

where each q_j with $j \leq k$ is actually some $p^2_1, \ldots, p^2_n$ and each q_j with $j > k$ is some $p^1_1, \ldots, p^1_n$.

Since $(\overline{T_{\mathbb{P}}})^1_i$ and $(\overline{T_{\mathbb{P}}})^2_i$ are composition of differentiable mappings, they are also differentiable mappings. Considering only $(\overline{T_{\mathbb{P}}})^1_i$, we will compute its partial derivatives distinguishing different cases:

[3]Notice that the variables have been denoted with the propositional symbols, abusing of notation.

(a) If $p_j = q_t$, with $t \leq h$, then

$$\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j} = (\vartheta^1)^{\alpha_w} \cdot \gamma_w \cdot q_t^{\gamma_w - 1} \cdot (q_1 \cdots q_{t-1} \cdot q_{t+1} \cdots q_h \cdot (1-q_{h+1}) \cdots (1-q_k))^{\gamma_w}$$

(b) When $p_j = q_t$, with $t > h$, we have:

$$\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j} = (\vartheta^1)^{\alpha_w} \cdot (-\gamma_w) \cdot (1 - q_t)^{\gamma_w - 1}(q_1 \cdots q_h \cdot (1 - q_{h+1}) \cdots (1 - q_{t-1}) \cdot (1 - q_{t+1}) \cdots (1 - q_k))^{\gamma_w}$$

(c) Otherwise, $\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j} = 0$.

Note that, by the definition of the MANLP $\mathbb{P}$, all propositional symbols appearing in the body of a rule are different.

From the computations above, we obtain that the sum of all partial derivatives evaluated in $(J^1(p_1), \ldots, J^1(p_n), J^2(p_1), \ldots, J^2(p_n))$ verifies the next inequality:

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| =$$

$$= \sum_{j=1}^{h}\left|\vartheta^{\alpha_w}\cdot\gamma_w\cdot J^1(q_j)^{\gamma_w-1}\cdot\Big(J^1(q_1)\cdots J^1(q_{j-1})\cdot J^1(q_{j+1})\cdots J^1(q_h)\cdot(1-J^2(q_{h+1}))\cdots(1-J^2(q_k))\Big)^{\gamma_w}\right|$$

$$+\sum_{j=h+1}^{k}\Big|\vartheta^{\alpha_w}\cdot(-\gamma_w)\cdot(1-J^2(q_j)^{\gamma_w-1})\cdot\Big(J^1(q_1)\cdots J^1(q_h)\cdot(1-J^2(q_{h+1}))\cdots$$

$$\cdots(1-J^2(q_{j-1}))\cdot\ (1-J^2(q_{j+1}))\cdots(1-J^2(q_k))\Big)^{\gamma_w}\Big|$$

$$\le\left(\sum_{j=1}^{h}\vartheta^{\alpha_w}\cdot\gamma_w\cdot J^1(q_j)^{\gamma_w-1}\cdot\Big(J^1(q_1)\cdots J^1(q_{j-1})\cdot J^1(q_{j+1})\cdots J^1(q_h)\Big)^{\gamma_w}\right)$$
$$+\sum_{j=h+1}^{k}\vartheta^{\alpha_w}\cdot\gamma_w\cdot(1-J^2(q_j)^{\gamma_w-1})(J^1(q_1)\cdots J^1(q_h))^{\gamma_w}$$

$$\le\left(\sum_{j=1}^{h}\vartheta^{\alpha_w}\cdot\gamma_w\cdot J^1(q_j)^{\gamma_w-1}\cdot\Big(J^1(q_1)\cdots J^1(q_{j-1})\cdot J^1(q_{j+1})\cdots J^1(q_h)\Big)^{\gamma_w}\right)$$
$$+\ \vartheta^{\alpha_w}\cdot\gamma_w\cdot(k-h)\cdot(J^1(q_1)\cdots J^1(q_h))^{\gamma_w}$$

$$\le\left(\sum_{j=1}^{h}\vartheta^{\alpha_w}\cdot\gamma_w\cdot I^1(q_j)^{\gamma_w-1}\cdot\Big(I^1(q_1)\cdots I^1(q_{j-1})\cdot I^1(q_{j+1})\cdots I^1(q_h)\Big)^{\gamma_w}\right)$$
$$+\ \vartheta^{\alpha_w}\cdot\gamma_w\cdot(k-h)\cdot(I^1(q_1)\cdots I^1(q_h))^{\gamma_w}$$

An analogous reasoning with respect to the second component of the immediate consequence operator leads us to conclude that:

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^2}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right|\le\beta$$

where

$$\beta \;=\; \sum_{j=1}^{h}(\vartheta^2)^{\beta_w}\cdot\delta_w\cdot I^2(q_j)^{\delta_w-1}\cdot(I^2(q_1)\cdots I^2(q_{j-1})\cdot I^2(q_{j+1})\cdots I^2(q_h))^{\delta_w}$$
$$+\ (\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(k-h)(I^2(q_1)\cdots I^2(q_h))^{\delta_w}$$

□

Based on the previous results, the following theorem is focused on the uniqueness of stable models. The proof will be based on demonstrating that $T_{\mathbb{P}}$ has only one fix-point in $\mathcal{C}([0,1])^n$. Considering Proposition 26 and Theorem 28, we can state that each stable model of $\mathbb{P}$ is a minimal fix-point of $T_{\mathbb{P}}$ and there exists at least one stable model of $\mathbb{P}$, respectively. These facts will lead us to conclude that the only fix-point of $T_{\mathbb{P}}$ is the unique stable model.

Theorem 34. *Let $\mathbb{P}$ be a finite MANLP defined on the multi-adjoint normal lattice $(\mathcal{C}([0,1]), \leq, \leftarrow^{\alpha_1\gamma_1}_{\beta_1\delta_1}, \&^{\alpha_1\gamma_1}_{\beta_1\delta_1}, \dots, \leftarrow^{\alpha_m\gamma_m}_{\beta_m\delta_m}, \&^{\alpha_m\gamma_m}_{\beta_m\delta_m}, \neg)$ such that the only possible operators in the body of the rules are $\&^{\alpha\gamma}_{\beta\delta}$ with $\alpha = \beta = \gamma = \delta = 1$, and $[\vartheta^1_p, \vartheta^2_p] = \max\{[\vartheta^1, \vartheta^2] \mid \langle p \leftarrow^{\alpha_w\gamma_w}_{\beta_w\delta_w} \mathcal{B}; [\vartheta^1, \vartheta^2]\rangle \in \mathbb{P}\}$. If the inequality*

$$\sum_{j=1}^{h} (\vartheta^2)^{\beta_w} \cdot \delta_w \cdot (\vartheta^2_{q_j})^{\delta_w - 1} \cdot \left(\vartheta^2_{q_1} \cdots \vartheta^2_{q_{j-1}} \cdot \vartheta^2_{q_{j+1}} \cdots \vartheta^2_{q_h}\right)^{\delta_w} + (\vartheta^2)^{\beta_w} \cdot \delta_w \cdot (k-h)(\vartheta^2_{q_1} \cdots \vartheta^2_{q_h})^{\delta_w} < 1$$

holds for every rule $\langle p \leftarrow^{\alpha_w\gamma_w}_{\beta_w\delta_w} q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k; [\vartheta^1, \vartheta^2]\rangle \in \mathbb{P}$, with $w \in \{1, \dots, m\}$, then there exists a unique stable model of $\mathbb{P}$.

Proof. Given the $\mathcal{C}([0,1])$-interpretation I_ϑ, which assigns the value $[\vartheta^1_p, \vartheta^2_p]$ to each propositional symbol $p \in \Pi_{\mathbb{P}}$, the natural number n representing the number of propositional symbols in $\Pi_{\mathbb{P}}$ and the set $A = \{\iota(J) \mid J \in \mathcal{C}([0,1])^n \text{ and } J \sqsubseteq I_\vartheta\}$, we will begin proving that $\overline{T_{\mathbb{P}}}$ is a contractive mapping in A with respect to the supremum norm $|| \cdot ||_\infty$. That is, we will demonstrate that there exists a real value $0 < \lambda < 1$ such that:

$$||\overline{T_{\mathbb{P}}}(\overline{J_1}) - \overline{T_{\mathbb{P}}}(\overline{J_2})||_\infty \leq ||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda \qquad (1)$$

for each pair of $\overline{J_1}, \overline{J_2} \in A$. This fact will allow us to apply Banach fix-point theorem and to ensure that $\overline{T_{\mathbb{P}}}$ has only one fix-point in A. To reach this purpose, we distinguish two cases:

Base Case: Given a multi-adjoint normal logic program $\mathbb{P}$, we will suppose that there exists at most one rule in $\mathbb{P}$ with head p, for each propositional symbol $p \in \Pi_{\mathbb{P}}$. In order to prove Equation (1), we will apply the mean value theorem [45] on each component of $\overline{T_{\mathbb{P}}} = [\overline{T_{\mathbb{P}}}^1, \overline{T_{\mathbb{P}}}^2]$. First of all, we have to prove that the conditions of this theorem are satisfied. Considering $\overline{T_{\mathbb{P}}}^1$ and $\overline{T_{\mathbb{P}}}^2$ as functions defined on $\mathbb{R}^n$ in the way mentioned in Lemma 33, both of them are differentiable functions in $\mathbb{R}^n$. Moreover, for all $\overline{J_1}, \overline{J_2} \in A$, the line segment $S(\overline{J_1}, \overline{J_2}) = \{(1-t) \cdot \overline{J_1} + t \cdot \overline{J_2} \mid 0 \leq t \leq 1\}$

is contained in A. Hence, we can apply the mean value theorem on each component of $\overline{T_{\mathbb{P}}}$, obtaining that:

$$||\overline{T_{\mathbb{P}}}^1(\overline{J_1})-\overline{T_{\mathbb{P}}}^1(\overline{J_2})||_\infty \leq ||\overline{J_1}-\overline{J_2}||_\infty \cdot \sup\{||D\overline{T_{\mathbb{P}}}^1(\overline{J})||_\infty \mid \overline{J} \in S(\overline{J_1},\overline{J_2})\} \quad (2)$$

$$||\overline{T_{\mathbb{P}}}^2(\overline{J_1})-\overline{T_{\mathbb{P}}}^2(\overline{J_2})||_\infty \leq ||\overline{J_1}-\overline{J_2}||_\infty \cdot \sup\{||D\overline{T_{\mathbb{P}}}^2(\overline{J})||_\infty \mid \overline{J} \in S(\overline{J_1},\overline{J_2})\} \quad (3)$$

where

$$\begin{aligned} ||D\overline{T_{\mathbb{P}}}^1(\overline{J})||_\infty &= \sup\{||D\overline{T_{\mathbb{P}}}^1(\overline{J})(x)||_\infty \mid ||x||_\infty \leq 1\} \\ ||D\overline{T_{\mathbb{P}}}^2(\overline{J})||_\infty &= \sup\{||D\overline{T_{\mathbb{P}}}^2(\overline{J})(x)||_\infty \mid ||x||_\infty \leq 1\} \\ ||x||_\infty &= \max\{|x_i| \mid x = (x_1,\ldots,x_n) \in \mathbb{R}^n\} \end{aligned}$$

In order to prove that $\overline{T_{\mathbb{P}}}^1$ is a contractive mapping in A, some previous considerations must be taken into account:

(a) The line segment $S(\overline{J_1},\overline{J_2}) = \{(1-t)\cdot\overline{J_1} + t\cdot\overline{J_2} \mid 0 \leq t \leq 1\}$ is a compact set and consequently $\sup\{||D\overline{T_{\mathbb{P}}}^1(\overline{J})||_\infty \mid \overline{J} \in S(\overline{J_1},\overline{J_2})\}$ is a maximum.

(b) Since only one rule with head p exists in $\mathbb{P}$, the conditions required in Lemma 33 are satisfied and, therefore, for each $\overline{J} \in A$, the following inequalities hold:

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| \leq$$

$$\begin{aligned} &\leq \sum_{j=1}^{h}(\vartheta^1)^{\alpha_w}\cdot\gamma_w\cdot I_\vartheta^1(q_j)^{\gamma_w-1}\cdot(I_\vartheta^1(q_1)\cdots I_\vartheta^1(q_{j-1})\cdot I_\vartheta^1(q_{j+1})\cdots I_\vartheta^1(q_h))^{\gamma_w} \\ &\quad + (\vartheta^1)^{\alpha_w}\cdot\gamma_w\cdot(k-h)(I_\vartheta^1(q_1)\cdots I_\vartheta^1(q_h))^{\gamma_w} \\ &= \sum_{j=1}^{h}(\vartheta^1)^{\alpha_w}\cdot\gamma_w\cdot(\vartheta_{q_j}^1)^{\gamma_w-1}\cdot\left(\vartheta_{q_1}^1\cdots\vartheta_{q_{j-1}}^1\cdot\vartheta_{q_{j+1}}^1\cdots\vartheta_{q_h}^1\right)^{\gamma_w} \\ &\quad + (\vartheta^1)^{\alpha_w}\cdot\gamma_w\cdot(k-h)(\vartheta_{q_1}^1\cdots\vartheta_{q_h}^1)^{\gamma_w} = \lambda^1 \end{aligned}$$

An analogous reasoning can be given for $\overline{T_{\mathbb{P}}}^2$, obtaining:

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^2}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| \leq$$

$$\begin{aligned}
\leq\ & \sum_{j=1}^{h}(\vartheta^2)^{\beta_w}\cdot\delta_w\cdot I_\vartheta^2(q_j)^{\delta_w-1}\cdot(I_\vartheta^2(q_1)\cdots I_\vartheta^2(q_{j-1})\cdot I_\vartheta^2(q_{j+1})\cdots I_\vartheta^2(q_h))^{\delta_w}\\
& + (\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(k-h)(I_\vartheta^2(q_1)\cdots I_\vartheta^2(q_h))^{\delta_w}\\
=\ & \sum_{j=1}^{h}(\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(\vartheta_{q_j}^2)^{\delta_w-1}\cdot\left(\vartheta_{q_1}^2\cdots\vartheta_{q_{j-1}}^2\cdot\vartheta_{q_{j+1}}^2\cdots\vartheta_{q_h}^2\right)^{\delta_w}+\\
& + (\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(k-h)(\vartheta_{q_1}^2\cdots\vartheta_{q_h}^2)^{\delta_w}=\lambda^2
\end{aligned}$$

Taking into account the hypothesis, for each $\overline{J}\in A$, we have that $\lambda^2<1$ and so,

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^2}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| \leq \lambda^2<1 \qquad (4)$$

According to the ordering defined on $\mathcal{C}([0,1])$ and considering that $\beta_w\leq\alpha_w$ y $\delta_w\leq\gamma_w$, we can ensure that $\lambda^1\leq\lambda^2$. Hence, by hypothesis, we deduce that:

$$\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| \leq \lambda^1<1 \qquad (5)$$

From Equation (5), for each $\overline{J}\in A$ and $x\in\mathbb{R}^n$ such that $||x||_\infty\leq 1$, we obtain:

$$||(D\overline{T_{\mathbb{P}}}^1(\overline{J}))(x)||_\infty =$$

$$\overset{(1)}{=}\max\left\{\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\cdot x_j\right| \mid i\in\{1,\ldots,n\}\right\}$$

$$\begin{aligned}
\leq\ & \max\left\{\sum_{l=1}^{2}\sum_{j=1}^{n}\left|\frac{\partial(\overline{T_{\mathbb{P}}})_i^1}{\partial p_j^l}(J^1(p_1),\ldots,J^1(p_n),J^2(p_1),\ldots,J^2(p_n))\right| \mid i\in\{1,\ldots,n\}\right\}\\
\leq\ & \lambda^1
\end{aligned}$$

for all $\overline{J} \in S(\overline{J_1}, \overline{J_2})$, where (1) is given by definition of $||\cdot||_\infty$ on matrices.

Considering Equation (2), we can conclude that:

$$\begin{aligned} ||\overline{T_{\mathbb{P}}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}}}^1(\overline{J_2})||_\infty &\leq ||\overline{J_1} - \overline{J_2}||_\infty \cdot \sup\{||D\overline{T_{\mathbb{P}}}^1(\overline{J})||_\infty \mid \overline{J} \in S(\overline{J_1}, \overline{J_2})\} \\ &\overset{(a)}{\leq} ||\overline{J_1} - \overline{J_2}||_\infty \cdot \max\{||D\overline{T_{\mathbb{P}}}^1(\overline{J})||_\infty \mid \overline{J} \in S(\overline{J_1}, \overline{J_2})\} \\ &\overset{(b)}{\leq} ||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^1 \end{aligned}$$

Therefore, $\overline{T_{\mathbb{P}}}^1$ is a contractive mapping in A whose Lipschitz constant is $\lambda^1 < 1$. Following an analagous reasoning, we deduce that $\overline{T_{\mathbb{P}}}^2$ is a contractive mapping in A with Lipschitz constant $\lambda^2 < 1$, that is:

$$||\overline{T_{\mathbb{P}}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}}}^2(\overline{J_2})||_\infty \leq ||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^2$$

After proving the contractivity of each one of the components of $\overline{T_{\mathbb{P}}}$ with respect to $||\cdot||_\infty$, we will prove that $\overline{T_{\mathbb{P}}}$ is contractive in A with respect to $||\cdot||_\infty$. First of all, by definition of the norm $||\cdot||_\infty$, we obtain:

$$||\overline{T_{\mathbb{P}}}(\overline{J_1}) - \overline{T_{\mathbb{P}}}(\overline{J_2})||_\infty = \max\left\{||\overline{T_{\mathbb{P}}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}}}^1(\overline{J_2})||_\infty, ||\overline{T_{\mathbb{P}}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}}}^2(\overline{J_2})||_\infty\right\} \quad (6)$$

Therefore, by the contractivity of $\overline{T_{\mathbb{P}}}^1$ and $\overline{T_{\mathbb{P}}}^2$, the following chain holds:

$$\begin{aligned} ||\overline{T_{\mathbb{P}}}(\overline{J_1}) - \overline{T_{\mathbb{P}}}(\overline{J_2})||_\infty &\leq \max\left\{||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^1, ||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^2\right\} \\ &\leq \max\left\{||\overline{J_1} - \overline{J_2}||_\infty, ||\overline{J_1} - \overline{J_2}||_\infty\right\} \cdot \max\{\lambda^1, \lambda^2\} \\ &= ||\overline{J_1} - \overline{J_2}||_\infty \cdot \max\{\lambda^1, \lambda^2\} \end{aligned}$$

Thus, the operator $\overline{T_{\mathbb{P}}}$ is a contractive mapping with Lipschitz constant $\max\{\lambda^1, \lambda^2\} = \lambda^2$, since $\lambda^1 \leq \lambda^2$. Finally, by Banach fix-point theorem, we conclude that $\overline{T_{\mathbb{P}}}$ has only one fix-point in A.

General Case: Now, we will prove the general case assuming that $\mathbb{P}$ is a general MANLP. By Proposition 21, we can ensure the existence of a partition of MANLPs $\{\mathbb{P}_\gamma\}_{\gamma\in\Gamma}$ such that:

- Two rules with the same head for each $\mathbb{P}_\gamma$ do not exist.
- It is satisfied that $T_{\mathbb{P}}(I)(p) = \sup\{T_{\mathbb{P}_\gamma}(I)(p) \mid \gamma \in \Gamma\}$.

Given a propositional symbol $p \in \Pi_{\mathbb{P}}$, we will denote by $\mathbb{P}_{\gamma_i^p}$, with $i \in \{1, \ldots, m_p\}$, the subprograms of the partition with a unique rule whose head is p, where m_p is the number of rules with head p.

Since $\mathbb{P}$ is a finite program and taking into account the properties of the partition, we obtain, for each $\mathcal{C}([0,1])$-interpretation J and each propositional symbol $p \in \Pi_{\mathbb{P}}$, the following chain:

$$T_{\mathbb{P}}(J)(p) = [T^1_{\mathbb{P}}(J)(p), T^2_{\mathbb{P}}(J)(p)] =$$

$$= \Big[\sup\{T^1_{\mathbb{P}_{\gamma_i^p}}(J)(p) \mid i \in \{1, \ldots, m_p\}\}, \sup\{T^2_{\mathbb{P}_{\gamma_i^p}}(J)(p) \mid i \in \{1, \ldots, m_p\}\}\Big]$$

$$= \Big[\max\{T^1_{\mathbb{P}_{\gamma_i^p}}(J)(p) \mid i \in \{1, \ldots, m_p\}\}, \max\{T^2_{\mathbb{P}_{\gamma_i^p}}(J)(p) \mid i \in \{1, \ldots, m_p\}\}\Big]$$

Now, we will work individually with each component of the immediate consequence operator. Given two $\mathcal{C}([0,1])$-interpretations J_1 and J_2, we will build two programs $\mathbb{P}^1$ and $\mathbb{P}^2$ from the rules of $\mathbb{P}$ such that, for each propositional symbol $p \in \Pi_{\mathbb{P}}$, the following inequalities are satisfied:

$$|T^1_{\mathbb{P}}(J_1)(p) - T^1_{\mathbb{P}}(J_2)(p)| \leq |T^1_{\mathbb{P}^1}(J_1)(p) - T^1_{\mathbb{P}^1}(J_2)(p)|$$

$$|T^2_{\mathbb{P}}(J_1)(p) - T^2_{\mathbb{P}}(J_2)(p)| \leq |T^2_{\mathbb{P}^2}(J_1)(p) - T^2_{\mathbb{P}^2}(J_2)(p)|$$

Specifically, for each propositional symbol $p \in \Pi_{\mathbb{P}}$, we will add a rule with head p from the original program to $\mathbb{P}^1$ and a rule with head p, which can be the same or other different rule, from the original program to $\mathbb{P}^2$.

Let us begin then with the computation of the program $\mathbb{P}^1$, which we suppose empty by default, this is, without any rule. Let $p \in \Pi_{\mathbb{P}}$. As $\mathbb{P}$ has a finite number of rules, there exist $\gamma_1, \gamma_2 \in \Gamma$ such that

$$T^1_{\mathbb{P}}(J_1)(p) = \max\{T^1_{\mathbb{P}_{\gamma_i^p}}(J_1)(p) \mid i \in \{1, \ldots, m_p\}\} = T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) \quad (7)$$

$$T^1_{\mathbb{P}}(J_2)(p) = \max\{T^2_{\mathbb{P}_{\gamma_i^p}}(J_2)(p) \mid i \in \{1, \ldots, m_p\}\} = T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p) \quad (8)$$

Suppose that $T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p) \leq T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p)$. Then, by Equation (8), the following chain of inequalities holds:

$$T^1_{\mathbb{P}_{\gamma_1}}(J_2)(p) \leq T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p) \leq T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p)$$

Therefore

$$|T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) - T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p)| \leq |T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) - T^1_{\mathbb{P}_{\gamma_1}}(J_2)(p)|$$

Consequently, we add the rule of the program $\mathbb{P}_{\gamma_1}$ with head p to the program $\mathbb{P}^1$.

On the other hand, if $T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) \leq T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p)$, taking into account Equation (7), we obtain the following chain:

$$T^1_{\mathbb{P}_{\gamma_2}}(J_1)(p) \leq T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) \leq T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p)$$

Thus

$$|T^1_{\mathbb{P}_{\gamma_1}}(J_1)(p) - T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p)| \leq |T^1_{\mathbb{P}_{\gamma_2}}(J_1)(p) - T^1_{\mathbb{P}_{\gamma_2}}(J_2)(p)|$$

In this case, we add the rule of the program $\mathbb{P}_{\gamma_2}$ with head p to $\mathbb{P}^1$.

Progressing with an analogous reasoning with the other propositional symbols, we obtain a program $\mathbb{P}^1$ such that for each propositional symbol $p \in \Pi_{\mathbb{P}}$ at most one rule with head p appears in $\mathbb{P}^1$ and

$$|T^1_{\mathbb{P}}(J_1)(p) - T^1_{\mathbb{P}}(J_2)(p)| \quad \leq \quad |T^1_{\mathbb{P}^1}(J_1)(p) - T^1_{\mathbb{P}^1}(J_2)(p)| \tag{9}$$

Likewise, it is analogously obtained a program $\mathbb{P}^2$ such that for each propositional symbol $p \in \Pi_{\mathbb{P}}$ we obtain that

$$|T^2_{\mathbb{P}}(J_1)(p) - T^2_{\mathbb{P}}(J_2)(p)| \quad \leq \quad |T^2_{\mathbb{P}^2}(J_1)(p) - T^2_{\mathbb{P}^2}(J_2)(p)| \tag{10}$$

Since Equations (9) and (10) hold, for all $p \in \Pi_{\mathbb{P}}$, we obtain that:

$$\begin{aligned} ||T^1_{\mathbb{P}}(J_1) - T^1_{\mathbb{P}}(J_2)||_\infty &\leq ||T^1_{\mathbb{P}^1}(J_1) - T^1_{\mathbb{P}^1}(J_2)||_\infty \\ ||T^2_{\mathbb{P}}(J_1) - T^2_{\mathbb{P}}(J_2)||_\infty &\leq ||T^2_{\mathbb{P}^2}(J_1) - T^2_{\mathbb{P}^2}(J_2)||_\infty \end{aligned}$$

Moreover, by the definition of $\overline{T_{\mathbb{P}}}$, the following inequalities are satisfied:

$$\begin{aligned} ||\overline{T_{\mathbb{P}}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}}}^1(\overline{J_2})||_\infty &\leq ||\overline{T_{\mathbb{P}^1}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}^1}}^1(\overline{J_2})||_\infty \\ ||\overline{T_{\mathbb{P}}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}}}^2(\overline{J_2})||_\infty &\leq ||\overline{T_{\mathbb{P}^2}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}^2}}^2(\overline{J_2})||_\infty \end{aligned}$$

These inequalities provide the contractivity of $\overline{T_{\mathbb{P}}}$ in A as we show next. By Equation (6) and by the definition of $\mathbb{P}^1$ and $\mathbb{P}^2$, we obtain that

$$\begin{aligned} ||\overline{T_{\mathbb{P}}}(\overline{J_1}) - \overline{T_{\mathbb{P}}}(\overline{J_2})||_\infty &= \max\Big\{||\overline{T_{\mathbb{P}}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}}}^1(\overline{J_2})||_\infty, ||\overline{T_{\mathbb{P}}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}}}^2(\overline{J_2})||_\infty\Big\} \\ &\leq \max\Big\{||\overline{T_{\mathbb{P}^1}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}^1}}^1(\overline{J_2})||_\infty\, ||\overline{T_{\mathbb{P}^2}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}^2}}^2(\overline{J_2})||_\infty\Big\} \end{aligned}$$

Moreover, since $\mathbb{P}^1$ and $\mathbb{P}^2$ are programs such that each propositional symbol $p \in \Pi_{\mathbb{P}}$ has at most one rule in $\mathbb{P}^1$ and one rule in $\mathbb{P}^2$ with head p, both programs are in the situation of the base case. As a consequence, $\overline{T_{\mathbb{P}^1}}^1$ and $\overline{T_{\mathbb{P}^2}}^2$ are both contractive mappings in A with Lipschitz constants λ^1 and λ^2, respectively. Taking into account the first component of $\overline{T_{\mathbb{P}^1}}$ and the second of $\overline{T_{\mathbb{P}^2}}$, we deduce that

$$\begin{aligned} ||\overline{T_{\mathbb{P}}}(\overline{J_1}) - \overline{T_{\mathbb{P}}}(\overline{J_2})||_\infty &\leq \max\left\{||\overline{T_{\mathbb{P}^1}}^1(\overline{J_1}) - \overline{T_{\mathbb{P}^1}}^1(\overline{J_2})||_\infty \, ||\overline{T_{\mathbb{P}^2}}^2(\overline{J_1}) - \overline{T_{\mathbb{P}^2}}^2(\overline{J_2})||_\infty\right\} \\ &\leq \max\left\{||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^1, ||\overline{J_1} - \overline{J_2}||_\infty \cdot \lambda^2\right\} \\ &\leq \max\left\{||\overline{J_1} - \overline{J_2}||_\infty, ||\overline{J_1} - \overline{J_2}||_\infty\right\} \cdot \max\{\lambda^1, \lambda^2\} \\ &= ||\overline{J_1} - \overline{J_2}||_\infty \cdot \max\{\lambda^1, \lambda^2\} \end{aligned}$$

Therefore, $\overline{T_{\mathbb{P}}}$ is a contractive mapping in A whose Lipschitz constant is equal to $\max\{\lambda^1, \lambda^2\} = \lambda^2$. Applying Banach fix-point theorem, we obtain that $\overline{T_{\mathbb{P}}}$ has a unique fix-point in A.

Now, we will prove that $T_{\mathbb{P}}$ has a unique fix-point in $\mathcal{C}([0,1])^n$. For all $\mathcal{C}([0,1])$-interpretation I being a fix-point of $T_{\mathbb{P}}$, we will demonstrate that $\overline{I} \in A$ and $\overline{I}$ is a fix-point of $\overline{T_{\mathbb{P}}}$.

We suppose that I is a fix-point of $T_{\mathbb{P}}$, that is, I is a $\mathcal{C}([0,1])$-interpretation such that $T_{\mathbb{P}}(I) = I$. On the one hand, considering the inclusion mapping ι, we can ensure that $\overline{T_{\mathbb{P}}}(\overline{I}) = \iota(T_{\mathbb{P}}(I)) = \iota(I) = \overline{I}$. Therefore, $\overline{I}$ is a fix-point of $\overline{T_{\mathbb{P}}}$. On the other hand, we will evince that $\overline{I} \in A$. Clearly, the inequality $T_{\mathbb{P}}(I)(p) \leq \max\{[\vartheta^1, \vartheta^2] \mid \langle p \leftarrow^{\alpha_w \gamma_w}_{\beta_w \delta_w} \mathcal{B}; [\vartheta^1, \vartheta^2]\rangle \in \mathbb{P}\}$ holds, for all $\mathcal{C}([0,1])$-interpretation I. As a consequence, $I = T_{\mathbb{P}}(I) \sqsubseteq I_\vartheta$ and so, by definition of set A, we obtain that $\overline{I} \in A$.

Finally, by Proposition 26 and Theorem 28, we can conclude that the only fix-point of $T_{\mathbb{P}}$ is actually the only stable model of the program. □

The following example shows a simple MANLP which satisfies the hypothesis of Theorem 34.

Example 35. Given the set of propositional symbols $\Pi_{\mathbb{P}} = \{p, q, s, t\}$ and the multi-adjoint normal lattice $(\mathcal{C}([0,1]), \leq, \leftarrow^{11}_{11}, \&^{11}_{11}, \leftarrow^{23}_{12}, \&^{23}_{12}, \leftarrow^{22}_{11}, \&^{22}_{11}, \neg)$, we will define the multi-adjoint normal logic program $\mathbb{P}$ as follows:

$$\begin{aligned} r_1 &: \langle p \leftarrow^{11}_{11} \neg q \; ; \; [0.7, 0.8]\rangle \\ r_2 &: \langle s \leftarrow^{23}_{12} p \; ; \; [0.4, 0.5]\rangle \\ r_3 &: \langle p \leftarrow^{22}_{11} s * \neg t \; ; \; [0.5, 0.6]\rangle \\ r_4 &: \langle q \leftarrow^{22}_{11} t * \neg p \; ; \; [0.7, 0.9]\rangle \end{aligned}$$

In order to apply Theorem 34 to the program $\mathbb{P}$, we need to verify that the inequality

$$\sum_{j=1}^{h}(\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(\vartheta^2_{q_j})^{\delta_w-1}\cdot\left(\vartheta^2_{q_1}\cdots\vartheta^2_{q_{j-1}}\cdot\vartheta^2_{q_{j+1}}\cdots\vartheta^2_{q_h}\right)^{\delta_w}+(\vartheta^2)^{\beta_w}\cdot\delta_w\cdot(k-h)(\vartheta^2_{q_1}\cdots\vartheta^2_{q_h})^{\delta_w}<1$$

holds for every rule $\langle p \leftarrow^{\alpha_w\gamma_w}_{\beta_w\delta_w} q_1 * \cdots * q_h * \neg q_{h+1} * \cdots * \neg q_k; [\vartheta^1, \vartheta^2]\rangle \in \mathbb{P}$.

Concerning rule r_1, we obtain that

$$0.9 < 1$$

For r_2, it holds that

$$0.5 \cdot 2 \cdot 0.9 = 0.9 < 1$$

For r_3 we obtain: $0.6 + 0.5 \cdot 0.6 = 0.9 < 1$. Finally, for r_3 we have

$$0.9 + 0.9 \cdot 0 = 0.9 < 1$$

Therefore, the program $\mathbb{P}$ satisfies the hypothesis of the uniqueness theorem (Theorem 34). As a consequence, we can ensure that it has a unique stable model.

Moreover, this stable model can be computed by iterating $T_{\mathbb{P}}$ under the minimum interpretation $I_\bot$. The following table shows the iterations of the immediate consequence operator, taking as first the entry interpretation constantly $\bot$, which is denoted as $I_\bot$.

	p	q	s	t
$I_\bot$	[0,0]	[0,0]	[0,0]	[0,0]
$T_{\mathbb{P}}(I_\bot)$	[0.7,0.9]	[0,0]	[0,0]	[0,0]
$T^2_{\mathbb{P}}(I_\bot)$	[0.7,0.9]	[0,0]	[0.05488,0.405]	[0,0]
$T^3_{\mathbb{P}}(I_\bot)$	[0.7,0.9]	[0,0]	[0.05488,0.405]	[0,0]

Thus, $T^2_{\mathbb{P}}(I_\bot)$ is the unique stable model of $\mathbb{P}$. □

The previous example has also shown how the unique stable model can be computed. We only need to iterate the operator $T_{\mathbb{P}}$ under $I_\bot$ and the obtained fix-point will be the stable model. As a consequence, when the hypotheses of the unicity theorem hold, the computational complexity for computing the unique stable model is the same as in the case of positive logic programs. Hence, in this case, the use of negation operators does not increase the complexity for obtaining consequences from the knowledge system represented by the multi-adjoint normal logic program.

5. Conclusions and future work

This paper has considered the philosophy of the multi-adjoint paradigm to introduce the syntax and semantics of a new and more flexible normal logic programming framework. We have proven that the stable models in multi-adjoint normal logic programs satisfy the more important properties of the classical and residuated case. Moreover, we have proven the existence of stable models of logic programs defined on a multi-adjoint normal lattice whose carrier is a convex compact set and the operators are continuous. Furthermore, a special kind of multi-adjoint normal logic program on the subinterval lattice $(\mathcal{C}([0,1]), \leq)$ has been introduced in which the uniqueness of stable models is ensured. Moreover, this stable model can easily be computed by iterating the immediate consequence operator from the minimum interpretation.

The introduced results on the existence and uniqueness of stable models can straightforwardly be applied to different useful frameworks, such as monotonic and residuated logic programming [13, 12], fuzzy logic programming [47] and possibilistic logic programming [14], in which a negation operator is considered in the language. In addition, we can use the introduced results in fuzzy answer set programming when the notion of x-consistency is included in our logical framework [35]. These results may also be applied to generalized annotated logic programs [20], when a negation operator is used. Taking into consideration the relation given in [22] to the fuzzy logic programming, we can define a new generalized annotated logic programming in which a negation operator (decreasing with respect to the ordering considered for the $T_{\mathbb{P}}$ operator) can be used and, as a consequence, a more flexible annotated logic is obtained. Moreover, the proposed framework in this paper will be also compared with the paraconsistent logic programming [1, 2], the disjunctive logic programming [6, 34, 36, 43] and the sorted logic programming [5, 11]. These relations are more complex and they will be studied in depth in the future. In addition, we will apply the obtained results to other logics with negation operators, such as Possibilistic Defeasible Logic Programming (PDeLP) [3].

The representation of knowledge from databases containing uncertainty and inconsistent information is an important task. Different authors [17, 21, 38, 48] have highlighted that, in order to handle this goal in a suitable way, it is necessary to distinguish what can be proved to be false from what is false because it cannot be proved true, which is called false by default. This difference can be obtained by the use of an explicit negation operator in normal logic programs. This philosophy is used by the paraconsistent logic programming framework given in [1, 2], where the explicit and default negations have been considered and related by the coherence principle. Therefore, the relation between paraconsistent logic programming [1, 2] and multi-adjoint normal logic programming will be also important for studying the notion of inconsistency in MANLPs.

Moreover, we will also adapt the definitions of coherence and inconsistency given in [27, 28] to the multi-adjoint framework, which are focused on the detection of plausible stable models. In particular, we will inspect measuring inconsistency in fuzzy answer set semantics for MANLPs. Furthermore, real-life applications will be considered in which the introduced flexible multi-adjoint framework will be applied.

[1] J. Alcântara, C. V. Damásio, and L. M. Pereira. Paraconsistent logic programs. *Lecture Notes in Artificial Intelligence*, 2424:345–356, 2002.

[2] J. Alcântara, C. V. Damásio, and L. M. Pereira. An encompassing framework for paraconsistent logic programs. *Journal of Applied Logic*, 3(1):67 – 95, 2005. A Paraconsistent Decagon.

[3] T. Alsinet, C. Chesñevar, L. Godo, S. Sandri, and G. Simari. Formalizing argumentative reasoning in a possibilistic logic programming setting with fuzzy unification. *International Journal of Approximate Reasoning*, 48(3):711–729, 2008.

[4] S. Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3(1):133–181, 1922.

[5] C. Baral, M. Gelfond, and N. Rushton. Probabilistic reasoning with answer sets. *Theory and Practice of Logic Programming*, 9(01):57–144, jan 2009.

[6] K. Bauters, S. Schockaert, M. D. Cock, and D. Vermeir. Characterizing and extending answer set semantics using possibility theory. *Theory and Practice of Logic Programming*, 15(01):79–116, jan 2014.

[7] M. E. Cornejo, D. Lobo, and J. Medina. Stable models in normal residuated logic programs. In J. Kacprzyk, L. Koczy, and J. Medina, editors, *7th European Symposium on Computational Intelligence and Mathematices (ESCIM 2015)*, pages 150–155, 2015.

[8] M. E. Cornejo, D. Lobo, and J. Medina. Towards multi-adjoint logic programming with negations. In L. Koczy and J. Medina, editors, *8th European Symposium on Computational Intelligence and Mathematics (ESCIM 2016)*, pages 24–29, 2016.

[9] M. E. Cornejo, J. Medina, and E. Ramírez-Poussa. A comparative study of adjoint triples. *Fuzzy Sets and Systems*, 211:1–14, 2013.

[10] M. E. Cornejo, J. Medina, and E. Ramírez-Poussa. Multi-adjoint algebras versus non-commutative residuated structures. *International Journal of Approximate Reasoning*, 66:119–138, 2015.

[11] C. Damásio, J. Medina, and M. Ojeda-Aciego. Termination of logic programs with imperfect information: applications and query procedure. *Journal of Applied Logic*, 5:435–458, 2007.

[12] C. V. Damásio and L. M. Pereira. Hybrid probabilistic logic programs as residuated

logic programs. In *Logics in Artificial Intelligence, JELIA'00*, pages 57–73. Lecture Notes in Artificial Intelligence 1919, 2000.

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

[14] D. Dubois, J. Lang, and H. Prade. Automated reasoning using possibilistic logic: semantics, belief revision, and variable certainty weights. *Knowledge and Data Engineering, IEEE Transactions on*, 6(1):64–71, feb 1994.

[15] M. Fitting. The family of stable models. *The Journal of Logic Programming*, pages 17(2–4):197–225, 1993.

[16] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *ICLP/SLP*, volume 88, pages 1070–1080, 1988.

[17] M. Gelfond and V. Lifschitz. Logic programs with classical negation. In D. H. Warren and P. Szeredi, editors, *Logic Programming*, pages 579–597. MIT Press, Cambridge, MA, USA, 1990.

[18] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9(3):365–385, 1991.

[19] P. Julián, G. Moreno, and J. Penabad. On fuzzy unfolding: A multi-adjoint approach. *Fuzzy Sets and Systems*, 154(1):16–33, 2005.

[20] M. Kifer and V. S. Subrahmanian. Theory of generalized annotated logic programming and its applications. *J. of Logic Programming*, 12:335–367, 1992.

[21] R. A. Kowalski and F. Sadri. Logic programs with exceptions. *New Generation Computing*, 9(3):387–400, Aug 1991.

[22] S. Krajči, R. Lencses, and P. Vojtáš. A comparison of fuzzy and annotated logic programming. *Fuzzy Sets and Systems*, 144(1):173 – 192, 2004.

[23] D. Leborgne. *Calcul différentiel et géométrie*. Presses Universitaires de France, 1982. ISBN 978-2-13-037495-4.

[24] J. Lloyd. *Foundations of Logic Programming*. Springer Verlag, 1987.

[25] Y. Loyer and U. Straccia. Epistemic foundation of stable model semantics. *Journal of Theory and Practice of Logic Programming*, pages 6:355–393, 2006.

[26] N. Madrid and M. Ojeda-Aciego. Towards a fuzzy answer set semantics for residuated logic programs. *Web Intelligence/IAT Workshops*, pages 260–264, 2008.

[27] N. Madrid and M. Ojeda-Aciego. On coherence and consistence in fuzzy answer set semantics for residuated logic programs. *Lect. Notes in Computer Science*, pages 5571:60–67, 2009.

[28] N. Madrid and M. Ojeda-Aciego. Measuring inconsistency in fuzzy answer set semantics. *IEEE Transactions on Fuzzy Systems*, 19(4):605–622, Aug. 2011.

[29] N. Madrid and M. Ojeda-Aciego. On the existence and unicity of stable models in normal residuated logic programs. *International Journal of Computer Mathematics*, 2012.

[30] J. Medina. Adjoint pairs on interval-valued fuzzy sets. In E. Hüllermeier, R. Kruse, and F. Hoffmann, editors, *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, volume 81 of *Communications in Computer and Information Science*, pages 430–439. Springer, 2010.

[31] J. Medina, E. Mérida-Casermeiro, and M. Ojeda-Aciego. A neural implementation of multi-adjoint logic programming. *Journal of Applied Logic*, 2(3):301–324, 2004.

[32] J. Medina, M. Ojeda-Aciego, and P. Vojtáš. A multi-adjoint logic approach to abductive reasoning. In *Logic Programming, ICLP'01*, pages 269–283. Lecture Notes in Computer Science 2237, 2001.

[33] J. Medina, M. Ojeda-Aciego, and P. Vojtáš. Multi-adjoint logic programming with continuous semantics. In *Logic Programming and Non-Monotonic Reasoning, LPNMR'01*, pages 351–364. Lecture Notes in Artificial Intelligence 2173, 2001.

[34] J. Minker and A. Rajasekar. A fixpoint semantics for disjunctive logic programs. *The Journal of Logic Programming*, 9(1):45 – 74, 1990.

[35] D. V. Nieuwenborgh, M. D. Cock, and D. Vermeir. An introduction to fuzzy answer set programming. *Annals of Mathematics and Artificial Intelligence*, 50(3-4):363–388, jul 2007.

[36] J. C. Nieves, M. Osorio, and U. Cortés. Semantics for possibilistic disjunctive programs. *Theory and Practice of Logic Programming*, 13(01):33–70, jul 2011.

[37] J. Pavelka. On fuzzy logic I, II, III. *Zeitschr. f. Math. Logik und Grundl. der Math.*, 25, 1979.

[38] L. M. Pereira and J. J. Alferes. Well founded semantics for logic programs with explicit negation. In *EUROPEAN CONFERENCE ON ARTIFICIAL INTELLIGENCE*, pages 102–106. John Wiley & Sons, 1992.

[39] T. Przymusinski. Well-founded semantics coincides with three-valued stable semantics. *Fundamenta Informaticae*, pages 13:445–463, 1990.

[40] U. Straccia. Query answering in normal logic programs under uncertainty. *Lect. Notes in Computer Science*, pages 3571:687–700, 2005.

[41] U. Straccia. Query answering under the any-world assumption for normal logic programs. *Lect. Notes in Computer Science*, pages 3571:687–700, 2006.

[42] U. Straccia. A top-down query answering procedure for normal logic programs under the any-world assumption. *Proc. of the 10th International Conference on Principles of Knowledge Representation*, pages 329–339, 2006.

[43] U. Straccia, M. Ojeda-Aciego, and 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):1881–1911, 2009.

[44] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.*, 5(2):285–309, 1955.

[45] M. Tsoy-Wo. *Classical Analysis on Normed Spaces.* World Scientific Publishing, 1995.

[46] A. Van Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *J. ACM*, 38(3):619–649, July 1991.

[47] P. Vojtáš. Fuzzy logic programming. *Fuzzy sets and systems*, 124(3):361–370, 2001.

[48] G. Wagner. *A database needs two kinds of negation*, pages 357–371. Springer Berlin Heidelberg, Berlin, Heidelberg, 1991.